Configuración y administración de redes Oracle® Solaris 11.1

Referencia: E36456–01
Octubre de 2012

ORACLE

130327@25097

Contenido

Prefacio

Bienvenido a *Configuración y administración de redes Oracle Solaris 11.1*. Este manual forma parte de la serie *Establecimiento de una red Oracle Solaris 11.1* que abarca procedimientos y temas básicos para configurar redes Oracle Solaris. En este manual, se da por sentado que ya instaló Oracle Solaris. Debe estar listo para configurar la red o para configurar el software de red que se necesite.

Quién debe utilizar este manual

Este manual está destinado a las personas encargadas de administrar sistemas que ejecutan Oracle Solaris configurado en red. Para utilizar este manual, se debe tener como mínimo dos años de experiencia en la administración de sistemas UNIX . Puede resultar útil participar en cursos de formación para administración de sistemas UNIX.

Acceso a Oracle Support

Los clientes de Oracle tienen acceso a soporte electrónico por medio de My Oracle Support. Para obtener más información, visite `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info` o, si tiene alguna discapacidad auditiva, visite `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs`.

Convenciones tipográficas

La siguiente tabla describe las convenciones tipográficas utilizadas en este manual.

TABLA P–1 Convenciones tipográficas

Tipos de letra	Descripción	Ejemplo
`AaBbCc123`	Los nombres de los comandos, los archivos, los directorios y los resultados que el equipo muestra en pantalla	Edite el archivo `.login`. Utilice el comando `ls -a` para mostrar todos los archivos. `nombre_sistema% tiene correo.`

TABLA P–1 Convenciones tipográficas *(Continuación)*

Tipos de letra	Descripción	Ejemplo
AaBbCc123	Lo que se escribe, en contraposición con la salida del equipo en pantalla	`nombre_sistema% su` `Contraseña:`
aabbcc123	Marcador de posición: sustituir por un valor o nombre real	El comando necesario para eliminar un archivo es `rm` *nombre_archivo*.
AaBbCc123	Títulos de los manuales, términos nuevos y palabras destacables	Consulte el capítulo 6 de la *Guía del usuario*. Una *copia en caché* es aquella que se almacena localmente. *No* guarde el archivo. **Nota:** algunos elementos destacados aparecen en negrita en línea.

Indicadores de los shells en los ejemplos de comandos

La tabla siguiente muestra los indicadores de sistema UNIX predeterminados y el indicador de superusuario de shells que se incluyen en los sistemas operativos Oracle Solaris. Tenga en cuenta que el indicador predeterminado del sistema que se muestra en los ejemplos de comandos varía según la versión de Oracle Solaris.

TABLA P–2 Indicadores de shell

Shell	Indicador
Shell Bash, shell Korn y shell Bourne	`$`
Shell Bash, shell Korn y shell Bourne para superusuario	`#`
Shell C	`machine_name%`
Shell C para superusuario	`machine_name#`

◆ ◆ ◆ CAPÍTULO 1

Planificación de la implementación de red

En este capítulo, se describen brevemente las distintas consideraciones que debe tener en cuenta al planificar la configuración de red. Estas cuestiones lo ayudarán a implementar la red de una manera organizada y rentable. Tenga en cuenta que los detalles sobre la planificación de la red están fuera del alcance de este manual. Únicamente se proporcionan instrucciones generales.

En este manual, se da por sentado que usted está familiarizado con los conceptos y la terminología básicos. Para obtener una descripción de la forma en que el conjunto de protocolos TCP/IP se implementa en Oracle Solaris 11, consulte "Pila de red en Oracle Solaris" de *Introducción a redes de Oracle Solaris 11*.

Planificación de la red (mapa de tareas)

En la siguiente tabla, se enumeran las distintas tareas para planificar la configuración de red.

Tarea	Descripción	Para obtener información
Identificar los requisitos de hardware de la topología de red planificada.	Determine los tipos de equipo que necesita para el sitio de red.	"Determinación del hardware de red" en la página 12 Para obtener información sobre un tipo de equipo específico, consulte la documentación del fabricante del equipo.
Determinar el tipo de direcciones IP que se utilizarán y obtener direcciones IP registradas.	Seleccione si está implementando una red puramente IPv4, una red IPv6 o una red que utiliza ambos tipos de direcciones IP. Obtenga direcciones IP exclusivas para comunicarse con redes públicas en Internet.	"Cómo decidir el formato de las direcciones IP para la red" en la página 13 "Cómo obtener el número de IP de la red" en la página 15

Tarea	Descripción	Para obtener información
Determinar un esquema de nomenclatura para identificar los hosts de la red y también el servicio de nombres que se utilizará.	Cree una lista de nombres para asignar a los sistemas de la red y decida si se utilizarán NIS, LDAP, DNS o las bases de datos de red en el directorio `/etc` local.	"Administración de nombres de host" en la página 16 "Selección de un servicio de nombres y de directorios" en la página 16
Si es necesario, establecer subdivisiones administrativas y diseñar una estrategia para subredes.	Decida si el sitio requiere la división de la red en subredes para prestar servicio a subdivisiones administrativas.	"Uso de subredes" en la página 17
Determinar dónde colocar los enrutadores en el diseño de la red.	Si la red es lo suficientemente grande como para requerir el uso de enrutadores, cree una topología de red que los admita.	"Planificación de enrutadores en la red" en la página 18
Decidir si se deben crear redes virtuales en el esquema de configuración de red general.	Es posible que deba crear redes virtuales dentro de un sistema para reducir el espacio utilizado por el hardware en la red.	*Uso de redes virtuales en Oracle Solaris 11.1*

Determinación del hardware de red

El número de sistemas que espera admitir afecta la configuración de la red. Es posible que su organización requiera una pequeña red de varias docenas de sistemas independientes ubicados en una única planta de un edificio. También es posible que requiera la configuración de una red con más de 1.000 sistemas ubicados en varios edificios. Esta configuración podría hacer necesaria la división de la red en subdivisiones denominadas *subredes*.

A continuación, se presentan algunas de las decisiones de planificación que debe tomar relacionadas con el hardware:

- La topología de red, el diseño y las conexiones del hardware de red
- El tipo y número de sistemas host que admite la red, incluidos los servidores que pueden ser necesarios
- Los dispositivos de red que se instalarán en estos sistemas
- El tipo de medios de red que se utilizarán, como Ethernet, etc.
- Si necesita puentes o enrutadores que extiendan este medio o conecten la red local a redes externas

Nota – Para obtener una descripción sobre cómo funcionan los enrutadores, consulte "Planificación de enrutadores en la red" en la página 18. Para obtener una descripción general de los puentes, consulte "Descripción general sobre puentes" de *Gestión del rendimiento de red de Oracle Solaris 11.1*.

Cómo decidir el formato de las direcciones IP para la red

Al planificar el esquema de direcciones de la red, debe tener en cuenta los siguientes factores:

- El tipo de dirección IP que desea utilizar: IPv4 o IPv6
- El número de sistemas potenciales de la red
- El número de sistemas que son enrutadores o sistemas de host múltiple, que requieren varias tarjetas de interfaz de red (NIC) con sus propias direcciones IP individuales
- Si se utilizarán direcciones privadas en la red
- Si habrá un servidor DHCP que administre las agrupaciones de direcciones IPv4

Brevemente, los tipos de direcciones IP incluyen los siguientes:

Direcciones IPv4

Estas direcciones de 32 bits son el formato original de direcciones IP para TCP/IP. Más adelante, IETF desarrolló direcciones de *enrutamiento entre dominios sin clase (CIDR)* como una solución de corto a mediano plazo para la escasez de direcciones IPv4 y la capacidad limitada de las tablas de enrutamiento de Internet globales.

Para obtener más información, consulte los siguientes recursos:

- Internet Protocol DARPA Internet Program Protocol Specification (`http://tools.ietf.org/html/rfc791`) (Especificación del protocolo de Internet de DARPA Internet Program)
- Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan (`http://tools.ietf.org/html/rfc4632`) (Enrutamiento entre dominios sin clase [CIDR]: plan de agregación y asignación de direcciones de Internet)

En la siguiente tabla, se proporcionan las subredes en formato de notación CIDR y en formato decimal con punto.

TABLA 1–1 Prefijos CIDR y sus equivalentes decimales

Prefijo de red CIDR	Equivalente de subred decimal con punto	Direcciones IP disponibles
/19	255.255.224.0	8,192

TABLA 1–1 Prefijos CIDR y sus equivalentes decimales *(Continuación)*

Prefijo de red CIDR	Equivalente de subred decimal con punto	Direcciones IP disponibles
/20	255.255.240.0	4,096
/21	255.255.248.0	2,048
/22	255.255.252.0	1,024
/23	255.255.254.0	512
/24	255.255.255.0	256
/25	255.255.255.128	128
/26	255.255.255.192	64
/27	255.255.255.224	32

Direcciones DHCP

El protocolo de configuración dinámica de sistemas (DHCP, Dynamic Host Configuration Protocol) permite a un sistema recibir información de configuración de un servidor DHCP, incluida una dirección IP, como parte del proceso de inicio. Los servidores DHCP cuentan con agrupaciones de direcciones IP desde las que se asignan direcciones a los clientes DHCP. Un sitio que utilice DHCP puede utilizar una agrupación de direcciones IP menor que la que se necesitaría si todos los clientes tuvieran asignada una dirección IP permanente. Puede configurar el servicio DHCP para administrar las direcciones IP del sitio, o parte de ellas. Para obtener más información, consulte el Capítulo 1, "Acerca de DHCP (descripción general)" de *Uso de DHCP en Oracle Solaris 11.1*.

Direcciones IPv6

Las direcciones IPv6 de 128 bits proporcionan un espacio de direcciones más grande que el que está disponible con IPv4. Al igual que con las direcciones IPv4 en formato CIDR, las direcciones IPv6 no tienen clase y utilizan prefijos para designar la parte de la dirección que define la red del sitio. Para obtener más información acerca de las direcciones IPv6, consulte Internet Protocol, Version 6 (IPv6) Specification (`http://tools.ietf.org/html/rfc2460`).

Direcciones privadas y prefijos de documentación

La IANA ha reservado un bloque de direcciones IPv4 y un prefijo de sitio IPv6 para utilizar en redes privadas. Las direcciones privadas se utilizan para tráfico de red dentro de una red privada. Estas direcciones también se utilizan en la documentación.

La tabla siguiente muestra los intervalos de direcciones IPv4 privadas y sus correspondientes máscaras de red.

Rango de direcciones IPv4	Máscara de red
10.0.0.0 - 10.255.255.255	10.0.0.0
172.16.0.0 - 172.31.255.255	172.16.0.0
192.168.0.0 - 192.168.255.255	192.168.0.0

Para las direcciones IPv6, el prefijo `2001:db8::/32` es un prefijo IPv6 especial que se utiliza específicamente para ejemplos de documentación. Los ejemplos de este manual utilizan direcciones IPv4 privadas y el prefijo de documentación de IPv6 reservado.

Cómo obtener el número de IP de la red

Una red IPv4 se define con una combinación de un número de red IPv4 más una *máscara de red*. Una red IPv6 se define mediante el *prefijo de sitio* y si cuenta con subredes mediante el *prefijo de subred*.

Para activar la red privada para que se comunique con redes externas en Internet, debe obtener un número de IP registrado para su red de la organización pertinente. Esta dirección pasará a ser el número de red para el esquema de direcciones IPv4 o el prefijo de sitio para el esquema de direcciones IPv6.

Los proveedores de servicios de Internet proporcionan direcciones IP para las redes cuyos precios se basan en los distintos niveles de servicio. Compare los diferentes ISP para determinar cuál de ellos proporciona el mejor servicio para su red. Los ISP normalmente ofrecen a las empresas direcciones asignadas dinámicamente o direcciones IP estáticas. Algunos ISP ofrecen direcciones tanto IPv4 como IPv6.

Si su sitio es un ISP, obtiene bloques de direcciones IP para los clientes a través de un registro de Internet (IR) para su configuración regional. La Autoridad de números asignados de Internet (IANA o Internet Assigned Numbers Authority) es la principal responsable de la delegación de direcciones IP registradas a los registros de Internet de todo el mundo. Cada IR cuenta con información de registro y plantillas para la configuración regional en la que el IR ofrece el servicio. Para obtener información sobre la IANA y sus IR, consulte la página de servicio de direcciones IP de IANA (`http://www.iana.org/ipaddress/ip-addresses.htm`).

Entidades de denominación en la red

Los protocolos TCP/IP localizan un sistema en una red utilizando su dirección IP. Sin embargo, un nombre de host le permite identificar sistemas más fácilmente que las direcciones IP. Los protocolos TCP/IP (y Oracle Solaris) requieren tanto la dirección IP como el nombre de host para identificar un sistema de forma exclusiva.

Desde el punto de vista del protocolo TCP/IP, una red es un conjunto de entidades con nombre. Un host es una entidad con un nombre. Un enrutador es una entidad con un nombre. La red es una entidad con un nombre. Del mismo modo, se puede asignar un nombre a un grupo o departamento en el que esté instalada la red, así como a una división, región o compañía. En teoría, la jerarquía de nombres que se pueden utilizar para identificar una red prácticamente no tiene límites. El nombre de dominio identifica un *dominio*.

Administración de nombres de host

Planifique un esquema de nomenclatura para los sistemas que compondrán la red. Para los sistemas que funcionan como servidores y que tienen varias NIC, se debe proporcionar al menos un nombre de host asociado con la dirección IP de su interfaz de red principal.

No puede haber dos máquinas en la red que tengan el mismo nombre de host. Por lo tanto, cada nombre de host debe ser exclusivo para cada sistema. Sin embargo, un host o un sistema con un nombre exclusivo asignado pueden tener varias direcciones IP.

Cuando planifique su red, realice una lista de las direcciones IP y sus nombres de host asociados para poder acceder a ellos fácilmente durante el proceso de configuración. Dicha lista le ayudará a verificar que todos los nombres de host sean exclusivos.

Selección de un servicio de nombres y de directorios

En Oracle Solaris, puede seleccionar entre tres tipos de servicios de nombres: archivos locales, NIS y DNS. Los servicios de nombres conservan información crítica sobre las máquinas de una red, como los nombres de host, las direcciones IP, las direcciones Ethernet, etc. También puede utilizar el servicio de directorios LDAP además del servicio de nombres o en lugar de él. Para obtener una introducción a los servicios de Oracle Solaris, consulte Parte I, "Acerca de los servicios de nombres y directorios" de *Trabajo con servicios de nombres y directorios en Oracle Solaris 11.1*.

Durante la instalación del sistema operativo, proporcione el nombre de host y la dirección IP del sistema autónomo, cliente o de servidor. El programa de instalación agrega esta información a la base de datos `hosts` para que el servicio de red la utilice al prestar servicio a la red.

La configuración de las bases de datos de red es imprescindible. Debe decidir qué servicio de nombres utilizará como parte del proceso de planificación de la red. Asimismo, la decisión de utilizar servicios de nombres también determina si organizará la red en un dominio administrativo.

Para un servicio de nombres, puede seleccionar una de las opciones siguientes:

- NIS o DNS. Los servicios de nombres NIS y DNS conservan bases de datos de red en varios servidores de la red. En *Trabajo con servicios de nombres y directorios en Oracle Solaris 11.1*, se describen estos servicios de nombres y se explica cómo configurar las bases de datos. Asimismo, la guía explica de forma pormenorizada los conceptos de "espacio de nombres" y "dominio administrativo".
- Archivos locales. Si no desea implementar NIS, LDAP o DNS, la red utiliza *archivos locales* para proporcionar el servicio de nombres. El término "archivos locales" hace referencia a la serie de archivos del directorio `/etc` que utilizan las bases de datos de red. En los procedimientos de este manual se presupone que está utilizando archivos locales para el servicio de nombres, a menos que se especifique lo contrario.

Nota – Si decide utilizar archivos locales como servicio de nombres para la red, puede configurar otro servicio de nombres posteriormente.

Nombres de dominio

Muchas redes organizan sus hosts y enrutadores en una jerarquía de dominios administrativos. Si utiliza el servicio de nombres NIS o DNS, debe seleccionar un nombre de dominio para la organización que sea exclusivo en todo el mundo. Para asegurarse de que su nombre de dominio sea exclusivo, debe registrarlo en InterNIC. Si tiene previsto utilizar DNS, también debe registrar su propio nombre de dominio en InterNIC.

La estructura del nombre de dominio es jerárquica. Un nuevo dominio normalmente se ubica debajo de un dominio relacionado que ya existe. Por ejemplo, el nombre de dominio para una compañía subsidiaria puede ubicarse debajo el dominio de su compañía principal. Si el nombre de dominio no tiene otra relación, una organización puede colocar su nombre de dominio directamente debajo de uno de los dominios existentes de nivel superior, como `.com`, `.org`, `.edu`, `.gov`, etc.

Uso de subredes

El uso de subredes está relacionado con la necesidad de contar con subdivisiones administrativas para abordar cuestiones de tamaño y control. Cuantos mas hosts y servidores haya en una red, más compleja será la tarea de administración. Al crear divisiones administrativas y utilizar subredes, la gestión de una red compleja resulta más fácil. La decisión de configurar subdivisiones administrativas para su red la determinan los factores siguientes:

- **Tamaño de la red**

  Las subredes también son útiles incluso en una red relativamente pequeña cuyas subdivisiones están ubicadas a lo largo de una amplia área geográfica.

- **Necesidades comunes compartidas por grupos de usuarios**

  Por ejemplo, posiblemente tenga una red que esté limitada a un único edificio y que admita un número relativamente pequeño de máquinas. Estos equipos se reparten en una serie de subredes. Cada subred admite grupos de usuarios con diferentes necesidades. En este ejemplo, puede utilizar una subdivisión administrativa para cada subred.

Planificación de enrutadores en la red

Tenga en cuenta que en el protocolo TCP/IP existen dos tipos de entidades en una red: hosts y enrutadores. Mientras que todas las redes requieren un host, no es necesario que tengan un enrutador. La topología física de la red determina la necesidad de enrutadores. En esta sección se introducen los conceptos de topología de red y enrutamiento. Estos conceptos son importantes cuando decide agregar otra red a su entorno de red.

Nota – Para obtener las tareas y los detalles completos para la configuración de enrutadores en las redes IPv4, consulte "Configuración de los componentes del sistema en la red" en la página 35. Para ver las tareas y detalles completos para la configuración de los enrutadores en las redes IPv6, consulte "Configuración de un enrutador IPv6" en la página 66.

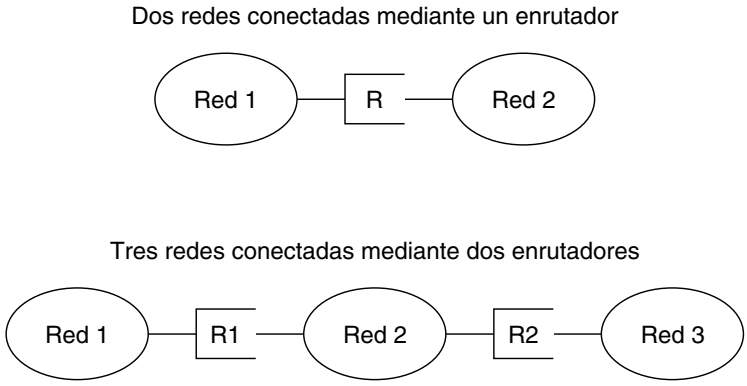
Descripción general de la topología de red

La topología de red describe cómo encajan las redes. Los enrutadores son las entidades que conectan las redes entre sí. Un enrutador es un equipo que tiene dos o más interfaces de red e implementa el reenvío de IP. Sin embargo, el sistema no puede funcionar como enrutador hasta que esté configurado tal como se describe en "Configuración de un enrutador IPv4" en la página 43.

Los enrutadores conectan dos o más redes para formar interredes mayores. Los enrutadores deben configurarse para transferir paquetes entre dos redes adyacentes. Los enrutadores también deben poder transferir paquetes a redes que se encuentran fuera de las redes adyacentes.

La figura siguiente muestra las partes básicas de una topología de red. La primera ilustración muestra una configuración sencilla de dos redes conectadas por un único enrutador. La segunda ilustración muestra una configuración de tres redes, interconectadas por dos enrutadores. En el primer ejemplo, el enrutador R une la red 1 y la red 2 en una interred mayor. En el segundo ejemplo, el enrutador 1 conecta las redes 1 y 2. El enrutador R2 conecta las redes 2 y 3. Las conexiones de una red que incluye las redes 1, 2 y 3.

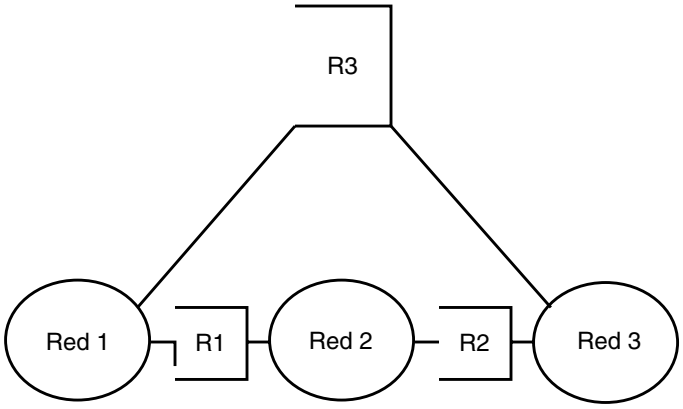
FIGURA 1–1 Topología de red básica



Además de unir las redes en interredes, los enrutadores transfieren los paquetes entre las redes que se basan en las direcciones de la red de destino. A medida que las interredes se hacen más complejas, cada enrutador debe tomar más decisiones sobre los destinos de los paquetes.

La figura siguiente muestra un caso más complejo. El enrutador R3 conecta las redes 1 y 3. La redundancia aumenta la fiabilidad. Si la red 2 no funciona, el enrutador R3 continua proporcionando una ruta entre las redes 1 y 3. Se pueden interconectar muchas redes. No obstante, las redes deben utilizar los mismos protocolos de red.

FIGURA 1–2 Topología de red que proporciona una ruta adicional entre las redes



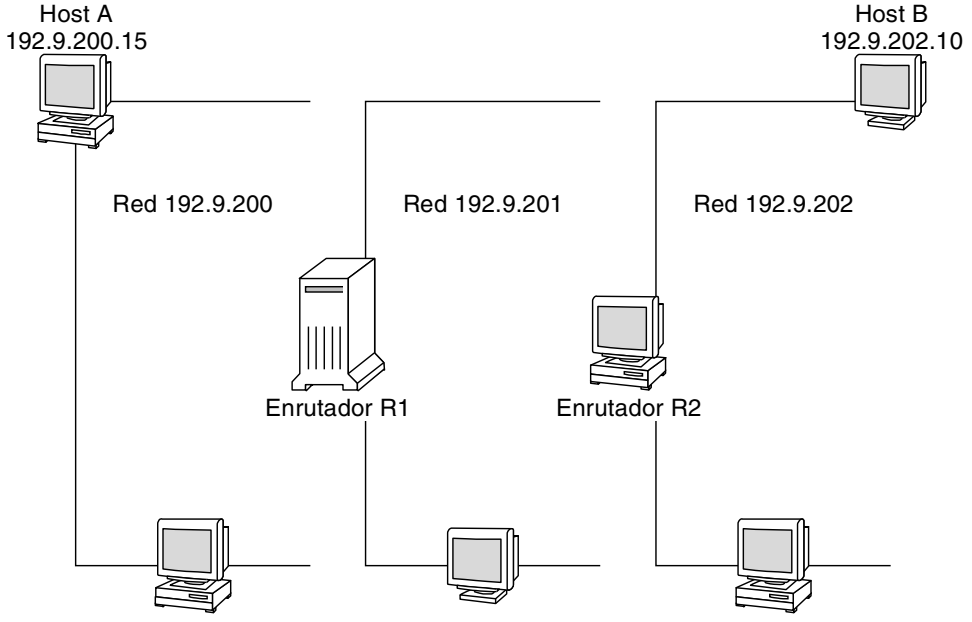
Cómo transfieren los paquetes los enrutadores

La dirección IP del receptor, que forma parte del encabezado del paquete, determina el modo en que se enruta el paquete. Si esta dirección incluye el número de red de la red local, el paquete va directamente al host con esa dirección IP. Si el número de red no es la red local, el paquete va al enrutador de la red local.

Los enrutadores contienen información de enrutamiento en las *tablas de enrutamiento*. Estas tablas contienen la dirección IP de los hosts y enrutadores de las redes a las que está conectado el enrutador. Las tablas también contienen punteros a esas redes. Cuando un enrutador recibe un paquete, comprueba su tabla de enrutamiento para determinar si la tabla incluye la dirección de destino en el encabezado. Si la tabla no contiene la dirección de destino, el enrutador envía el paquete a otro enrutador que aparezca en la tabla de enrutamiento. Si desea más información sobre los enrutadores, consulte "Configuración de un enrutador IPv4" en la página 43.

La figura siguiente muestra una topología de red con tres redes que están conectadas con dos enrutadores.

FIGURA 1–3 Topología de red con tres redes interconectadas



El enrutador R1 conecta las redes `192.9.200` y `192.9.201`. El enrutador R2 conecta las redes `192.9.201` y `192.9.202`.

Si el host A de la red `192.9.200` envía un mensaje al host B de la red `192.9.202`, tienen lugar los siguientes eventos:

1. El host A envía un paquete a través de la red `192.9.200`. El encabezado del paquete contiene la dirección IPv4 del host B receptor, `192.9.202.10`.
2. Ninguno de los equipos de la red `192.9.200` tiene la dirección IPv4 `192.9.202.10`. Por tanto, el enrutador R1 acepta el paquete.
3. El enrutador R1 examina sus tablas de enrutamiento. Ningún equipo de la red `192.9.201` tiene la dirección `192.9.202.10`. Sin embargo, las tablas de enrutamiento incluyen el enrutador R2.
4. A continuación, R1 selecciona R2 como enrutador para el "siguiente salto". R1 envía el paquete a R2.
5. Como R2 conecta la red `192.9.201` en `192.9.202`, R2 tiene la información de enrutamiento para el host B. El enrutador R2 envía el paquete a la red `192.9.202`, donde el host B acepta el paquete.

Implementación de redes virtuales

Esta versión de Oracle Solaris admite la creación de redes virtuales en una única red al configurar zonas y tarjetas de red virtual (VNIC). Las VNIC son interfaces de red que se crean además de las NIC. La combinación de zonas y VNIC es una manera eficaz de consolidar un centro de datos enorme que contiene un gran número de sistemas físicos en menos sistemas. Para obtener más información sobre las redes virtuales, consulte *Uso de redes virtuales en Oracle Solaris 11.1*.

◆ ◆ ◆ CAPÍTULO 2

Consideraciones para el uso de direcciones IPv6

Este capítulo complementa al Capítulo 1, "Planificación de la implementación de red" y describe consideraciones adicionales que deben tenerse en cuenta al decidir utilizar direcciones IPv6 en la red.

Si tiene previsto utilizar direcciones IPv6 además de direcciones IPv4, asegúrese de que el ISP actual admita ambos tipos de direcciones. De lo contrario, deberá encontrar un ISP independiente para admitir direcciones IPv6.

Para obtener una introducción a los conceptos relativos a IPv6, consulte los siguientes recursos; consulte Internet Protocol, Version 6 (IPv6) Specification (`http://www.ietf.org/rfc/rfc2460.txt`).

Planificación de IPv6 (mapa de tareas)

En la tabla siguiente, se enumeran diferentes consideraciones que deben tenerse en cuenta al implementar IPv6 en la red.

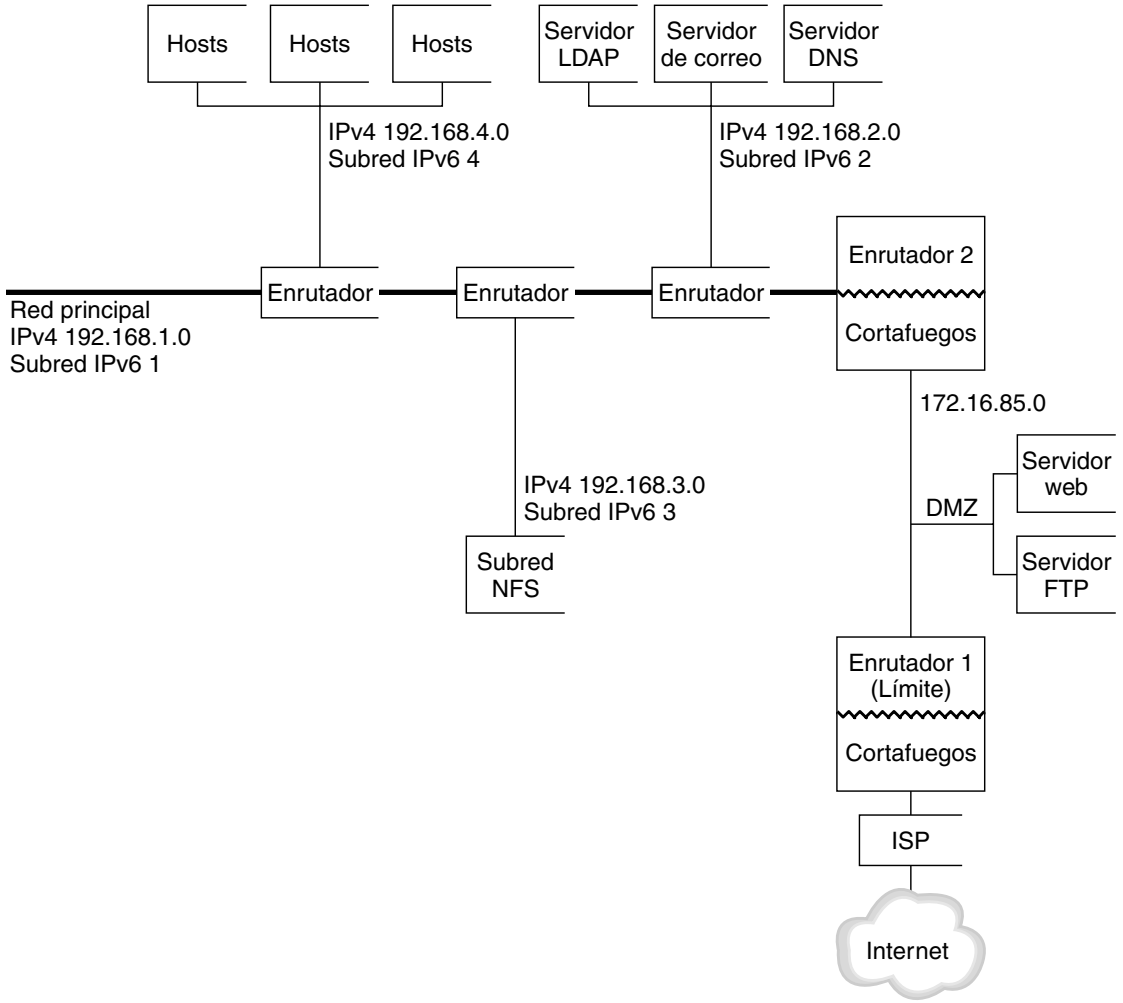
Tarea	Descripción	Para obtener instrucciones
Preparar el hardware para admitir IPv6.	Compruebe que el hardware se pueda actualizar a IPv6.	"Cómo garantizar la compatibilidad de hardware para IPv6" en la página 26
Asegurarse de que las aplicaciones estén preparadas para funcionar con IPv6.	Verifique que las aplicaciones puedan funcionar en un entorno IPv6.	"Configuración de servicios de red para admitir IPv6" en la página 29
Diseñar un plan para el uso de túneles.	Establezca los enrutadores que deben ejecutar túneles a otras subredes o redes externas.	"Planificación para el uso de túneles en la red" en la página 31

Tarea	Descripción	Para obtener instrucciones
Planificar cómo proteger las redes y desarrollar una política de seguridad IPv6.	Por motivos de seguridad, se precisa un plan de direcciones para la DMZ y sus entidades antes de configurar IPv6. Decida cómo implementará la seguridad, por ejemplo, con un filtro IP, una arquitectura de seguridad IP (IPsec), el intercambio de claves de Internet (IKE) y otras funciones de seguridad de esta versión.	"Aspectos relacionados con la seguridad en la implementación de IPv6" en la página 31 *Protección de la red en Oracle Solaris 11.1*
Crear un plan de direcciones para sistemas de la red.	Se debe planificar la dirección de servidores, enrutadores y hosts antes de configurar IPv6. Este paso implica obtener un prefijo de sitio para la red, además de planificar subredes IPv6, si es necesario.	"Creación de un plan de direcciones IPv6 para nodos" en la página 27

Situación hipotética de topología de red IPv6

Por lo general, IPv6 se utiliza en una topología de red mixta que también utiliza IPv4, como se muestra en la figura siguiente. Esta figura se utiliza como referencia en la descripción de las tareas de configuración de IPv6 de las secciones siguientes.

FIGURA 2–1 Situación hipotética de topología de red IPv6



La situación hipotética de red empresarial se compone de cinco subredes con cuatro direcciones IPv4 ya configuradas. Los vínculos de la red se corresponden directamente con las subredes administrativas. Las cuatro redes internas se muestran con direcciones IPv4 privadas en formato RFC 1918, solución habitual ante la falta de direcciones IPv4. Estas redes internas se basan en el siguiente esquema de direcciones:

- La subred 1 es la red principal interna `192.168.1`.
- La subred 2 es la red interna `192.168.2`, con LDAP, `sendmail` y servidores DNS.
- La subred 3 es la red interna `192.168.3`, con los servidores NFS de la empresa.

- La subred 4 es la red interna `192.168.4`, que contiene hosts para los empleados de la empresa.

La red pública externa `172.16.85` funciona como DMZ de la corporación. Esta red contiene servidores web, servidores FTP anónimos y demás recursos que la empresa ofrece al entorno exterior. El enrutador 2 ejecuta un cortafuegos y separa la red pública `172.16.85` de la red principal interna. En el otro extremo de la DMZ, el enrutador 1 ejecuta un cortafuegos y actúa como enrutador de límite de la empresa.

En la Figura 2–1, la DMZ pública presenta la dirección privada RFC 1918 `172.16.85`. En un entorno real, la DMZ pública debe tener registrada una dirección IPv4. La mayoría de los sitios de IPv4 emplean una combinación de direcciones públicas y direcciones privadas RFC 1918. Sin embargo, en el ámbito de IPv6 el concepto de direcciones públicas y privadas es distinto. Debido a que IPv6 dispone de mucho más espacio de direcciones, las direcciones públicas IPv6 se utilizan en redes públicas y privadas.

La pila doble de protocolos de Oracle Solaris permite operaciones simultáneas de IPv4 e IPv6. Puede ejecutar correctamente operaciones relacionadas con IPv4 durante la implementación de IPv6 en la red y después de esta implementación. Al implementar IPv6 en una red operativa que ya utiliza IPv4, asegúrese de no interrumpir las operaciones en curso.

En las secciones siguientes, se describen las áreas que debe tener en cuenta al prepararse para implementar IPv6.

Cómo garantizar la compatibilidad de hardware para IPv6

Consulte la documentación de los fabricantes para conocer la compatibilidad de IPv6 con los siguientes tipos de hardware:

- Enrutadores
- Cortafuegos
- Servidores
- Conmutadores

Nota – Todos los procedimientos de este manual suponen que los equipos, en especial los enrutadores, se pueden actualizar a IPv6.

Algunos modelos de enrutador no se pueden actualizar a IPv6. Para obtener más información y una solución alternativa, consulte "IPv4 Router Cannot Be Upgraded to IPv6" de *Troubleshooting Network Issues*.

Para cada NIC de los servidores IPv6, configure manualmente la parte del ID de interfaz de la dirección IPv6, en lugar de obtener automáticamente el ID con el protocolo de descubrimiento de vecinos. De esta forma, si se reemplaza una NIC, se puede aplicar el mismo ID de interfaz a la

NIC de reemplazo. Es posible que un ID diferente generado automáticamente por el protocolo de descubrimiento de vecinos cause un comportamiento inesperado en el servidor.

Preparación de un plan de direcciones IPv6

Desarrollar un plan de direcciones es importante en la transición de IPv4 a IPv6. Para esta tarea se necesitan los siguientes requisitos previos:

- "Obtención de un prefijo de sitio" en la página 27
- "Creación del esquema de numeración de IPv6" en la página 27

Obtención de un prefijo de sitio

Debe obtenerse un prefijo de sitio antes de configurar IPv6. El prefijo de sitio se emplea en la derivación de direcciones IPv6 para todos los nodos de la implementación de IPv6.

Un ISP que admita IPv6 puede brindar a las empresas prefijos de sitio de IPv6 de 48 bits. Si el ISP sólo admite IPv4, se puede buscar otro que sea compatible con IPv6 y mantener el ISP actual para IPv4. En tal caso, existen las siguientes soluciones alternativas. Para obtener más información, consulte "Current ISP Does Not Support IPv6" de *Troubleshooting Network Issues*.

Si su organización es un ISP, los prefijos de sitio de sus clientes se obtienen del pertinente registro de Internet. Para obtener más información, consulte la página de IANA (Internet Assigned Numbers Authority) (`http://www.iana.org`).

Creación del esquema de numeración de IPv6

A menos que la red IPv6 que se proponga sea totalmente nueva, la topología de IPv4 ya configurada sirve de base para el esquema de numeración de IPv6.

Creación de un plan de direcciones IPv6 para nodos

En la mayoría de los hosts, la configuración automática sin estado de direcciones IPv6 para sus interfaces constituye una estrategia válida y eficaz. Cuando el host recibe el prefijo de sitio del enrutador más próximo, el protocolo ND genera de forma automática direcciones IPv6 para cada interfaz del host.

Los servidores necesitan direcciones IPv6 estables. Si no configura manualmente las direcciones IPv6 de un servidor, siempre que se reemplaza una tarjeta NIC del servidor se configura automáticamente una dirección IPv6. Al crear direcciones para servidores debe tenerse en cuenta lo siguiente:

- Proporcione a los servidores unos ID de interfaz descriptivos y estables. Un método consiste en aplicar un sistema de numeración consecutiva a los ID de interfaz. Por ejemplo, la interfaz interna del servidor LDAP en la Figura 2–1 podría ser `2001:db8:3c4d:2::2`.
- Si habitualmente no cambia la numeración de la red IPv4, es buena idea utilizar como ID de interfaz las direcciones IPv4 ya creadas de los enrutadores y servidores. En la Figura 2–1, suponga que la interfaz del enrutador 1 con la DMZ tiene la dirección IPv4 `123.456.789.111`. La dirección IPv4 puede convertirse a hexadecimal y aplicar el resultado como ID de interfaz. El nuevo ID de interfaz será `::7bc8:156F`.

  Este planteamiento se utiliza sólo si se es el propietario de la dirección IPv4 registrada, en lugar de haber obtenido la dirección de un ISP. Si utiliza una dirección IPv4 proporcionada por un ISP, se crea una dependencia que puede causar problemas en caso de cambiar los ISP.

Debido al número limitado de direcciones IPv4, antes un diseñador de redes debía tener en cuenta si iba a utilizar direcciones registradas globales y direcciones RFC 1918 privadas. No obstante, el concepto de direcciones IPv4 globales y privadas no es aplicable a las direcciones IPv6 . Puede utilizar direcciones unidifusión globales, que incluyen el prefijo de sitio, en todos los vínculos de la red, incluida la DMZ pública.

Creación de un esquema de numeración para subredes

Inicie el esquema de numeración asignando las subredes IPv4 ya configuradas a subredes IPv6 equivalentes. Por ejemplo, fíjese en las subredes de la Figura 2–1. Las subredes 1–4 utilizan la designación de redes privadas IPv4 de RFC 1918 para los primeros 16 bits de sus direcciones, además de los dígitos 1–4 para indicar la subred. A modo de ejemplo, suponga que el prefijo de IPv6 `2001:db8:3c4d/48` se ha asignado al sitio.

La tabla siguiente muestra la asignación de prefijos de IPv4 privados a prefijos de IPv6.

Prefijo de subred IPv4	Prefijo de subred IPv6 equivalente
`192.168.1.0/24`	`2001:db8:3c4d:1::/64`
`192.168.2.0/24`	`2001:db8:3c4d:2::/64`
`192.168.3.0/24`	`2001:db8:3c4d:3::/64`
`192.168.4.0/24`	`2001:db8:3c4d:4::/64`

Configuración de servicios de red para admitir IPv6

Los siguientes servicios de red IPv4 típicos de la versión actual de Oracle Solaris admiten IPv6:

- `sendmail`
- NFS
- HTTP (versiones Apache 2 u Orion)
- DNS
- LDAP

El servicio de correo IMAP sólo es apto para IPv4.

Los nodos configurados para IPv6 pueden ejecutar servicios de IPv4. Al activar IPv6, no todos los servicios aceptan conexiones IPv6. Los servicios conectados a IPv6 aceptarán una conexión. Los servicios que no estén conectados a IPv6 seguirán funcionando con la mitad de IPv4 de la pila de protocolos.

Al actualizar los servicios a IPv6 pueden surgir algunos problemas. Para obtener detalles, consulte "Problems After Upgrading Services to IPv6" de *Troubleshooting Network Issues*.

▼ Cómo preparar servicios de red para admitir IPv6

1. **Actualice los servicios de red siguientes para que admitan IPv6:**
   - Servidores de correo
   - Servidores NIS
   - NFS

   Nota – LDAP admite IPv6 sin tener que realizar tareas de configuración propias de IPv6.

2. **Verifique que el hardware del cortafuegos ya esté preparado para IPv6.**

   Para obtener instrucciones, consulte la documentación pertinente sobre servidores de seguridad.

3. **Verifique que otros servicios de la red se hayan conectado a IPv6.**

   Para obtener más información, consulte la publicidad adicional y la documentación relativa al software.

4. **Si el sitio implementa los servicios siguientes, asegúrese de haber tomado las medidas apropiadas:**
   - Cortafuegos

Para poder admitir IPv6, quizá deba incrementar la severidad de las directrices ya establecidas para IPv4. Para otros aspectos sobre seguridad, consulte "Aspectos relacionados con la seguridad en la implementación de IPv6" en la página 31.

- Correo

  En los registros MX para DNS, quizá deba agregar la dirección IPv6 del servidor de correo.

- DNS

  Para cuestiones específicas de DNS, consulte "Cómo preparar DNS para admitir IPv6" en la página 30.

- IPQoS

  En un host, emplee las mismas directrices DiffServ que se usaban en IPv4. Para obtener más información, consulte "Módulo clasificador" de *Gestión de calidad de servicio IP en Oracle Solaris 11.1*.

5 Audite los servicios de red que ofrezca un nodo antes de convertir a IPv6 dicho nodo.

▼ Cómo preparar DNS para admitir IPv6

La versión actual de Oracle Solaris admite resolución de DNS desde el lado del cliente y del servidor. Efectúe el procedimiento siguiente con el fin de preparar IPv6 para servicios de DNS.

Para obtener más información relativa a la compatibilidad de DNS con IPv6, consulte *Trabajo con servicios de nombres y directorios en Oracle Solaris 11.1*.

1 Compruebe que el servidor DNS que ejecuta la resolución de nombres recursivos esté en una pila doble (IPv4 e IPv6) o sólo en IPv4.

2 En el servidor DNS, rellene la base de datos de DNS con los pertinentes registros AAAA de base de datos de IPv6 en la zona de reenvío.

Nota – Los servidores que ejecutan varios servicios fundamentales necesitan atención especial. Verifique que la red funcione correctamente. Compruebe también que todos los servicios fundamentales tengan conexión con IPv6. A continuación, agregue la dirección IPv6 del servidor a la base de datos de DNS.

---

3 Incorpore los registros PTR relativos a los registros AAAA en la zona inversa.

4 Agregue datos sólo de IPv4, o de IPv6 e IPv4, en el registro NS que describe zonas.

Planificación para el uso de túneles en la red

La implementación de IPv6 permite varias configuraciones de túneles para actuar como mecanismos de transición cuando la red migra a una combinación de IPv4 e IPv6. Los túneles posibilitan la comunicación entre redes IPv6 aisladas. Como en Internet se ejecuta mayoritariamente IPv4, los paquetes de IPv6 del sitio deben desplazarse por Internet a través de túneles hacia las redes IPv6 de destino.

A continuación se presentan varias de las situaciones hipotéticas más destacadas sobre el uso de túneles en la topología de red IPv6:

- El ISP del que adquiere servicios IPv6 permite crear un túnel desde el enrutador de límite del sitio hasta la red del ISP. La Figura 2–1 muestra un túnel de esta clase. En tal caso, se debe ejecutar IPv6 manual a través de un túnel de IPv4.
- Se administra una red distribuida de gran tamaño con conectividad IPv4. Para conectar los sitios distribuidos que utilizan IPv6, puede ejecutar un túnel de 6to4 desde el enrutador de límite de cada subred.
- En ocasiones, un enrutador de la infraestructura no se puede actualizar a IPv6. En tal caso, la alternativa es crear un túnel manual en el enrutador de IPv4 con dos enrutadores de IPv6 como puntos finales.

Para conocer los procedimientos para la configuración de túneles, consulte "Configuración de túneles (mapa de tareas)" en la página 112. Para obtener información conceptual relativa a los túneles, consulte "Descripción general de túneles IP" en la página 103.

Aspectos relacionados con la seguridad en la implementación de IPv6

Al implementar IPv6 en una red ya configurada, debe tener la precaución de no poner en riesgo la seguridad del sitio. Durante la sucesivas fases en la implementación de IPv6, tenga en cuenta los siguientes aspectos relacionados con la seguridad:

- Los paquetes de IPv6 e IPv4 necesitan la misma cantidad de filtrado.
- A menudo, los paquetes de IPv6 pasan por un túnel a través de un cortafuegos. Por lo tanto, debe aplicar cualquiera de las siguientes situaciones hipotéticas:
  - Haga que el cortafuegos inspeccione el contenido en el túnel.
  - Coloque un cortafuegos de IPv6 con reglas parecidas en el punto final del túnel del extremo opuesto.
- Determinados mecanismos de transición utilizan IPv6 en UDP a través de túneles de IPv4. Dichos mecanismos pueden resultar peligrosos al cortocircuitarse el cortafuegos.

- Los nodos de IPv6 son globalmente asequibles desde fuera de la red empresarial. Si la política de seguridad prohíbe el acceso público, debe establecer reglas más estrictas con relación al cortafuegos. Por ejemplo, podría configurar un cortafuegos con estado.

Este manual proporciona funciones de seguridad válidas en una implementación de IPv6.

- La función de IPsec (IP architecture security, arquitectura de seguridad IP) posibilita la protección criptográfica de paquetes IPv6. Para obtener más información, consulte el Capítulo 6, "Arquitectura de seguridad IP (descripción general)" de *Protección de la red en Oracle Solaris 11.1*.
- La función IKE (Internet Key Exchange, intercambio de claves en Internet) permite el uso de autenticación de claves públicas para paquetes de IPv6. Para obtener más información, consulte el Capítulo 9, "Intercambio de claves de Internet (descripción general)" de *Protección de la red en Oracle Solaris 11.1*.

◆ ◆ ◆ CAPÍTULO 3

Configuración de una red IPv4

La configuración de red se compone de dos etapas: ensamblado del hardware y configuración de los daemons, los archivos y los servicios que implementan el protocolo TCP/IP.

En este capítulo, se explica cómo configurar una red que implementa servicios y direcciones IPv4.

Muchas de las tareas de este capítulo se aplican a redes activadas tanto para IPv4 como para IPv6. Las tareas que son específicas para las redes IPv6, se incluyen en el Capítulo 4, "Activación de IPv6 en una red".

Nota – Antes de configurar TCP/IP, revise las distintas tareas de planificación que se enumeran en el Capítulo 1, "Planificación de la implementación de red". Si planea utilizar direcciones IPv6, consulte también el Capítulo 2, "Consideraciones para el uso de direcciones IPv6".

Este capítulo contiene la información siguiente:

Configuración de red (mapa de tareas)

La tabla siguiente muestra las tareas adicionales requeridas después de cambiar de una configuración de red sin subredes a una red que utiliza subredes. La tabla incluye una descripción de lo que hace cada tarea y la sección de la documentación actual en que se detalla el procedimiento correspondiente.

Tarea	Descripción	Para obtener instrucciones
Configurar las interfaces IP del sistema.	Asigna direcciones IP a las interfaces IP del sistema.	"Cómo configurar una interfaz IP" de *Conexión de sistemas mediante la configuración de redes fijas en Oracle Solaris 11.1*
Configurar un sistema para el modo de archivos locales.	Edita archivos de configuración específicos en el directorio `/etc` del sistema y configura el servicio SMF `nis/domain`.	"Cómo configurar un sistema para el modo de archivos locales" en la página 40
Configurar un servidor de configuración de red.	Activa el daemon `in.tftp` y edita otros archivos de configuración en el directorio `/etc` del sistema.	"Cómo instalar un servidor de configuración de red" en la página 42
Configurar un sistema para el modo de cliente de red.	Edita archivos de configuración en el directorio `/etc` del sistema.	"Cómo configurar un sistema para el modo de cliente de red" en la página 41
Especificar una estrategia de enrutamiento para el cliente de red.	Configura sistemas para que utilicen el enrutamiento estático o el enrutamiento dinámico.	"Cómo activar el enrutamiento estático en un host de interfaz única" en la página 52 y "Cómo activar el enrutamiento dinámico en un sistema de interfaz única" en la página 53

Antes de comenzar la configuración de red

En esta versión de Oracle Solaris, la configuración de red de un sistema se gestiona mediante un *perfil de configuración de red (NCP)* activo. La configuración de red del sistema es automática si el NCP activo es reactivo, por ejemplo, el NCP `automatic`. Si el NCP activo es `DefaultFixed`, el modo de configuración de red del sistema es fijo. El sistema con configuración de red reactiva se comporta de manera diferente que con la configuración de red fija.

Cualquier configuración que realice se aplica al NCP activo. Por lo tanto, antes de realizar cualquier procedimiento de configuración, debe saber qué NCP está activo. Como consecuencia, el sistema se comportará como se espera después de completar los procedimientos de configuración. Para determinar qué NCP está activo en un sistema, escriba el siguiente comando:

```
# netadm list
TYPE        PROFILE        STATE
ncp         DefaultFixed   online
ncp         Automatic      disabled
loc         Automatic      offline
loc         NoNet          offline
loc         User           offline
loc         DefaultFixed   online
```

El perfil cuyo estado aparece como en línea (online) es el NCP activo del sistema.

Para obtener información más detallada sobre el NCP en el sistema, utilice la opción `-x` con el comando `netadm`.

```
netadm list -x
TYPE        PROFILE        STATE          AUXILIARY STATE
ncp         DefaultFixed   online         active
ncp         Automatic      disabled       disabled by administrator
loc         Automatic      offline        conditions for activation are unmet
loc         NoNet          offline        conditions for activation are unmet
loc         User           offline        conditions for activation are unmet
loc         DefaultFixed   online         active
```

Para cambiar entre los tipos de perfiles, por ejemplo, de un perfil reactivo a un perfil fijo, escriba el siguiente comando:

```
# netadm enable -p ncp NCP-name
```

donde *NCP-name* es el nombre de un tipo de NCP.

Para obtener una introducción a la configuración de red gestionada por perfiles, consulte "Perfiles de configuración de red" de *Introducción a redes de Oracle Solaris 11*. Para obtener descripciones detalladas de los NCP, consulte *Conexión de sistemas mediante la configuración de redes reactivas en Oracle Solaris 11.1*.

Configuración de los componentes del sistema en la red

Al configurar sistemas de red, necesita la siguiente información de configuración:

- Nombre de host de cada sistema.
- Dirección IP y máscara de red de cada sistema. Si la red está subdividida en subredes, debe contar con los números de subred y el esquema de direcciones IP que se aplicarán a los sistemas en cada subred, incluidas sus respectivas máscaras de red.
- Nombre de dominio al que pertenece cada sistema.
- Dirección del enrutador predeterminado.

  Esta información se facilita en caso de tener una topología de red simple con un único enrutador conectado a cada red. También se facilita esta información si los enrutadores no ejecutan protocolos de enrutamiento como RDISC (Router Discovery Server Protocol) o RIP (Router Information Protocol). Para obtener más información acerca de los enrutadores y para obtener una lista de los protocolos de enrutamiento que admite Oracle Solaris, consulte "Protocolos de enrutamiento en Oracle Solaris" en la página 129.

Nota – Puede configurar la red durante la instalación de Oracle Solaris. Para obtener instrucciones, consulte *Instalación de sistemas Oracle Solaris 11.1*.

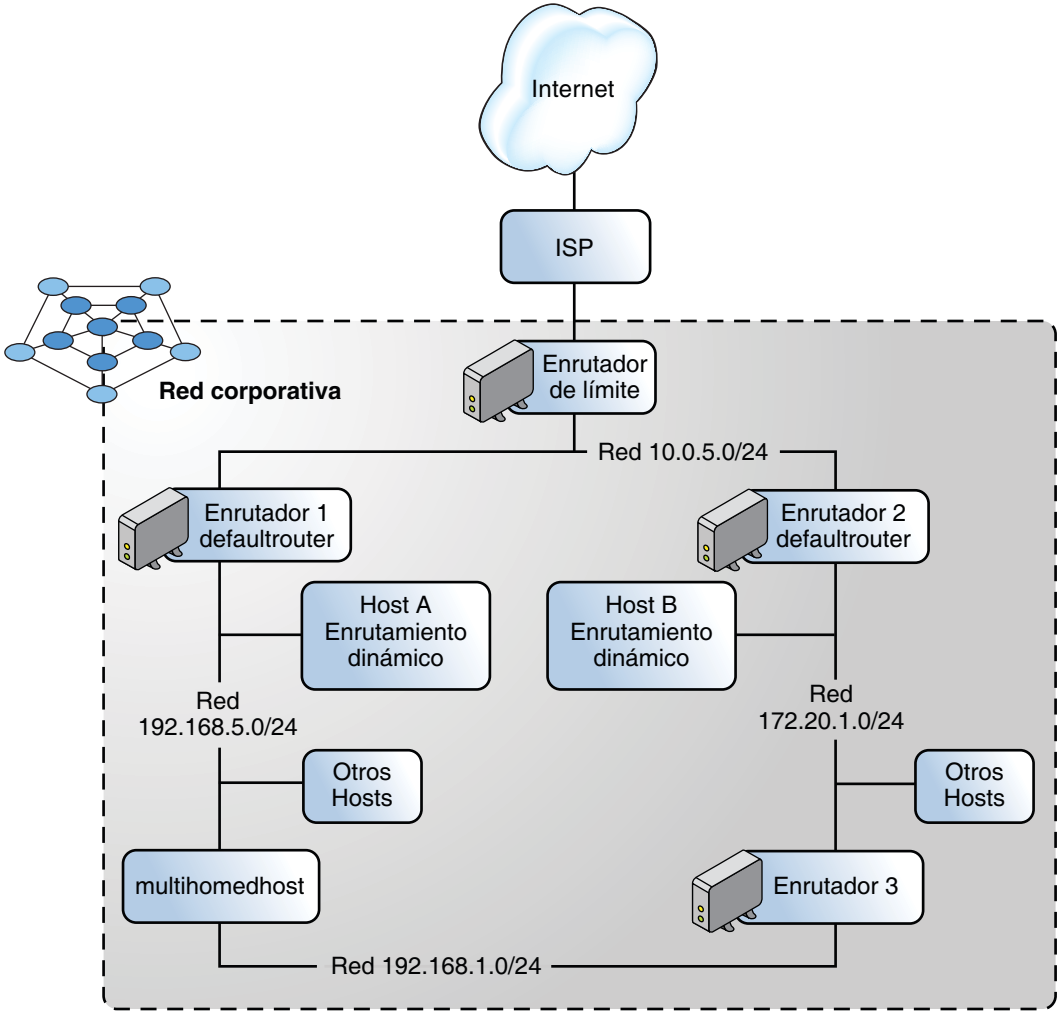
En este documento, los procedimientos suponen que la red se configura después de haber instalado el sistema operativo.

Utilice la Figura 3–1 que se incluye en la siguiente sección como referencia para configurar los componentes del sistema de la red.

Topología de sistemas autónomos IPv4

Los sitios con varios enrutadores y redes normalmente administran su topología de red como dominio de enrutamiento único, o *sistema autónomo (SA)*.

FIGURA 3–1 Sistema autónomo con varios enrutadores IPv4



En la Figura 3–1, se muestra un AS que está dividido en tres redes locales, `10.0.5.0`, `172.16.1.0` y `192.168.5.0`. La red se compone de los siguientes tipos de sistemas:

- Los enrutadores utilizan protocolos de enrutamiento para gestionar la forma en que los paquetes de red se dirigen o se enrutan desde el origen hasta los destinos dentro de la red local o en redes externas. Para obtener información sobre los protocolos de enrutamiento admitidos en Oracle Solaris, consulte "Tablas de protocolos de enrutamiento en Oracle Solaris" en la página 130.

  A continuación, se describen los tipos de enrutadores:

- El *enrutador de límite* conecta la red local, como `10.0.5.0`, externamente a un proveedor de servicios.
- Los *enrutadores predeterminados* gestionan el enrutamiento de paquetes en la red local, que, a su vez, puede incluir varias redes locales. Por ejemplo, en la Figura 3–1, el enrutador 1 actúa como enrutador predeterminado para `192.168.5`. En el mismo momento, el enrutador 1 también está conectado a la red interna `10.0.5.0`. Las interfaces del enrutador 2 se conectan a las redes internas `10.0.5.0` y `172.16.1.0`.
- Los *enrutadores de reenvío de paquetes* reenvían paquetes entre redes internas, pero no ejecutan protocolos de enrutamiento. En la Figura 3–1, el enrutador 3 es un enrutador de reenvío de paquetes con conexiones a las redes `172.16.1` y `192.168.5`.

- Sistemas cliente
  - Sistemas de host múltiple o sistemas que tienen varias NIC. En Oracle Solaris, de manera predeterminada, estos sistemas pueden reenviar paquetes a otros sistemas del mismo segmento de red.
  - Los sistemas de interfaz única confían en los enrutadores locales para reenviar paquetes y para recibir información de configuración.

Configuración de los modos de configuración del sistema

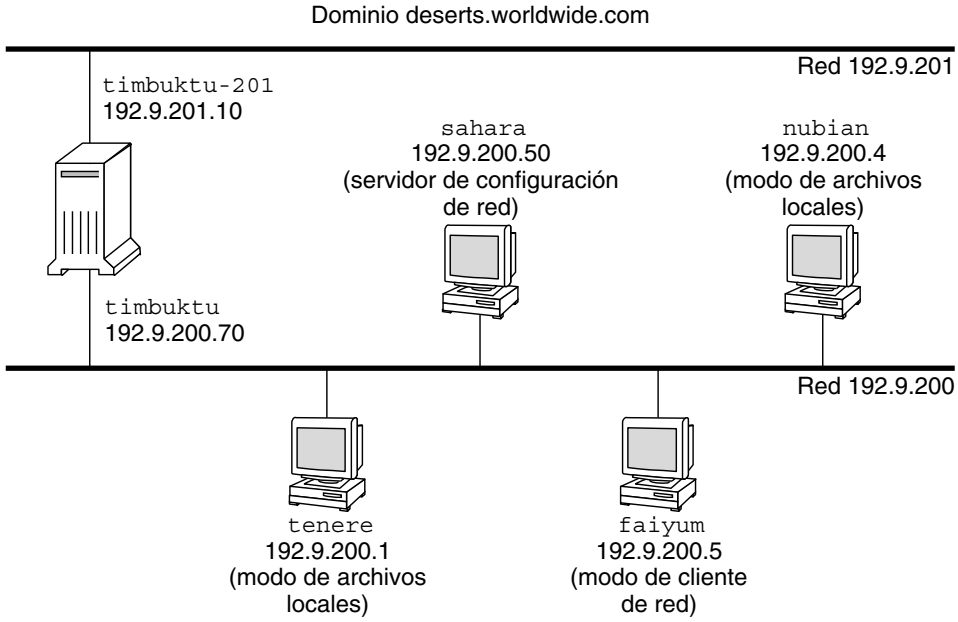
En esta sección, se describen los procedimientos para configurar un sistema para que se ejecute en *modo de archivos locales* o en *modo de cliente de red*. Al ejecutar el sistema en modo de archivos locales, el sistema obtiene toda la información de configuración TCP/IP de los archivos que se encuentran en el directorio local. En el modo de cliente de red, la información de configuración se proporciona para todos los sistemas de la red mediante un servidor de configuración de red remota.

Por lo general, los servidores de la red se ejecutan en modo de archivos locales, como los siguientes:

- Servidores de configuración de red
- Servidores NFS
- Servidores de nombres que proporcionan servicios NIS, LDAP o DNS
- Servidores de correo
- Enrutadores

Los clientes se pueden ejecutar en cualquiera de los dos modos. Por lo tanto, en la red puede existir una combinación de estos modos con los cuales se configuran distintos sistemas, como se muestran en la figura siguiente.

FIGURA 3–2 Sistemas en un escenario de topología de red IPv4



La Figura 3–2 muestra los sistemas en una red `192.9.200`.

- Todos los sistemas pertenecen al dominio organizativo `deserts.worldwide.com`.
- `sahara` es un servidor de configuración. Como servidor, se ejecuta en modo de archivos locales, donde la información de configuración TCP/IP se obtiene del disco local del sistema.

Nota – Si los clientes se configuran para ejecutarse en modo de cliente de red, se debe configurar al menos un servidor de configuración de red que proporcionará la información de configuración a esos clientes.

- `tenere`, `nubian` y `faiyum` son clientes en la red. `tenere` y `nubian` se ejecutan en modo de archivos locales. Independientemente del disco local de `faiyum`, el sistema se configura para funcionar en modo de cliente de red.
- `timbuktu` está configurado como enrutador y, por lo tanto, funciona en modo de archivos locales. El sistema incluye dos NIC, cada una con sus propias interfaces IP configuradas. La primera interfaz IP se denomina `timbuktu` y se conecta a la red `192.9.200`. La segunda interfaz IP se denomina `timbuktu-201` y se conecta a la red `192.9.201`.

▼ Cómo configurar un sistema para el modo de archivos locales

Use este procedimiento para configurar cualquier sistema para que se ejecute en modo de archivos locales.

1. **Configure las interfaces IP del sistema con las direcciones IP asignadas.**

   Consulte "Cómo configurar una interfaz IP" de *Conexión de sistemas mediante la configuración de redes fijas en Oracle Solaris 11.1* para conocer el procedimiento.

2. **Compruebe que se haya configurado el nombre de host correcto en el archivo `/etc/nodename`.**

3. **Compruebe que las entradas del archivo `/etc/inet/hosts` sean actuales.**

   El programa de instalación de Oracle Solaris crea entradas para la interfaz de red principal, la dirección en bucle y, si es preciso, cualquier interfaz adicional configurada durante la instalación.

   Este archivo también debe incluir el nombre del enrutador predeterminado y la dirección IP del enrutador.

   a. **(Opcional) Agregue las direcciones IP y los nombres correspondientes para las interfaces de red que se hayan agregado al sistema tras la instalación.**

   b. **(Opcional) Si el sistema de archivos `/usr` está montado en NFS, agregue la dirección o las direcciones IP del servidor de archivos.**

4. **Especifique el dominio completo del sistema como una propiedad del servicio SMF `nis/domain`.**

   Por ejemplo, especifique `deserts.worldwide.com` como el valor para la propiedad `domainname` del servicio SMF `nis/domain` de la siguiente manera:

   ```
   # domainname domainname
   ```

   Este paso produce un cambio persistente.

5. **Escriba el nombre de enrutador en el archivo `/etc/defaultrouter`.**

6. **Agregue la información de la máscara de red, si corresponde.**

   Nota – Si está usando servicios DHCP, omita este paso.

   a. **Escriba el número de red y la máscara de red en el archivo `/etc/inet/netmasks`.**

      Para crear entradas, utilice el formato *número_red, máscara_red*. Por ejemplo, para el número de red de clase C `192.168.83`, escribiría:

      ```
      192.168.83.0    255.255.255.0
      ```

Para las direcciones CIDR, convierta el prefijo de red en la representación decimal con punto equivalente. Los prefijos de red y sus equivalentes decimales con punto se incluyen en la Tabla 1–1. Por ejemplo, utilice lo siguiente para expresar el prefijo de red CIDR `192.168.3.0/22`.

```
192.168.3.0     255.255.252.0
```

b. Cambie el orden de consulta para las máscaras de red en la propiedad SMF del conmutador de modo que primero se busque en los archivos locales y, luego, refresque la instancia.

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

7 Reinicie el sistema.

▼ Cómo configurar un sistema para el modo de cliente de red

Realice el procedimiento siguiente en cada host que desee configurar en modo de cliente de red.

Antes de empezar Los clientes de red reciben la información de configuración de los servidores de configuración de red. Por lo tanto, antes de configurar un sistema como un cliente de red, debe asegurarse de que haya como mínimo un servidor de configuración de red para la red.

1 Conviértase en administrador.

Para obtener más información, consulte "Cómo usar los derechos administrativos que tiene asignados" de *Administración de Oracle Solaris 11.1: servicios de seguridad*.

2 Configure las interfaces IP del sistema con las direcciones IP asignadas.

Consulte "Cómo configurar una interfaz IP" de *Conexión de sistemas mediante la configuración de redes fijas en Oracle Solaris 11.1* para conocer el procedimiento.

3 Asegúrese de que el archivo `/etc/inet/hosts` contenga únicamente el nombre y la dirección IP de `localhost` de la interfaz de red en bucle de retorno.

```
# cat /etc/inet/hosts
# Internet host table
#
127.0.0.1       localhost
```

4 Elimine los valores asignados a la propiedad `domainname` del servicio SMF `nis/domain`.

```
# domainname ''
```

Este paso produce un cambio persistente.

5 Asegúrese de que las rutas de búsqueda en el servicio `name-service/switch` del cliente reflejen los mismos requisitos de servicio para su red.

▼ Cómo instalar un servidor de configuración de red

Puede encontrar información sobre cómo configurar servidores de instalación y servidores de inicio en *Instalación de sistemas Oracle Solaris 11.1*.

1 Conviértase en administrador.

Para obtener más información, consulte "Cómo usar los derechos administrativos que tiene asignados" de *Administración de Oracle Solaris 11.1: servicios de seguridad*.

2 Active el daemon `in.tftpd` de la siguiente manera:

a. Navegue hasta el directorio raíz (`/`) del servidor de configuración de red designado.

b. Cree el directorio `/tftpboot`:

```
# mkdir /tftpboot
```

Este comando configura el sistema como servidor TFTP, bootparams y RARP.

c. Cree un vínculo simbólico al directorio.

```
# ln -s /tftpboot/. /tftpboot/tftpboot
```

3 Agregue la línea `tftp` en el archivo `/etc/inetd.conf`.

La línea debe decir lo siguiente:

```
tftp dgram udp6 wait root /usr/sbin/in.tftpd in.tftpd -s /tftpboot
```

Esta línea impide que `in.tftpd` recupere archivos que no sean los que se encuentran en `/tftpboot`.

4 En la base de datos `/etc/hosts`, agregue los nombres de host y las direcciones IP de todos los clientes de la red.

5 En la base de datos `/etc/ethers`, cree entradas para cada sistema de la red que se ejecuta en modo de cliente de red.

Las entradas en esta base de datos tienen el formato siguiente:

MAC Address *host name* *#comment*

Para obtener más información, consulte la página del comando man `ethers(4)`.

6 En la base de datos `/etc/bootparams`, cree una entrada para cada sistema de la red que se ejecuta en modo de cliente de red.

Para obtener información sobre cómo editar esta base de datos, consulte la página del comando man `bootparams(4)`.

7 **Convierta la entrada `/etc/inetd.conf` en un manifiesto de servicios de la utilidad de gestión de servicios (SMF) y active el servicio resultante.**

```
# /usr/sbin/inetconv
```

8 **Compruebe que `in.tftpd` funcione correctamente.**

```
# svcs network/tftp/udp6
```

Obtendrá un resultado similar al siguiente:

```
STATE          STIME    FMRI
online         18:22:21 svc:/network/tftp/udp6:default
```

Más información

Administración del daemon `in.tftpd`

La utilidad de gestión de servicios administra el daemon `in.tftpd`. Las acciones administrativas de `in.tftpd`, como la activación, la desactivación o la solicitud de reinicio, pueden llevarse a cabo utilizando el comando `svcadm`. La responsabilidad de iniciar y reiniciar este servicio se delega al comando `inetd` . Utilice el comando `inetadm` para realizar cambios de configuración y ver la información de configuración para `in.tftpd`. Puede consultar el estado del servicio con el comando `svcs`. Para obtener una descripción general de la utilidad de gestión de servicios, consulte el Capítulo 1, "Gestión de servicios (descripción general)" de *Gestión de servicios y errores en Oracle Solaris 11.1*.

Configuración de un enrutador IPv4

Un enrutador proporciona la interfaz entre dos o más redes. Por lo tanto, debe asignar un nombre y una dirección IP exclusivos a cada interfaz de red física del enrutador. Por tanto, cada enrutador tiene un nombre de host y una dirección IP asociados con su interfaz de red principal, además de otro nombre exclusivo y dirección IP, como mínimo, para cada interfaz de red adicional.

También puede utilizar el siguiente procedimiento para configurar un sistema sólo con una interfaz física (de modo predeterminado, un host) como enrutador. Puede configurar un sistema con una sola interfaz como enrutador si el sistema actúa como punto final en un enlace PPP, tal como se describe en "Planificación de un enlace de PPP por marcación telefónica" de *Gestión de redes seriales con UUCP y PPP en Oracle Solaris 11.1*.

▼ Configuración de un enrutador IPv4

Las instrucciones siguientes presuponen que está configurando interfaces para el enrutador tras la instalación.

Antes de empezar Después de que el enrutador se haya instalado físicamente en la red, configure el enrutador para que funcione en el modo de archivos locales, como se describe en "Cómo configurar un sistema para el modo de archivos locales" en la página 40. Con esta configuración, los enrutadores se reiniciarán si el servidor de configuración de red no funciona.

1. **Conviértase en administrador.**

   Para obtener más información, consulte "Cómo usar los derechos administrativos que tiene asignados" de *Administración de Oracle Solaris 11.1: servicios de seguridad*.

2. **Configure las interfaces IP en las tarjetas NIC en el sistema.**

   Para obtener pasos detallados para configurar interfaces IP, consulte "Cómo configurar una interfaz IP" de *Conexión de sistemas mediante la configuración de redes fijas en Oracle Solaris 11.1*.

   Asegúrese de que cada interfaz IP esté configurada con la dirección IP de la red para la cual el sistema enrutará los paquetes. De esta manera, si el sistema presta servicio a las redes `192.168.5.0` y `10.0.5.0`, se debe configurar una NIC para cada red.

Precaución – Si desea configurar enrutadores IPv4 para que utilicen DHCP, debe tener amplios conocimientos sobre la administración DHCP.

3. **Agregue el nombre de host y la dirección IP de cada interfaz al archivo `/etc/inet/hosts`.**

   Por ejemplo, suponga que los nombres que asignó a las dos interfaces del enrutador 1 son `krakatoa` y `krakatoa-1`, respectivamente. Las entradas del archivo `/etc/inet/hosts` serían las siguientes:

   ```
   192.168.5.1      krakatoa        #interface for network 192.168.5.0
   10.0.5.1         krakatoa-1      #interface for network 10.0.5.0
   ```

4. **Siga el resto de los pasos para configurar este enrutador para que se ejecute en modo de archivos locales.**

   Consulte "Cómo configurar un sistema para el modo de archivos locales" en la página 40.

5. **Si el enrutador está conectado a cualquier red con subredes, agregue el número de red y la máscara de red al archivo `/etc/inet/netmasks`.**

   Por ejemplo, para la notación de direcciones IPv4 tradicional, como `192.168.5.0`, debe escribir:

   ```
   192.168.5.0    255.255.255.0
   ```

6. **Active el reenvío de paquetes IPv4 en el enrutador.**

   ```
   # ipadm set-prop -p forwarding=on ipv4
   ```

7. **(Opcional) Inicie un protocolo de enrutamiento.**

   Utilice una de las siguientes sintaxis del comando:

- `# routeadm -e ipv4-routing -u`
- `# svcadm enable route:default`

  El FMRI SMF asociado con el daemon `in.routed` es `svc:/network/routing/route`.

Cuando inicia un protocolo de enrutamiento, el daemon de enrutamiento `/usr/sbin/in.routed` actualiza automáticamente la tabla de enrutamiento. Este proceso se conoce como *enrutamiento dinámico*. Para obtener más información sobre los tipos de enrutamiento, consulte "Tablas y tipos de enrutamiento" en la página 46. Para obtener información sobre el comando `routeadm`, consulte la página del comando man `routeadm(1M)`.

Ejemplo 3–1 Configuración del enrutador predeterminado para una red

Este ejemplo se basa en la Figura 3–1. El enrutador 2 contiene dos conexiones de red cableadas, una conexión a la red `172.16.1.0` y otra a la red `10.0.5.0`. El ejemplo muestra cómo configurar el enrutador 2 para que sea el enrutador predeterminado de la red `172.16.1.0`. El ejemplo también supone que el enrutador 2 se configuró para funcionar en modo de archivos locales, como se describe en "Cómo configurar un sistema para el modo de archivos locales" en la página 40.

Una vez se haya convertido en superusuario o haya asumido un rol equivalente, debe determinar el estado de las interfaces del sistema.

```
# dladm show-link
LINK     CLASS     MTU     STATE   BRIDGE   OVER
net0     phys      1500    up      --       --
net1     phys      1500    up      --       --
net2     phys      1500    up      --       --
# ipadm show-addr
ADDROBJ           TYPE     STATE        ADDR
lo0/v4            static   ok           127.0.0.1/8
net0/v4           static   ok           172.16.1.10/24
```

Únicamente `net0` se configuró con una dirección IP. Para convertir el enrutador 2 en el enrutador predeterminado, debe conectar físicamente la interfaz `net1` a la red `10.0.5.0`.

```
# ipadm create-ip net1
# ipadm create-addr -a 10.0.5.10/24 net1
# ipadm show-addr
ADDROBJ           TYPE     STATE        ADDR
lo0/v4            static   ok           127.0.0.1/8
net0/v4           static   ok           172.16.1.10/24
net1/v4           static   ok           10.0.5.10/24
```

A continuación, deberá actualizar las siguientes bases de datos de red con información sobre la interfaz recientemente configurada y la red a la que está conectada:

```
# vi /etc/inet/hosts
127.0.0.1       localhost
172.16.1.10        router2        #interface for network 172.16.1
```

```
10.0.5.10          router2-out    #interface for network 10.0.5
# vi /etc/inet/netmasks
172.16.1.0    255.255.255.0
10.0.5.0      255.255.255.0
```

Por último, active el reenvío de paquetes y el daemon de enrutamiento `in.routed`.

```
# ipadm set-prop -p forwarding=on ipv4
# svcadm enable route:default
```

Ahora el reenvío de paquetes IPv4 y el enrutamiento dinámico mediante RIP están activados en el enrutador 2. Sin embargo, la configuración predeterminada del enrutador para la red `172.16.1.0` aún no está completa. Debe hacer lo siguiente:

- Modifique cada host de la red `172.16.1.0` de modo que obtenga la información de enrutamiento del nuevo enrutador predeterminado. Para más información, consulte "Cómo activar el enrutamiento estático en un host de interfaz única" en la página 52.
- Defina una ruta estática para el enrutador de límite en la tabla de enrutamiento del enrutador 2. Para obtener más información, consulte "Tablas y tipos de enrutamiento" en la página 46.

Tablas y tipos de enrutamiento

Tanto los enrutadores como los hosts mantienen una *tabla de enrutamiento*. La tabla de enrutamiento enumera las direcciones IP de las redes que conoce el sistema, incluida la red local predeterminada del sistema. La tabla también enumera la dirección IP de un sistema de portal para cada red conocida. La *puerta de enlace* es un sistema que puede recibir paquetes salientes y reenviarlos un salto más allá de la red local.

La siguiente es una tabla de enrutamiento simple para un sistema en una red de sólo IPv4:

```
Routing Table: IPv4
  Destination           Gateway           Flags  Ref   Use   Interface
-------------------- -------------------- ----- ----- ------ ---------
default              172.16.1.10          UG        1    532   net0
224.0.0.0            10.0.5.100           U         1      0   net1
10.0.0.0             10.0.5.100           U         1      0   net1
127.0.0.1            127.0.0.1            UH        1     57   lo0
```

En un sistema Oracle Solaris, puede configurar dos tipos de enrutamiento: estático y dinámico. Puede configurar uno o ambos tipos de enrutamiento en un único sistema. Un sistema que implementa el *enrutamiento dinámico* se basa en protocolos de enrutamiento, como RIP para redes IPv4 y RIPng para redes IPv6, para enrutar el tráfico de red y actualizar información enrutamiento en la tabla. Con el *enrutamiento estático*, la información de enrutamiento se mantiene de forma manual con el comando `route`. Para obtener más información al respecto, consulte la página del comando man `route(1M)`.

Al configurar el enrutamiento para la red local o el sistema autónomo, considere el tipo de enrutamiento que desea para los hosts y enrutadores específicos.

La tabla siguiente muestra los diversos tipos de enrutamiento y las redes para las que es adecuado cada tipo.

Tipo de enrutamiento	Recomendado para
Estático	Hosts y redes de tamaño reducido que obtienen las rutas de un enrutador predeterminado, y enrutadores predeterminados que sólo necesitan conocer uno o dos enrutadores en los siguientes saltos.
Dinámico	Interredes de mayor tamaño, enrutadores en redes locales con múltiples hosts y hosts de sistemas autónomos de gran tamaño. El enrutamiento dinámico es la mejor opción para los sistemas en la mayoría de las redes.
Estático y dinámico combinados	Enrutadores que conectan una red con enrutamiento estático y una red con enrutamiento dinámico, y enrutadores de límite que conectan un sistema autónomo interior con redes externas. La combinación del enrutamiento estático y dinámico en un sistema es una práctica habitual.

El SA que se muestra en la Figura 3–1 combina el enrutamiento estático y el dinámico.

Nota – Dos rutas al mismo destino no hacen que el sistema ejecute automáticamente la función de equilibrio de carga o conmutación por error. Si necesita estas funciones, utilice IPMP, tal como se describe en el Capítulo 5, "Introducción a IPMP" de *Gestión del rendimiento de red de Oracle Solaris 11.1*.

▼ Cómo agregar una ruta estática a la tabla de enrutamiento

1 Visualice el estado actual de la tabla de enrutamiento.

Utilice su cuenta de usuario habitual para ejecutar la forma siguiente del comando `netstat`:

```
% netstat -rn
```

Obtendrá un resultado similar al siguiente:

```
Routing Table: IPv4
  Destination           Gateway           Flags  Ref   Use   Interface
-------------------- -------------------- ----- ----- ------ ---------
192.168.5.125        192.168.5.10          U      1   5879   net0
224.0.0.0            198.168.5.10          U      1  0       net0
default              192.168.5.10          UG     1  91908
127.0.0.1            127.0.0.1             UH     1  811302   lo0
```

2 Conviértase en administrador.

Para obtener más información, consulte "Cómo usar los derechos administrativos que tiene asignados" de *Administración de Oracle Solaris 11.1: servicios de seguridad*.

3 (Opcional) Vacíe las entradas existentes en la tabla de enrutamiento.

```
# route flush
```

4 Agregue una ruta que persista tras el reinicio del sistema.

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

`-p`	Crea una ruta que debe persistir tras el reinicio del sistema. Si desea que la ruta sea válida sólo para la sesión actual, no utilice la opción `-p`.
`-net` *dirección_red*	Especifica que la ruta se dirige a la red con la dirección de *dirección_red*.
`-gateway` *dirección_portal*	Indica que el sistema de portal para la ruta especificada tiene la dirección IP *dirección_portal*.

Ejemplo 3–2 Cómo agregar una ruta estática a la tabla de enrutamiento

En el siguiente ejemplo, se muestra cómo agregar una ruta estática al enrutador 2 de la Figura 3–1. La ruta estática es necesaria para el enrutador de límite del SA: `10.0.5.150`.

Para ver la tabla de enrutamiento del enrutador 2, debe configurar lo siguiente:

```
# netstat -rn
Routing Table: IPv4
  Destination           Gateway           Flags  Ref   Use   Interface
-------------------- -------------------- ----- ----- ------ ---------
default              172.16.1.10          UG        1    249 ce0
224.0.0.0            172.16.1.10          U         1      0 ce0
10.0.5.0             10.0.5.20            U         1     78 bge0
127.0.0.1            127.0.0.1            UH        1     57 lo0
```

La tabla de enrutamiento indica las dos rutas que conoce el enrutador 2. El enrutador predeterminado utiliza la interfaz `172.16.1.10` del enrutador 2 como puerta de enlace. El segundo enrutador, `10.0.5.0`, fue detectado por el daemon `in.routed` que se ejecuta en el enrutador 2. El portal de esta ruta es el enrutador 1, con la dirección IP `10.0.5.20`.

Para agregar una segunda ruta a la red `10.0.5.0`, que tiene su portal como enrutador de límite, debe configurar lo siguiente:

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

Ahora la tabla de enrutamiento cuenta con una ruta para el enrutador de límite, que tiene la dirección IP `10.0.5.150/24`.

```
# netstat -rn
Routing Table: IPv4
  Destination           Gateway           Flags  Ref   Use   Interface
```

```
-------------------- -------------------- ----- ----- ------ ---------
default              172.16.1.10          UG        1    249 ce0
224.0.0.0            172.16.1.10          U         1      0 ce0
10.0.5.0             10.0.5.20            U         1     78 bge0
10.0.5.0             10.0.5.150           U         1    375 bge0
127.0.0.1            127.0.0.1            UH        1     57 lo0
```

Configuración de hosts múltiples

En Oracle Solaris, un sistema con más de una interfaz se considera un *host múltiple*. Las interfaces de un host múltiple se conectan a distintas subredes, ya sea en redes físicas diferentes o en la misma red física.

En un sistema cuyas múltiples interfaces se conectan a la misma subred, es necesario configurar primero las interfaces en un grupo IPMP. De lo contrario, el sistema no puede ser un host múltiple. Para obtener más información sobre IPMP, consulte el Capítulo 5, "Introducción a IPMP" de *Gestión del rendimiento de red de Oracle Solaris 11.1*.

Un host múltiple no reenvía paquetes IP, pero se puede configurar para ejecutar protocolos de enrutamiento. Normalmente se configuran los siguientes tipos de sistemas como hosts múltiples:

- Los servidores NFS, especialmente los que funcionan como grandes centros de datos, se pueden conectar a más de una red para que una agrupación de usuarios de gran tamaño pueda compartir archivos. No es necesario que estos servidores mantengan tablas de enrutamiento.
- Los servidores de bases de datos pueden tener varias interfaces de red para proporcionar recursos a una agrupación de usuarios de gran tamaño, como los servidores NFS.
- Los portales de cortafuegos son sistemas que proporcionan conexión entre la red de una compañía y las redes públicas como Internet. Los administradores configuran los cortafuegos como una medida de seguridad. Cuando se configura el host como un cortafuegos, no transfiere paquetes entre las redes conectadas a las interfaces del host. Sin embargo, el host puede seguir ofreciendo los servicios TCP/IP estándar, como `ssh`, a los usuarios autorizados.

Nota – Cuando los hosts múltiples tienen distintos tipos de cortafuegos en cualquiera de sus interfaces, procure evitar la interrupción involuntaria de los paquetes del host. Este problema sucede especialmente con los cortafuegos con estado. Una solución podría ser configurar los cortafuegos sin estado. Para obtener más información sobre cortafuegos, consulte "Sistemas de cortafuegos" de *Administración de Oracle Solaris 11.1: servicios de seguridad* o la documentación de su cortafuegos de otro proveedor.

▼ Cómo crear un host múltiple

1 **Conviértase en administrador.**

Para obtener más información, consulte "Cómo usar los derechos administrativos que tiene asignados" de *Administración de Oracle Solaris 11.1: servicios de seguridad*.

2 **Configure cada interfaz de red adicional que no haya sido configurada como parte de la instalación de Oracle Solaris.**

Consulte "Cómo configurar una interfaz IP" de *Conexión de sistemas mediante la configuración de redes fijas en Oracle Solaris 11.1*.

3 **Si el reenvío de paquetes está activado, desactive este servicio.**

```
# ipadm show-prop -p forwarding ipv4
PROTO PROPERTY     PERM CURRENT      PERSISTENT   DEFAULT      POSSIBLE
ipv4  forwarding   rw   on           --           off          on,off

ipadm set-prop -p forwarding=off ipv4
```

4 **(Opcional) Active el enrutamiento dinámico para el host múltiple.**

Utilice una de las siguientes sintaxis del comando:

- # **routeadm -e ipv4-routing -u**
- # **svcadm enable route:default**

  El FMRI SMF asociado con el daemon `in.routed` es `svc:/network/routing/route`.

Ejemplo 3–3 Configuración de un host múltiple

En el siguiente ejemplo, se muestra cómo configurar el host múltiple que aparece en la Figura 3–1. En el ejemplo, el sistema tiene el nombre de host `hostc`. Este host cuenta con dos interfaces, que están conectadas a la red `192.168.5.0`.

Para empezar, debe mostrar el estado de las interfaces del sistema.

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

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

El comando `dladm show-link` informa que `hostc` tiene dos enlaces de datos. Sin embargo, únicamente `net0` se configuró con una dirección IP. Para configurar `hostc` como host múltiple,

debe configurar net1 con una dirección IP en la misma red 192.168.5.0. Asegúrese de que la NIC física subyacente de net1 esté conectada físicamente a la red.

```
# ipadm create-ip net1
# ipadm create-addr static -a 192.168.5.85/24 net1
# ipadm show-addr
ADDROBJ        TYPE     STATE        ADDR
lo0/v4         static   ok           127.0.0.1/8
net0/v4        static   ok           192.168.5.82/24
net1/v4        static   ok           192.168.5.85/24
```

A continuación, debe agregar la interfaz net1 a la base de datos /etc/hosts:

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

Luego, debe desactivar el reenvío de paquetes si este servicio se está ejecutando en hostc:

```
# ipadm show-prop -p forwarding ipv4
PROTO PROPERTY     PERM CURRENT   PERSISTENT   DEFAULT      POSSIBLE
ipv4  forwarding   rw   on        --           off          on,off

# ipadm set-prop -p forwarding=off ipv4

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

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

El comando routeadm informa que el enrutamiento dinámico a través del daemon in.routed está actualmente desactivado.

Configuración del enrutamiento para sistemas de interfaz única

Los sistemas de interfaz única se pueden configurar con enrutamiento estático o enrutamiento dinámico. Con el enrutamiento estático, el host debe confiar en los servicios de un enrutador predeterminado para obtener información de enrutamiento. Los procedimientos siguientes contienen las instrucciones para activar ambos tipos de enrutamiento.

▼ Cómo activar el enrutamiento estático en un host de interfaz única

También puede utilizar el procedimiento siguiente para configurar enrutamiento estático en un host múltiple.

1. **Conviértase en administrador.**

   Para obtener más información, consulte "Cómo usar los derechos administrativos que tiene asignados" de *Administración de Oracle Solaris 11.1: servicios de seguridad*.

2. **Configure las interfaces IP del sistema con una dirección IP para la red a la que pertenece el sistema.**

   Para obtener instrucciones, consulte "Cómo configurar una interfaz IP" de *Conexión de sistemas mediante la configuración de redes fijas en Oracle Solaris 11.1*.

3. **Con un editor de textos, cree o modifique el archivo `/etc/defaultrouter` agregando la dirección IP del enrutador que utilizará el sistema.**

4. **Agregue una entrada para el enrutador predeterminado en el archivo `/etc/inet/hosts` local.**

5. **Asegúrese de que el enrutamiento esté desactivado.**

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

          Routing services   "route:default ripng:default"

# svcadm disable route:default
```

6. **Asegúrese de que el reenvío de paquetes esté desactivado.**

```
# # ipadm show-prop -p forwarding ipv4
PROTO PROPERTY     PERM CURRENT   PERSISTENT   DEFAULT      POSSIBLE
ipv4  forwarding   rw   on        --           off          on,off

# ipadm set-prop -p forwarding=off ipv4
```

Ejemplo 3–4 Configuración del enrutamiento estático en un sistema de interfaz única

En el siguiente ejemplo, se muestra cómo configurar el enrutamiento estático para `hostb`, un sistema de interfaz única en la red `172.16.1.0`, como se muestra en la Figura 3–1. `hostb` necesita utilizar el enrutador 2 como el enrutador predeterminado. El ejemplo supone que ya se configuró la interfaz IP del sistema.

Primero, debe iniciar sesión en `hostb` con derechos de administrador. A continuación, debe determinar si el archivo `/etc/defaultrouter` está presente en el sistema:

```
# cd /etc
# ls | grep defaultrouter

# vi /etc/defaultrouter
172.16.1.10
```

La dirección IP `172.16.1.10` pertenece al enrutador 2.

```
# vi /etc/inet/hosts
127.0.0.1          localhost
172.16.1.18        host2    #primary network interface for host2
172.16.1.10        router2  #default router for host2

# ipadm show-prop -p forwarding ipv4
PROTO PROPERTY     PERM CURRENT   PERSISTENT   DEFAULT      POSSIBLE
ipv4  forwarding   rw   on          --           off          on,off

# ipadm set-prop -p forwarding=off ipv4

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

          Routing services   "route:default ripng:default"

# svcadm disable route:default
```

Cómo activar el enrutamiento dinámico en un sistema de interfaz única

El enrutamiento dinámico que utiliza un protocolo de enrutamiento es la manera más sencilla de gestionar el enrutamiento en un sistema.

1. **Conviértase en administrador.**

   Para obtener más información, consulte "Cómo usar los derechos administrativos que tiene asignados" de *Administración de Oracle Solaris 11.1: servicios de seguridad*.

2. **Configure las interfaces IP del sistema con una dirección IP para la red a la que pertenece el sistema.**

   Para obtener instrucciones, consulte "Cómo configurar una interfaz IP" de *Conexión de sistemas mediante la configuración de redes fijas en Oracle Solaris 11.1*.

3. **Suprima las entradas en el archivo `/etc/defaultrouter`.**

   Un archivo `/etc/defaultrouter` vacío obliga al sistema a utilizar el enrutamiento dinámico.

4. **Asegúrese de que el reenvío de paquetes esté desactivado.**

   ```
   # ipadm set-prop -p forwarding=off ipv4
   ```

5 **Active los protocolos de enrutamiento en el sistema.**

Utilice uno de los siguientes comandos:

- `# routeadm -e ipv4-routing -u`
- `# svcadm enable route:default`

Ejemplo 3–5 Ejecución del enrutamiento dinámico en un sistema de interfaz única

En el ejemplo siguiente, se muestra cómo configurar el enrutamiento dinámico para el comando `hosta`, un sistema de interfaz única en la red `192.168.5.0` que se muestra en la Figura 3–1. El sistema utiliza el enrutador 1 como enrutador predeterminado. El ejemplo supone que ya se configuró la interfaz IP del sistema.

Primero, debe iniciar sesión en `hosta` con derechos de administrador. A continuación, debe eliminar el archivo `/etc/defaultrouter` si está presente en el sistema:

```
# cd /etc
# ls | grep defaultrouter
defaultrouter

# rm defaultrouter

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

           Routing services   "route:default ripng:default"

# svcadm enable route:default

# ipadm show-prop -p forwarding ipv4
PROTO PROPERTY     PERM CURRENT   PERSISTENT   DEFAULT      POSSIBLE
ipv4  forwarding   rw   on          --           off          on,off

# ipadm set-prop -p forwarding=off ipv4
```

Cómo agregar una subred a una red

Si está cambiando de una red que no utiliza una subred a una red que utiliza una subred, realice las tareas de la siguiente lista. Esta lista supone que ya se ha preparado un esquema de subred.

- Asigne las direcciones IP con el nuevo número de subred a los sistemas que pertenecen a esa subred.

  Para obtener referencias, consulte "Cómo configurar una interfaz IP" de *Conexión de sistemas mediante la configuración de redes fijas en Oracle Solaris 11.1*.

- Agregue la dirección IP y la máscara de red correctas al archivo `/etc/netmasks` de cada sistema.
- Revise el archivo `/etc/inet/hosts` de cada sistema con la dirección IP correcta que corresponde a los nombres de host.
- Reinicie todos los sistemas de la subred.

El siguiente procedimiento está estrechamente relacionado con las subredes. Si implementa subredes mucho tiempo después de haber configurado originalmente la red sin subredes, realice el siguiente procedimiento para implementar los cambios.

▼ Cómo cambiar la dirección IPv4 y otros parámetros de configuración de red

Este procedimiento explica cómo modificar la dirección IPv4, el nombre de host y otros parámetros de red en un sistema instalado previamente. Siga el procedimiento para modificar la dirección IP de un servidor o sistema autónomo en red. El procedimiento no se aplica a los clientes o dispositivos en red. Estos pasos crean una configuración que persiste a pesar de los reinicios.

Nota – Las instrucciones tienen la finalidad de cambiar la dirección IPv4 de la interfaz de red principal. Para agregar otra interfaz al sistema, consulte "Cómo configurar una interfaz IP" de *Conexión de sistemas mediante la configuración de redes fijas en Oracle Solaris 11.1*.

En la mayoría de los casos, los pasos siguientes utilizan la notación decimal con punto de IPv4 tradicional para especificar la dirección IPv4 y la máscara de subred. También puede utilizar la notación CIDR para especificar la dirección IPv4 en todos los archivos aplicables de este procedimiento.

1 Conviértase en administrador.

Para obtener más información, consulte "Cómo usar los derechos administrativos que tiene asignados" de *Administración de Oracle Solaris 11.1: servicios de seguridad*.

2 Modifique la dirección IP con el comando `pntadm`.

Con el comando `ipadm`, no puede modificar una dirección de IP directamente. Primero suprima el objeto de dirección que representa la dirección IP que desea modificar. A continuación, asigne una nueva dirección mediante la misma dirección nombre de objeto.

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

3. **Si corresponde, modifique la entrada de nombre de host en el servicio SMF `system/identity:node`:**

   ```
   # hostname newhostname
   ```

   Este paso produce un cambio persistente.

4. **Si la máscara de subred ha cambiado, modifique las entradas de subred en el archivo `/etc/netmasks`.**

5. **Si la dirección de subred ha cambiado, cambie la dirección IP del enrutador predeterminado en `/etc/defaultrouter` a la dirección del nuevo enrutador predeterminado de la subred.**

6. **Reinicie el sistema.**

   ```
   # reboot -- -r
   ```

Ejemplo 3–6 Cambio de la dirección IP y el nombre de host

En este ejemplo, se muestra cómo cambiar el nombre de un host, la dirección IP de la interfaz de red principal y la máscara de subred. La dirección IP de la interfaz de red principal `net0` cambia de `10.0.0.14` a `192.168.34.100`.

```
# ipadm show-addr
ADDROBJ      TYPE     STATE   ADDR
lo0/v4       static   ok      127.0.0.1/8
net0/v4      static   ok      10.0.0.14/24

# ipadm delete-addr net0/v4
# ipadm create-addr -a 192.168.34.100/24 net0
# hostname mynewhostname

# ipadm show-addr
ADDROBJ         TYPE     STATE   ADDR
lo0/v4          static   ok      127.0.0.1/8
net0/v4         static   ok      192.168.34.100/24

# hostname
mynewhostname
```

Véase también Para cambiar la dirección IP de una interfaz que no sea la interfaz de red principal, consulte "Cómo configurar una interfaz IP" de *Conexión de sistemas mediante la configuración de redes fijas en Oracle Solaris 11.1*.

Supervisión y modificación de los servicios de capa de transporte

Los protocolos de capa de transporte TCP, SCTP y UDP son parte del paquete Oracle Solaris estándar. Estos protocolos normalmente no requieren ninguna intervención para ejecutarse correctamente. Sin embargo, las circunstancias de su sitio podrían requerir el registro o la modificación de los servicios que ejecutan los protocolos de capa de transporte. Luego, debe modificar los perfiles de estos servicios usando la utilidad de gestión de servicios (SMF), que se describe en el Capítulo 1, "Gestión de servicios (descripción general)" de *Gestión de servicios y errores en Oracle Solaris 11.1*.

El daemon `inetd` se encarga de iniciar los servicios estándar de Internet cuando se inicia un sistema. Estos servicios incluyen aplicaciones que utilizan TCP, SCTP o UDP como protocolo de capa de transporte. Puede modificar los servicios de Internet existentes o agregar servicios nuevos con los comandos SMF. Para más información sobre `inetd`, consulte "Daemon de servicios de Internet `inetd`" en la página 127.

Las operaciones que requieren protocolos de capa de transporte incluyen:

- Registrar todas las conexiones TCP entrantes
- Agregar servicios que ejecutan un protocolo de capa de transporte, utilizando SCTP a modo de ejemplo
- Configurar la función de envoltorios TCP para el control de acceso

Para obtener información detallada sobre el daemon `inetd`, consulte la página del comando man `inetd`(1M).

▼ Cómo registrar las direcciones IP de todas las conexiones TCP entrantes

1 Conviértase en administrador.

Para obtener más información, consulte "Cómo usar los derechos administrativos que tiene asignados" de *Administración de Oracle Solaris 11.1: servicios de seguridad*.

2 Active el seguimiento TCP para todos los servicios que administre `inetd`.

```
# inetadm -M tcp_trace=TRUE
```

▼ Cómo agregar servicios que utilicen el protocolo SCTP

El protocolo de transporte SCTP ofrece servicios a los protocolos de capa de modo similar a TCP. Sin embargo, SCTP permite la comunicación entre dos sistemas, que pueden ser (uno o ambos) de host múltiple. La conexión SCTP se denomina *asociación*. En una asociación, una aplicación divide los datos que se transmitirán en uno o más flujos de mensajes, o en *múltiples flujos*. Una conexión SCTP puede realizarse en los puntos finales con varias direcciones IP, lo cual es especialmente importante en las aplicaciones de telefonía. Las posibilidades que ofrece el host múltiple de SCTP constituyen una consideración de seguridad si el sitio utiliza filtro IP o IPsec. En la página del comando man `sctp(7P)` se describen algunas de estas consideraciones.

De modo predeterminado, SCTP se incluye en Oracle Solaris y no requiere ninguna configuración adicional. Sin embargo, es posible que tenga que configurar de modo explícito determinados servicios de capa de la aplicación para que utilicen SCTP. Algunas aplicaciones de ejemplo son `echo` y `discard`. El procedimiento siguiente muestra cómo agregar un servicio echo que utilice un socket de estilo uno a uno SCTP.

Nota – También puede utilizar el procedimiento siguiente para agregar servicios para los protocolos de capa de transporte TCP y UDP.

La tarea siguiente muestra cómo agregar un servicio SCTP `inet` que administre el daemon `inetd` al repositorio SMF. La tarea muestra cómo utilizar los comandos de la utilidad de gestión de servicios (SMF) para agregar el servicio.

- Para obtener información sobre los comandos SMF, consulte "Utilidades administrativas de la línea de comandos de la SMF" de *Gestión de servicios y errores en Oracle Solaris 11.1*.
- Para obtener información sobre la sintaxis, consulte las páginas del comando man para los comandos SMF, como se describe en el procedimiento.
- Para obtener información detallada sobre SMF, consulte la página del comando man `smf(5)`.

Antes de empezar Antes de llevar a cabo el procedimiento siguiente, cree un archivo manifest para el servicio. El procedimiento utiliza como ejemplo un archivo manifest para el servicio `echo` que se denomina `echo.sctp.xml`.

1 Inicie sesión en el sistema local con una cuenta de usuario con privilegios de escritura para los archivos del sistema.

2 Edite el archivo `/etc/services` y agregue una definición para el nuevo servicio.

Utilice la siguiente sintaxis para la definición del servicio.

service-name |*port/protocol* | *aliases*

3 Agregue el nuevo servicio.

Vaya al directorio en el que se encuentra el manifiesto del servicio y escriba lo siguiente:

```
# cd dir-name
# svccfg import service-manifest-name
```

Para ver la sintaxis completa de `svccfg`, consulte la página del comando man `svccfg(1M)`.

Supongamos que desea agregar un nuevo servicio SCTP `echo` utilizando el manifiesto `echo.sctp.xml` que se encuentra en el directorio `service.dir`. Debe escribir lo siguiente:

```
# cd service.dir
# svccfg import echo.sctp.xml
```

4 Compruebe que se haya agregado el manifiesto del servicio:

```
# svcs FMRI
```

Para el argumento *FMRI*, utilice el Fault Managed Resource Identifier (FMRI) del manifiesto del servicio. Por ejemplo, para el servicio SCTP `echo`, debe utilizar el comando siguiente:

```
# svcs svc:/network/echo:sctp_stream
```

El resultado que obtendrá será similar al siguiente:

```
    STATE          STIME    FMRI
disabled       16:17:00 svc:/network/echo:sctp_stream
```

Si desea obtener información detallada sobre el comando `svcs`, consulte la página del comando man `svcs(1)`.

El resultado indica que el nuevo manifiesto del servicio está desactivado.

5 Enumere las propiedades del servicio para determinar si debe realizar modificaciones.

```
# inetadm -l FMRI
```

Para obtener información detallada sobre el comando `inetadm`, consulte la página del comando man `inetadm(1M)`.

Por ejemplo, para el servicio SCTP `echo`, debe escribir lo siguiente:

```
# inetadm -l svc:/network/echo:sctp_stream
SCOPE    NAME=VALUE
             name="echo"
             endpoint_type="stream"
             proto="sctp"
             isrpc=FALSE
             wait=FALSE
             exec="/usr/lib/inet/in.echod -s"
         .
         .
         default  tcp_trace=FALSE
           default  tcp_wrappers=FALSE
```

6. **Active el nuevo servicio:**

   ```
   # inetadm -e FMRI
   ```

7. **Compruebe que el servicio esté activado:**

   Por ejemplo, para el nuevo servicio echo, debe escribir:

   ```
   # inetadm | grep sctp_stream
   .
   .
       enabled   online         svc:/network/echo:sctp_stream
   ```

Ejemplo 3–7 Cómo agregar un servicio que utilice el protocolo de transporte SCTP

El siguiente ejemplo muestra los comandos para utilizar las entradas de archivo necesarias para que el servicio echo utilice el protocolo de capa de transporte SCTP.

```
$ cat /etc/services
.
.
echo            7/tcp
echo            7/udp
echo            7/sctp

# cd service.dir

    # svccfg import echo.sctp.xml

# svcs network/echo*
    STATE          STIME    FMRI
    disabled       15:46:44 svc:/network/echo:dgram
    disabled       15:46:44 svc:/network/echo:stream
    disabled       16:17:00 svc:/network/echo:sctp_stream

# inetadm -l svc:/network/echo:sctp_stream
    SCOPE    NAME=VALUE
             name="echo"
             endpoint_type="stream"
             proto="sctp"
             isrpc=FALSE
             wait=FALSE
             exec="/usr/lib/inet/in.echod -s"
             user="root"
    default  bind_addr=""
    default  bind_fail_max=-1
    default  bind_fail_interval=-1
    default  max_con_rate=-1
    default  max_copies=-1
    default  con_rate_offline=-1
    default  failrate_cnt=40
    default  failrate_interval=60
    default  inherit_env=TRUE
    default  tcp_trace=FALSE
    default  tcp_wrappers=FALSE
```

```
# inetadm -e svc:/network/echo:sctp_stream

# inetadm | grep echo
    disabled  disabled       svc:/network/echo:stream
    disabled  disabled       svc:/network/echo:dgram
    enabled   online         svc:/network/echo:sctp_stream
```

▼ Cómo utilizar los envoltorios TCP para controlar el acceso a los servicios TCP

El programa `tcpd` implementa *envoltorios TCP*. Los envoltorios TCP incorporan una medida de seguridad para los daemons de servicio como `ftpd` al permanecer entre el daemon y las solicitudes de servicio entrantes. Los envoltorios TCP registran los intentos de conexión correctos e incorrectos. Asimismo, los envoltorios TCP pueden proporcionar control de acceso, y permitir o denegar la conexión en función del lugar donde se origine la solicitud. Puede utilizar los envoltorios TCP para proteger los daemons como SSH, Telnet o FTP. La aplicación `sendmail` también puede utilizar envoltorios TCP, como se describe en "Compatibilidad con envoltorios TCP de la versión 8.12 de sendmail" de *Gestión de servicios sendmail en Oracle Solaris 11.1*.

1. **Conviértase en administrador.**

   Para obtener más información, consulte "Cómo usar los derechos administrativos que tiene asignados" de *Administración de Oracle Solaris 11.1: servicios de seguridad*.

2. **Active los envoltorios TCP.**

   ```
   # inetadm -M tcp_wrappers=TRUE
   ```

3. **Configure la política de control de acceso de los envoltorios TCP tal como se describe en la página del comando man `hosts_access`(3).**

   Esta página del comando man se puede encontrar en el directorio `/usr/sfw/man`.

◆ ◆ ◆ CAPÍTULO 4

Activación de IPv6 en una red

En este capítulo, se presentan las tareas para activar IPv6 en una red. Se tratan los temas principales siguientes:

Configuración de una interfaz de IPv6

Como paso inicial para usar IPv6 en una red, configure IPv6 en la interfaz IP del sistema.

En el proceso de instalación de Oracle Solaris, IPv6 se puede activar en una o varias interfaces del sistema. Si activó la compatibilidad con IPv6 durante la instalación, una vez que se completa la instalación, se crean los siguientes archivos y tablas relacionados con IPv6:

- El servicio SMF `name-service/switch` se modificó para permitir consultas mediante direcciones IPv6.
- Se crea la tabla de directrices de selección de direcciones IPv6. En esta tabla se da prioridad al formato de direcciones IP que debe utilizarse en las transmisiones a través de una interfaz activada para IPv6.

En esta sección, se describe cómo activar IPv6 en las interfaces después de que se completa la instalación de Oracle Solaris.

▼ Cómo configurar un sistema para IPv6

Comience el proceso de configuración de IPv6. Para ello, active IPv6 en las interfaces de todos los sistemas que se convertirán en nodos de IPv6. Al principio, la interfaz obtiene su dirección IPv6 mediante el proceso de configuración automática, como se explica en "Proceso de configuración automática" en la página 146. Posteriormente, puede adaptar a su conveniencia la configuración del nodo a partir de su función en la red IPv6 como host, servidor o enrutador.

Nota – Si la interfaz se ubica en el mismo vínculo como enrutador que anuncia un prefijo de IPv6, la interfaz obtiene el prefijo de sitio como parte de sus direcciones configuradas automáticamente. Para obtener más información, consulte "Cómo configurar un enrutador activado para IPv6" en la página 66.

En el procedimiento siguiente se explica cómo activar IPv6 para una interfaz incorporada después de instalar Oracle Solaris.

1 **Configure la interfaz IP con los comandos adecuados.**

Consulte "Cómo configurar una interfaz IP" de *Conexión de sistemas mediante la configuración de redes fijas en Oracle Solaris 11.1*.

Nota – Al asignar la dirección IP, asegúrese de utilizar la opción correcta para asignar una dirección IPv6:

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

Para agregar más direcciones, utilice la sintaxis siguiente:

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

---

2 **Inicie el daemon de IPv6 `in.ndpd`.**

```
# /usr/lib/inet/in.ndpd
```

3 **(Opcional) Cree una ruta IPv6 estática predeterminada.**

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

4 **(Opcional) Cree un archivo `/etc/inet/ndpd.conf` que defina parámetros para variables de interfaz en el nodo.**

Si tiene que crear direcciones temporales para la interfaz del host, consulte "Uso de direcciones temporales para una interfaz" en la página 69. Para obtener más información sobre `/etc/inet/ndpd.conf`, consulte la página del comando man `ndpd.conf`(4) y "Archivo de configuración `ndpd.conf`" en la página 133.

5 **(Opcional) Para visualizar el estado de las interfaces IP con sus configuraciones IPv6, escriba el comando siguiente:**

```
# ipadm show-addr
```

Ejemplo 4–1 Activación de una interfaz para IPv6 tras la instalación

En este ejemplo, se muestra cómo activar IPv6 en la interfaz `net0`. Antes de comenzar, compruebe el estado de todas las interfaces configuradas en el sistema.

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

Para este sistema, únicamente está configurada la interfaz `net0`. Active IPv6 en esta interfaz de la forma que se indica a continuación:

```
# ipadm create-addr -T addrconf net0
# ipadm create-addr -a 2001:db8:3c4d:15:203/64 net0
# /usr/lib/inet/in.ndpd

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

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

Pasos siguientes

- Para configurar el nodo de IPv6 como enrutador, consulte "Configuración de un enrutador IPv6" en la página 66.
- Para anular la configuración automática de direcciones en el nodo, consulte "Cómo desactivar la configuración automática de direcciones IPv6" en la página 65.
- Para adaptar el nodo como servidor, tenga en cuenta las sugerencias de "Administración de interfaces activadas para IPv6 en servidores" en la página 74.

▼ Cómo desactivar la configuración automática de direcciones IPv6

En general, la configuración automática de direcciones se emplea para generar las direcciones IPv6 de las interfaces de hosts y servidores. No obstante, en ocasiones quizá quiera desactivar la configuración automática de direcciones, sobre todo a la hora de configurar manualmente un token, como se explica en "Configuración de un token IPv6" en la página 72.

1. **Cree un archivo `/etc/inet/ndpd.conf` para el nodo.**

   El archivo `/etc/inet/ndpd.conf` define las variables de interfaz del nodo en particular. Este archivo debería contener lo siguiente a fin de desactivar la configuración automática de direcciones para una interfaz en el servidor:

   interface `StatelessAddrConf false`

   Para desactivar la configuración automática de todas las interfaces, utilice la siguiente entrada:

   ```
   ifdefault StatelessAddrConf false
   ```

   Para obtener más información sobre `/etc/inet/ndpd.conf`, consulte la página del comando man `ndpd.conf(4)` y "Archivo de configuración `ndpd.conf`" en la página 133.

2. **Actualice el daemon de IPv6 con los cambios.**

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

Configuración de un enrutador IPv6

En esta sección, se describen las tareas para configurar un enrutador IPv6. Según los requisitos del sitio, es posible que deba realizar únicamente tareas seleccionadas.

▼ Cómo configurar un enrutador activado para IPv6

El siguiente procedimiento supone que ya ha configurado el sistema para IPv6. Para conocer los procedimientos, consulte "Configuración de una interfaz de IPv6" en la página 63.

1. **Configure el reenvío de paquetes IPv6 en todas las interfaces del enrutador.**

   ```
   # ipadm set-prop -p forwarding=on ipv6
   ```

2. **Inicie el daemon de enrutamiento.**

   El daemon `in.ripngd` se encarga del enrutamiento de IPv6. Active el enrutamiento de IPv6 mediante cualquiera de las opciones siguientes:

   - Utilice el comando `routeadm`:

     ```
     # routeadm -e ipv6-routing -u
     ```

   - Utilice el comando SMF adecuado:

     ```
     # svcadm enable ripng:default
     ```

   Para obtener información sobre la sintaxis del comando `routeadm`, consulte la página del comando man `routeadm(1M)`.

3 Cree el archivo `/etc/inet/ndpd.conf`.

Especifique el prefijo de sitio que debe anunciar el enrutador y demás datos de configuración en `/etc/inet/ndpd.conf`. El daemon `in.ndpd` lee este archivo e implementa el protocolo de descubrimiento de vecinos de IPv6.

Para obtener una lista de variables y valores admitidos, consulte "Archivo de configuración `ndpd.conf`" en la página 133 y la página del comando man `ndpd.conf(4)`.

4 Escriba el texto siguiente en el archivo `/etc/inet/ndpd.conf`:

```
ifdefault AdvSendAdvertisements true
prefixdefault AdvOnLinkFlag on AdvAutonomousFlag on
```

Este texto indica al daemon `in.ndpd` que envíe anuncios de enrutador en todas las interfaces del enrutador que se hayan configurado para IPv6.

5 Agregue texto adicional al archivo `/etc/inet/ndpd.conf` para configurar el prefijo de sitio en las distintas interfaces del enrutador.

El texto debe tener el formato siguiente:

`prefix` *global-routing-prefix*`:`*subnet ID*`/64` *interface*

En el siguiente archivo de ejemplo `/etc/inet/ndpd.conf`, se configura el enrutador para que anuncie el prefijo de sitio `2001:0db8:3c4d::/48` en las interfaces `net0` y `net1`.

```
ifdefault AdvSendAdvertisements true
prefixdefault AdvOnLinkFlag on AdvAutonomousFlag on

if net0 AdvSendAdvertisements 1
prefix 2001:0db8:3c4d:15::0/64 net0

if net1 AdvSendAdvertisements 1
prefix 2001:0db8:3c4d:16::0/64 net1
```

6 Reinicie el sistema.

El enrutador de IPv6 comienza a anunciar en el vínculo cualquier prefijo de sitio que esté en el archivo `ndpd.conf`.

Ejemplo 4–2 Salida de `ipadm show-addr` que muestra interfaces IPv6

En el ejemplo siguiente, se muestra la salida del comando `ipadm show-addr` después de finalizar el procedimiento de "Configuración de un enrutador IPv6" en la página 66.

```
ADDROBJ      TYPE       STATE   ADDR
lo0/v4       static     ok      127.0.0.1/8
net0/v4      static     ok      172.16.15.232/24
net1/v4      static     ok      172.16.16.220/24
net0/v6      addrconf   ok      fe80::203:baff:fe11:b115/10
lo0/v6       static     ok      ::1/128
net0/v6a     static     ok      2001:db8:3c4d:15:203:baff:fe11:b115/64
net1/v6      addrconf   ok      fe80::203:baff:fe11:b116/10
net1/v6a     static     ok      2001:db8:3c4d:16:203:baff:fe11:b116/64
```

En este ejemplo, cada interfaz configurada para IPv6 dispone ahora de dos direcciones. La entrada con el nombre de objeto de dirección, como *interface*`/v6`, muestra la dirección de enlace local de esa interfaz. La entrada con el nombre de objeto de dirección, como *interface* `/v6add` muestra una dirección IPv6 global. Esta dirección incluye el prefijo de sitio configurado en el archivo `/etc/ndpd.conf`, además del ID de interfaz. Tenga en cuenta que la designación `v6add` es una cadena definida de forma aleatoria. Puede definir otras cadenas para la segunda parte del nombre de objeto de dirección, siempre que la *interface* refleje la interfaz donde se están creando las direcciones IPv6, por ejemplo `net0/mystring`, `net0/ipv6addr`, etc.

Véase también

- Para configurar túneles desde los enrutadores identificados en su topología de red IPv6, consulte "Configuración y administración de túneles con el comando `dladm`" en la página 112.
- Para obtener información sobre cómo configurar conmutadores y concentradores en la red, consulte la documentación del fabricante.
- Para configurar hosts de IPv6, consulte "Modificación de la configuración de una interfaz de IPv6 para hosts y servidores" en la página 68.
- Para mejorar la compatibilidad de IPv6 en los servidores, consulte "Administración de interfaces activadas para IPv6 en servidores" en la página 74.
- Para obtener más información sobre comandos, archivos y daemons de IPv6, consulte "Implementación de IPv6 en Oracle Solaris" en la página 133.

Modificación de la configuración de una interfaz de IPv6 para hosts y servidores

Esta sección explica el procedimiento para modificar la configuración de interfaces activadas para IPv6 en nodos que son hosts o servidores. En la mayoría de los casos, deberá utilizar la configuración automática de direcciones para interfaces activadas para IPv6. Sin embargo, la dirección IPv6 de una interfaz se puede modificar, si hace falta, como se explica en las tareas de la presente sección.

Debe realizar tres tareas generales en el siguiente orden:

1. Desactivar la configuración automática de direcciones IPv6. Consulte "Cómo desactivar la configuración automática de direcciones IPv6" en la página 65.
2. Crear una dirección temporal para un host. Consulte "Cómo configurar una dirección temporal" en la página 69.
3. Configurar un token IPv6 para el ID de interfaz. Consulte "Cómo configurar un token IPv6 especificado por el usuario" en la página 72.

Uso de direcciones temporales para una interfaz

Una *dirección temporal* IPv6 emplea un número de 64 bits generado aleatoriamente como ID de interfaz, en lugar de la dirección MAC de la interfaz. Puede utilizar direcciones temporales para las interfaces de un nodo IPv6 que desee mantener anónimas. Por ejemplo, puede utilizar direcciones temporales para las interfaces de un host que deba acceder a servidores web públicos. Las direcciones temporales implementan mejoras de privacidad de IPv6. Estas mejoras se describen en RFC 3041, que está disponible en "Privacy Extensions for Stateless Address Autoconfiguration in IPv6" (`http://www.ietf.org/rfc/rfc3041.txt?number=3041`).

Las direcciones temporales se activan en el archivo `/etc/inet/ndpd.conf` para una o varias interfaces, si es necesario. Sin embargo, a diferencia de las direcciones IPv6 estándar configuradas automáticamente, una dirección temporal consta del prefijo de subred de 64 bits y un número de 64 bits generado aleatoriamente. Ese número aleatorio constituye el segmento de ID de interfaz de la dirección IPv6. Una dirección local de vínculo no se genera con la dirección temporal como ID de interfaz.

Las direcciones temporales tienen un *periodo de vida preferente* predeterminado de un día. Al activar la generación de direcciones temporales, también puede configurar las variables siguientes en el archivo `/etc/inet/ndpd.conf`:

periodo de vida válido TmpValidLifetime	Lapso durante el cual existe la dirección temporal; una vez transcurrido, la dirección se suprime del host.
periodo de vida preferente TmpPreferredLifetime	Tiempo transcurrido antes de prescindir de la dirección temporal. Ese lapso de tiempo debe ser más breve que el periodo de vida válido.
regeneración de direcciones	Intervalo de tiempo antes de la conclusión del periodo de vida preferente durante el cual el host debe generar otra dirección temporal.

La duración de las direcciones temporales se especifica de la manera siguiente:

n	*n* cantidad de segundos, que es el valor predeterminado
n `h`	*n* cantidad de horas (`h`)
n `d`	*n* cantidad de días (`d`)

▼ Cómo configurar una dirección temporal

1. **Si es necesario, active IPv6 en las interfaces del host.**

   Consulte "Cómo configurar un sistema para IPv6" en la página 64.

2. **Edite el archivo `/etc/inet/ndpd.conf` para activar la generación de direcciones temporales.**

- Para configurar direcciones temporales en todas las interfaces de un host, agregue la línea siguiente en el archivo `/etc/inet/ndpd.conf`:

  `ifdefault TmpAddrsEnabled true`
- Para configurar una dirección temporal para una determinada interfaz, agregue la línea siguiente en el archivo `/etc/inet/ndpd.conf`:

  `if` *interface* **`TmpAddrsEnabled true`**

3 (Opcional) Especifique el periodo de vida válido de la dirección temporal.

`ifdefault TmpValidLifetime` *duration*

Esta sintaxis especifica el periodo de vida válido de todas las interfaces en un host. El valor de *duración* debe especificarse en segundos, horas o días. El periodo de vida válido predeterminado es 7 días. `TmpValidLifetime` también puede usarse con las palabras clave `if` *interface* para especificar el periodo de vida válido de una dirección temporal relativa a una determinada interfaz.

4 (Opcional) Especifique un periodo de vida preferente para la dirección temporal; una vez transcurrido, se prescinde de la dirección.

`if` *interface* **`TmpPreferredLifetime`** *duration*

Esta sintaxis especifica el periodo de vida preferente de la dirección temporal de una determinada interfaz. El periodo de vida preferente predeterminado es un día. `TmpPreferredLifetime` también se puede utilizar con la palabra clave `ifdefault` para indicar el periodo de vida preferente de las direcciones temporales relativas a todas las interfaces de un host.

Nota – La selección de direcciones predeterminadas otorga una prioridad inferior a las direcciones IPv6 que se han descartado. Si se prescinde de una dirección IPv6 temporal, la selección de direcciones predeterminadas elige una dirección no descartada como dirección de origen de un paquete. Una dirección no descartada podría ser la dirección IPv6 generada de manera automática o, posiblemente, la dirección IPv4 de la interfaz. Para obtener más información sobre la selección de direcciones predeterminadas, consulte "Administración de selección de direcciones predeterminadas" en la página 99.

---

5 (Opcional) Especifique el tiempo de generación antes del descarte de direcciones durante el cual el host debe generar otra dirección temporal.

`ifdefault TmpRegenAdvance duration`

Esta sintaxis indica el tiempo de generación antes del descarte de dirección de las direcciones temporales relativas a todas las interfaces de un host. El valor predeterminado es 5 segundos.

6. **Cambie la configuración del daemon `in.ndpd`.**

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

7. **Verifique que las direcciones temporales se hayan creado con el comando `ipadm show-addr`, como se muestra en el Ejemplo 4–4.**

   La salida del comando muestra el indicador `t` en el campo `CURRENT` de las direcciones temporales.

Ejemplo 4–3 Variables de direcciones temporales en el archivo `/etc/inet/ndpd.conf`

En el ejemplo siguiente se muestra un segmento de un archivo `/etc/inet/ndpd.conf` con direcciones temporales activadas para la interfaz de red principal.

```
ifdefault TmpAddrsEnabled true

ifdefault TmpValidLifetime 14d

ifdefault TmpPreferredLifetime 7d

ifdefault TmpRegenAdvance 6s
```

Ejemplo 4–4 Salida del comando `ipadm show-addr` con direcciones temporales activadas

En este ejemplo, se muestra la salida del comando `ipadm show-addr` después de crear direcciones temporales. Tenga en cuenta que en la salida de ejemplo únicamente se incluye información relacionada con IPv6.

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

Tenga en cuenta que para el objeto de dirección `net0/?`, el indicador `t` se configura en el campo `CURRENT`. El indicador señala que la dirección correspondiente tiene un ID de interfaz temporal.

Véase también

- Para configurar la compatibilidad del servicio de nombres para direcciones IPv6, consulte "Configuración de la compatibilidad con el servicio de nombres para IPv6" en la página 75.
- Para configurar direcciones IPv6 para un servidor, consulte "Cómo configurar un token IPv6 especificado por el usuario" en la página 72.
- Para supervisar actividades en los nodos IPv6, consulte el Capítulo 5, "Administración de una red TCP/IP".

Configuración de un token IPv6

El ID de interfaz de 64 bits de una dirección IPv6 también se denomina *token*. Durante la configuración automática de direcciones, el token se asocia con la dirección MAC de la interfaz. En la mayoría de los casos, los nodos sin enrutadores, es decir los hosts y servidores IPv6, deben utilizar sus tokens configurados automáticamente.

No obstante, el uso de tokens configurados automáticamente puede comportar problemas en servidores cuyas interfaces se intercambien de manera rutinaria como parte de la administración de sistemas. Si se cambia la tarjeta de interfaz, también se cambia la dirección MAC. Como consecuencia, los servidores que necesiten direcciones IP estables pueden tener problemas. Las distintas partes de la infraestructura de red, por ejemplo DNS o NIS, pueden tener guardadas determinadas direcciones IPv6 para las interfaces del servidor.

Para prevenir los problemas de cambio de dirección, puede configurar manualmente un token para emplearse como ID de interfaz en una dirección IPv6. Para crear el token, especifique un número hexadecimal de 64 bits o menos para ocupar la parte del ID de interfaz de la dirección IPv6. En la subsiguiente configuración automática de direcciones, el descubrimiento de vecinos no crea un ID de interfaz que se base en la dirección MAC de la interfaz. En lugar de ello, el token creado manualmente se convierte en el ID de interfaz. Este token queda asignado a la interfaz, incluso si se sustituye una tarjeta.

Nota – La diferencia entre los tokens especificados por el usuario y las direcciones temporales es que estas segundas se generan aleatoriamente, no las crea el usuario.

▼ Cómo configurar un token IPv6 especificado por el usuario

Las instrucciones siguientes suelen ser útiles en el caso de servidores cuyas interfaces se reemplazan de manera rutinaria. También son aptas para configurar tokens especificados por el usuario en cualquier nodo de IPv6.

1 Verifique que la interfaz que desea configurar con un token exista y que no haya direcciones IPv6 configuradas en la interfaz.

Nota – Asegúrese de que la interfaz no tenga configurada ninguna dirección IPv6.

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

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

En esta salida, se muestra que la interfaz de red `net0` existe y no que tiene configurada ninguna dirección IPv6.

2 **Cree uno o varios números hexadecimales de 64 bits para utilizar como tokens para las interfaces del nodo con el formato *xxxx*:*xxxx*:*xxxx*:*xxxx*.**

3 **Configure cada interfaz con un token.**

Utilice la forma siguiente del comando `ipadm` para cada interfaz que deba tener un ID de interfaz especificado por el usuario (token):

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

Por ejemplo, utilice el comando siguiente para configurar la interfaz `net0` con un token:

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

Nota – Después de crear el objeto de dirección con el token, no se puede modificar el token.

4 **Actualice el daemon de IPv6 con los cambios.**

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

Ejemplo 4–5 Configuración de un token especificado por el usuario en una interfaz de IPv6

En el ejemplo siguiente, se muestra que `net0` se configura con una dirección IPv6 y un token.

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

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

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

Después de configurar el token, el objeto de dirección `net0/v6` tiene una dirección de enlace local y una dirección con `1a:2b:3c:4d` configurado para este ID de interfaz. Tenga en cuenta que este token no puede ser modificado para esta interfaz después de la creación de `net0/v6`.

Véase también

- Para actualizar los servicios de nombres con las direcciones IPv6 del servidor, consulte "Configuración de la compatibilidad con el servicio de nombres para IPv6" en la página 75.

- Para supervisar el rendimiento del servidor, consulte el Capítulo 5, "Administración de una red TCP/IP".

Administración de interfaces activadas para IPv6 en servidores

Si tiene previsto implementar IPv6 en un servidor, debe adoptar una serie de medidas al activar IPv6 en las interfaces del servidor. Las decisiones repercuten en la estrategia que se aplica en la configuración de los ID de interfaz, o *tokens*, de una dirección IPv6 de interfaz.

▼ Cómo activar IPv6 en las interfaces de un servidor

Este procedimiento proporciona pasos generales para activar IPv6 en los servidores de la red. Algunos de los pasos pueden variar según cómo desea implementar IPv6.

1. **Active IPv6 en las interfaces IP del servidor.**

   Para conocer los procedimientos, consulte "Configuración de una interfaz de IPv6" en la página 63.

2. **Compruebe que el prefijo de subred IPv6 esté configurado en un enrutador en el mismo vínculo que el servidor.**

   Para obtener más información, consulte "Configuración de un enrutador IPv6" en la página 66.

3. **Aplique la estrategia pertinente relativa al ID de interfaz en las interfaces activadas para IPv6 del servidor.**

   De forma predeterminada, la configuración automática de direcciones IPv6 utiliza la dirección MAC de una interfaz al crear la parte del ID de interfaz de la dirección IPv6. Si se conoce bien la dirección IPv6 de la interfaz, el intercambio de interfaces puede resultar problemático. La dirección MAC de la nueva interfaz será distinta. En el proceso de configuración automática de direcciones, se genera un nuevo ID de interfaz.

   - En una interfaz activada para IPv6 que no tenga previsto reemplazar, utilice la dirección IPv6 configurada automáticamente, como se explica en "Proceso de configuración automática" en la página 146.
   - En el caso de interfaces activadas para IPv6 que deben figurar como anónimas fuera de la red local, plantee la posibilidad de utilizar para el ID de interfaz un token generado aleatoriamente. Para obtener instrucciones y un ejemplo, consulte "Cómo configurar una dirección temporal" en la página 69.
   - En las interfaces activadas para IPv6 que tenga previsto intercambiar con regularidad, cree tokens para los ID de interfaz. Para obtener instrucciones y un ejemplo, consulte "Cómo configurar un token IPv6 especificado por el usuario" en la página 72.

Configuración de la compatibilidad con el servicio de nombres para IPv6

En esta sección se explica cómo configurar los servicios de nombres DNS y NIS para admitir los servicios de IPv6.

Nota – LDAP admite IPv6 sin tener que realizar tareas de configuración propias de IPv6.

Para obtener detalles completos sobre la administración de DNS, NIS y LDAP, consulte *Trabajo con servicios de nombres y directorios en Oracle Solaris 11.1*.

▼ Cómo agregar direcciones IPv6 a DNS

1. **Edite el pertinente archivo de zona de DNS agregando registros de AAAA por cada nodo activado para IPv6:**

   hostname `IN AAAA` *host-address*

2. **Edite el archivo de zona inversa de DNS y agregue registros PTR:**

   hostaddress `IN PTR` *hostname*

   Para obtener más información sobre administración de DNS, consulte *Trabajo con servicios de nombres y directorios en Oracle Solaris 11.1*.

Ejemplo 4–6 Archivo de zona inversa de DNS

En este ejemplo se muestra una dirección IPv6 en el archivo de zona inversa.

```
$ORIGIN    ip6.int.
8.2.5.0.2.1.e.f.f.f.9.2.0.0.a.0.6.5.2.9.0.0.0.0.0.0.0.0.2.0.0.0 \
    IN        PTR        vallejo.Eng.apex.COM.
```

▼ Cómo visualizar información sobre servicios de nombres de IPv6

El comando `nslookup` se utiliza para visualizar información sobre servicios de nombres de IPv6.

1. **Desde la cuenta de usuario, ejecute el comando `nslookup`.**

   ```
   % /usr/sbin/nslookup
   ```

Se muestran la dirección y el nombre de servidor predeterminados, seguidos del símbolo de comillas angulares del comando `nslookup`.

2 **Visualice información de un determinado host. Para ello, en el símbolo de comillas angulares escriba los comandos siguientes:**

```
>set q=any
>hostname
```

3 **Escriba el comando siguiente para ver sólo registros AAAA:**

```
>set q=AAAA
hostname
```

4 **Salga del comando `nslookup`. Para ello, escriba `exit`.**

Ejemplo 4–7 Uso del comando `nslookup` para visualizar información relativa a IPv6

En este ejemplo se muestra el resultado del comando `nslookup` en un entorno de red IPv6.

```
%  /usr/sbin/nslookup
Default Server:  dnsserve.local.com
Address:  10.10.50.85
> set q=AAAA
> host85
Server:  dnsserve.local.com
Address:  10.10.50.85

host85.local.com      IPv6 address = 2::9256:a00:fe12:528
> exit
```

▼ Cómo verificar que los registros PTR de DNS IPv6 se actualicen correctamente

En este procedimiento, el comando `nslookup` se utiliza para visualizar los registros PTR relativos a DNS IPv6.

1 **En la cuenta de usuario, ejecute el comando `nslookup`.**

```
% /usr/sbin/nslookup
```

Se muestran la dirección y el nombre de servidor predeterminados, seguidos del símbolo de comillas angulares del comando `nslookup`.

2 **En el símbolo de comillas angulares, escriba lo siguiente para ver los registros PTR:**

```
>set q=PTR
```

3 **Salga del comando. Para ello, escriba `exit`.**

Ejemplo 4–8 Uso del comando `nslookup` para visualizar registros PTR

El ejemplo siguiente muestra la visualización de registros PTR generada a partir del comando `nslookup`.

```
%  /usr/sbin/nslookup
Default Server:  space1999.Eng.apex.COM
Address:  192.168.15.78
> set q=PTR
> 8.2.5.0.2.1.e.f.f.f.0.2.0.0.a.0.6.5.2.9.0.0.0.0.0.0.0.0.2.0.0.0.ip6.int

8.2.5.0.2.1.e.f.f.f.0.2.0.0.a.0.6.5.2.9.0.0.0.0.0.0.0.0.2.0.0.0.ip6.int name =
vallejo.ipv6.Eng.apex.COM
ip6.int nameserver = space1999.Eng.apex.COM
> exit
```

▼ Cómo visualizar información de IPv6 mediante NIS

En este procedimiento, el comando `ypmatch` se utiliza para visualizar información relativa a IPv6 mediante NIS:

- **En la cuenta de usuario, escriba lo siguiente para visualizar direcciones IPv6 en NIS:**

  `% ypmatch` *hostname* `hosts .`*byname*

  Aparece la información relativa al *nombre_host* especificado.

◆ ◆ ◆ CAPÍTULO 5

Administración de una red TCP/IP

El presente capítulo presenta tareas para la administración de redes TCP/IP. Contiene los temas siguientes:

Nota – Para supervisar las interfaces de red, consulte "Supervisión de direcciones e interfaces IP" de *Conexión de sistemas mediante la configuración de redes fijas en Oracle Solaris 11.1*.

Las tareas dan por sentado que se dispone de una red TCP/IP operativa, ya sea IPv4y o IPv4/IPv6 de doble pila. Si desea implementar IPv6 en el sistema pero no lo ha hecho, para obtener más información consulte los capítulos siguientes:

- Para planificar una implementación de IPv6, consulte el Capítulo 2, "Consideraciones para el uso de direcciones IPv6".
- Para configurar IPv6 y crear un entorno de red de pila doble, consulte el Capítulo 4, "Activación de IPv6 en una red".

Tareas de administración principales de TCP/IP (mapa de tareas)

La tabla siguiente muestra diversas tareas (por ejemplo, mostrar información de red) para la administración de la red tras la configuración inicial. La tabla incluye una descripción de lo que hace cada tarea y la sección de la documentación actual en que se detalla el procedimiento correspondiente.

Tarea	Descripción	Para obtener información
Visualizar estadísticas según el protocolo.	Supervisar el rendimiento de los protocolos de red en un determinado sistema.	"Cómo visualizar estadísticas por protocolo" en la página 81
Visualizar el estado de la red.	Supervisar el sistema visualizando todos los sockets y las entradas de la tabla de enrutamiento. En la salida figuran la familia de direcciones inet4 de IPv4 y la familia de direcciones inet6 de IPv6.	"Cómo visualizar el estado de los sockets" en la página 84
Visualizar el estado de las interfaces de red.	Supervisar el rendimiento de las interfaces de red, útil para resolver problemas de transmisiones.	"Cómo visualizar el estado de interfaces de red" en la página 83
Visualizar el estado de la transmisión de paquetes.	Supervisar el estado de los paquetes conforme se van transmitiendo.	"Cómo visualizar el estado de las transmisiones de paquetes de un determinado tipo de dirección" en la página 86
Controlar la salida en pantalla de los comandos relacionados con IPv6.	Controla la salida de los comandos `ping`, `netstat` y `traceroute`. Se crea un archivo denominado `inet_type`. En este archivo, se establece la variable `DEFAULT_IP`.	"Cómo controlar la salida de visualización de comandos relacionados con IP" en la página 89
Supervisar el tráfico de la red.	Se visualizan todos los paquetes IP mediante el comando `snoop`.	"Cómo supervisar tráfico de redes IPv6" en la página 95
Efectuar el seguimiento de todas las rutas conocidas en los enrutadores de la red.	Se utiliza el comando `traceroute` para mostrar todas las rutas.	"Cómo efectuar el seguimiento de todas las rutas" en la página 92

Nota – Para supervisar las interfaces de red, consulte "Supervisión de direcciones e interfaces IP" de *Conexión de sistemas mediante la configuración de redes fijas en Oracle Solaris 11.1*.

Supervisión del estado de la red con el comando netstat

El comando netstat genera visualizaciones que muestran el estado de la red y estadísticas de protocolo. El estado de los protocolos TCP, SCTP y los puntos finales de UDP puede visualizarse en formato de tabla. También puede visualizarse información sobre la tabla de enrutamiento e información de interfaces.

El comando netstat muestra varios tipos de datos de red, según la opción de línea de comandos que se haya seleccionado. Estas visualizaciones son sumamente útiles para administrar sistemas. A continuación se muestra la sintaxis básica del comando netstat:

netstat [-m] [-n] [-s] [-i | -r] [-f *familia_direcciones*]

En esta sección se describen las opciones que más se usan del comando netstat. Para obtener más información sobre todas las opciones de netstat, consulte la página del comando man netstat(1M).

▼ Cómo visualizar estadísticas por protocolo

La opción netstat -s muestra estadísticas de los protocolos UDP, TCP, SCTP, ICMP e IP.

Nota – Puede utilizar su cuenta de usuario de Oracle Solaris para obtener salidas del comando netstat.

- **Visualice el estado del protocolo.**

```
$ netstat -s
```

Ejemplo 5–1 Estadísticas de protocolos de red

En el ejemplo siguiente se muestra la salida del comando netstat -s. Se han truncado algunas partes. La salida puede indicar áreas en que el protocolo tiene problemas. Por ejemplo, la información estadística de ICMPv4 e ICMPv6 puede indicar dónde ha encontrado errores el protocolo ICMP.

```
RAWIP
        rawipInDatagrams    =  4701     rawipInErrors       =     0
        rawipInCksumErrs    =     0     rawipOutDatagrams   =     4
        rawipOutErrors      =     0

UDP
        udpInDatagrams      = 10091     udpInErrors         =     0
        udpOutDatagrams     = 15772     udpOutErrors        =     0

TCP     tcpRtoAlgorithm     =     4     tcpRtoMin           =   400
        tcpRtoMax           = 60000     tcpMaxConn          =    -1
```

```
        .
        .
        tcpListenDrop       =     0     tcpListenDropQ0     =     0
        tcpHalfOpenDrop     =     0     tcpOutSackRetrans   =     0

IPv4    ipForwarding        =     2     ipDefaultTTL        =   255
        ipInReceives        =300182     ipInHdrErrors       =     0
        ipInAddrErrors      =     0     ipInCksumErrs       =     0
        .
        .
        ipsecInFailed       =     0     ipInIPv6            =     0
        ipOutIPv6           =     3     ipOutSwitchIPv6     =     0

IPv6    ipv6Forwarding      =     2     ipv6DefaultHopLimit =   255
        ipv6InReceives      = 13986     ipv6InHdrErrors     =     0
        ipv6InTooBigErrors  =     0     ipv6InNoRoutes      =     0
        .
        .
        rawipInOverflows    =     0     ipv6InIPv4          =     0

       ipv6OutIPv4         =     0     ipv6OutSwitchIPv4   =     0

ICMPv4  icmpInMsgs          = 43593     icmpInErrors        =     0
        icmpInCksumErrs     =     0     icmpInUnknowns      =     0
        .
        .
        icmpInOverflows     =     0

ICMPv6  icmp6InMsgs         = 13612     icmp6InErrors       =     0
        icmp6InDestUnreachs =     0     icmp6InAdminProhibs =     0
        .
        .
        icmp6OutGroupQueries=     0     icmp6OutGroupResps  =     2
        icmp6OutGroupReds   =     0

IGMP:
      12287 messages received
          0 messages received with too few bytes
          0 messages received with bad checksum
      12287 membership queries received
SCTP  sctpRtoAlgorithm     =  vanj
      sctpRtoMin           =  1000
      sctpRtoMax           = 60000
      sctpRtoInitial       =  3000
      sctpTimHearBeatProbe =     2
      sctpTimHearBeatDrop  =     0
      sctpListenDrop       =     0
      sctpInClosed         =     0
```

▼ Cómo visualizar el estado de protocolos de transporte

El comando `netstat` permite visualizar información sobre el estado de los protocolos de transporte. Para obtener más información, consulte la página del comando man `netstat(1M)`.

1 Visualice el estado de los protocolos de transporte TCP y SCTP en un sistema.

```
$ netstat
```

2 Visualice el estado de un determinado protocolo de transporte en un sistema.

```
$ netstat -P transport-protocol
```

Los valores de la variable *protocolo_transporte* son `tcp`, `sctp` o `udp`.

Ejemplo 5–2 Visualización del estado de los protocolos de transporte TCP y SCTP

En este ejemplo se muestra la salida del comando `netstat` básico. Sólo se muestra información de IPv4.

```
$ netstat

TCP: IPv4
   Local Address      Remote Address    Swind Send-Q  Rwind Recv-Q      State
----------------- -------------------- ----- ------  ----- ------     -------
lhost-1.login      abc.def.local.Sun.COM.980 49640      0     49640    0 ESTABLISHED
lhost-1.login      ghi.jkl.local.Sun.COM.1020 49640     1     49640    0 ESTABLISHED
remhost-1.1014     mno.pqr.remote.Sun.COM.nfsd 49640    0     49640    0 TIME_WAIT
SCTP:
Local Address    Remote Address  Swind  Send-Q  Rwind  Recv-Q StrsI/O  State
---------------- --------------  -----  ------ ------ ------  ------   -------
 *.echo            0.0.0.0            0       0 102400      0   128/1   LISTEN
 *.discard         0.0.0.0            0       0 102400      0   128/1   LISTEN
 *.9001            0.0.0.0            0       0 102400      0   128/1   LISTEN
```

Ejemplo 5–3 Visualización del estado de un determinado protocolo de transporte

En este ejemplo se muestran los resultados que se obtienen al especificar la opción `-P` del comando `netstat`.

```
$ netstat -P tcp

TCP: IPv4
   Local Address      Remote Address    Swind Send-Q  Rwind Recv-Q      State
----------------- -------------------- ----- ------  ----- ------     -------
lhost-1.login      abc.def.local.Sun.COM.980 49640      0     49640    0 ESTABLISHED
lhost.login        ghi.jkl.local.Sun.COM.1020 49640     1     49640    0 ESTABLISHED
remhost.1014       mno.pqr.remote.Sun.COM.nfsd 49640    0     49640    0 TIME_WAIT

TCP: IPv6
 Local Address    Remote Address        Swind Send-Q Rwind Recv-Q   State If
---------------- ---------------------- ------ ----- ------ ------------ -----
localhost.38983   localhost.32777       49152      0 49152      0 ESTABLISHED
localhost.32777   localhost.38983       49152      0 49152      0 ESTABLISHED
localhost.38986   localhost.38980       49152      0 49152      0 ESTABLISHED
```

▼ Cómo visualizar el estado de interfaces de red

La opción `i` del comando `netstat` muestra el estado de las interfaces de red que se configuran en el sistema local. Esta opción permite determinar la cantidad de paquetes que transmite un sistema y que recibe cada red.

- **Visualice el estado de las interfaces de red.**

```
$ netstat -i
```

Ejemplo 5–4 Visualización del estado de las interfaces de red

En el ejemplo siguiente se muestra el estado de un flujo de paquetes IPv4 e IPv6 a través de las interfaces del host.

Por ejemplo, la cantidad de paquetes de entrada (`Ipkts`) que aparece en un servidor puede aumentar cada vez que un cliente intenta iniciar, mientras que la cantidad de paquetes de salida (`Opkts`) no se modifica. De esta salida puede inferirse que el servidor está viendo los paquetes de solicitud de inicio del cliente. Sin embargo, parece que el servidor no sabe responder. Esta confusión podría deberse a una dirección incorrecta en la base de datos `hosts` o `ethers`.

No obstante, si la cantidad de paquetes de entrada permanece invariable, el equipo no ve los paquetes. De este resultado puede inferirse otra clase de error, posiblemente un problema de hardware.

```
Name  Mtu  Net/Dest      Address        Ipkts  Ierrs Opkts  Oerrs Collis Queue
lo0   8232 loopback      localhost      142    0     142    0     0      0
net0  1500 host58        host58        1106302 0     52419  0     0      0

Name  Mtu  Net/Dest      Address                   Ipkts  Ierrs Opkts  Oerrs Collis
lo0   8252 localhost     localhost                  142    0     142    0     0
net0  1500 fe80::a00:20ff:feb9:4c54/10 fe80::a00:20ff:feb9:4c54 1106305 0 52422 0  0
```

▼ Cómo visualizar el estado de los sockets

Mediante la opción `-a` del comando `netstat` se puede visualizar el estado de los sockets en el host local.

- **Escriba lo siguiente para visualizar el estado de los sockets y las entradas de tabla de enrutador:**

  Puede emplear su cuenta de usuario para ejecutar esta opción de `netstat`.

```
% netstat -a
```

Ejemplo 5–5 Visualización de todos los sockets y las entradas de tabla de enrutador

La salida del comando `netstat -a` muestra estadísticas exhaustivas. En el ejemplo siguiente se muestran partes de una salida típica de `netstat -a`.

```
UDP: IPv4
   Local Address         Remote Address     State
-------------------- -------------------- -------
      *.bootpc                            Idle
host85.bootpc                             Idle
      *.*                                 Unbound
```

```
      *.*                                Unbound
      *.sunrpc                           Idle
      *.*                                Unbound
      *.32771                            Idle
      *.sunrpc                           Idle
      *.*                                Unbound
      *.32775                            Idle
      *.time                             Idle
       .
       .
      *.daytime                          Idle
      *.echo                             Idle
      *.discard                          Idle

UDP: IPv6
   Local Address                 Remote Address                  State      If
-------------------------------- -------------------------------- ---------- -----
      *.*                                                        Unbound
      *.*                                                        Unbound
      *.sunrpc                                                   Idle
      *.*                                                        Unbound
      *.32771                                                    Idle
      *.32778                                                    Idle
      *.syslog                                                   Idle
      .
      .
TCP: IPv4
   Local Address        Remote Address    Swind Send-Q Rwind Recv-Q  State
-------------------- -------------------- ----- ------ ----- ------ -------
      *.*                  *.*               0      0 49152      0 IDLE
localhost.4999             *.*               0      0 49152      0 LISTEN
      *.sunrpc             *.*               0      0 49152      0 LISTEN
      *.*                  *.*               0      0 49152      0 IDLE
      *.sunrpc             *.*               0      0 49152      0 LISTEN
      .
      .
      *.printer            *.*               0      0 49152      0 LISTEN
      *.time               *.*               0      0 49152      0 LISTEN
      *.daytime            *.*               0      0 49152      0 LISTEN
      *.echo               *.*               0      0 49152      0 LISTEN
      *.discard            *.*               0      0 49152      0 LISTEN
      *.chargen            *.*               0      0 49152      0 LISTEN
      *.shell              *.*               0      0 49152      0 LISTEN
      *.shell              *.*               0      0 49152      0 LISTEN
      *.kshell             *.*               0      0 49152      0 LISTEN
      *.login
       .
       .
           *.*               0      0 49152      0 LISTEN
   *TCP: IPv6
 Local Address            Remote Address        Swind Send-Q Rwind Recv-Q   State If
----------------------- ----------------------- ----- ------ ----- ------    ----
   *.*                      *.*                   0      0 49152      0      IDLE
   *.sunrpc                 *.*                   0      0 49152      0      LISTEN
   *.*                      *.*                   0      0 49152      0      IDLE
   *.32774                  *.*                   0      0 49152
```

▼ Cómo visualizar el estado de las transmisiones de paquetes de un determinado tipo de dirección

Utilice la opción -f del comando netstat para ver estadísticas relacionadas con transmisiones de paquetes de una determinada familia de direcciones.

- **Visualice estadísticas de transmisiones de paquetes de IPv4 o IPv6.**

  ```
  $ netstat -f inet | inet6
  ```

  Para ver información sobre transmisiones de IPv4, escriba inet como argumento de netstat -f. Utilice inet6 como argumento de netstat -f para ver información de IPv6.

Ejemplo 5–6 Estado de transmisión de paquetes de IPv4

En el ejemplo siguiente se muestra la salida del comando netstat -f inet.

```
TCP: IPv4
   Local Address        Remote Address    Swind Send-Q Rwind Recv-Q  State
-------------------- -------------------- ----- ------ ----- ------ -------
host58.734         host19.nfsd       49640      0 49640      0 ESTABLISHED
host58.38063       host19.32782      49640      0 49640      0 CLOSE_WAIT
host58.38146       host41.43601      49640      0 49640      0 ESTABLISHED
host58.996         remote-host.login 49640      0 49206      0 ESTABLISHED
```

Ejemplo 5–7 Estado de transmisión de paquetes de IPv6

En el ejemplo siguiente se muestra la salida del comando netstat -f inet6.

```
TCP: IPv6
 Local Address          Remote Address        Swind Send-Q Rwind Recv-Q   State    If
------------------ ------------------------ ----- ------ ----- ------ --------- -----
localhost.38065         localhost.32792       49152   0 49152      0    ESTABLISHED
localhost.32792         localhost.38065       49152   0 49152      0    ESTABLISHED
localhost.38089         localhost.38057       49152   0 49152      0    ESTABLISHED
```

▼ Cómo visualizar el estado de rutas conocidas

La opción -r del comando netstat muestra la tabla de rutas del host local. En esta tabla se muestra el estado de todas las rutas de las que el host tiene conocimiento. Esta opción de netstat puede ejecutarse desde la cuenta de usuario.

- **Visualice la tabla de rutas IP.**

  ```
  $ netstat -r
  ```

Ejemplo 5–8 Salida de tabla de rutas con el comando netstat

En el ejemplo siguiente se muestra la salida del comando netstat -r.

```
Routing Table: IPv4
  Destination           Gateway           Flags  Ref   Use   Interface
-------------------- -------------------- ----- ----- ------ ---------
host15               myhost               U         1  31059  net0
10.0.0.14            myhost               U         1      0  net0
default              distantrouter        UG        1      2  net0
localhost            localhost            UH        42019361  lo0

Routing Table: IPv6
  Destination/Mask      Gateway                           Flags Ref  Use   If
--------------------- --------------------------          ----- --- ------ -----
2002:0a00:3010:2::/64 2002:0a00:3010:2:1b2b:3c4c:5e6e:abcd U     1   0      net0:1
fe80::/10             fe80::1a2b:3c4d:5e6f:12a2           U     1   23     net0
ff00::/8              fe80::1a2b:3c4d:5e6f:12a2           U     1   0      net0
default               fe80::1a2b:3c4d:5e6f:12a2           UG    1   0      net0
localhost             localhost                           UH    9   21832  lo0
```

La tabla siguiente describe el significado de los distintos parámetros de salida de pantalla del comando `netstat -r`.

Parámetro	Descripción
`Destination` `Destination/Mask`	Indica el host que es el punto final de destino de la ruta. La tabla de ruta IPv6 muestra el prefijo de un punto final de túnel 6to4 (`2002:0a00:3010:2::/64`) como punto final de destino de la ruta.
`Gateway`	Especifica el portal que se usa para enviar paquetes.
`Flags`	Indica el estado actual de la ruta. El indicador `U` especifica que la ruta está activa. El indicador `G` especifica que la ruta es a un portal.
`Use`	Muestra la cantidad de paquetes enviados.
`Interface`	Indica la interfaz concreta del host local que es el punto final de origen de la transmisión.

Sondeo de hosts remotos con el comando `ping`

El comando `ping` se usa para determinar el estado de un host remoto. Al ejecutar el comando `ping`, el protocolo ICMP envía al host un determinado datagrama para solicitar una respuesta. El protocolo ICMP se ocupa de los errores en las redes TCP/IP. Al utilizar `ping`, se puede saber si el host remoto dispone de conexión IP.

A continuación se muestra la sintaxis básica del comando `ping`:

`/usr/sbin/ping` *host [timeout]*

En esta sintaxis, *host* corresponde al nombre del host remoto. El argumento *tiempo_espera* opcional indica el tiempo en segundos para que el comando `ping` siga intentando contactar con el host remoto. El valor predeterminado es de 20 segundos. Para obtener más información sobre sintaxis y opciones, consulte la página del comando man `ping(1M)`.

▼ Cómo determinar si un host remoto está en ejecución

- **Escriba la forma siguiente del comando `ping`:**

  `$ ping` *hostname*

  Si el host *nombre_host* acepta transmisiones ICMP, se muestra el mensaje siguiente:

  hostname `is alive`

  Este mensaje indica que *nombre_host* ha respondido a la solicitud de ICMP. Sin embargo, si *nombre_host* está desconectado o no puede recibir los paquetes de ICMP, el comando `ping` genera la respuesta siguiente:

  `no answer from` *hostname*

▼ Cómo determinar si un host descarta paquetes

Utilice la opción `-s` del comando `ping` para determinar si un host remoto está en ejecución y por otro lado pierde paquetes.

- **Escriba la forma siguiente del comando `ping`:**

  `$ ping -s` *hostname*

Ejemplo 5–9 Salida de `ping` para la detección de paquetes descartados

El comando `ping -s` *nombre_host* envía constantemente paquetes al host especificado hasta que se envía un carácter de interrupción o finaliza el tiempo de espera. Las respuestas que aparecen en pantalla tienen un aspecto parecido al siguiente:

```
& ping -s host1.domain8
PING host1.domain8 : 56 data bytes
64 bytes from host1.domain8.COM (172.16.83.64): icmp_seq=0. time=1.67 ms
64 bytes from host1.domain8.COM (172.16.83.64): icmp_seq=1. time=1.02 ms
64 bytes from host1.domain8.COM (172.16.83.64): icmp_seq=2. time=0.986 ms
64 bytes from host1.domain8.COM (172.16.83.64): icmp_seq=3. time=0.921 ms
64 bytes from host1.domain8.COM (172.16.83.64): icmp_seq=4. time=1.16 ms
64 bytes from host1.domain8.COM (172.16.83.64): icmp_seq=5. time=1.00 ms
64 bytes from host1.domain8.COM (172.16.83.64): icmp_seq=5. time=1.980 ms

^C

----host1.domain8  PING Statistics----
7 packets transmitted, 7 packets received, 0% packet loss
round-trip (ms)  min/avg/max/stddev = 0.921/1.11/1.67/0.26
```

La estadística de pérdida de paquetes indica si el host ha descartado paquetes. Si falla el comando `ping`, compruebe el estado de la red que indican los comandos `ipadm` y `netstat`.

Consulte "Supervisión de direcciones e interfaces IP" de *Conexión de sistemas mediante la configuración de redes fijas en Oracle Solaris 11.1* y "Supervisión del estado de la red con el comando `netstat`" en la página 81.

Administración y registro de la visualización del estado de la red

Las tareas siguientes enseñan a comprobar el estado de la red mediante comandos de red perfectamente conocidos.

▼ Cómo controlar la salida de visualización de comandos relacionados con IP

Puede controlar la salida del comando `netstat` para visualizar información de IPv4 únicamente, o información de IPv4 y de IPv6.

1. **Cree el archivo `/etc/default/inet_type`.**

2. **Agregue una de las entradas siguientes a `/etc/default/inet_type`, según lo que necesite la red:**

   - Para visualizar únicamente información de IPv4:

     ```
     DEFAULT_IP=IP_VERSION4
     ```

   - Para visualizar información de IPv4 e IPv6:

     ```
     DEFAULT_IP=BOTH
     ```

     o

     ```
     DEFAULT_IP=IP_VERSION6
     ```

     Para obtener más información acerca del archivo `inet_type`, consulte la página del comando man `inet_type(4)`.

Nota – El indicador `-f` del comando `netstat` sustituye los valores establecidos en el archivo `inet_type`.

Ejemplo 5–10 Control de la salida para seleccionar información de IPv4 e IPv6

- Si especifica la variable `DEFAULT_IP=BOTH` o `DEFAULT_IP=IP_VERSION6` en el archivo `inet_type`, en principio debe obtenerse la salida siguiente:

```
% ipadm show-addr
ADDROBJ       TYPE       STATE   ADDR
lo0/v4        static     ok      127.0.0.1/8
net0/v4       static     ok      10.46.86.54/24
lo0/v6        static     ok      ::1/128
net0/v6       addrconf   ok      fe80::a00:fe73:56a8/10
net0/v6add    static     ok      2001:db8:3c4d:5:a00:fe73:56a8/64
```

- Si se especifica la variable `DEFAULT_IP=IP_VERSION4` en el archivo `inet_type`, debe obtener el siguiente resultado:

```
% ipadm show-addr
ADDROBJ       TYPE       STATE   ADDR
lo0/v4        static     ok      127.0.0.1/8
net0/v4       static     ok      10.46.86.54/24
```

▼ Cómo registrar acciones del daemon de rutas de IPv4

Si tiene la impresión de que el comando `routed`, daemon de rutas de IPv4, funciona de modo incorrecto, inicie un registro que efectúe el seguimiento de la actividad del daemon. El registro incluye todas las transferencias de paquetes al iniciarse el daemon `routed`.

- **Cree un archivo de registro de acciones de daemon de enrutamiento:**

```
# /usr/sbin/in.routed /var/log-file-name
```

Precaución – En una red que esté ocupada, este comando puede generar salida casi continua.

Ejemplo 5–11 Registro de red del daemon `in.routed`

En el ejemplo siguiente se muestra el comienzo del archivo de registro que se crea mediante el procedimiento "Cómo registrar acciones del daemon de rutas de IPv4" en la página 90.

```
-- 2003/11/18 16:47:00.000000 --
Tracing actions started
RCVBUF=61440
Add interface lo0  #1   127.0.0.1      -->127.0.0.1/32
   <UP|LOOPBACK|RUNNING|MULTICAST|IPv4> <PASSIVE>
Add interface net0 #2   10.10.48.112    -->10.10.48.0/25
    <UP|BROADCAST|RUNNING|MULTICAST|IPv4>
turn on RIP
Add    10.0.0.0        -->10.10.48.112      metric=0  net0  <NET_SYN>
Add    10.10.48.85/25  -->10.10.48.112      metric=0  net0  <IF|NOPROP>
```

▼ Cómo efectuar el seguimiento de las actividades del daemon de descubrimiento cercano de IPv6

Si tiene la impresión de que el daemon `in.ndpd` funciona de modo incorrecto, inicie un registro que efectúe el seguimiento de la actividad del daemon. Dicho seguimiento se refleja en la salida estándar hasta su conclusión. En el seguimiento figuran todas las transferencias de paquetes al iniciarse el daemon `in.ndpd`.

1. **Inicie el seguimiento del daemon `in.ndpd`.**

   ```
   # /usr/lib/inet/in.ndpd -t
   ```

2. **Concluya el seguimiento a su conveniencia. Para ello, pulse las teclas Control+C.**

Ejemplo 5–12 Seguimiento del daemon `in.ndpd`

En la salida siguiente se muestra el inicio de un seguimiento del daemon `in.ndpd`.

```
# /usr/lib/inet/in.ndpd -t
Nov 18 17:27:28 Sending solicitation to  ff02::2 (16 bytes) on net0
Nov 18 17:27:28         Source LLA: len 6 <08:00:20:b9:4c:54>
Nov 18 17:27:28 Received valid advert from fe80::a00:20ff:fee9:2d27 (88 bytes) on net0
Nov 18 17:27:28         Max hop limit: 0
Nov 18 17:27:28         Managed address configuration: Not set
Nov 18 17:27:28         Other configuration flag: Not set
Nov 18 17:27:28         Router lifetime: 1800
Nov 18 17:27:28         Reachable timer: 0
Nov 18 17:27:28         Reachable retrans timer: 0
Nov 18 17:27:28         Source LLA: len 6 <08:00:20:e9:2d:27>
Nov 18 17:27:28         Prefix: 2001:08db:3c4d:1::/64
Nov 18 17:27:28                 On link flag:Set
Nov 18 17:27:28                 Auto addrconf flag:Set
Nov 18 17:27:28                 Valid time: 2592000
Nov 18 17:27:28                 Preferred time: 604800
Nov 18 17:27:28         Prefix: 2002:0a00:3010:2::/64
Nov 18 17:27:28                 On link flag:Set
Nov 18 17:27:28                 Auto addrconf flag:Set
Nov 18 17:27:28                 Valid time: 2592000
Nov 18 17:27:28                 Preferred time: 604800
```

Visualización de información de enrutamiento con el comando `traceroute`

El comando `traceroute` efectúa el seguimiento de la ruta que sigue un paquete de IP en dirección a un sistema remoto. Para obtener más información sobre `traceroute`, consulte la página del comando man `traceroute(1M)`.

El comando `traceroute` se usa para descubrir cualquier error de configuración de enrutamiento y errores de ruta de enrutamiento. Si no se puede conectar con un determinado

host, el comando traceroute sirve para comprobar la ruta que sigue el paquete hasta el host remoto y detectar los errores que pudiera haber.

Asimismo, el comando traceroute muestra el tiempo de ida y vuelta en cada portal de la ruta del host de destino. Esta información resulta útil para analizar dónde hay tráfico lento entre dos host.

▼ Cómo saber la ruta de un host remoto

- **Para descubrir la ruta de un sistema remoto, escriba lo siguiente:**

  `% traceroute` *destination-hostname*

  Esta forma del comando traceroute se puede ejecutar desde la cuenta de usuario.

Ejemplo 5–13 Uso del comando traceroute para mostrar la ruta de un host remoto

La salida siguiente del comando traceroute muestra la ruta de siete saltos de un paquete que va del sistema local nearhost al sistema remoto farhost. También muestra los intervalos de tiempo que emplea el paquete en atravesar cada salto.

```
istanbul% traceroute farhost.faraway.com
    traceroute to farhost.faraway.com (172.16.64.39), 30 hops max, 40 byte packets
     1  frbldg7c-86 (172.16.86.1)  1.516 ms  1.283 ms  1.362 ms
     2  bldg1a-001 (172.16.1.211)  2.277 ms  1.773 ms  2.186 ms
     3  bldg4-bldg1 (172.16.4.42)  1.978 ms  1.986 ms  13.996 ms
     4  bldg6-bldg4 (172.16.4.49)  2.655 ms  3.042 ms  2.344 ms
     5  ferbldg11a-001 (172.16.1.236)  2.636 ms  3.432 ms  3.830 ms
     6  frbldg12b-153 (172.16.153.72)  3.452 ms  3.146 ms  2.962 ms
     7  sanfrancisco (172.16.64.39)  3.430 ms  3.312 ms  3.451 ms
```

▼ Cómo efectuar el seguimiento de todas las rutas

Este procedimiento emplea la opción -a del comando traceroute para realizar el seguimiento de todas las rutas.

- **Escriba el comando siguiente en el sistema local:**

  `% traceroute -a`*host-name*

  Esta forma del comando traceroute se puede ejecutar desde la cuenta de usuario.

Ejemplo 5–14 Seguimiento de todas las rutas de un host de doble pila

En este ejemplo figuran todas las rutas de un host de doble pila.

```
% traceroute -a v6host.remote.com
traceroute: Warning: Multiple interfaces found; using 2::56:a0:a8 @ eri0:2
traceroute to v6host (2001:db8:4a3b::102:a00:fe79:19b0),30 hops max, 60 byte packets
```

```
 1  v6-rout86 (2001:db8:4a3b:56:a00:fe1f:59a1)  35.534 ms  56.998 ms *
 2  2001:db8::255:0:c0a8:717  32.659 ms  39.444 ms *
 3  farhost.faraway.COM (2001:db8:4a3b::103:a00:fe9a:ce7b)  401.518 ms  7.143 ms *
 4  distant.remote.com (2001:db8:4a3b::100:a00:fe7c:cf35)  113.034 ms  7.949 ms *
 5  v6host (2001:db8:4a3b::102:a00:fe79:19b0)  66.111 ms *  36.965 ms

traceroute to v6host.remote.com  (192.168.10.75),30 hops max,40 byte packets
 1  v6-rout86 (172.16.86.1)  4.360 ms  3.452 ms  3.479 ms
 2  flrmpj17u.here.COM (172.16.17.131)  4.062 ms  3.848 ms  3.505 ms
 3  farhost.farway.com (10.0.0.23)  4.773 ms *  4.294 ms
 4  distant.remote.com (192.168.10.104)  5.128 ms  5.362 ms *
 5  v6host  (192.168.15.85)  7.298 ms  5.444 ms *
```

Control de transferencias de paquetes con el comando snoop

El comando snoop es apto para supervisar el estado de las transferencias de datos. El comando snoop captura paquetes de red y muestra su contenido en el formato que se especifica. Los paquetes se pueden visualizar nada más recibirse o se pueden guardar en un archivo. Si el comando snoop escribe en un archivo intermedio, es improbable que haya pérdidas de paquete en situaciones de seguimiento ocupado. El propio comando snoop se utiliza para interpretar el archivo.

Para capturar paquetes en y desde la interfaz predeterminada en modo promiscuo, se debe adquirir la función de administración de redes o convertirse en superusuario. En el formato resumido, snoop sólo muestra los datos relativos al protocolo de nivel más alto. Por ejemplo, un paquete de NFS muestra únicamente información de NFS. Se suprime la información subyacente de RPC, UDP, IP y Ethernet; sin embargo, se puede visualizar en caso de elegir cualquiera de las opciones detalladas.

Utilice el comando snoop con frecuencia y buen criterio para familiarizarse con el comportamiento normal del sistema. Para obtener asistencia en el análisis de paquetes, busque documentación técnica reciente y funciones de petición de comentarios; asimismo, solicite el consejo de un experto en un ámbito determinado, por ejemplo NFS o NIS. Para obtener más información sobre el comando snoop y sus opciones, consulte la página del comando man snoop(1M).

▼ Cómo comprobar paquetes de todas las interfaces

1. **Imprima la información relativa a las interfaces conectadas al sistema.**

   ```
   # ipadm show-if
   ```

   El comando snoop suele utilizar el primer dispositivo que no es de bucle de retorno, en general la interfaz de red principal.

2. **Comience a capturar paquetes escribiendo el comando snoop sin argumentos, como se muestra en el Ejemplo 5–15.**

3 Para detener el proceso, pulse Control+C.

Ejemplo 5–15 Salida del comando snoop

La salida básica que genera el comando `snoop` se parece a la siguiente en el caso de un host de doble pila.

```
% snoop
Using device /dev/net (promiscuous mode)
router5.local.com -> router5.local.com ARP R 10.0.0.13, router5.local.com is
    0:10:7b:31:37:80
router5.local.com -> BROADCAST     TFTP Read "network-confg" (octet)
myhost -> DNSserver.local.com      DNS C 192.168.10.10.in-addr.arpa. Internet PTR ?
DNSserver.local.com  myhost        DNS R 192.168.10.10.in-addr.arpa. Internet PTR
   niserve2.
.
.
.
fe80::a00:20ff:febb:e09 -> ff02::9 RIPng R (5 destinations)
```

Los paquetes que se capturan en esta salida muestran una sección de inicio de sesión remoto, incluidas las búsquedas en los servidores NIS y DNS para resolver direcciones. También se incluyen paquetes ARP periódicos del enrutador local y anuncios de la dirección local de vínculos IPv6 en el comando `in.ripngd`.

▼ Cómo capturar la salida del comando `snoop` en un archivo

1 Capture una sesión de `snoop` en un archivo.

`# snoop -o` *filename*

Por ejemplo:

```
# snoop -o /tmp/cap
Using device /dev/eri (promiscuous mode)
30 snoop: 30 packets captured
```

En el ejemplo, se han capturado 30 paquetes en un archivo que se denomina `/tmp/cap`. El archivo se puede ubicar en cualquier directorio que disponga de suficiente espacio en disco. La cantidad de paquetes capturados se muestra en la línea de comandos, y permite pulsar Control+C para cancelar en cualquier momento.

El comando `snoop` crea una evidente carga de red en el equipo host que puede distorsionar el resultado. Para ver el resultado real, `snoop` debe ejecutarse desde otro sistema.

2 Inspeccione el archivo de capturas de la salida del comando `snoop`.

`# snoop -i` *filename*

Ejemplo 5–16 Contenido de un archivo de capturas de la salida del comando snoop

La salida siguiente muestra distintas capturas que se pueden recibir como salida del comando `snoop -i`.

```
# snoop -i /tmp/cap
1   0.00000 fe80::a00:20ff:fee9:2d27 -> fe80::a00:20ff:fecd:4375
    ICMPv6 Neighbor advertisement
...
10  0.91493    10.0.0.40 -> (broadcast)  ARP C Who is 10.0.0.40, 10.0.0.40 ?
34  0.43690 nearserver.here.com  -> 224.0.1.1  IP  D=224.0.1.1 S=10.0.0.40 LEN=28,
      ID=47453, TO =0x0, TTL=1
35  0.00034  10.0.0.40 -> 224.0.1.1    IP  D=224.0.1.1 S=10.0.0.40 LEN=28, ID=57376,
     TOS=0x0, TTL=47
```

▼ Cómo comprobar paquetes entre un cliente y un servidor IPv4

1. **Establezca un sistema snoop fuera de un concentrador conectado al cliente o al servidor.**

   El tercer sistema (sistema snoop) comprueba todo el tráfico involucrado, de manera que el seguimiento de snoop refleja lo que sucede realmente en la conexión.

2. **Escriba el comando snoop con opciones y guarde la salida que se genere en un archivo.**

3. **Inspeccione e interprete la salida.**

   Consulte RFC 1761, Snoop Version 2 Packet Capture File Format (`http://www.ietf.org/rfc/rfc1761.txt?number=1761`) para obtener más información sobre el archivo de capturas del comando snoop.

▼ Cómo supervisar tráfico de redes IPv6

El comando snoop puede utilizarse para supervisar únicamente paquetes de IPv6.

- **Capture paquetes de IPv6.**

  ```
  # snoop ip6
  ```

  Para obtener más información sobre el comando snoop, consulte la página del comando man `snoop(1M)`.

Ejemplo 5–17 Visualización sólo de tráfico de redes IPv6

En el ejemplo siguiente se muestra una salida típica que puede recibirse tras ejecutar el comando `snoop ip6` en un nodo.

```
# snoop ip6
fe80::a00:20ff:fecd:4374 -> ff02::1:ffe9:2d27 ICMPv6 Neighbor solicitation
fe80::a00:20ff:fee9:2d27 -> fe80::a00:20ff:fecd:4375 ICMPv6 Neighbor
      solicitation
fe80::a00:20ff:fee9:2d27 -> fe80::a00:20ff:fecd:4375 ICMPv6 Neighbor
      solicitation
fe80::a00:20ff:febb:e09 -> ff02::9      RIPng R (11 destinations)
fe80::a00:20ff:fee9:2d27 -> ff02::1:ffcd:4375 ICMPv6 Neighbor solicitation
```

Supervisión de paquetes mediante dispositivos de capa IP

Los dispositivos de capa IP se agregan en Oracle Solaris para mejorar la observabilidad IP. Estos dispositivos ofrecen acceso a todos los paquetes con direcciones que están asociadas con la interfaz de red del sistema. Las direcciones incluyen direcciones locales y direcciones que están alojadas en interfaces que no son de bucle de retorno o interfaces lógicas. El tráfico observable puede incluir tanto direcciones IPv4 como direcciones IPv6. Por lo tanto, se puede supervisar todo el tráfico destinado al sistema. El tráfico puede incluir tráfico IP en bucle de retorno, paquetes de máquinas remotas, paquetes que se envían desde el sistema o todo el tráfico reenviado.

Con los dispositivos de capa IP, un administrador de una zona global puede supervisar el tráfico entre zonas y dentro de una zona. Un administrador de una zona no global también puede observar el tráfico que envía y recibe esa zona.

Para supervisar el tráfico en la capa IP, se agrega una nueva opción, `-I`, al comando `snoop`. Esta opción especifica que el comando debe utilizar los dispositivos de capa IP nuevos, en lugar del dispositivo subyacente de capa de enlace, para visualizar los datos de tráfico.

▼ Cómo comprobar paquetes en la capa IP

1 Si es necesario, imprima la información relativa a las interfaces conectadas al sistema.

```
# ipadm show-if
```

2 Capture el tráfico IP en una interfaz específica.

```
# snoop -I interface [-V | -v]
```

Ejemplos de comprobación de paquetes

Todos los ejemplos se basan en la siguiente configuración del sistema:

```
# ipadm show-addr
ADDROBJ      TYPE       STATE   ADDR
lo0/v4       static     ok      127.0.0.1/8
net0/v4      static     ok      192.68.25.5/24
lo0/?        static     ok      127.0.0.1/8
```

```
net0/?       static     ok      172.0.0.3/24
net0/?       static     ok      172.0.0.1/24
lo0/?        static     ok      127.0.0.1/8
```

Suponga que dos zonas, `sandbox` y `toybox`, están utilizando las siguientes direcciones IP:

- `sandbox`: `172.0.0.3`
- `toybox`: `172.0.0.1`

Puede emitir el comando `snoop -I` en las distintas interfaces del sistema. La información de paquetes que se visualiza depende de si usted es administrador de la zona global o de la zona no global.

EJEMPLO 5–18 Tráfico en la interfaz en bucle de retorno

```
# snoop -I lo0
Using device ipnet/lo0 (promiscuous mode)
   localhost -> localhost    ICMP Echo request (ID: 5550 Sequence number: 0)
   localhost -> localhost    ICMP Echo reply (ID: 5550 Sequence number: 0)
```

Para generar una salida detallada, utilice la opción `-v`.

```
# snoop -v -I lo0
Using device ipnet/lo0 (promiscuous mode)
IPNET:  ----- IPNET Header -----
IPNET:
IPNET:  Packet 1 arrived at 10:40:33.68506
IPNET:  Packet size = 108 bytes
IPNET:  dli_version = 1
IPNET:  dli_type = 4
IPNET:  dli_srczone = 0
IPNET:  dli_dstzone = 0
IPNET:
IP:   ----- IP Header -----
IP:
IP:   Version = 4
IP:   Header length = 20 bytes
...
```

La compatibilidad para la observación de paquetes en la capa IP implementa un encabezado ipnet nuevo que precede a los paquetes que se están observando. Se indican los ID de origen y de destino. El ID '0' indica que el tráfico se genera en la zona global.

EJEMPLO 5–19 Flujo de paquetes en el dispositivo net0 en las zonas locales

```
# snoop -I net0
Using device ipnet/net0 (promiscuous mode)
toybox -> sandbox TCP D=22 S=62117 Syn Seq=195630514 Len=0 Win=49152 Options=<mss
sandbox -> toybox TCP D=62117 S=22 Syn Ack=195630515 Seq=195794440 Len=0 Win=49152
toybox -> sandbox TCP D=22 S=62117 Ack=195794441 Seq=195630515 Len=0 Win=49152
sandbox -> toybox TCP D=62117 S=22 Push Ack=195630515 Seq=195794441 Len=20 Win=491
```

EJEMPLO 5–19 Flujo de paquetes en el dispositivo net0 en las zonas locales *(Continuación)*

La salida muestra el tráfico que se produce en las distintas zonas dentro del sistema. Puede ver todos los paquetes que están asociados con las direcciones IP `net0`, incluidos los paquetes que se transfieren localmente a otras zonas. Si genera una salida detallada, puede ver las zonas que forman parte del flujo de paquetes.

```
# snoop -I net0 -v port 22
IPNET:  ----- IPNET Header -----
IPNET:
IPNET:  Packet 5 arrived at 15:16:50.85262
IPNET:  Packet size = 64 bytes
IPNET:  dli_version = 1
IPNET:  dli_type = 0
IPNET:  dli_srczone = 0
IPNET:  dli_dstzone = 1
IPNET:
IP:   ----- IP Header -----
IP:
IP:   Version = 4
IP:   Header length = 20 bytes
IP:   Type of service = 0x00
IP:         xxx. .... = 0 (precedence)
IP:         ...0 .... = normal delay
IP:         .... 0... = normal throughput
IP:         .... .0.. = normal reliability
IP:         .... ..0. = not ECN capable transport
IP:         .... ...0 = no ECN congestion experienced
IP:   Total length = 40 bytes
IP:   Identification = 22629
IP:   Flags = 0x4
IP:         .1.. .... = do not fragment
IP:         ..0. .... = last fragment
IP:   Fragment offset = 0 bytes
IP:   Time to live = 64 seconds/hops
IP:   Protocol = 6 (TCP)
IP:   Header checksum = 0000
IP:   Source address = 172.0.0.1, 172.0.0.1
IP:   Destination address = 172.0.0.3, 172.0.0.3
IP:   No options
IP:
TCP:  ----- TCP Header -----
TCP:
TCP:  Source port = 46919
TCP:  Destination port = 22
TCP:  Sequence number = 3295338550
TCP:  Acknowledgement number = 3295417957
TCP:  Data offset = 20 bytes
TCP:  Flags = 0x10
TCP:        0... .... = No ECN congestion window reduced
TCP:        .0.. .... = No ECN echo
TCP:        ..0. .... = No urgent pointer
TCP:        ...1 .... = Acknowledgement
TCP         .... 0... = No push
TCP         .... .0.. = No reset
TCP:        .... ..0. = No Syn
```

EJEMPLO 5–19 Flujo de paquetes en el dispositivo net0 en las zonas locales *(Continuación)*

```
TCP:        .... ...0 = No Fin
TCP:  Window = 49152
TCP:  Checksum = 0x0014
TCP:  Urgent pointer = 0
TCP:  No options
TCP:
```

El encabezado ipnet indica que el paquete proviene de la zonal global (ID 0) y se dirige a Sandbox (ID 1).

EJEMPLO 5–20 Observación del tráfico mediante la identificación de la zona

```
# snoop -I hme0 sandboxsnoop -I net0 sandbox
Using device ipnet/hme0 (promiscuous mode)
toybox -> sandbox TCP D=22 S=61658 Syn Seq=374055417 Len=0 Win=49152 Options=<mss
sandbox -> toybox TCP D=61658 S=22 Syn Ack=374055418 Seq=374124525 Len=0 Win=49152
toybox -> sandbox TCP D=22 S=61658 Ack=374124526 Seq=374055418 Len=0 Win=49152
#
```

La capacidad de observar paquetes identificando la zona es útil en sistemas que tienen varias zonas. En la actualidad, únicamente se puede identificar la zona con el ID de zona. No se admite el uso de snoop con nombres de zonas.

Administración de selección de direcciones predeterminadas

Oracle Solaris permite que una misma interfaz tenga varias direcciones IP. Por ejemplo, tecnologías como IPMP permiten la conexión de varias tarjetas de interfaz de red en la misma capa de vínculo IP. Ese vínculo puede tener una o varias direcciones IP. Además, las interfaces en sistemas compatibles con IPv6 disponen de una dirección IPv6 local de vínculo, como mínimo una dirección de enrutamiento IPv6 y una dirección IPv4 para al menos una interfaz.

Cuando el sistema inicia una transacción, una aplicación realiza una llamada al socket `getaddrinfo`. `getaddrinfo` descubre la posible dirección que está en uso en el sistema de destino. El núcleo da prioridad a esta lista a fin de buscar el destino más idóneo para el paquete. Este proceso se denomina *ordenación de direcciones de destino*. A continuación, el núcleo de Oracle Solaris selecciona el formato correspondiente para la dirección de origen, a partir de la dirección de destino más apropiada para el paquete. El proceso se denomina *selección de direcciones*. Para obtener más información sobre la ordenación de direcciones de destino, consulte la página del comando man `getaddrinfo`(3SOCKET).

Los sistemas IPv4 y de doble pila IPv4/IPv6 deben realizar una selección de direcciones predeterminadas. En la mayoría de los casos, no hace falta cambiar los mecanismos de selección de direcciones predeterminadas. Sin embargo, quizá deba cambiar la prioridad de los formatos de direcciones para poder admitir IPMP o preferir los formatos de direcciones 6to4, por ejemplo.

▼ Cómo administrar la tabla de directrices de selección de direcciones IPv6

A continuación se explica el procedimiento para modificar la tabla de directrices de selección de direcciones. Para obtener información sobre la selección de direcciones IPv6 predeterminadas, consulte Comando ipaddrsel.

Precaución – La tabla de directrices de selección de direcciones IPv6 no se debe modificar salvo por los motivos que se exponen en la tarea siguiente. Una tabla de directrices mal configurada puede ocasionar problemas en la red. Efectúe una copia de seguridad de la tabla de directrices, como en el procedimiento siguiente.

1 Revise la tabla de directrices de selección de direcciones IPv6 actual.

```
# ipaddrsel
# Prefix                  Precedence Label
::1/128                           50 Loopback
::/0                              40 Default
2002::/16                         30 6to4
::/96                             20 IPv4_Compatible
::ffff:0.0.0.0/96                 10 IPv4
```

2 Efectúe una copia de seguridad de la tabla de directrices de direcciones predeterminadas.

```
# cp /etc/inet/ipaddrsel.conf /etc/inet/ipaddrsel.conf.orig
```

3 Si desea personalizar la tabla, utilice un editor de textos en el archivo `/etc/inet/ipaddrsel.conf`.

Utilice la sintaxis siguiente para las entradas del archivo `/etc/inet/ipaddrsel`:

prefix/prefix-length precedence label [# comment]

A continuación se muestran varias de las modificaciones habituales que podría querer aplicar a la tabla de directrices:

- Asignar la máxima prioridad a las direcciones 6to4.

  ```
  2002::/16                       50 6to4
  ::1/128                         45 Loopback
  ```

  El formato de dirección 6to4 ahora tiene la prioridad más alta: 50. Bucle, que anteriormente presentaba una prioridad de 50, ahora presenta una prioridad de 45. Los demás formatos de direcciones siguen igual.

- Designar una dirección de origen concreta que se deba utilizar en las comunicaciones con una determinada dirección de destino.

  ```
  ::1/128                         50 Loopback
  2001:1111:1111::1/128           40 ClientNet
  2001:2222:2222::/48             40 ClientNet
  ::/0                            40 Default
  ```

Esta entrada en concreto es útil para los host que cuentan sólo con una interfaz física. En este caso, `2001:1111:1111::1/128` se prefiere como dirección de origen de todos los paquetes cuyo destino previsto es la red `2001:2222:2222::/48`. La prioridad 40 otorga una posición preferente a la dirección de origen `2001:1111:1111::1/128` en relación con los demás formatos de direcciones configurados para la interfaz.

- Favorecer direcciones IPv4 respecto a direcciones IPv6.

```
::ffff:0.0.0.0/96                60 IPv4
::1/128                          50 Loopback
.
.
```

El formato de IPv4 `::ffff:0.0.0.0/96` ha cambiado su prioridad predeterminada de 10 a 60, la prioridad máxima de la tabla.

4 Cargue en el núcleo la tabla de directrices modificada.

```
ipaddrsel -f /etc/inet/ipaddrsel.conf
```

5 Si la tabla de directrices modificada presenta problemas, restaure la tabla predeterminada de directrices de selección de direcciones IPv6.

```
# ipaddrsel -d
```

▼ Cómo modificar la tabla de selección de direcciones IPv6 sólo para la sesión actual

Si edita el archivo `/etc/inet/ipaddrsel.conf`, las modificaciones que efectúe se mantendrán después de cada reinicio. Si quiere aplicar las modificaciones únicamente en la sesión actual, siga este procedimiento.

1 Copie el contenido de `/etc/inet/ipaddrsel` en *nombre_archivo*; *nombre_archivo* es el archivo que haya seleccionado.

```
# cp /etc/inet/ipaddrsel filename
```

2 Modifique la tabla de directrices de *nombre_archivo* a su conveniencia.

3 Cargue en el núcleo la tabla de directrices modificada.

```
# ipaddrsel -f filename
```

El núcleo emplea la nueva tabla de directrices hasta que se vuelva a iniciar el sistema.

◆ ◆ ◆ CAPÍTULO 6

Configuración de túneles IP

En este capítulo, se presentan descripciones de túneles IP y procedimientos para configurar y mantener túneles en Oracle Solaris.

Descripción general de túneles IP

Los túneles IP proporcionan un medio para transportar paquetes de datos entre dominios cuando el protocolo en esos dominios no está admitido por redes intermediarias. Por ejemplo, con la introducción del protocolo IPv6, las redes IPv6 requieren una manera de comunicarse más allá de sus límites en un entorno donde la mayoría de las redes utilizan el protocolo IPv4. La comunicación es posible gracias al uso de túneles. El túnel IP proporciona un enlace virtual entre dos nodos a los que se puede acceder mediante IP. De esta forma, el enlace se puede utilizar para transportar paquetes IPv6 en redes IPv4 para permitir la comunicación IPv6 entre los dos sitios IPv6.

Administración de túnel IP en Oracle Solaris 11

En esta versión de Oracle Solaris, se revisó la administración de túneles para que sea coherente con el nuevo modelo de administración de enlaces de datos de red. Ahora, los túneles se crean y se configuran con nuevos subcomandos `dladm`. Los túneles ahora también pueden utilizar otras funciones de enlaces de datos del modelo de administración nuevo. Por ejemplo, la compatibilidad con nombres elegidos administrativamente permite que se asignen nombres significativos a los túneles. Para obtener más información sobre los subcomandos `dladm`, consulte la página del comando man `dladm(1M)`.

Tipos de túneles

La creación de túneles implica la encapsulación de un paquete IP dentro de otro paquete. Esta encapsulación permite que el paquete llegue a destino a través de redes intermediarias que no admiten el protocolo del paquete.

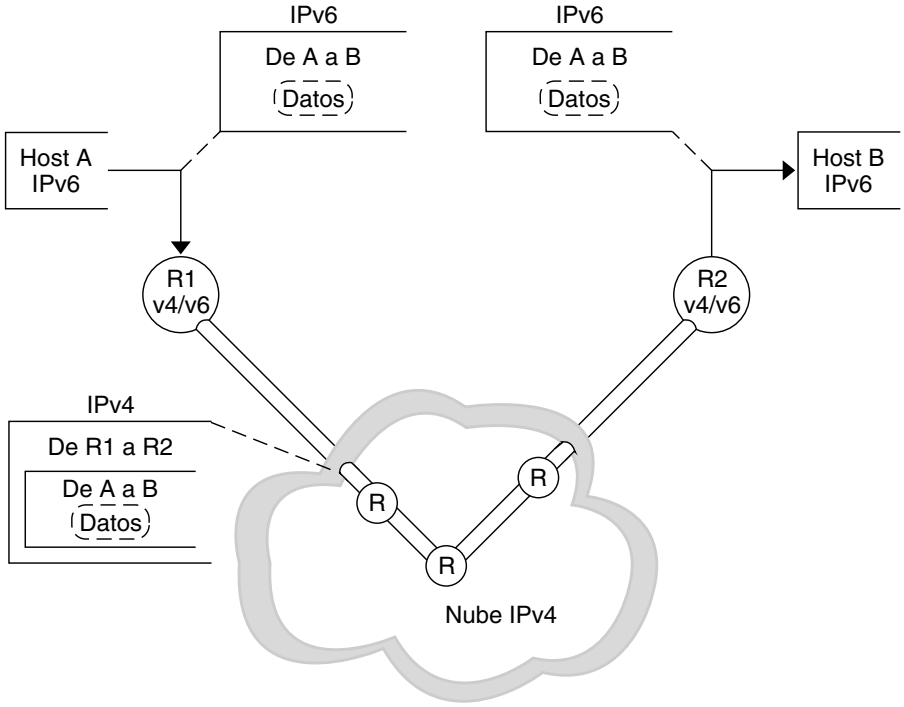
Los túneles varían según el tipo de encapsulación de paquetes. En Oracle Solaris, se admiten los siguientes tipos de paquetes:

- *Túneles IPv4*: los paquetes IPv4 o IPv6 se encapsulan en un encabezado IPv4 y se envían a un destino IPv4 de unidifusión preconfigurado. Para indicar más específicamente los paquetes que pasan por el túnel, los túneles IPv4 también se denominan *IPv4 en túneles IPv4* o *IPv6 en túneles IPv4*.
- *Túneles IPv6*: los paquetes IPv4 o IPv6 se encapsulan en un encabezado IPv6 y se envían a un destino IPv6 de unidifusión preconfigurado. Para indicar más específicamente los paquetes que pasan por el tunnel, los túneles IPv6 también se denominan *IPv4 en túneles IPv6* o *IPv6 en túneles IPv6*.
- *Túneles 6to4*: los paquetes IPv6 se encapsulan en un encabezado IPv4 y se envían a un destino IPv4 que se determina automáticamente por paquete. La determinación se basa en un algoritmo definido en el protocolo 6to4.

Túneles en los entornos de red IPv6 e IPv4 combinados

La mayoría de los sitios tienen dominios IPv6 que se comunican con otros dominios IPv6 atravesando redes IPv4, que son más prevalentes que las redes de sólo IPv6. En la figura siguiente, se ilustra el mecanismo de creación de túneles entre dos hosts IPv6 a través de enrutadores IPv4; esto se indica con una "R."

FIGURA 6–1 Mecanismo de creación de túneles IPv6



En la figura, el túnel está compuesto por dos enrutadores configurados para tener un enlace de punto a punto virtual entre los dos enrutadores en la red IPv4.

Un paquete IPv6 está encapsulado dentro de un paquete IPv4. El enrutador de límite de la red IPv6 configura un túnel de extremo a extremo a través de varias redes IPv4 hasta el enrutador de límite de la red IPv6 de destino. El paquete es transportado por el túnel hasta el enrutador de límite de destino, donde se desencapsula. A continuación, el enrutador reenvía el paquete IPv6 separado al nodo de destino.

Túneles 6to4

Oracle Solaris incluye túneles 6to4 como método provisional preferido para realizar la transición de direcciones IPv4 a IPv6. Los túneles 6to4 permiten que los sitios IPv6 aislados se comuniquen a través de un túnel automático en una red IPv4 que no admite IPv6. Para utilizar túneles 6to4 debe configurar un enrutador de límite de sistema en la red IPv6 como un punto final del túnel automático 6to4. Después, el enrutador 6to4 puede participar en un túnel hasta otra ubicación 6to4, o, si es necesario, hasta un ubicación IPv6 nativa, no 6to4.

Esta sección proporciona material de referencia sobre los siguientes temas 6to4:

- Configuración de un túnel 6to4
- Descripción del flujo de paquetes a través de un túnel 6to4
- Configuración de un túnel entre un enrutador 6to4 y un enrutador de reenvío 6to4
- Puntos que considerar antes de configurar la compatibilidad con enrutador de reenvío 6to4

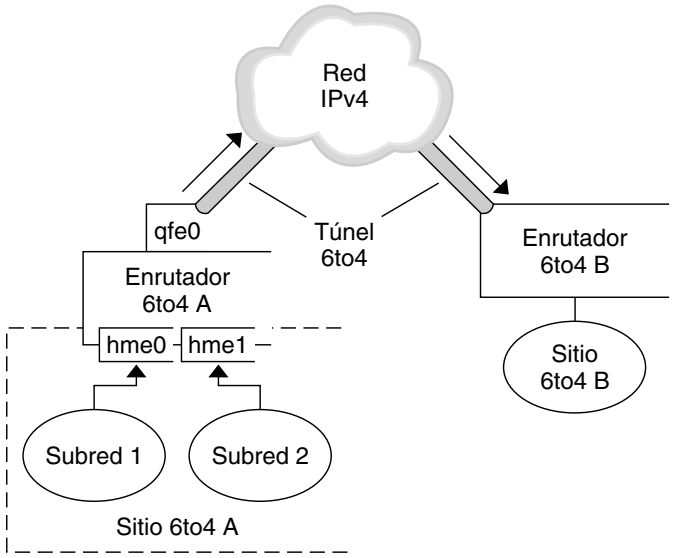
La tabla siguiente describe tareas adicionales para configurar túneles 6to4 y los recursos para obtener información adicional útil.

Tarea o detalle	Para obtener información
Tareas para configurar un túnel 6to4	"Cómo configurar un túnel 6to4" en la página 117
RCF relacionado con 6to4	RFC 3056, "Connection of IPv6 Domains via IPv4 Clouds" (`http://www.ietf.org/rfc/rfc3056.txt`)
Información detallada sobre el comando `6to4relay`, que permite utilizar túneles hasta un enrutador de reenvío 6to4	`6to4relay`(1M)
Cuestiones de seguridad de 6to4	Security Considerations for 6to4 (`http://www.ietf.org/rfc/rfc3964.txt`)

Configuración de un túnel 6to4

Un túnel 6to4 proporciona conectividad IPv6 a todas las ubicaciones 6to4 en cualquier parte. Asimismo, el túnel ejerce como vínculo con todas las ubicaciones IPv6, incluida Internet IPv6 nativa, siempre que el enrutador se configure para reenviar a un enrutador de repetición. La figura siguiente ilustra la forma en que un túnel 6to4 proporciona esta clase de conectividad entre sitios 6to4.

FIGURA 6–2 Túnel entre dos ubicaciones 6to4



En la figura, se muestran dos redes 6to4 aisladas: sitio A y sitio B. Cada sitio tiene configurado un enrutador con una conexión externa a una red IPv4. Un túnel 6to4 en la red IPv4 proporciona una conexión para vincular ubicaciones 6to4.

Antes de que una ubicación IPv6 pueda convertirse en 6to4, debe configurar al menos una interfaz de enrutador para que admite 6to4. Esta interfaz debe proporcionar la conexión externa a la red IPv4. La dirección configurada en `qfe0` debe ser única globalmente. En esta figura, la interfaz `qfe0` del enrutador de límite de sistema encaminador A conecta la ubicación de sitio A con la red IPv4. La interfaz `qfe0` ya debe estar configurada con una dirección IPv4 antes de que sea posible configurar `qfe0` como una pseudointerfaz 6to4.

En la figura, sitio A 6to4 está compuesto por dos subredes conectadas a las interfaces `hme0` y `hme1` en el enrutador A. Todos los hosts IPv6 de la subredes del sitio A se reconfiguran automáticamente con direcciones derivadas de 6to4 al recibir el anuncio del enrutador A.

La ubicación de sitio B es otra ubicación 6to4 aislada. Para recibir correctamente tráfico de la ubicación de sitio A, se debe configurar un enrutador de límite en la ubicación sitio B para admitir 6to4. De no ser así, los paquetes que recibe el enrutador de sitio A no se reconocen y se descartan.

Flujo de paquetes a través del túnel 6to4

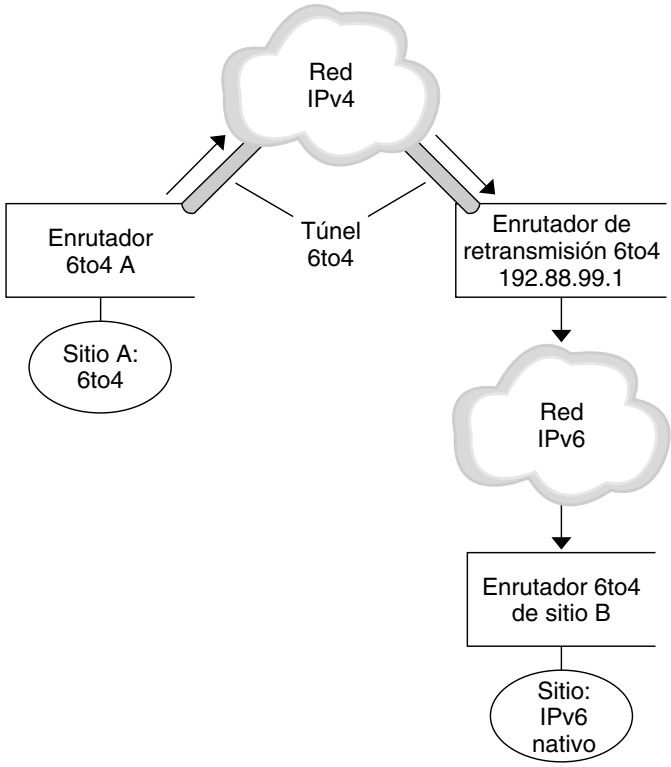
Esta sección describe el flujo de paquetes entre un hosts en una ubicación 6to4 y un host en una ubicación 6to4 remota. Esta situación hipotética utiliza la topología de la Figura 6–2. En el ejemplo se considera que los enrutadores y hosts 6to4 ya están configurados.

1. Un host en la subred 1 de la ubicación de sitio A de 6to4 envía una transmisión, con un host de la ubicación sitio B de 6to4 como destino. El encabezado de cada paquete tiene una dirección de origen derivada de 6to4 y una dirección de destino derivada de 6to4.
2. El enrutador de la ubicación sitio A encapsula cada paquete 6to4 dentro de un encabezado IPv4. En este proceso, el enrutador establece la dirección IPv4 de destino del encabezado de encapsulado en la dirección de enrutador de la ubicación de sitio B. En cada paquete de IPv6 que pasa por la interfaz de túnel, la dirección de destino de IPv6 también contiene la dirección de destino de IPv4. De este modo, el enrutador puede determinar la dirección IPv4 de destino que se establece en el encabezado de encapsulado. Después, el enrutador utiliza procedimientos estándar IPv4 para reenviar los paquetes a través de la red IPv4.
3. Cualquier enrutador IPv4 que encuentren los paquetes en su camino utilizará la dirección de destino IPv4 del paquete para reenviarlo. Esta dirección es la dirección IPv4 globalmente única de la interfaz del encaminador B, que también funciona como pseudointerfaz 6to4.
4. Los paquetes de sitio A llegan al encaminador B, que desencapsula los paquetes IPv6 del encabezado IPv4.
5. A continuación, el encaminador B utiliza la dirección de destino del paquete IPv6 para reenviar los paquetes al receptor en el sitio B.

Consideraciones para túneles hasta un enrutador de reenvío 6to4

Los enrutadores de reenvío 6to4 funcionan como puntos finales para túneles desde enrutadores 6to4 que necesitan comunicarse con redes IPv6 nativas, no 6to4. Los enrutadores de reenvío son básicamente puentes entre la ubicación 6to4 y ubicaciones IPv6 nativas. Debido a que esta solución puede llegar a ser muy insegura, Oracle Solaris no tiene la admisión de enrutadores 6to4 activada. No obstante, si es necesario establecer un túnel de este tipo en su ubicación, puede utilizar el comando `6to4relay` para activar la situación hipotética siguiente de creación de túneles.

FIGURA 6–3 Túnel desde una ubicación 6to4 hasta un enrutador de reenvío 6to4



En la Figura 6–3, el sitio A 6to4 necesita comunicarse con un nodo en el sitio B IPv6 nativo. En la figura, se muestra la ruta de tráfico del sitio A al túnel 6to4 a través de una red IPv4. Los puntos finales del túnel son el encaminador A de 6to4 y un enrutador de reenvío 6to4. Más allá del enrutador de reenvío 6to4 se encuentra la red IPv6, a la que está conectada la ubicación de sitio B IPv6.

Flujo de paquetes entre una ubicación 6to4 y una ubicación IPv6 nativa

En esta sección se describe el flujo de paquetes desde una ubicación 6to4 hasta una ubicación IPv6 nativa. Esta situación hipotética utiliza la topología de la Figura 6–3.

1. Un host en el sitio A 6to4 envía una transmisión que especifica como destino un host en el sitio B IPv6 nativo. El encabezado de cada paquete tiene una dirección derivada de 6to4 como dirección de origen. La dirección de destino es una dirección IPv6 estándar.

2. El enrutador 6to4 de la ubicación de sitio A encapsula cada paquete dentro de un encabezado IPv4, que tiene la dirección IPv4 del enrutador de reenvío 6to4 como destino. El enrutador 6to4 utiliza procedimientos IPv4 estándar para reenviar el paquete a través de la red IPv4. Cualquier enrutador IPv4 que encuentren los paquetes en su camino los reenviará al enrutador de reenvío 6to4.
3. El enrutador de reenvío 6to4 de difusión por proximidad más cercano físicamente a la ubicación de sitio A recibe los paquetes destinados al grupo de difusión por proximidad `192.88.99.1`.

Nota – Los enrutadores de reenvío 6to4 que forman parte del grupo de difusión por proximidad de enrutador de reenvío 6to4 tienen la dirección IP `192.88.99.1`. Esta dirección de difusión por proximidad es la dirección predeterminada de enrutadores de reenvío 6to4. Si necesita utilizar un enrutador de reenvío 6to4 específico, puede anular la dirección predeterminada y especificar la dirección IPv4 del enrutador.

---

4. El enrutador de reenvío desencapsula el encabezado IPv4 de los paquetes 6to4 y, de este modo, revela la dirección de destino IPv6 nativa.
5. A continuación, el enrutador de relé envía los paquetes que ahora son de sólo IPv6 a la red IPv6, donde, en última instancia, un enrutador del sitio B recupera los paquetes. Luego, el enrutador reenvía los paquetes al nodo IPv6 de destino.

Implementación de túneles

Para implementar adecuadamente los túneles IP, debe realizar dos tareas principales. Primero, debe crear el enlace de túnel. Luego, debe configurar una interfaz IP en el túnel. En esta sección, se describen brevemente los requisitos para crear túneles y sus correspondientes interfaces IP.

Requisitos para crear túneles

Para crear túneles correctamente, debe tener cumplir los siguientes requisitos:

- Si utiliza nombres de host en lugar de direcciones IP literales, estos nombres deben remitir a direcciones IP válidas compatibles con el tipo de túnel.
- El túnel IPv4 o IPv6 que cree no debe compartir la misma dirección de origen ni la misma dirección de destino con otro túnel configurado.
- El túnel IPv4 o IPv6 que cree no debe compartir la misma dirección de origen con un túnel 6to4 existente.
- Si crea un túnel 6to4, ese túnel no debe compartir la misma dirección de origen con otro túnel configurado.

Para obtener información sobre la configuración de túneles en la red, consulte "Planificación para el uso de túneles en la red" en la página 31.

Requisitos para túneles e interfaces IP

Cada tipo de túnel tiene requisitos de direcciones IP específicos en la interfaz IP que se configure en el túnel. Los requisitos se resumen en la tabla siguiente.

TABLA 6–1 Requisitos de túneles e interfaces IP

Tipo de túnel	Interfaz IP permitida en el túnel	Requisito de interfaz IP
Túnel IPv4	Interfaz IPv4	Las direcciones locales y remotas se especifican manualmente.
	Interfaz IPv6	Las direcciones locales y remotas de enlace local se configuran automáticamente al emitir el comando `ipadm create-addr -T addrconf`. Para obtener detalles, consulte la página del comando man `ipadm`(1M).
Túnel IPv6	Interfaz IPv4	Las direcciones locales y remotas se especifican manualmente.
	Interfaz IPv6	Las direcciones locales y remotas de enlace local se configuran automáticamente al emitir el comando `ipadm create-addr -T addrconf`. Para obtener detalles, consulte la página del comando man `ipadm`(1M).
Túnel 6to4	Interfaz IPv6 únicamente	La dirección IPv6 predeterminada se selecciona automáticamente al ejecutar el comando `ipadm create-ip`. Para obtener detalles, consulte la página del comando man `ipadm`(1M).

Para sustituir la dirección de interfaz IPv6 predeterminada de los túneles 6to4, puede especificar una dirección IPv6 diferente con el comando `ipadm`.

De manera similar, para sustituir las direcciones de enlace local configuradas automáticamente para las interfaces IPv6 en túneles IPv4 o IPv6, puede especificar distintas direcciones de origen y de destino en el archivo host del túnel.

Configuración y administración de túneles con el comando dladm

En esta sección, se describen los procedimientos que utiliza el comando dladm para configurar túneles.

Subcomandos dladm

A partir de esta versión de Oracle Solaris, la administración de túneles es independiente de la configuración de la interfaz IP. El aspecto de enlace de datos de los túneles IP ahora se administra con el comando dladm. Además, la configuración de la interfaz IP, incluida la interfaz de túnel IP, se realiza con el comando ipadm.

Para configurar túneles IP se utilizan los siguientes subcomandos de dladm:

- create-iptun
- modify-iptun
- show-iptun
- delete-iptun
- set-linkprop

Para obtener detalles sobre el comando dladm, consulte la página del comando man dladm(1M).

Nota – La administración de túneles IP está estrechamente relacionada con la configuración de IPsec. Por ejemplo, las redes privadas virtuales (VPN) IPsec constituyen uno de los principales usos de la creación de túneles IP. Para obtener más información sobre la seguridad en Oracle Solaris, consulte el Capítulo 6, "Arquitectura de seguridad IP (descripción general)" de *Protección de la red en Oracle Solaris 11.1*. Para configurar IPsec, consulte el Capítulo 7, "Configuración de IPsec (tareas)" de *Protección de la red en Oracle Solaris 11.1*.

Configuración de túneles (mapa de tareas)

Tarea	Descripción	Para obtener instrucciones
Crear un túnel IP.	Configure el túnel que se utilizará para las comunicaciones entre redes.	"Cómo crear y configurar un túnel IP" en la página 113
Modificar la configuración de un túnel.	Cambie los parámetros originales del túnel, como la dirección de origen o de destino del túnel.	"Cómo modificar una configuración de túnel IP" en la página 120

Tarea	Descripción	Para obtener instrucciones
Visualizar la configuración de un túnel.	Muestre la información de configuración de un túnel específico o de todos los túneles IP del sistema.	"Cómo visualizar una configuración de túnel IP" en la página 122
Suprimir un túnel.	Suprima la configuración de un túnel.	"Cómo suprimir un túnel IP" en la página 123

▼ Cómo crear y configurar un túnel IP

1 Cree el túnel.

```
# dladm create-iptun [-t] -T type -a [local|remote]=addr,... tunnel-link
```

Para este comando, están disponibles las opciones o los argumentos siguientes:

`-t` — Crea un túnel temporal. De manera predeterminada, el comando crea un túnel persistente.

Nota – Si desea configurar una interfaz IP persistente en el túnel, debe crear un túnel persistente y no utilizar la opción `-t`.

`-T` *tipo* — Especifica el tipo de túnel que desea crear. Este argumento es necesario para crear todos los tipos de túneles.

`-a [local|remote]=`*dirección,...* — Especifica los nombres de host o las direcciones IP literales que corresponden a la dirección local y a la dirección de túnel remota. Las direcciones deben ser válidas y ya deben estar creadas en el sistema. Según el tipo de túnel, debe especificar una sola dirección o ambas direcciones (locales y remotas). Si especifica direcciones locales y remotas, debe separarlas con una coma.

- Los túneles `IPv4` requieren direcciones IPv4 locales y remotas para funcionar.
- Los túneles `IPv6` requieren direcciones IPv6 locales y remotas para funcionar.
- Los túneles `6to4` requieren una dirección IPv4 local para funcionar.

Nota – Para configuraciones de enlace de datos de túneles IP, si está utilizando nombres de host para las direcciones, estos nombres de host se guardan en el almacenamiento de la configuración. Durante un inicio posterior del sistema, si el nombre remite a direcciones IP distintas de las direcciones IP utilizadas cuando se creó el túnel, el túnel adquiere una nueva configuración.

enlace_túnel

Especifica el enlace de túnel IP. Al admitir nombres significativos en una administración de enlace de red, los nombres de los túneles ya no se restringen al tipo de túnel que se está creando. En cambio, se puede asignar a un túnel cualquier nombre elegido administrativamente. Los nombres de túneles está formados por una cadena y el número de punto físico de conexión (PPA), por ejemplo, *mitúnel0*. Para conocer las reglas que rigen la asignación de nombres descriptivos, consulte "Reglas para nombres de enlaces válidos" de *Introducción a redes de Oracle Solaris 11*.

Si no especifica el enlace de túnel, el nombre se proporciona automáticamente según las convenciones de denominación siguientes:

- Para túneles IPv4: `ip.tun#`
- Para túneles IPv6: `ip6.tun#`
- Para túneles 6to4: `ip.6to4tun#`

corresponde al número de PPA más bajo disponible para el tipo de túnel que se está creando.

2 (Opcional) Configure valores para el límite de salto o el límite de encapsulación.

```
# dladm set-linkprop -p [hoplimit=value] [encaplimit=value] tunnel-link
```

hoplimit — Especifica el límite de salto de la interfaz de túnel para la creación de túneles en IPv6. El valor de *hoplimit* es equivalente al campo de tiempo de vida (TTL) de IPv4 para la creación de túneles en IPv4.

encaplimit — Especifica el número de niveles de creación de túneles anidados permitidos para un paquete. Esta opción se aplica únicamente a túneles IPv6.

Especifica el número de niveles de creación de túneles anidados permitidos para un paquete. Esta opción se aplica únicamente a túneles IPv6.

Nota – Los valores establecidos para `hoplimit` y `encaplimit` deben estar dentro de rangos aceptables. `hoplimit` y `encaplimit` son propiedades de enlace de túnel. Por lo tanto, estas propiedades se administran con los mismos subcomandos `dladm` que otras propiedades de enlace. Los subcomandos son `dladm set-linkprop`, `dladm reset-linkprop` y `dladm show-linkprop`. Consulte la página del comando man `dladm`(1M) para conocer los distintos subcomandos que se utilizan con el comando `dladm` para administrar enlaces.

3 Cree una interfaz IP en el túnel.

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

Donde *interfaz_túnel* utiliza el mismo nombre que el enlace de túnel.

4 Asigne direcciones IP locales y remotas a la interfaz de túnel.

```
# ipadm create-addr [-t] -a local=address,remote=address  interface
```

`-t`	Indica una configuración IP temporal en lugar de una configuración IP persistente en el túnel. Si no utiliza esta opción, la configuración de la interfaz IP es persistente.
`-a local=`*dirección* `,remote=`*dirección*	Especifica las direcciones IP de la interfaz de túnel. Se requieren direcciones IP de origen y de destino, representadas por `local` y `remote`. Las direcciones locales y remotas pueden ser direcciones IPv4 o IPv6.
interface	Especifica la interfaz de túnel.

Para obtener más información sobre el comando `ipadm` y las diferentes opciones para configurar las interfaces IP, incluidas las interfaces de túneles, consulte la página del comando man `ipadm`(1M) y *Conexión de sistemas mediante la configuración de redes fijas en Oracle Solaris 11.1*.

5 Agregue la información sobre la configuración del túnel al archivo `/etc/hosts`.

6 (Opcional) Verifique el estado de la configuración de la interfaz IP del túnel.

```
# ipadm show-addr interface
```

Ejemplo 6–1 Creación de una interfaz IPv6 en un túnel IPv4

En este ejemplo, se muestra cómo crear una IPv6 persistente a través de un túnel IPv4.

```
# dladm create-iptun -T ipv4 -a local=63.1.2.3,remote=192.4.5.6 private0
# dladm set-linkprop -p hoplimit=200 private0
# ipadm create-ip private0
```

```
# ipadm create-addr -T addrconf private0
# ipadm show-addr private/
ADDROBJ       TYPE     STATE   ADDR
private0/v6   static   ok      fe80::a08:392e/10 --> fe80::8191:9a56
```

Para agregar direcciones alternativas, use la misma sintaxis. Por ejemplo, puede agregar una dirección global de la siguiente manera:

```
# ipadm create-addr -a local=2001:db8:4728::1, \
remote=2001:db8:4728::2 private0
# ipadm show-addr private0/
ADDROBJ          TYPE       STATE   ADDR
private0/v6      addrconf   ok      fe80::a08:392e/10 --> fe80::8191:9a56
private0/v6a     static     ok      2001:db8:4728::1 --> 2001:db8:4728::2
```

Tenga en cuenta que el prefijo `2001:db8` para las direcciones IPv6 es un prefijo IPv6 especial que se utiliza específicamente para ejemplos de documentación.

Ejemplo 6–2 Creación de una interfaz IPv4 en un túnel IPv4

En este ejemplo, se muestra cómo crear una IPv4 persistente a través de un túnel IPv4.

```
# dladm create-iptun -T ipv4 -a local=63.1.2.3,remote=192.4.5.6 vpn0
# ipadm create-ip vpn0
# ipadm create-addr -a local=10.0.0.1,remote=10.0.0.2 vpn0
# ipadm show-addr
ADDROBJ   TYPE     STATE   ADDR
lo0/v4    static   ok      127.0.0.1
vpn0/v4   static   ok      10.0.0.1-->10.0.0.2
```

Puede configurar, además, una política IPsec para proporcionar conexiones seguras para los paquetes que pasan por este túnel. Para obtener información sobre la configuración de IPsec, consulte el Capítulo 7, "Configuración de IPsec (tareas)" de *Protección de la red en Oracle Solaris 11.1*.

Ejemplo 6–3 Creación de una interfaz IPv6 en un túnel IPv6

En este ejemplo, se muestra cómo crear una IPv6 persistente a través de un túnel IPv6.

```
# dladm create-iptun -T ipv6 -a local=2001:db8:feed::1234,remote=2001:db8:beef::4321 \
tun0
# ipadm create-ip tun0
# ipadm create-addr -T addrconf tun0
# ipadm show-addr
ADDROBJ   TYPE       STATE   ADDR
lo0/v6    static     ok      ::1/128
tun0/v6   addrconf   ok      2001:db8:feed::1234 --> 2001:db8:beef::4321
```

Par agregar direcciones, como una dirección global o direcciones locales y remotas alternativas, utilice el comando `ipadm` de la siguiente manera:

```
# ipadm create-addr \
-a local=2001:db8::4728:56bc,remote=2001:db8::1428:57ab tun0
# ipadm show-addr tun0
ADDROBJ    TYPE      STATE ADDR
tun0/v6    addrconf  ok    2001:db8:feed::1234 --> 2001:db8:beef::4321
tun0/v6a   static    ok    2001:db8::4728:56bc --> 2001:db8::1428:57ab
```

▼ Cómo configurar un túnel 6to4

En túneles 6to4, un enrutador 6to4 debe actuar como enrutador IPv6 para los nodos de los sitios 6to4 de la red. Por lo tanto, al configurar un enrutador 6to4, ese enrutador también debe estar configurado como enrutador IPv6 en las interfaces físicas. Para obtener más información sobre los enrutadores IPv6, consulte "Enrutamiento de IPv6" en la página 151.

1 Cree un túnel 6to4.

 # dladm create-iptun -T 6to4 -a local=*address tunnel-link*

Para este comando, están disponibles las opciones o los argumentos siguientes:

-a local=*address*	Especifica la dirección local del túnel, que ya debe existir en el sistema para ser una dirección válida.
enlace_túnel	Especifica el enlace de túnel IP. Al admitir nombres significativos en una administración de enlace de red, los nombres de los túneles ya no se restringen al tipo de túnel que se está creando. En cambio, se puede asignar a un túnel cualquier nombre elegido administrativamente. Los nombres de túneles está formados por una cadena y el número de PPA, por ejemplo, *mitúnel0*. Para conocer las reglas que rigen la asignación de nombres descriptivos, consulte "Reglas para nombres de enlaces válidos" de *Introducción a redes de Oracle Solaris 11*.

2 Cree la interfaz IP del túnel.

 # ipadm create-ip *tunnel-interface*

Donde *interfaz_túnel* utiliza el mismo nombre que el enlace de túnel.

3 (Opcional) Agregue direcciones IPv6 alternativas para el uso del túnel.

4 Agregue las siguientes dos líneas para editar el archivo /etc/inet/ndpd.conf para anunciar el enrutamiento 6to4:

```
if subnet-interface AdvSendAdvertisements 1
IPv6-address  subnet-interface
```

La primera línea especifica la subred que recibe el anuncio. Donde *interfaz_subred* se refiere al enlace al que está conectada la subred. La dirección IPv6 de la segunda línea debe tener el prefijo 6to4 `2000` que se utiliza para direcciones IPv6 en túneles 6to4.

Para obtener información detallada sobre el archivo ndpd.conf, consulte la página del comando man ndpd.conf(4).

5 **Active el reenvío de IPv6.**

```
# ipadm set-prop -p forwarding=on ipv6
```

6 **Reinicie el enrutador.**

También puede enviar un comando sighup al daemon /etc/inet/in.ndpd para que empiece a enviar anuncios de enrutador. Los nodos IPv6 de cada subred que recibirá el prefijo 6to4 se autoconfiguran con las nuevas direcciones derivadas 6to4.

7 **Agregue las nuevas direcciones derivadas 6to4 de los nodos al servicio de nombre utilizado en la ubicación 6to4.**

Si necesita instrucciones, consulte "Configuración de la compatibilidad con el servicio de nombres para IPv6" en la página 75.

Ejemplo 6–4 Creación de un túnel 6to4

En este ejemplo, la interfaz de subred es bge0, a la que se referirá /etc/inet/ndpd.conf en el paso correspondiente.

En este ejemplo, se muestra cómo crear un túnel 6to4. Tenga en cuenta que únicamente las interfaces IPv6 se pueden configurar en túneles 6to4.

```
# dladm create-iptun -T 6to4 -a local=192.168.35.10 tun0
# ipadm create-ip tun0
# ipadm show-addr
ADDROBJ       TYPE     STATE   ADDR
lo0/v4        static   ok      127.0.0.1/8
net0/v4       static   ok      192.168.35.10/24
lo0/v6        static   ok      ::1/128
tun0/_a       static   ok      2002:c0a8:57bc::1/64

# ipadm create-addr -a 2002:c0a8:230a::2/16 tun0
# ipadm create-addr -a 2002:c0a8:230a::3/16 tun0
# ipadm show-addr tun0
ADDROBJ       TYPE     STATE   ADDR
lo0/v4        static   ok      127.0.0.1/8
net0/v4       static   ok      192.168.35.10/24
lo0/v6        static   ok      ::1/128
tun0/_a       static   ok      2002:c0a8:57bc::1/64
tun0/v6       static   ok      2002:c0a8:230a::2/16
tun0/v6a      static   ok      2002:c0a8:230a::3/16

# vi /etc/inet/ndpd.conf
if bge0 AdvSendAdvertisements 1
2002:c0a8:57bc::1/64 bge0

# ipadm set-prop -p forwarding=on ipv6
```

Tenga en cuenta que para los túneles 6to4, el prefijo para la dirección IPv6 es 2002.

▼ Cómo configurar un túnel 6to4 hasta un enrutador de reenvío 6to4

Precaución – Por problemas graves de seguridad, Oracle Solaris tiene desactivada la compatibilidad con enrutadores de reenvío. Consulte "Security Issues When Tunneling to a 6to4 Relay Router" de *Troubleshooting Network Issues*.

Antes de empezar

Antes de activar un túnel hasta un enrutador de reenvío 6to4, debe haber realizado las siguientes tareas:

- Configurar un enrutador 6to4 en el sitio, como se explica en "Cómo crear y configurar un túnel IP" en la página 113
- Revisar los problemas de seguridad relacionados con el establecimiento de un túnel hasta un enrutador de reenvío 6to4

1 Active un túnel hasta el enrutador de reenvío 6to4 utilizando uno de los siguientes formatos:

- Activar un túnel a un enrutador de reenvío 6to4 de difusión por proximidad.

  ```
  # /usr/sbin/6to4relay -e
  ```

  La opción -e establece un túnel entre el enrutador 6to4 y un enrutador de reenvío 6to4 de difusión por proximidad. Los enrutadores de reenvío 6to4 de difusión por proximidad tienen la dirección IPv4 192.88.99.1. El enrutador de reenvío de difusión por proximidad que se encuentre más cerca físicamente de su ubicación pasa a ser el punto final del túnel 6to4. Este enrutador de reenvío gestiona el reenvío de paquetes entre su ubicación 6to4 y una ubicación IPv6 nativa.

  Si necesita información detallada sobre enrutadores de reenvío 6to4 de difusión por proximidad, consulte RFC 3068, "An Anycast Prefix for 6to 4 Relay Routers" (ftp://ftp.rfc-editor.org/in-notes/rfc3068.txt).

- Active un túnel hasta un enrutador de reenvío 6to4 específico.

  ```
  # /usr/sbin/6to4relay -e -a relay-router-address
  ```

  La opción -a indica que a continuación se especifica una dirección de un enrutador determinado. Reemplace *dirección_enrutador_reenvío* con la dirección IPv4 del enrutador de reenvío 6to4 específico con el que quiera establecer un túnel.

El túnel hasta el enrutador de reenvío 6to4 permanece activo hasta que se elimine la pseudointerfaz de túnel 6to4.

2. **Suprima el túnel hasta el enrutador de reenvío 6to4 cuando ya no sea necesario:**

   ```
   # /usr/sbin/6to4relay -d
   ```

3. **(Optativo) Haga que el túnel hasta el enrutador de reenvío 6to4 se mantenga al reiniciar.**

   Es posible que en su ubicación sea necesario restablecer el túnel hasta el enrutador de reenvío 6to4 cada vez que se reinicia en enrutador 6to4. Para ello, debe hacer lo siguiente:

   a. **Edite el archivo `/etc/default/inetinit`.**

      La línea que se debe modificar se encuentra al final del archivo.

   b. **Cambie el valor "NO" de la línea `ACCEPT6TO4RELAY=NO` por "YES".**

   c. **(Optativo) Cree un túnel a un enrutador de reenvío 6to4 específico que se mantenga al reiniciar.**

      En el parámetro `RELAY6TO4ADDR`, cambie la dirección `192.88.99.1` por la dirección IPv4 del enrutador de reenvío 6to4 que quiera usar.

Ejemplo 6–5 Obtención de información de estado sobre la compatibilidad con enrutador de reenvío 6to4

Puede usar el comando `/usr/bin/6to4relay` para averiguar si la compatibilidad con enrutadores de reenvío 6to4 está activada. El siguiente ejemplo muestra el resultado cuando la compatibilidad con enrutadores de reenvío 6to4 está desactivada, que es la opción predeterminada en Oracle Solaris:

```
# /usr/sbin/6to4relay
6to4relay: 6to4 Relay Router communication support is disabled.
```

Si la compatibilidad con enrutadores de reenvío 6to4 está activada, recibirá el siguiente resultado:

```
# /usr/sbin/6to4relay
6to4relay: 6to4 Relay Router communication support is enabled.
IPv4 remote address of Relay Router=192.88.99.1
```

▼ Cómo modificar una configuración de túnel IP

- **Cambie la configuración del túnel.**

  ```
  # dladm modify-iptun -a [local|remote]=addr,... tunnel-link
  ```

  No puede modificar el tipo de un túnel existente. Por lo tanto, la opción `-T` *tipo* no se permite para este comando. Únicamente pueden modificarse los parámetros de túnel siguientes:

`-a [local|remote]=`*dirección,...*

Especifica los nombres de host o las direcciones IP literales que corresponden a la dirección local y a la dirección de túnel remota. Según el tipo de túnel, debe especificar una sola dirección o ambas direcciones (locales y remotas). Si especifica direcciones locales y remotas, debe separarlas con una coma.

- Los túneles `IPv4` requieren direcciones IPv4 locales y remotas para funcionar.
- Los túneles `IPv6` requieren direcciones IPv6 locales y remotas para funcionar.
- Los túneles `6to4` requieren una dirección IPv4 local para funcionar.

Para configuraciones de enlace de datos de túneles IP, si está utilizando nombres de host para las direcciones, estos nombres de host se guardan en el almacenamiento de la configuración. Durante un inicio posterior del sistema, si el nombre remite a direcciones IP distintas de las direcciones IP utilizadas cuando se creó el túnel, el túnel adquiere una nueva configuración.

Si está cambiando las direcciones locales y remotas del túnel, asegúrese de que estas direcciones sean coherentes con el tipo de túnel que está modificando.

Nota – Si desea cambiar el nombre del enlace de túnel, no utilice el subcomando `modify-iptun`. En cambio, utilice `dladm rename-link`.

`# dladm rename-link` *old-tunnel-link new-tunnel-link*

De manera similar, no utilice el comando `modify-iptun` para cambiar las propiedades del túnel, como `hoplimit` o `encaplimit`. En cambio, utilice el comando `dladm set-linkprop` para configurar valores para estas propiedades.

Ejemplo 6–6 Modificación de la dirección y las propiedades de un túnel

Este ejemplo consta de dos procedimientos. En primer lugar, las direcciones locales y remotas del túnel IPv4 `vpn0` se cambian temporalmente. Cuando el sistema se reinicia más adelante, el túnel vuelve a utilizar las direcciones originales. En un segundo procedimiento, el valor de `hoplimit` de `vpn0` se cambia a 60.

```
# dladm modify-iptun -t -a local=10.8.48.149,remote=192.1.2.3 vpn0
```

```
# dladm set-linkprop -p hoplimit=60 vpn0
```

▼ Cómo visualizar una configuración de túnel IP

- **Visualice la configuración del túnel IP.**

  # dladm show-iptun [-p] -o *fields* **[***tunnel-link***]**

  Con el comando, se pueden utilizar las siguientes opciones:

`-p`	Muestra la información en un formato que la máquina puede analizar. Este argumento es opcional.
`-o` *campos*	Muestra campos seleccionados que proporcionan información de un túnel específico.
enlace_túnel	Especifica el túnel cuya información de configuración desea visualizar. Este argumento es opcional. Si omite el nombre del túnel, el comando muestra la información sobre todos los túneles del sistema.

Ejemplo 6–7 Visualización de información sobre todos los túneles

En este ejemplo, únicamente existe un túnel en el sistema.

```
# dladm show-iptun
LINK     TYPE     FLAGS     LOCAL          REMOTE
tun0     6to4     --        192.168.35.10  --
vpn0     ipv4     --        10.8.48.149    192.1.2.3
```

Ejemplo 6–8 Visualización de campos seleccionados en un formato que la máquina puede analizar

En este ejemplo, únicamente se muestran campos específicos con información del túnel.

```
# dladm show-iptun -p -o link,type,local
tun0:6to4:192.168.35.10
vpn0:ipv4:10.8.48.149
```

▼ Cómo visualizar las propiedades de un túnel IP

- **Visualice las propiedades del enlace de túnel.**

  # dladm show-linkprop [-c] [-o *fields***] [***tunnel-link***]**

  Con el comando, se pueden utilizar las siguientes opciones:

`-c`	Muestra la información en un formato que la máquina puede analizar. Este argumento es opcional.

`-o` *campos*	Muestra campos seleccionados que proporcionan información sobre las propiedades del enlace.
enlace_túnel	Especifica el túnel cuya información de propiedades se desea visualizar. Este argumento es opcional. Si omite el nombre del túnel, el comando muestra la información sobre todos los túneles del sistema.

Ejemplo 6–9 Visualización de las propiedades de un túnel

En este ejemplo, se muestra cómo visualizar todas las propiedades del enlace de un túnel.

```
# dladm show-linkprop tun0
LINK     PROPERTY     PERM     VALUE     DEFAULT     POSSIBLE
tun0     autopush     --       --        --          --
tun0     zone         rw       --        --          --
tun0     state        r-       up        up          up,down
tun0     mtu          r-       65515     --          576-65495
tun0     maxbw        rw       --        --          --
tun0     cpus         rw       --        --          --
tun0     priority     rw       high      high        low,medium,high
tun0     hoplimit     rw       64        64          1-255
```

▼ Cómo suprimir un túnel IP

1 Utilice la sintaxis adecuada para desconectar la interfaz IP configurada en el túnel según el tipo de interfaz.

`# ipadm delete-ip` *tunnel-link*

Nota – Para suprimir correctamente un túnel, no puede conectarse en el túnel ninguna interfaz IP existente.

---

2 Suprima el túnel IP.

`# dladm delete-iptun` *tunnel-link*

La única opción para este comando es `-t`, que suprime el túnel temporalmente. Al reiniciar el sistema, se restaura el túnel.

Ejemplo 6–10 Supresión de un túnel IPv6 configurado con una interfaz IPv6

En este ejemplo, se suprime permanentemente un túnel persistente.

```
# ipadm delete-ip ip6.tun0
# dladm delete-iptun ip6.tun0
```

◆ ◆ ◆ CAPÍTULO 7

Referencia de IPv4

Este capítulo proporciona información de referencia sobre la red TCP/IP para los archivos de configuración de la red, incluidos los tipos, su finalidad y el formato de las entradas de archivo.

El capítulo contiene la información siguiente:

Archivos de configuración TCP/IP

En una red, la información de configuración se almacena en distintos archivos y bases de datos que regulan la forma en que funciona la red. En esta sección, se proporciona una breve descripción de estos archivos. Algunos archivos requieren actualización y mantenimiento a medida que se implementan cambios en la red. Otros archivos requieren muy poca o ninguna administración.

`/etc/defaultrouter` Este archivo contiene los nombres de interfaces IP de los enrutadores que están directamente conectados a la red. La existencia de este archivo en el sistema es opcional. Si existe el archivo, el sistema está configurado para admitir el enrutamiento estático.

`/etc/inet/hosts` Este archivo contiene las direcciones IPv4 en la red junto con los nombres de las interfaces correspondientes en las que están configuradas las direcciones. Si utiliza el servicio de nombres NIS o DNS, o el servicio de directorios LDAP, la información de host se almacena en una base de datos diferente, como `hosts.byname`, que existe en los servidores. Para obtener más información, consulte *Trabajo con servicios de nombres y directorios en Oracle Solaris 11.1*.

`/etc/inet/netmasks`	Este archivo contiene el número de red, como `192.168.0.0`, y la información de máscara de red de ese número de red, como `255.255.255.0`. En una red que utiliza NIS o LDAP, esta información se almacena en una base de datos de máscara de red en los servidores. Consulte la página del comando man `netmasks(4)` para obtener más información.
`/etc/bootparams`	Este archivo contiene los parámetros que determinan los procesos de inicio para los sistemas que están configurados para iniciarse en modo de cliente de red. Para obtener más información, consulte "Configuración de los modos de configuración del sistema" en la página 38. El archivo sirve de base para la creación de la base de datos `bootparams` que el servicio de nombres usa cuando no se está utilizando el modo de archivos locales. Para obtener información específica sobre el contenido y el formato de este archivo, consulte la página del comando man `bootparams(4)`.
`/etc/ethers`	El archivo asocia los nombres de host con las direcciones MAC. El archivo sirve de base para la creación de una base de datos `ethers` que se utiliza en la red donde los sistemas están configurados como clientes de red. Para obtener más información, consulte la página del comando man `ethers(4)`.
`/etc/inet/networks`	Este archivo asocia nombres de red con números de red. También se pueden agregar comentarios para ofrecer una aclaración adicional de cada entrada en la base de datos. Este archivo permite que las aplicaciones utilicen y muestren los nombres de red en lugar de los números de red. Por ejemplo, el programa `netstat` utiliza la información de esta base de datos para producir tablas de estado. Se deben incluir en este archivo todas las subredes que se conectan a la red local mediante enrutadores. Para obtener más información, consulte la página del comando man `networks(4)`.
`/etc/inet/protocols`	Este archivo enumera los protocolos TCP/IP instalados en el sistema, además de sus números de protocolo. Este archivo rara vez requiere administración. Para obtener más información, consulte la página del comando man `protocols(4)`.
`/etc/inet/services`	Este archivo enumera los nombres de los servicios TCP y UDP, además de sus números de puerto conocidos. Los programas que llaman a los servicios de red utilizan esta lista. Por lo general, este archivo no requiere ninguna administración. Para obtener más información, consulte la página del comando man `services(4)`.

Daemon de servicios de Internet `inetd`

El daemon `inetd` inicia los servicios de Internet cuando se inicia un sistema, y puede reiniciar un servicio mientras el sistema está en ejecución. Con la utilidad de gestión de servicios (SMF), podrá modificar los servicios de Internet estándar o hacer que el daemon `inetd` inicie servicios adicionales.

Utilice los comandos SMF siguientes para administrar los servicios iniciados por el comando `inetd`:

`svcadm` Para las acciones de un servicio, como activar, desactivar o reiniciar. Para ver más detalles, consulte la página del comando man `svcadm`(1M).

`svcs` Para consultar el estado de un servicio. Para ver más detalles, consulte la página del comando man `svcs`(1).

`inetadm` Para ver y modificar las propiedades de un servicio. Si desea más información, consulte la página del comando man `inetadm`(1M).

El valor de campo `proto` del perfil `inetadm` de un servicio específico indica el protocolo de capa de transporte en el que se ejecuta el servicio. Si el servicio está activado sólo para IPv4, el campo `proto` debe especificarse como `tcp`, `udp` o `sctp`.

- Para obtener instrucciones sobre cómo usar los comandos SMF, consulte "Utilidades administrativas de la línea de comandos de la SMF" de *Gestión de servicios y errores en Oracle Solaris 11.1*.
- Para ver una tarea que utilice comandos SMF para agregar un servicio que se ejecute con SCTP, consulte "Cómo agregar servicios que utilicen el protocolo SCTP" en la página 58.
- Para obtener información sobre cómo agregar servicios que manejen solicitudes IPv4 e IPv6, consulte "Daemon de servicios de Internet `inetd`" en la página 127.

El servicio SMF name-service/switch

El servicio SMF `name-service/switch` define el orden de búsqueda de información de configuración en las bases de datos de red. Parte de la información de configuración de red que antes estaba almacenada en los archivos de configuración, como el dominio predeterminado, se convirtió en las propiedades de este servicio SMF. Las propiedades de este servicio SMF determinan la implementación de los servicios de nombres en el sistema. Las propiedades se enumeran de la siguiente manera:

```
% svccfg -s name-service/switch listprop config
config                      application
config/value_authorization  astring             solaris.smf.value.name-service.switch
config/default              astring             files
config/password             astring             "files nis"
```

```
config/group              astring          "files nis"
config/host               astring          "files dns nis"
config/network            astring          "nis [NOTFOUND=return] files"
config/protocol           astring          "nis [NOTFOUND=return] files"
config/rpc                astring          "nis [NOTFOUND=return] files"
config/ether              astring          "nis [NOTFOUND=return] files"
config/netmask            astring          "files nis"
config/bootparam          astring          "nis [NOTFOUND=return] files"
config/publickey          astring          "nis [NOTFOUND=return] files"
config/netgroup           astring          nis
config/automount          astring          "files nis"
config/alias              astring          "files nis"
config/service            astring          "files nis"
config/printer            astring          "user nis"
config/auth_attr          astring          "files nis"
config/prof_attr          astring          "files nis"
config/project            astring          "files nis"
```

Los valores establecidos para cada una de las propiedades determinan en qué servicio de nombres se debe buscar la información que puede afectar a los usuarios de la red, como contraseñas, alias o máscaras de red. En el ejemplo, las propiedades de montaje automático y de contraseñas están establecidas en `files` y `nis`. De esta manera, la información de montaje automático y de contraseñas se obtiene de los archivos y del servicio NIS.

Si desea cambiar de un servicio de nombres a otro, debe configurar las propiedades pertinentes del servicio SMF `name-service/switch` para activar el servicio de nombres seleccionado.

Por ejemplo, suponga que desea utilizar el servicio de nombres LDAP en la red. Se deben configurar las siguientes propiedades del servicio SMF:

- `config/default` se debe configurar para utilizar archivos y LDAP.
- `config/host` se debe configurar para utilizar archivos y DNS.
- `config/netgroup` se debe configurar para utilizar LDAP.
- `config/printer` se debe configurar para utilizar usuarios, archivos y LDAP.

Por lo tanto, debe escribir los comandos siguientes para configurar estas propiedades correctamente.

```
# svccfg -s name-service/switch setprop config/default = astring: '"files ldap"'
# svccfg -s name-service/switch setprop config/host = astring: '"files dns"'
# svccfg -s name-service/switch setprop config/netgroup = astring: '"ldap"'
# svccfg -s name-service/switch setprop config/printer = astring: '"user files ldap"'
# svccfg -s name-service/switch:default refresh
```

Para obtener detalles completos sobre el cambio de servicios de nombres, consulte *Trabajo con servicios de nombres y directorios en Oracle Solaris 11.1*.

Cómo afectan los servicios de nombres a las bases de datos de red

El formato de la base de datos de red depende del tipo de servicio de nombres que seleccione para la red. Por ejemplo, la base de datos `hosts` contiene como mínimo el nombre de host y la dirección IPv4 del sistema local, así como cualquier interfaz de red que esté conectada directamente al sistema local. Sin embargo, la base de datos `hosts` puede contener otras direcciones IPv4 y nombres de host, según el tipo de servicio de nombres de la red.

Las bases de datos de red se utilizan de la siguiente manera:

- Las redes que utilizan archivos locales para el servicio de nombres dependen de los archivos de los directorios `/etc/inet` y `/etc`.
- NIS utiliza bases de datos denominadas mapas NIS.
- DNS utiliza registros con información de host.

Nota – Los archivos de datos e inicio DNS no se corresponden directamente con las bases de datos de red.

Consulte *Trabajo con servicios de nombres y directorios en Oracle Solaris 11.1* para obtener información sobre correspondencias de bases de datos de red en NIS, DNS y LDAP.

Protocolos de enrutamiento en Oracle Solaris

Esta sección describe dos protocolos de enrutamiento que admite Oracle Solaris: el protocolo de información de enrutamiento (RIP) y el ICMP Router Discovery (RDISC). RIP y RDISC son protocolos TCP/IP estándar. Para obtener una lista completa de los protocolos de enrutamiento disponibles en Oracle Solaris, consulte la Tabla 7–1 and Tabla 7–2.

Protocolo de información de enrutamiento (RIP)

RIP se implementa mediante el daemon de enrutamiento `in.routed`, que se inicia automáticamente al iniciar el sistema. Cuando se ejecuta en un enrutador con la opción `s` especificada, el comando `in.routed` rellena la tabla de enrutamiento del núcleo con una ruta a cada red accesible y comunica la posibilidad de acceso mediante todas las interfaces de red.

Cuando se ejecuta en un host con la opción `q` especificada, `in.routed` extrae la información de enrutamiento pero no comunica las posibilidades de acceso. En los hosts, la información de enrutamiento se puede extraer de dos modos:

- *No* se especifica el indicador `S` ("S" mayúscula: "Modo de ahorro de espacio"). El comando `in.routed` genera una tabla de enrutamiento completa, al igual que en un enrutador.
- Se especifica el indicador `S`. El comando `in.routed` crea una tabla de núcleo mínima, que contiene una única ruta predeterminada para cada enrutador disponible.

Protocolo ICMP Router Discovery (RDISC)

Los hosts utilizan RDISC para obtener información de enrutamiento de los enrutadores. De este modo, cuando los hosts ejecutan RDISC, los enrutadores también deben ejecutar otro protocolo, como RIP, para poder intercambiar información de enrutadores.

RDISC se implementa mediante el comando `in.routed`, que debe ejecutarse tanto en los enrutadores como en los hosts. En los hosts, `in.routed` utiliza RDISC para descubrir las rutas predeterminadas de los enrutadores que se dan a conocer a través de RDISC. En los enrutadores, `in.routed` utiliza RDISC para dar a conocer las rutas predeterminadas a los hosts en las redes conectadas directamente. Consulte las página del comando man `in.routed(1M)` y `gateways(4)`.

Tablas de protocolos de enrutamiento en Oracle Solaris

En la siguiente tabla, se enumeran todos los protocolos de enrutamiento admitidos en Oracle Solaris.

TABLA 7–1 Protocolos de enrutamiento de Oracle Solaris

Protocolo	Daemon asociado	Descripción	Para obtener instrucciones
Protocolo de información de enrutamiento (RIP)	`in.routed`	IGP que enruta paquetes IPv4 y mantiene una tabla de enrutamiento	"Configuración de un enrutador IPv4" en la página 43
Descubrimiento de enrutador de protocolo de mensajes de control de Internet (ICMP)	`in.routed`	Lo utilizan los hosts para descubrir la presencia de un enrutador en la red	"Cómo activar el enrutamiento estático en un host de interfaz única" en la página 52 y "Cómo activar el enrutamiento dinámico en un sistema de interfaz única" en la página 53
Protocolo de información de enrutamiento, nueva generación (RIPng)	`in.ripngd`	IGP que enruta paquetes IPv6 y mantiene una tabla de enrutamiento	"Cómo configurar un enrutador activado para IPv6" en la página 66

TABLA 7–1 Protocolos de enrutamiento de Oracle Solaris *(Continuación)*

Protocolo	Daemon asociado	Descripción	Para obtener instrucciones
Protocolo de descubrimiento de vecinos (ND)	`in.ndpd`	Advierte la presencia de un enrutador IPv6 y descubre la presencia de hosts IPv6 en una red	"Configuración de una interfaz de IPv6" en la página 63

En la siguiente tabla, se muestran los conjunto de protocolos de enrutamiento de código abierto Quagga que también se admiten en Oracle Solaris.

TABLA 7–2 Protocolos Quagga de código abierto

Protocolo	Daemon	Descripción
Protocolo RIP	`ripd`	Protocolo IGP vector-distancia para IPv4 que enruta paquetes IPv4 y muestra su tabla de enrutamiento a los vecinos.
RIPng	`ripngd`	Protocolo IGP vector-distancia para IPv6. Enruta paquetes IPv6 y mantiene una tabla de enrutamiento.
Protocolo Abrir primero la ruta más corta (OSPF)	`ospfd`	Protocolo IGP de estado de vínculo IPv4 para el enrutamiento de paquetes y las redes de gran disponibilidad.
Protocolo de portal de límite (BGP)	`bgpd`	Protocolo EGP para IPv4 y IPv6 para el enrutamiento en dominios administrativos.

◆ ◆ ◆ CAPÍTULO 8

Referencia de IPv6

Este capítulo proporciona la siguiente información de referencia relativa a la implementación de IPv6 en Oracle Solaris.

Para obtener información sobre las tareas relativas a la configuración de una red activada para IPv6, consulte el Capítulo 4, "Activación de IPv6 en una red". Para obtener información completa sobre los túneles IP, consulte el Capítulo 6, "Configuración de túneles IP".

Implementación de IPv6 en Oracle Solaris

Esta sección describe los archivos, comandos y daemons que activan IPv6 en Oracle Solaris.

Archivos de configuración de IPv6

Esta sección describe los archivos de configuración que forman parte de una implementación de IPv6:

Archivo de configuración `ndpd.conf`

El archivo `/etc/inet/ndpd.conf` se utiliza para configurar opciones empleadas por el daemon del protocolo ND `in.ndpd`. En el caso de un enrutador, `ndpd.conf` se utiliza sobre todo para

configurar el prefijo de sitio que se debe anunciar en el vínculo. En lo que respecta a un host, `ndpd.conf` se usa para desactivar la configuración automática de redes o para configurar direcciones temporales.

La tabla siguiente muestra las palabras clave que se utilizan en el archivo `ndpd.conf`.

TABLA 8–1 Palabras clave de `/etc/inet/ndpd.conf`

Variable	Descripción
`ifdefault`	Especifica el comportamiento de enrutador en todas las interfaces. Utilice la sintaxis siguiente para establecer los parámetros de enrutador y los valores correspondientes: `ifdefault [`*variable-value*`]`
`prefixdefault`	Especifica el comportamiento predeterminado para los anuncios de prefijo. Utilice la sintaxis siguiente para establecer los parámetros de enrutador y los valores correspondientes: `prefixdefault [`*variable-value*`]`
`if`	Establece los parámetros según la interfaz. Use la sintaxis siguiente: `if` *interface* `[`*variable-value*`]`
`prefix`	Anuncia información de prefijo según la interfaz. Use la sintaxis siguiente: `prefix` *prefix/length interface* `[`*variable-value*`]`

En el archivo `ndpd.conf`, las palabras clave de esta tabla se usan con un conjunto de variables de configuración de enrutador. Puede encontrar una definición detallada de estas variables en RFC 2461, Neighbor Discovery for IP Version 6 (IPv6) (`http://www.ietf.org/rfc/rfc2461.txt?number=2461`).

En la siguiente tabla aparecen las variables necesarias para configurar una interfaz, junto con breves definiciones.

TABLA 8–2 Variables de configuración de interfaz de `/etc/inet/ndpd.conf`

Variable	Predeterminado	Definición
`AdvRetransTimer`	0	Especifica el valor del campo RetransTimer en los mensajes de anuncio que envía el enrutador.
`AdvCurHopLimit`	Diámetro actual de Internet	Especifica el valor que se debe colocar en el límite de salto actual de los mensajes de anuncio que envía el enrutador.
`AdvDefaultLifetime`	3 + `MaxRtrAdvInterval`	Especifica la vida útil predeterminada de los anuncios de enrutador.
`AdvLinkMTU`	0	Especifica el valor de MTU (Maximum Transmission Unit, unidad de transmisión máxima) que debe enviar el enrutador. El cero indica que el enrutador no especifica opciones de MTU.

TABLA 8–2 Variables de configuración de interfaz de `/etc/inet/ndpd.conf` *(Continuación)*

Variable	Predeterminado	Definición
`AdvManaged Flag`	Falso	Indica el valor que se debe colocar en el indicador Manage Address Configuration del anuncio de enrutador.
`AdvOtherConfigFlag`	Falso	Indica el valor que se debe colocar en el indicador Other Stateful Configuration del anuncio de enrutador.
`AdvReachableTime`	0	Especifica el valor del campo ReachableTime en los mensajes de anuncio que envía el enrutador.
`AdvSendAdvertisements`	Falso	Indica si el nodo debe enviar anuncios y responder a solicitudes de enrutador. Esta variable se debe establecer en "TRUE" en el archivo `ndpd.conf` para activar funciones de anuncio de enrutador. Para obtener más información, consulte "Cómo configurar un enrutador activado para IPv6" en la página 66.
`DupAddrDetect Transmits`	1	Define la cantidad de mensajes consecutivos de solicitudes de vecino que el protocolo ND debe enviar durante la detección de direcciones duplicadas de la dirección del nodo local.
`MaxRtrAdvInterval`	600 segundos	Especifica el intervalo máximo de tiempo de espera entre el envío de anuncios multidifusión no solicitados.
`MinRtrAdvInterval`	200 segundos	Especifica el intervalo mínimo de espera entre el envío de anuncios multidifusión no solicitados.
`StatelessAddrConf`	Verdadero	Controla si el nodo configura su dirección IPv6 mediante la configuración automática de direcciones sin estado. Si en el archivo `ndpd.conf` se declara False, la dirección se debe configurar manualmente. Para obtener más información, consulte "Cómo configurar un token IPv6 especificado por el usuario" en la página 72.
`TmpAddrsEnabled`	Falso	Indica si se debe crear una dirección temporal para todas las interfaces o para una determinada interfaz de un nodo. Para obtener más información, consulte "Cómo configurar una dirección temporal" en la página 69.
`TmpMaxDesyncFactor`	600 segundos	Especifica un valor aleatorio que se debe sustraer de la variable de vida útil preferente `TmpPreferredLifetime` al iniciarse `in.ndpd`. La finalidad de la variable `TmpMaxDesyncFactor` es impedir que todos los sistemas de la red vuelvan a generar sus direcciones temporales al mismo tiempo. `TmpMaxDesyncFactor` permite modificar el límite superior de ese valor aleatorio.
`TmpPreferredLifetime`	Falso	Establece la vida útil preferente de una dirección temporal. Para obtener más información, consulte "Cómo configurar una dirección temporal" en la página 69.
`TmpRegenAdvance`	Falso	Especifica el tiempo de demora antes de descartar una dirección temporal. Para obtener más información, consulte "Cómo configurar una dirección temporal" en la página 69.

TABLA 8–2 Variables de configuración de interfaz de `/etc/inet/ndpd.conf` *(Continuación)*

Variable	Predeterminado	Definición
`TmpValidLifetime`	Falso	Establece la vida útil válida de una dirección temporal. Para obtener más información, consulte "Cómo configurar una dirección temporal" en la página 69.

En la siguiente tabla se muestran las variables que se utilizan para configurar prefijos IPv6.

TABLA 8–3 Variables de configuración de prefijo de `/etc/inet/ndpd.conf`

Variable	Predeterminado	Definición
`AdvAutonomousFlag`	Verdadero	Especifica el valor que se debe colocar en el campo AutonomousFlag en la opción de información de prefijo.
`AdvOnLinkFlag`	Verdadero	Especifica el valor que se debe colocar en el indicador on-link ("L-bit") en la opción de información de prefijo.
`AdvPreferredExpiration`	No establecido	Especifica la fecha de caducidad preferente del prefijo.
`AdvPreferredLifetime`	604800 segundos	Especifica el valor que se debe colocar en el campo PreferredLifetime en la opción de información de prefijo.
`AdvValidExpiration`	No establecido	Especifica la fecha de caducidad válida del prefijo.
`AdvValidLifetime`	2592000 segundos	Especifica la vida útil válida del prefijo que se configura.

EJEMPLO 8–1 Archivo `/etc/inet/ndpd.conf`

En el ejemplo siguiente se muestra el modo de utilizar las palabras clave y las variables de configuración en el archivo `ndpd.conf`. Elimine el comentario (#) para activar la variable.

```
# ifdefault       [variable-value ]*
# prefixdefault [variable-value ]*
# if ifname    [variable-value ]*
# prefix prefix/length ifname
#
#  Per interface configuration variables
#
#DupAddrDetectTransmits
#AdvSendAdvertisements
#MaxRtrAdvInterval
#MinRtrAdvInterval
#AdvManagedFlag
#AdvOtherConfigFlag
#AdvLinkMTU
#AdvReachableTime
#AdvRetransTimer
#AdvCurHopLimit
#AdvDefaultLifetime
#
# Per Prefix:  AdvPrefixList configuration variables
```

EJEMPLO 8–1 Archivo `/etc/inet/ndpd.conf` *(Continuación)*

```
#
#
#AdvValidLifetime
#AdvOnLinkFlag
#AdvPreferredLifetime
#AdvAutonomousFlag
#AdvValidExpiration
#AdvPreferredExpiration

ifdefault AdvReachableTime 30000 AdvRetransTimer 2000
prefixdefault AdvValidLifetime 240m AdvPreferredLifetime 120m

if qe0 AdvSendAdvertisements 1
prefix 2:0:0:56::/64 qe0
prefix fec0:0:0:56::/64 qe0

if qe1 AdvSendAdvertisements 1
prefix 2:0:0:55::/64 qe1
prefix fec0:0:0:56::/64 qe1

if hme1 AdvSendAdvertisements 1
prefix  2002:8192:56bb:1::/64 qfe0

if hme1 AdvSendAdvertisements 1
prefix  2002:8192:56bb:2::/64 hme1
```

Archivo de configuración `/etc/inet/ipaddrsel.conf`

El archivo `/etc/inet/ipaddrsel.conf` contiene la tabla de directrices de selección de direcciones predeterminadas de IPv6. Al instalar Oracle Solaris activado para IPv6, este archivo incluye el contenido que se muestra en la Tabla 8–4.

El contenido de `/etc/inet/ipaddrsel.conf` se puede editar. Ahora bien, en la mayoría de los casos no es conveniente modificarlo. Si hace falta realizar cambios, consulte el procedimiento "Cómo administrar la tabla de directrices de selección de direcciones IPv6" en la página 100. Para obtener más información sobre `ippaddrsel.conf`, consulte "Motivos para modificar la tabla de directrices de selección de direcciones IPv6" en la página 138 y la página del comando man `ipaddrsel.conf(4)`.

Comandos relacionados con IPv6

Esta sección describe comandos que se agregan con la implementación de IPv6 en Oracle Solaris. Asimismo, se especifican las modificaciones realizadas en los comandos para poder admitir IPv6.

Comando `ipaddrsel`

El comando `ipaddrsel` permite modificar la tabla de directrices de selección de direcciones predeterminadas de IPv6.

El núcleo de Oracle Solaris utiliza la tabla de directrices de selección de direcciones predeterminadas de IPv6 para ordenar direcciones de destino y seleccionar direcciones de origen en un encabezado de paquetes de IPv6. El archivo `/etc/inet/ipaddrsel.conf` contiene la tabla de políticas.

En la tabla siguiente se enumeran los formatos de direcciones predeterminadas y las correspondientes prioridades en la tabla de directrices. En la página del comando man `inet6(7P)` hay más información referente a aspectos técnicos sobre la selección de direcciones IPv6.

TABLA 8–4 Tabla de directrices de selección de direcciones IPv6

Prefijo	Prioridad	Definición
`::1/128`	50	Bucle de retorno
`::/0`	40	Predeterminado
`2002::/16`	30	6to4
`::/96`	20	Compatible con IPv4
`::ffff:0:0/96`	10	IPv4

En esta tabla, los prefijos de IPv6 (`::1/128` y `::/0`) tienen prioridad sobre las direcciones 6to4 (`2002::/16`) y las direcciones IPv4 (`::/96` y `::ffff:0:0/96`). Así pues, de forma predeterminada, el núcleo selecciona la dirección IPv6 global de la interfaz para paquetes que se dirigen a otro destino de IPv6. La dirección IPv4 de la interfaz tiene una prioridad inferior, sobre todo en cuanto a paquetes que se dirigen a un destino de IPv6. A partir de la dirección IPv6 de origen seleccionada, el núcleo también utiliza el formato de IPv6 para la dirección de destino.

Motivos para modificar la tabla de directrices de selección de direcciones IPv6

En la mayoría de los casos, no se necesita cambiar la tabla de directrices de selección de direcciones predeterminadas de IPv6. Para administrar la tabla de directrices, se utiliza el comando `ipaddrsel`.

La tabla de directrices podría modificarse en alguno de los supuestos siguientes:

- Si el sistema tiene una interfaz que se emplea para un túnel de 6to4, puede otorgar mayor prioridad a las direcciones 6to4.
- Si desea utilizar una determinada dirección de origen sólo para comunicarse con una determinada dirección de destino, puede agregar dichas direcciones a la tabla de directrices. A continuación, mediante el comando `ipadm`, etiquete estas direcciones en función de sus preferencias. Para obtener más información sobre el comando `ipadm`, consulte la página del comando man `ipadm(1M)`.

- Si quiere otorgar más prioridad a las direcciones IPv4 respecto a las de IPv6, la prioridad de `::ffff:0:0/96` puede cambiarse por un número superior.
- Si debe asignar mayor prioridad a direcciones descartadas, tales direcciones se pueden incorporar a la tabla de directrices. Por ejemplo, las direcciones locales de sitio ahora se descartan en IPv6. Estas direcciones tienen el prefijo `fec0::/10`. La tabla de directrices se puede modificar para conceder mayor prioridad a las direcciones locales de sitio.

Para obtener más información sobre el comando `ipaddrsel`, consulte la página del comando man `ipaddrsel`(1M).

Comando `6to4relay`

El establecimiento de túneles de 6to4 permite las comunicaciones entre sitios de 6to4 que están aislados. Sin embargo, para transferir paquetes con un sitio de IPv6 nativo que no sea de 6to4, el enrutador de 6to4 debe establecer un túnel con un enrutador de relé de 6to4. Así, el *enrutador de relé de 6to4* reenvía los paquetes de 6to4 a la red IPv6 y, en última instancia, al sitio de IPv6 nativo. Si el sitio activado para 6to4 debe intercambiar datos con sitio de IPv6 nativo, utilice el comando `6to4relay` para activar el túnel correspondiente.

Como el uso de enrutadores de relé no es seguro, en Oracle Solaris de manera predeterminada se desactiva el establecimiento de túneles con un enrutador de relé. Antes de implementar esta situación hipotética, debe tener muy en cuenta los problemas que comporta crear un túnel con un enrutador de relé de 6to4. Para obtener más información sobre enrutadores de relé de 6to4, consulte "Consideraciones para túneles hasta un enrutador de reenvío 6to4" en la página 108. Si decide activar la compatibilidad con enrutadores de relé 6to4, consulte "Cómo crear y configurar un túnel IP" en la página 113 para conocer los procedimientos relacionados.

Sintaxis de 6to4relay

El comando `6to4relay` presenta la sintaxis siguiente:

```
6to4relay -e [-a IPv4-address] -d -h
```

`-e`	Activa el uso de túneles entre el enrutador de 6to4 y un enrutador de relé de 6to4 de difusión por proximidad. Así, la dirección de punto final de túnel se establece en `192.88.99.1`, que es la predeterminada para el grupo de difusión por proximidad de enrutadores de relé de 6to4.
`-a` *dirección_IPv4*	Activa el uso de túneles entre el enrutador de 6to4 y un enrutador de relé de 6to4 con la *dirección_IPv4* que se especifique.
`-d`	Anula la admisión del establecimiento de túneles con el enrutador de relé de 6to4, que es el predeterminado de Oracle Solaris.
`-h`	Muestra la ayuda del comando `6to4relay`.

Para obtener más información, consulte la página del comando man `6to4relay`(1M).

EJEMPLO 8–2 Pantalla de estado predeterminado de admisión de enrutador de relé de 6to4

El comando `6to4relay`, sin argumentos, muestra el estado actual de la admisión de enrutadores de relé de 6to4. Este ejemplo ilustra el valor predeterminado de la implementación de IPv6 en Oracle Solaris.

```
# /usr/sbin/6to4relay
6to4relay:6to4 Relay Router communication support is disabled
```

EJEMPLO 8–3 Pantalla de estado con admisión activada de enrutadores de relé de 6to4

Si se activa la admisión de enrutadores de relé, `6to4relay` muestra la salida siguiente:

```
# /usr/sbin/6to4relay
6to4relay:6to4 Relay Router communication support is enabled
IPv4 destination address of Relay Router=192.88.99.1
```

EJEMPLO 8–4 Pantalla de estado con un enrutador de relé de 6to4 especificado

Si se especifica la opción `-a` y una dirección IPv4 en el comando `6to4relay`, en lugar de `-192.88.99.1` se muestra la dirección IPv4 que se proporciona con `a`.

`6to4relay` no indica la ejecución correcta de las opciones de `-dirección_IPv4 -d`, `-e` y *a*. Ahora bien, `6to4relay` muestra cualquier mensaje de error que se pudiera generar durante la ejecución de dichas opciones.

Modificaciones del comando `netstat` para admitir IPv6

El comando `netstat` muestra el estado de redes IPv4 e IPv6. Puede elegir la información de protocolo que se visualizará; para ello, establezca el valor de `DEFAULT_IP` en el archivo `/etc/default/inet_type` o recurra a la opción de línea de comandos `-f`. Si se aplica un valor permanente de `DEFAULT_IP`, se garantiza que `netstat` muestre únicamente información relativa a IPv4. Este valor puede anularse mediante la opción `-f`. Para obtener más información sobre el archivo `inet_type`, consulte la página del comando man `inet_type(4)`.

La opción `-p` del comando `netstat` muestra la tabla de red a soporte, que es la tabla ARP para IPv4 y la caché interna para IPv6. Consulte la página del comando man `netstat(1M)` para obtener más información. Consulte "Cómo visualizar el estado de los sockets" en la página 84 para obtener descripciones de procedimientos que utilizan este comando.

Modificaciones del comando snoop para admitir IPv6

El comando `snoop` puede capturar paquetes de IPv4 e IPv6. Este comando puede mostrar encabezados de IPv6, encabezados de extensiones de IPv6, encabezados de ICMPv6 y datos de protocolo ND. De manera predeterminada, el comando `snoop` muestra paquetes de IPv4 e IPv6. Si especifica la palabra clave de protocolo `ip` o `ip6`, el comando `snoop` muestra sólo paquetes de IPv4 o IPv6, respectivamente. La opción para filtrar IPv6 permite filtrar en todos los paquetes,

tanto de IPv4 como IPv6, y mostrar únicamente los paquetes de IPv6. Consulte la página del comando man `snoop`(1M) para obtener más información. Consulte "Cómo supervisar tráfico de redes IPv6" en la página 95 para obtener información sobre procedimientos que utilizan el comando `snoop`.

Modificaciones del comando `route` para admitir IPv6

El comando `route` funciona en rutas IPv4 e IPv6; el valor predeterminado son las rutas IPv4. Si la opción `-inet6` de la línea de comandos se utiliza inmediatamente después del comando `route`, las operaciones se llevan a cabo en rutas IPv6. Consulte la página del comando man `route`(1M) para obtener más información.

Modificaciones del comando `ping` para admitir IPv6

El comando `ping` utiliza protocolos IPv4 e IPv6 para sondear hosts de destino. La selección de protocolo depende de las direcciones que devuelve el servidor de nombres en relación con el host de destino específico. De forma predeterminada, si el servidor de nombres devuelve una dirección IPv6 para el host de destino, el comando `ping` utiliza el protocolo IPv6. Si el servidor devuelve sólo una dirección IPv4, el comando `ping` emplea el protocolo IPv4. Si desea anular esta acción, utilice la opción de línea de comandos `-A` para indicar el protocolo que debe usarse.

Para obtener más información, consulte la página del comando man `ping`(1M) Para obtener información sobre procedimientos que utilicen el comando `ping` , consulte "Sondeo de hosts remotos con el comando `ping`" en la página 87.

Modificaciones del comando `traceroute` para admitir IPv6

El comando `traceroute` efectúa el seguimiento de las rutas IPv4 e IPv6 de un determinado host. En una perspectiva de protocolos, `traceroute` utiliza el mismo algoritmo que `ping`. Si desea anular esta selección, utilice la opción de línea de comandos `-A`. Puede efectuar el seguimiento de cada ruta en cada dirección de un host con varias direcciones permanentes mediante la opción de línea de comandos `-a`.

Para obtener más información, consulte la página del comando man `traceroute`(1M) Para obtener información sobre procedimientos que usan el comando `traceroute`, consulte "Visualización de información de enrutamiento con el comando `traceroute`" en la página 91.

Daemons relacionados con IPv6

Esta sección trata sobre los daemons relacionados con IPv6.

Daemon `in.ndpd`, para el protocolo ND

El daemon `in.ndpd` implementa el protocolo ND de IPv6 y el descubrimiento de enrutadores. Asimismo, implementa la configuración automática de direcciones para IPv6. A continuación se muestran las opciones admitidas de `in.ndpd`.

`-a`	Desactiva la configuración automática de direcciones sin estado y con estado.
`-d`	Activa la depuración.
`-f` *config-file*	Especifica un archivo desde el cual leer los datos de configuración, en lugar del archivo predeterminado `/etc/inet/ndpd.conf`.
`-t`	Activa el seguimiento de paquetes de todos los paquetes entrantes y salientes.

El daemon `in.ndpd` lo controlan parámetros que se establecen en el archivo de configuración `/etc/inet/ndpd.conf` y los pertinentes parámetros del archivo de inicio de `/var/inet/ndpd_state.`*interface*.

Si existe el archivo `/etc/inet/ndpd.conf`, se analiza y utiliza para configurar un nodo como enrutador. En la Tabla 8–1 figuran las palabras clave válidas que podrían aparecer en este archivo. Si se inicia un host, podría suceder que los enrutadores no estuvieran disponibles de manera inmediata. Los paquetes anunciados por el enrutador podrían perderse. Asimismo, los paquetes anunciados quizá no se comuniquen con el host.

El archivo `/var/inet/ndpd_state.`*interface* es un archivo de estado. Cada nodo lo actualiza periódicamente. Si el nodo falla y se reinicia, el nodo puede configurar sus interfaces si no hay enrutadores. Este archivo contiene las direcciones de interfaz, la última vez que se modificó el archivo y el tiempo que este archivo será válido. Asimismo, el archivo contiene otros parámetros que se "aprenden" a partir de anteriores anuncios de enrutador.

Nota – No es necesario modificar el contenido de archivos de estado. El daemon `in.ndpd` mantiene los archivos de estado de forma automática.

Consulte las páginas de comando man `in.ndpd(1M)` y `ndpd.conf(4)` para obtener listas de variables de configuración y valores permitidos.

Daemon `in.ripngd`, para enrutamiento de IPv6

El daemon `in.ripngd` implementa el protocolo de información de enrutamiento de próxima generación (RIPng) para enrutadores IPv6. RIPng define el equivalente de IPv6 de RIP. Si se configura un enrutador de IPv6 con el comando `routeadm` y se activa el enrutamiento de IPv6, el daemon `in.ripngd` implementa el protocolo RIPng en el enrutador.

A continuación se muestran las opciones admitidas del protocolo RIPng.

`-p` *n* *n* especifica el número de puerto UDP que se utiliza para enviar o recibir paquetes de RIPnG.

`-P` Suprime el uso de valores negativos.

`-q` Suprime información de enrutamiento.

`-s` Fuerza la información de enrutamiento aun en caso de que el daemon funcione como enrutador.

`-t` Imprime todos los paquetes enviados y recibidos en la salida estándar.

`-v` Imprime todos los cambios a la tabla de enrutamiento en una salida estándar, que incluye indicadores de fecha y hora.

Daemon `inetd` y servicios de IPv6

Una aplicación de servidores activada para IPv6 puede asumir solicitudes de IPv4 e IPv6, o únicamente de IPv6. El servidor controla siempre las solicitudes mediante un socket de IPv6. Además, el servidor emplea el mismo protocolo que el del cliente correspondiente.

Si desea agregar o modificar un servicio de IPv6, emplee los comandos disponibles en la utilidad de gestión de servicios (SMF).

- Para obtener información sobre los comandos SMF, consulte "Utilidades administrativas de la línea de comandos de la SMF" de *Gestión de servicios y errores en Oracle Solaris 11.1*.
- Para ver una tarea de ejemplo que utilice SMF en la configuración de un manifiesto de servicio de IPv4 que se ejecute en SCTP, consulte "Cómo agregar servicios que utilicen el protocolo SCTP" en la página 58.

Si desea configurar un servicio de IPv6, asegúrese de que el valor del campo `proto` del perfil `inetadm` relativo a ese servicio presente el valor correspondiente:

- Si necesita un servicio que controle solicitudes de IPv4 e IPv6, elija `tcp6`, `udp6` o `sctp`. Un valor de `proto` de `tcp6`, `udp6` o `sctp6` hace que `inetd` pase en un socket de IPv6 al servidor. El servidor contiene una dirección asignada a IPv4 en caso de que un cliente IPv4 tenga una solicitud.
- Si necesita un servicio que únicamente controle solicitudes de IPv6, elija `tcp6only` o `udp6only`. Si se asigna cualquiera de estos valores a `proto`, `inetd` pasa el servidor a un socket de IPv6.

Si reemplaza un comando de Oracle Solaris por otra implementación, compruebe que la implementación de ese servicio admita IPv6. Si la implementación no admite IPv6, el valor de `proto` debe especificarse como `tcp`, `udp` o `sctp`.

A continuación se muestra un perfil generado tras la ejecución de `inetadm` para un manifiesto de servicio `echo` que admite IPv4 e IPv6, y se ejecuta mediante SCTP:

```
# inetadm -l svc:/network/echo:sctp_stream
    SCOPE    NAME=VALUE      name="echo"
             endpoint_type="stream"
             proto="sctp6"
             isrpc=FALSE
             wait=FALSE
             exec="/usr/lib/inet/in.echod -s"
             user="root"
    default  bind_addr=""
    default  bind_fail_max=-1
    default  bind_fail_interval=-1
    default  max_con_rate=-1
    default  max_copies=-1
    default  con_rate_offline=-1
    default  failrate_cnt=40
    default  failrate_interval=60
    default  inherit_env=TRUE
    default  tcp_trace=FALSE
    default  tcp_wrappers=FALSE
```

Si desea cambiar el valor del campo `proto`, aplique la sintaxis siguiente:

```
# inetadm -m FMRI proto="transport-protocols"
```

Todos los servidores que se proporcionan con el software Oracle Solaris necesitan sólo una entrada de perfil que especifique `proto` como `tcp6`, `udp6` o `sctp6`. No obstante, el servidor de shell remoto (`shell`) y el servidor de ejecución remoto (`exec`) se componen en la actualidad de una sola instancia de servicio, que necesita un valor de `proto` que contenga los valores de `tcp` y `tcp6only`. Por ejemplo, para establecer el valor de `proto` para `shell`, debe ejecutarse el comando siguiente:

```
# inetadm -m network/shell:default proto="tcp,tcp6only"
```

Para obtener más información sobre la escritura en servidores activados para IPv6 que utilizan sockets, consulte las extensiones de IPv6 de Socket API en la *Programming Interfaces Guide*.

Puntos que tener en cuenta al configurar un servicio para IPv6

Al agregar o modificar un servicio para IPv6, tenga en cuenta lo siguiente:

- El valor de `proto` debe establecerse en `tcp6`, `sctp6` o `udp6` para permitir conexiones IPv4 o IPv6. Si el valor de`proto` se establece en `tcp`, `sctp` o `udp`, el servicio utiliza sólo IPv4.
- Si bien puede agregar una instancia de servicio que utilice sockets SCTP de uno a varios estilos para `inetd`, no es recomendable. `inetd` no funciona con sockets SCTP de uno a varios estilos.
- Si un servicio necesita dos entradas debido a diferencias en las propiedades de `wait-status` o `exec`, debe crear dos instancias o servicios a partir del servicio original.

Protocolo ND de IPv6

IPv6 introduce el protocolo ND (Neighbor Discovery), tal como se describe en RFC 2461, Neighbor Discovery for IP Version 6 (IPv6) (`http://www.ietf.org/rfc/rfc2461.txt?number=2461`).

Esta sección trata sobre las características siguientes del protocolo ND:

Mensajes de ICMP del protocolo ND

El protocolo ND define cinco mensajes nuevos de ICMP (Internet Control Message Protocol). Dichos mensajes tienen los objetivos siguientes:

- **Solicitud de enrutador**: al activarse una interfaz, los hosts pueden enviar mensajes de solicitud de enrutador. Se solicita a los enrutadores que generen inmediatamente anuncios de enrutador, en lugar de hacerlo la próxima vez que se hubiera programado.
- **Anuncio de enrutador**: los enrutadores anuncian su presencia, así como varios parámetros de vínculos y de Internet. Los enrutadores anuncian de manera periódica o como respuesta a un mensaje de solicitud de enrutador. Los anuncios de enrutador contienen prefijos que se usan para la determinación de onlinks o configuración de direcciones, un valor de límite de salto propuesto, etcétera.
- **Solicitud de vecino**: los nodos envían mensajes de solicitud de vecino para determinar la dirección de capa de vínculo de un vecino. Los mensajes de solicitud de vecino también sirven para verificar que se pueda contactar con un vecino mediante una dirección de capa de vínculo almacenada en caché. Asimismo, las solicitudes de vecino se usan para detectar direcciones duplicadas.
- **Anuncio de vecino**: un nodo envía mensajes de anuncio de vecino como respuesta a un mensaje de solicitud de vecino. El nodo también puede enviar anuncios de vecino no solicitados para anunciar un cambio de dirección de capa de vínculo.
- **Redirección**: los enrutadores emplean mensajes de redirección para indicar a los hosts el mejor primer salto para acceder a un destino, o para indicar que el destino está en el mismo vínculo.

Proceso de configuración automática

Esta sección proporciona una descripción general de los pasos habituales que realizan las interfaces durante la configuración automática. La configuración automática se efectúa sólo en vínculos que permiten multidifusión.

1. Una interfaz que permite multidifusión se activa, por ejemplo, al iniciar el sistema de un nodo.
2. El nodo empieza el proceso de configuración automática generando una dirección local de vínculo para la interfaz.

   La dirección local de vínculo se forma a partir de la dirección MAC de la interfaz.
3. El nodo envía un mensaje de solicitud de vecino que contiene la dirección local de vínculo provisional como destino.

   La finalidad del mensaje es verificar que otro nodo del vínculo no esté utilizando ya la dirección de prueba. Tras verificarla, la dirección local de vínculo puede asignarse a una interfaz.

   a. Si la dirección propuesta ya la usa otro nodo, dicho nodo genera un anuncio de vecino para informar de ello.

   b. Si otro nodo intenta utilizar la misma dirección, dicho nodo también envía una solicitud de vecino para el destino.

      La cantidad de transmisiones y retransmisiones de solicitudes de vecino, así como el retraso entre solicitudes consecutivas, dependen de cada vínculo. Si es preciso, establezca estos parámetros.
4. Si un nodo determina que la dirección local de vínculo de prueba no es exclusiva, se detiene el proceso de configuración automática. De ser así, la dirección local de vínculo de la interfaz se debe configurar manualmente.

   Para simplificar la recuperación, puede especificar otro ID de interfaz que anule el predeterminado. De este modo, el mecanismo de configuración automática puede reanudar su funcionamiento con el nuevo ID de interfaz, que en principio es exclusivo.
5. Si un nodo determina que la dirección local de vínculo de prueba es exclusiva, el nodo la asigna a la interfaz.

   En ese momento, el nodo dispone de conectividad IP con nodos vecinos. Los demás pasos de la configuración automática los efectúan solamente hosts.

Obtención de un anuncio de enrutador

La fase siguiente de la configuración automática consiste en obtener un anuncio de enrutador o determinar que no hay enrutadores. Si hay enrutadores, éstos envían anuncios de enrutador para indicar la clase de configuración automática que debe ejecutar un host.

Los enrutadores envían periódicamente solicitudes de enrutador. No obstante, el retraso entre los sucesivos anuncios suele ser superior a lo que puede esperar un host que efectúa la configuración automática. Para obtener rápidamente un anuncio, el host envía una o varias solicitudes de enrutador al grupo multidifusión de todos los enrutadores.

Variables en la configuración de prefijos

Los anuncios de enrutador pueden contener también variables de prefijo con información que la configuración automática de direcciones emplea en la generación de prefijos. El campo de configuración automática de direcciones sin estado de los anuncios de enrutador se procesa de manera independiente. El indicador de configuración de direcciones, un campo de opción que contiene información de prefijo, indica si la opción se aplica también a la configuración automática sin estado. Si se aplica el campo de opción, otros campos de opciones contienen un prefijo de subred con valores continuamente vigentes. Estos valores indican la duración que tendrán la validez y preferencia de las direcciones creadas a partir del prefijo.

Debido a que los enrutadores generan periódicamente anuncios de enrutador, los hosts reciben anuncios nuevos de manera constante. Los hosts activados para IPv6 procesan la información que hay en cada anuncio. Los hosts se agregan a la información. También ponen al día la información recibida en anuncios anteriores.

Exclusividad de las direcciones

Por motivos de seguridad, antes de asignarse a la interfaz debe verificarse que todas las direcciones sean exclusivas. Es distinto en el caso de direcciones creadas con configuración automática sin estado. La exclusividad de una dirección la determina la parte de la dirección formada por un ID de interfaz. Por eso, si un nodo ya ha comprobado la exclusividad de una dirección local de vínculo, no hace falta verificar las direcciones adicionales una a una. Las direcciones deben crearse a partir del mismo ID de interfaz. Por su parte, debe comprobarse la exclusividad de todas las direcciones que se obtengan manualmente. Los administradores de sistemas de algunos sitios consideran que el esfuerzo y los recursos dedicados a detectar direcciones duplicadas son mayores que sus ventajas. En estos sitios, la detección de direcciones duplicadas se puede desactivar estableciendo un indicador de configuración según la interfaz.

Para acelerar el proceso de configuración automática, un host puede generar su propia dirección local de vínculo y verificar su exclusividad, mientras el host espera un anuncio de enrutador. Un enrutador podría retrasar durante unos segundos la respuesta a una solicitud de enrutador. Por lo tanto, el tiempo total que se necesita para completar la configuración automática puede ser considerablemente superior si los dos pasos se realizan en serie.

Solicitud e inasequibilidad de vecinos

El protocolo ND utiliza mensajes de *solicitud de vecino* para determinar si la misma dirección unidifusión tiene asignado más de un nodo. La *detección de inasequibilidad de vecinos* descubre el error de un vecino o de la ruta de reenvío del vecino. Esta clase de detección precisa la confirmación positiva de que los paquetes que se envían a un vecino lleguen realmente a su destino. Asimismo, la detección de inasequibilidad de vecinos determina que la capa IP del nodo procese correctamente los paquetes.

La detección de inasequibilidad de vecinos utiliza la confirmación a partir de dos puntos de referencia: los protocolos de capa superior y los mensajes de solicitud de vecino. Si es posible, los protocolos de capa superior brindan la confirmación positiva de que una conexión *avanza en el reenvío*. Por ejemplo, si se reciben reconocimientos de TCP, se confirma la correcta entrega de los datos enviados con anterioridad.

Si un nodo no obtiene una confirmación positiva de los protocolos de capa superior, dicho nodo envía mensajes de solicitud de vecino unidifusión. Estos mensajes solicitan anuncios de vecino como confirmación de asequibilidad a partir del próximo salto. Para reducir el tráfico redundante en la red, los mensajes sonda se envían sólo a los vecinos a los que el nodo esté enviando paquetes.

Algoritmo de detección de direcciones duplicadas

Para asegurarse de que todas las direcciones configuradas puedan ser exclusivas en un determinado vínculo, los nodos ejecutan en las direcciones un algoritmo de *detección de direcciones duplicadas*. Los nodos deben ejecutar el algoritmo antes de asignar las direcciones a una interfaz. El algoritmo de detección de direcciones duplicadas se ejecuta en todas las direcciones.

El proceso de configuración automática que se describe en esta sección detección de direcciones duplicadas sólo es válido para hosts, no para enrutadores. Debido a que la configuración automática de hosts emplea información anunciada por enrutadores, éstos se deben configurar por otros medios. Sin embargo, los enrutadores generan direcciones locales de vínculo mediante el mecanismo que se explica en este capítulo. Además, en principio los enrutadores deben superar correctamente el algoritmo de detección de direcciones duplicadas en todas las direcciones antes de asignar la dirección a una interfaz.

Anuncios de proxy

Un enrutador que acepta paquetes de parte de una dirección de destino puede ejecutar anuncios que no se anulan. El enrutador puede aceptar paquetes de parte de una dirección de destino que sea incapaz de responder a solicitudes de destino. En la actualidad no se especifica el uso de

proxy. Ahora bien, el anuncio de proxy se puede utilizar para ocuparse de casos como nodos móviles que se han desplazado fuera del vínculo. El uso de proxy no se ha concebido como mecanismo general para controlador nodos que no implementen este protocolo.

Equilibrio de la carga entrante

Los nodos con interfaces duplicadas quizá deban equilibrar la carga de la recepción de paquetes entrantes en las distintas interfaces de red del mismo vínculo. Estos nodos disponen de varias direcciones locales de vínculo asignadas a la misma interfaz. Por ejemplo, un solo controlador de red puede representar a varias tarjetas de interfaz de red como una única interfaz lógica que dispone de varias direcciones locales de vínculo.

El equilibrio de carga se controla permitiendo que los enrutadores omitan la dirección local de vínculo de origen de los paquetes de anuncio de enrutador. Por consiguiente, los vecinos deben emplear mensajes de solicitud de vecino para aprender las direcciones locales de vínculo de los enrutadores. Los mensajes de anuncio de vecino devueltos pueden contener direcciones locales de vínculo diferentes, en función del que haya emitido la solicitud.

Cambio de dirección local de vínculo

Un nodo que sepa que se ha modificado su dirección local de vínculo puede enviar paquetes de anuncios de vecinos multidifusión no solicitados. El nodo puede enviar paquetes multidifusión a todos los nodos para actualizar las direcciones locales de vínculo almacenadas en caché que ya no sean válidas. El envío de anuncios no solicitados es una simple mejora del rendimiento. El algoritmo de detección de inasequibilidad de vecinos se asegura de que todos los nodos descubran la nueva dirección de manera fiable, aunque ello comporte un retraso algo mayor.

Comparación del protocolo ND con ARP y protocolos relacionados con IPv4

El funcionamiento del protocolo ND de IPv6 equivale a combinar los siguientes protocolos de IPv4: ARP (Address Resolution Protocol), ICMP (Internet Control Message Protocol), Router Discovery e ICMP Redirect. IPv4 carece de un protocolo general establecido y de un mecanismo para detectar la inasequibilidad de vecinos. Sin embargo, los requisitos de host especifican determinados algoritmos para la detección de portales inactivos. La detección de portales inactivos es un subconjunto de los problemas que soluciona la detección de inasequibilidad de vecinos.

En la lista siguiente se comparan el protocolo ND con el conjunto correspondiente de protocolos de IPv4.

- El descubrimiento de enrutador forma parte del conjunto básico de protocolos de IPv6. Los hosts de IPv6 no necesitan aplicar el comando `snoop` a los protocolos de enrutamiento para buscar un enrutador. IPv4 utiliza ARP, descubrimiento de enrutadores ICMP y redirección de ICMP para el descubrimiento de enrutador.
- Los anuncios de enrutador de IPv6 llevan direcciones locales de vínculo. Para resolver la dirección local de vínculo no hace falta un intercambio adicional de paquetes.
- Los anuncios de enrutador llevan los prefijos de sitio para un vínculo. No hace falta un mecanismo aparte para configurar la máscara de red, como sucede con IPv4.
- Los anuncios de enrutador permiten la configuración automática de direcciones. En IPv4 no se implementa la configuración automática.
- El protocolo ND permite que los enrutadores de IPv6 anuncien una unidad de transmisión máxima (MTU, Maximum Transmission Unit) para hosts para utilizarse en el vínculo. Por lo tanto, todos los nodos emplean el mismo valor de MTU en los vínculos que carecen de una MTU bien definida. Podría ser que los hosts de IPv4 de una misma red tuvieran distintas MTU.
- A diferencia de las direcciones de emisión IPv4, las multidifusiones de resolución de direcciones IPv6 se distribuyen en cuatro mil millones (2^32) de direcciones multidifusión, lo cual reduce significativamente las interrupciones por resolución de direcciones en nodos que no sean el de destino. Además, no es recomendable interrumpir sistemas que no sean IPv6.
- Las redirecciones de IPv6 contienen la dirección local de vínculo del primer salto nuevo. Al recibir una redirección no hace falta una resolución de direcciones aparte.
- Una misma red IPv6 puede tener asociados varios prefijos de sitio. De forma predeterminada, los hosts aprenden todos los prefijos de sitio locales a partir de anuncios de enrutador. Sin embargo, es posible configurar los enrutadores para que omitan todos o algunos prefijos de anuncios de enrutador. En esos casos, los hosts dan por sentado que los destinos se encuentran en redes remotas. Por lo tanto, los hosts envían el tráfico a enrutadores. Así pues, un enrutador puede ejecutar redirecciones si es preciso.
- A diferencia de IPv4, el destinatario de un mensaje de redirección de IPv6 da por sentado que el próximo salto nuevo se da en la red local. En IPv4, un host hace caso omiso de los mensajes de redirección que especifiquen un próximo salto que no se ubique en la red local, conforme a la máscara de red. El mecanismo de redirección de IPv6 es análogo a la función XRedirect de IPv4. El mecanismo de redirección es útil en vínculos de soportes compartidos y de no emisión. En esta clase de redes, los nodos no deben comprobar todos los prefijos de destinos de vínculo local.
- La detección de inasequibilidad de vecinos de IPv6 mejora la distribución de paquetes si hay enrutadores que funcionan mal. Esta capacidad mejora la distribución de paquetes en vínculos con particiones o que funcionan parcialmente mal. Asimismo, mejora la distribución de paquetes en nodos que modifican sus direcciones locales de vínculo. Por ejemplo, los nodos móviles pueden salir de la red local sin perder ninguna clase de

conectividad debido a memorias caché de ARP que hayan quedado obsoletas. IPv4 carece de método equivalente para la detección de inasequibilidad de vecinos.

- A diferencia de ARP, el protocolo ND detecta errores parciales en vínculos mediante la detección de inasequibilidad de vecinos. El protocolo ND evita el envío de tráfico a vecinos si no existe conectividad bidireccional.
- Las direcciones locales de vínculo permiten la identificación exclusiva de enrutadores y los hosts de IPv6 mantienen las asociaciones de enrutador. La capacidad de identificar enrutadores es necesaria en anuncios de enrutador y mensajes de redirección. Los hosts deben mantener asociaciones de enrutador si el sitio emplea prefijos globales nuevos. IPv4 carece de un método equiparable para la identificación de enrutadores.
- Debido a que los mensajes de protocolo ND tienen un límite de salto de 255 en la recepción, dicho protocolo es inmune a ataques de spoofing provenientes de nodos que no están en el vínculo. Por el contrario, los nodos que no están en vínculos de IPv4 pueden enviar mensajes de redirección de ICMP. Asimismo, los nodos que no están en vínculos de IPv4 pueden enviar mensajes de anuncio de enrutador.
- La colocación de resolución de direcciones en la capa de ICMP hace que el protocolo ND sea más independiente en cuanto a soportes que ARP. por consiguiente, se pueden utilizar la autenticación IP y los mecanismos de seguridad estándar.

Enrutamiento de IPv6

El enrutamiento de IPv6 es casi idéntico al de IPv4 en la dirección de enrutamiento entre dominios sin clase (CIDR). La única diferencia estriba en que las direcciones son IPv6 de 128 bits, en lugar de IPv4 de 32 bits. Con extensiones sumamente sencillas, todos los algoritmos de enrutamiento de IPv4, por ejemplo OSPF, RIP, IDRP e IS-IS, son válidos para enrutar IPv6.

Asimismo, IPv6 incluye extensiones sencillas de enrutamiento que admiten nuevas y potentes posibilidades de enrutamiento. A continuación se describen las nuevas funciones de enrutamiento:

- La selección del proveedor se basa en las directrices, el rendimiento, los costes, etcétera
- Movilidad de los hosts, enrutamiento a la ubicación actual
- Redireccionamiento automático, enrutamiento a la dirección nueva

Para acceder a las nuevas funciones de enrutamiento, debe crear secuencias de direcciones IPv6 que utilicen la opción de enrutamiento de IPv6. Un origen de IPv6 utiliza la opción de enrutamiento para obtener uno o varios nodos intermedios, o un grupo topológico, que debe visitarse en dirección al destino del paquete. Es una función muy parecida a las opciones de ruta de registro y ruta holgada fija en origen de IPv4.

Para que las secuencias de direcciones sean una función general, los hosts de IPv6 deben, en la mayoría de los casos, invertir las rutas de un paquete que reciba un host. El paquete se debe autenticar correctamente mediante el encabezado de autenticación de IPv6. El paquete debe contener secuencias de direcciones para devolver el paquete al emisor. Esta técnica obliga a que las implementaciones de hosts de IPv6 admitan el control y la inversión de las rutas de origen. El control y la inversión de las rutas de origen es la clave que permite a los proveedores trabajar con los hosts que implementan las nuevas funciones de IPv6 como la selección de proveedor y las direcciones extendidas.

Anuncio de enrutador

En vínculos con capacidad multidifusión y punto a punto, cada enrutador envía, de forma periódica, al grupo multidifusión un paquete de anuncios de enrutador que informa de su disponibilidad. Un host recibe anuncios de enrutador de todos los enrutadores, y confecciona una lista de enrutadores predeterminados. Los enrutadores generan anuncios de enrutador con la suficiente frecuencia para que los hosts aprendan su presencia en pocos minutos. Sin embargo, los enrutadores no anuncian con suficiente frecuencia como para que una falta de anuncios permita detectar un error de enrutador. La detección de errores es factible mediante un algoritmo de detección independiente que determina la inasequibilidad de vecinos.

Prefijos de anuncio de enrutador

Los anuncios de enrutador contienen una lista de prefijos de subred que se usan para determinar si un host se encuentra en el mismo vínculo que el enrutador. La lista de prefijos también se utiliza en la configuración de direcciones autónomas. Los indicadores que se asocian con los prefijos especifican el uso concreto de un determinado prefijo. Los hosts utilizan los prefijos del vínculo anunciados para configurar y mantener una lista que se emplea para decidir si el destino de un paquete se encuentra en el vínculo o fuera de un enrutador. Un destino puede encontrarse en un vínculo aunque dicho destino no aparezca en ningún prefijo del vínculo que esté anunciado. En esos casos, un enrutador puede enviar una redirección. La redirección indica al remitente que el destino es un vecino.

Los anuncios de enrutador, y los indicadores de prefijo, permiten a los enrutadores informar a los hosts sobre cómo efectuar la configuración automática de direcciones sin estado.

Mensajes de anuncio de enrutador

Los mensajes de anuncio de enrutador contienen también parámetros de Internet, por ejemplo el límite de salto que los hosts deben emplear en los paquetes salientes. También es posible que los mensajes de anuncio de enrutador contengan parámetros de vínculo, por ejemplo la MTU de vínculo. Esta función permite la administración centralizada de los parámetros importantes. Los parámetros se pueden establecer en enrutadores y propagarse automáticamente a todos los hosts que estén conectados.

Los nodos llevan a cabo la resolución de direcciones enviando al grupo de multidifusión una solicitud de vecino que pide al nodo de destino que devuelva su dirección de capa de vínculo. Los mensajes de solicitud de vecino multidifusión se envían a la dirección multidifusión de nodo solicitado de la dirección de destino. El destino devuelve su dirección de capa de vínculo en un mensaje de anuncio de vecino unidifusión. Para que el iniciador y el destino resuelvan sus respectivas direcciones de capa de vínculo basta con un solo par de paquetes de solicitud-respuesta. El iniciador incluye su dirección de capa de vínculo en la solicitud de vecino.

Extensiones de IPv6 para servicios de nombres de Oracle Solaris

En esta sección se describen los cambios de denominación incorporados con la implementación de IPv6. Puede almacenar direcciones IPv6 en cualquiera de los servicios de nombres de Oracle Solaris, NIS, LDAP, DNS y archivos. También puede utilizar NIS en transportes IPv6 RPC para recuperar datos de NIS.

Extensiones de DNS para IPv6

El registro de recursos AAAA, propio de IPv6, se ha especificado en la RFC 1886 *DNS Extensions to Support IP Version 6*. Este registro AAAA asigna un nombre de host en una dirección IPv6 de 128 bits. El registro PTR se sigue usando en IPv6 para asignar direcciones IP en nombres de host. Las cuatro porciones de 32 bits de las direcciones de 128 bits se invierten para una dirección IPv6. Cada porción se convierte a su correspondiente valor ASCII hexadecimal. A continuación, se agrega `ip6.int`.

Cambios en los comandos de servicio de nombres

Para admitir IPv6, busque direcciones IPv6 con los comandos del servicio de nombres vigente. Por ejemplo, el comando `ypmatch` funciona con las nuevas asignaciones NIS. El comando `nslookup` busca los nuevos registros AAAA en DNS.

Admisión de NFS y RPC IPv6

NFS y Remote Procedure Call (RPC) son programas totalmente compatibles con IPv6. No han cambiado los comandos ya existentes relacionados con los servicios de NFS. Además, la mayoría de las aplicaciones RPC también funcionan con IPv6 sin cambios. Es posible que haya que actualizar algunas aplicaciones RPC avanzadas con reconocimiento de transporte.

Admisión de IPv6 en ATM

Oracle Solaris admite IPv6 en ATM, PVC (Permanent Virtual Circuits, circuitos virtuales permanentes) y SVC (Static Switched Virtual Circuits, circuitos virtuales conmutados estáticos).

Índice

C

D

E

M

N

O

P

R

S

T

U

V