Troubleshooting Network Administration Issues in Oracle® Solaris 11.2

ORACLE®

Contents

Using This Documentation

- **Overview** – Describes tasks for troubleshooting network configuration issues in the Oracle Solaris operating system (OS).
- **Audience** – System administrators.
- **Required knowledge** – Understanding of basic and advanced network administration concepts and practices.

Product Documentation Library

Late-breaking information and known issues for this product are included in the documentation library at http://www.oracle.com/pls/topic/lookup?ctx=E36784.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Feedback

Provide feedback about this documentation at http://www.oracle.com/goto/docfeedback.

♦ ♦ ♦ **C H A P T E R   1**

Troubleshooting Network Administration Issues

This chapter describes how to troubleshoot various problems that can occur on a network, including issues with network configuration, network connectivity, and various error conditions.

If you are transitioning from Oracle Solaris 10 to Oracle Solaris 11, see Chapter 7, "Managing Network Configuration," in "Transitioning From Oracle Solaris 10 to Oracle Solaris 11.2 " for more information about network administration in this release.

This chapter contains the following topics:

- "Answers to Common Network Administration Questions" on page 7
- "Troubleshooting Network Connectivity and Configuration Issues" on page 11
- "Troubleshooting Interface Configuration Error Conditions" on page 16
- "Troubleshooting Issues With IPv6 Deployment" on page 18
- "Troubleshooting IPMP Configuration" on page 20
- "Troubleshooting Issues With VRRP and the Oracle Solaris Bundled IP Filter" on page 22
- "Resources for Monitoring and Detecting Problems on a TCP/IP Network" on page 22
- "Monitoring Network Traffic on Datalinks and Flows" on page 24

Answers to Common Network Administration Questions

Refer to the following troubleshooting information when using the *fixed mode* for network administration. For information about troubleshooting network administration issues when using the *reactive mode*, see "Answers to Common Profile-Based Network Configuration Questions" on page 25. For further details, see "About Network Configuration Modes" in "Configuring and Administering Network Components in Oracle Solaris 11.2 ".

Question: How do I determine which networking mode my system is using after an installation?

Answer: The networking mode is determined by the profile that is activated during installation. If the `DefaultFixed` profile is activated, you are in the fixed mode. If the `Automatic` profile is activated, you are in the reactive mode. To determine which mode is currently active on your system, use the `netadm list` command as follows:

```
# netadm list
```

Question: My system defaulted to reactive mode after an installation. How can I fix this problem?

Answer: You need to switch to the fixed mode by enabling the `DefaultFixed` profile. Use the `netadm` command as follows to switch the active profile:

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

Question: I manually configured my system during an installation and the `netadm list` command shows that I am using the fixed mode, but my system's network is still not configured correctly. What should I do?

Answer: The answer depends on which network component is not configured correctly. When in the fixed mode, you use the `dladm` and `ipadm` commands to configure the network. Given the type of configuration parameters that can be set at installation time, most likely an IP interface or address is not configured correctly.

To determine which network components need to be reconfigured, start by displaying the current network configuration as follows:

```
# ipadm
```

If the IP address is incorrect, you will need to delete that address and then create the correct IP address, for example, a static IP address or a DHCP address.

The following example shows how to delete the IPv6 `addrconf` portion of an IP configuration. In this example the IPv6 `addrconf` address is determined by running the `ipadm` command:

```
# ipadm
NAME            CLASS/TYPE STATE      UNDER     ADDR
lo0             loopback   ok         --        --
lo0/v4          static     ok         --        127.0.0.1/8
lo0/v6          static     ok         --        ::1/128
net0            ip         ok         --        --
net0/v4         dhcp       ok         --        10.1.1.10/24
net0/v6         addrconf   ok         --        fe80::8:20ff:fe90:10df/10
# ipadm delete-addr net2/v6
# ipadm
NAME            CLASS/TYPE STATE      UNDER     ADDR
lo0             loopback   ok         --        --
lo0/v4          static     ok         --        127.0.0.1/8
lo0/v6          static     ok         --        ::1/128
net0            ip         ok         --        --
net0/v4         dhcp       ok         --        10.1.1.10/24
```

Then, set just the netmask property without deleting other existing IP configuration as follows:

```
# ipadm set-addrprop -p prefixlen=len addrobj-name
```

See Chapter 3, "Configuring and Administering IP Interfaces and Addresses in Oracle Solaris," in "Configuring and Administering Network Components in Oracle Solaris 11.2 " for complete instructions.

Question: How do I configure a persistent default route on my system?

Answer: Because the /etc/defaultrouter file is deprecated in Oracle Solaris 11, you can no longer manage default routes by editing this file. Also, after a fresh installation, you can no longer check this file to determine the system's default route.

Display and configure routes (default or otherwise) as follows:

- Display routes that are created persistently as follows:

  ```
  # route -p show
  ```
- Add a persistent default route as follows:

  ```
  # route -p add default ip-address
  ```
- Display the currently active routes on a system as follows:

  ```
  # netstat -rn
  ```

See "Creating Persistent (Static) Routes" in "Configuring and Administering Network Components in Oracle Solaris 11.2 ".

Question: How do I display the MAC address of a system?

Answer: Display the MAC addresses of the physical links in a system as follows:

```
# dladm show-phys -m
```

In Oracle Solaris 10, the ifconfig command is used to display similar information.

Display the MAC addresses of all of the links in a system (physical and non-physical) as follows:

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

Question: I can no longer use the dladm show-dev command to display the physical links that are in my system? What command do I use now?

Answer: Use the dladm show-phys command as follows:

```
# dladm show-phys
LINK            MEDIA           STATE    SPEED  DUPLEX   DEVICE
net0            Ethernet        up       0      unknown  vnet0
```

Question: How do I display the mapping between link names, devices, and locations on a system?

Answer: Use the `dladm show-phys` command with the `-L` option as follows:

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

Question: What command do I use to determine the MTU range my system supports?

Answer: Use the `ipadm show-ifprop` command to determine this information, as shown in the following example. The last column displays the supported MTU ranges.

```
# ipadm show-ifprop -p mtu interface
```

Question: What if the naming services settings on my system are lost or not configured correctly after an installation?

Answer: If you using the fixed mode, the naming services configuration should be what you specified during the installation. In this release, naming services are configured through the Service Management Facility (SMF). See Chapter 4, "Administering Naming and Directory Services on an Oracle Solaris Client," in "Configuring and Administering Network Components in Oracle Solaris 11.2 " for instructions on how to configure naming services and how to import naming services configuration on a client system after an installation.

Note - If you are using the reactive mode, see "Creating Locations" in "Configuring and Administering Network Components in Oracle Solaris 11.2 ".

Question: How can I start over and reconfigure all of the network settings for my system?

Answer: You can unconfigure and reconfigure an Oracle Solaris instance, including network settings, as follows:

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

Question: What is the difference between creating a virtual LAN (VLAN) with the `dladm create-vlan` command and a virtual NIC (VNIC) with the `dladm create-vnic -v VID ...` command? Also, what are the unique features of both commands that would dictate the use of one rather than the other?

Answer: Depending on your networking needs and what you trying to accomplish, you would use each of these features for different purposes.

A VLAN is a subdivision of a LAN at the datalink layer (L2) of the network stack. VLANs enable you to divide your network into subnetworks without having to add to the physical network environment. So, the subnetworks are virtual and they share the same physical network resources. VLANs facilitate network administration by using smaller groups that are easier to maintain.

VNICs are virtual network devices that use the same datalink interface as a physical network interface card (NIC). You configure VNICs over an underlying datalink. When VNICs are configured, they behave like physical NICs. Depending on the network interface that is in use, you can explicitly assign a MAC address to a VNIC other than the default address.

For more information about which network administration strategies to use, see Chapter 1, "Summary of Oracle Solaris Network Administration," in "Strategies for Network Administration in Oracle Solaris 11.2 ".

Troubleshooting Network Connectivity and Configuration Issues

The following are general guidelines for troubleshooting network connectivity and configuration issues.

One of the first signs of trouble on a network is loss of communications by one or more of the hosts. If a host does not to come up the first time it is added to the network, the problem might be a faulty NIC or a problem with network daemon that is managed by SMF.

If a single host that previously was connected to the network suddenly develops a network problem, the problem could be the system's network interface configuration. If the hosts on a network can communicate with each other but not with other networks, the problem could be the router. Another problem could be with the other network.

Performing Basic Network Configuration Checks

You can troubleshoot network configuration problems with a single system by using the `dladm` and `ipadm` commands. These two commands, when used without any options, provide useful information about your current network configuration.

The following are some of the ways in which you can use these commands to troubleshoot configuration issues:

- Use the `dladm` command to display general information about all of the datalinks that are on a system:

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

- Display information about the mapping between the datalinks, their generic names, and the corresponding device instances as follows:

```
# dladm show-phys
LINK       MEDIA       STATE     SPEED     DUPLEX      DEVICE
net0       Ethernet    up        1000      full        e1000g0
```

- Use the `ipadm` command to display general information about all of the IP interfaces that are on a system:

```
# ipadm
NAME           CLASS/TYPE STATE      UNDER   ADDR
lo0            loopback   ok         --      --
 lo0/v4         static     ok         --      127.0.0.1/8
 lo0/v6         static     ok         --      ::1/128
net0           ip         ok         --      --
 net0/v4        static     ok         --      10.132.146.233/24
```

- Use the `ipadm show-if` *interface* command to display information about a specific IP interface:

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

- Display information about all of the interfaces on a system as follows:

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

- Display information about all of the IP addresses on the system as follows:

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

- Use the `ipadm show-addr` *interface* command to display information about a specific interface's IP address:

```
# ipadm show-addr net0
ADDROBJ         TYPE      STATE      ADDR
net0/v4         dhcp      ok         10.153.123.225/24
```

- Display the properties of a specific IP address as follows::

```
# ipadm show-addrprop net1/v4
```

```
ADDROBJ         PROPERTY    PERM CURRENT         PERSISTENT      DEFAULT    POSSIBLE
net0/v4         broadcast  r-   10.153.123.255  --              10.255.255.255  --
net0/v4         deprecated rw   off             --              off   on,off
net0/v4         prefixlen  rw   24              --              8    1-30,32
net0/v4         private    rw   off             --              off   on,off
net0/v4         reqhost    r-   --              --              --    --
net0/v4         transmit   rw   on              --              on    on,off
net0/v4         zone       rw   global          --              global
```

For more information, see the ipadm(1M) man page.

Verifying That Network Services and Daemons Are Running

A critical step in troubleshooting issues with network connectivity is to determine the current status of all of the SMF network services that are running on the system.

You can verify the current status of all of the SMF network services that are running on the system as follows:

$ **svcs svc:/network/***

If the command output shows that a service is disabled or in a maintenance state, you can obtain more information about that particular service as follows:

$ **svcs -xv** *service-name*

For example, you would obtain more information about the svc:/network/loopback:default SMF network service as follows:

```
$ svcs -xv svc:/network/loopback:default
svc:/network/loopback:default (loopback network interface)
 State: online since Thu Dec 05 19:30:54 2013
   See: man -M /usr/share/man -s 1M ifconfig
   See: /system/volatile/network-loopback:default.log
Impact: None.
```

Running Basic Network Diagnostic Checks

Less obvious causes of network problems are those that degrade network performance. If the network is having problems, you can run a series of software checks to diagnose and fix basic problems. For example, you can use the ping command to quantify problems, such as the loss of packets by a host. Or, you can use the netstat command to display routing tables and protocol statistics. For more information about the various methods that you can use to troubleshoot these types of networking problems, see "Resources for Monitoring and Detecting Problems on a TCP/IP Network" on page 22.

For information about performing network diagnostics by using the network monitor utility, see Chapter 4, "Performing Network Diagnostics With the network-monitor Transport Module Utility".

Third-party network diagnostic programs also provide a number of tools for troubleshooting network issues. Refer to the third-party product documentation for specifics.

▼ How to Perform Basic Network Software Checking

1. **Use the netstat command to display network information.**

   The netstat command displays a variety of useful information for troubleshooting network connectivity issues. The type of information that is displayed depends on the options that you use. See "Monitoring Network Status With the netstat Command" in "Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 " and the netstat(1M) man page.

2. **Check the hosts database to ensure that all of the entries are correct and current.**

   For more information about the /etc/inet/hosts database, see the hosts(4) man page.

3. **Try to connect to the local host by using the telnet command.**

   For more information, see the telnet(1) man page.

4. **Ensure that the inetd network daemon is running.**

   ```
   # /usr/bin/pgrep inetd
   883
   ```

   The previous output indicates that the inetd daemon is running on the system with the process ID 883.

5. **If IPv6 is enabled on your network, verify that the in.ndpd daemon is running.**

   ```
   # /usr/bin/pgrep in.ndpd
   822
   ```

   The previous output indicates that the inetd daemon is running on the system with the process ID 882.

6. **Check the system's router and routing information.**

   - **Display a system's persistent route as follows:**

     ```
     # route -p show
     ```

     See "Troubleshooting Issues When Adding a Persistent Route" on page 15.

■ **Display the configuration that is in the routing table as follows:**

```
# netstat -nr
```

Troubleshooting Issues When Adding a Persistent Route

The route command is used to manage the network routing tables. Use the -p option to ensure that any changes you make to the network routing tables persist across system reboots.

Note - It is important to use care when adding persistent routes to make sure that any routes that are being added do not conflict with the existing persistent configuration.

Check whether a route already exists in the persistent configuration as follows:

```
# route -p show
persistent: route add default 10.153.123.1 -ifp net0
```

If the route already exists in the persistent configuration, the information in the network routing tables (which is not persistent configuration) might differ from the persistent configuration.

The following example illustrates this point further. In this example, an attempt is made to add a persistent route to net1. However, the command fails because a persistent route already exists for net0, per the previous example's output.

```
# route -p add default 10.153.123.1 -ifp net1
add net default: gateway 10.153.123.1
add persistent net default: gateway 10.153.123.1: entry exists
Warning: persistent route might not be consistent with routing table.
```

Running the route -p show command again reveals that the persistent route did not change and is still configured for net0, as shown in the following output:

```
# route -p show
persistent: route add default 10.153.123.1 -ifp net0
```

However, the command did change the routing tables in the kernel to use net1, as shown in the following output:

```
# netstat -nr

Routing Table: IPv4
  Destination         Gateway          Flags  Ref    Use Interface
------------------- ------------------- ----- ----- ---------- ---------
default            10.153.123.1        UG      2        1 net1
10.153.123.0       10.153.123.78       U       3        0 net1
127.0.0.1          127.0.0.1           UH      2      466 lo0
```

.
.
.

Therefore, it is always best to delete any existing persistent route configuration prior to adding a new route. See "Creating Persistent (Static) Routes" in "Configuring and Administering Network Components in Oracle Solaris 11.2 " for more information.

Troubleshooting Interface Configuration Error Conditions

This section contains the following topics:

- "Cannot Assign IP address by Using the `ipadm create-addr` command" on page 16
- "Error Message: `cannot create address object: Invalid argument provided` Displayed During IP Address Configuration" on page 16
- "Error Message: `cannot create address: Persistent operation on temporary object` Displayed During IP Interface Configuration" on page 17

Cannot Assign IP address by Using the `ipadm create-addr` command

With the traditional `ifconfig` command that is used for network configuration in Oracle Solaris 10, you can plumb and assign an IP address by using a single command. In Oracle Solaris 11, you use the `ipadm` command to configure IP interfaces and addresses.

The following example assumes that a static IP address is being assigned to an interface. This process involves two steps. First, create or plumb the IP interface by using the `ipadm create-ip` command. Then, assign an IP address to the interface by using the `ipadm create-addr` command:

```
# ipadm create-ip interface
# ipadm create-addr -T addr-type -a address addrobj
```

Error Message: `cannot create address object: Invalid argument provided` Displayed During IP Address Configuration

The address object identifies a specific IP address that is bound to an IP interface. The address object is a unique identifier for each IP address on the IP interface. You must specify a different address object to identify a second IP address that you assign to the same IP interface. If you

want to use the same address object name, you must delete the first instance of the address object before assigning it to a different IP address.

Use one of the following methods:

- Specify a different address object to identify a second IP address as follows:

```
# ipadm show-addr
ADDROBJ   TYPE     STATE   ADR
lo0       static   ok      127.0.0.1/10
net0/v4   static   ok      192.168.10.1


# ipadm create-addr -T static -a 192.168.10.5 net0/v4b
# ipadm show-addr
ADDROBJ    TYPE     STATE   ADR
lo0        static   ok      127.0.0.1/10
net0/v4    static   ok      192.168.10.1
net0/v4b   static   ok      192.168.10.5
```

- Delete the first instance of the address object, then assign the same address object to a different IP address, as follows:

```
# ipadm show-addr
ADDROBJ   TYPE     STATE   ADR
lo0       static   ok      127.0.0.1/10
net0/v4   static   ok      192.168.10.1
# ipadm delete-addr net0/v4
# ipadm create-addr -T static -a 192.168.10.5 net0/v4
# ipadm show-addr
ADDROBJ   TYPE     STATE   ADR
lo0       static   ok      127.0.0.1/10
net0/v4   static   ok      192.168.10.5
```

Error Message: `cannot create address: Persistent operation on temporary object` Displayed During IP Interface Configuration

By default, the `ipadm` command creates persistent network configuration. If the IP interface that you are configuring was created as a temporary interface, you cannot use the `ipadm` command to configure persistent settings on that interface. After verifying that the interface you are configuring is temporary, delete that interface and recreate it persistently. You can then resume configuring the interface as follows:

```
# ipadm show-if -o all
IFNAME   CLASS      STATE   ACTIVE   CURRENT        PERSISTENT   OVER
lo0      loopback   ok      yes      -m46-v------   46--         --
```

```
net0    ip      ok      yes     bm4--------- ----         --
```

The absence of a 4 flag for an IPv4 configuration or a 6 flag for an IPv6 configuration in the PERSISTENT field indicates that net0 was created as a temporary interface.

```
# ipadm delete-ip net0
# ipadm create-ip net0
# ipadm create-addr -T static -a 192.168.1.10 net0/v4
ipadm: cannot create address: Persistent operation on temporary object
```

Troubleshooting Issues With IPv6 Deployment

Refer to the following information if you encounter any issues while planning and deploying IPv6 at your site. For specific planning tasks, see Chapter 2, "Planning for Using IPv6 Addresses," in "Planning for Network Deployment in Oracle Solaris 11.2 ".

IPv6 Interface Is Not Configured Correctly

The existence of an IPv6 interface does not necessarily mean the system is using IPv6. The interface is not brought up until you actually configure an IPv6 address on that interface.

For example, the following output of the ifconfig command shows that the inet6 net0 interface has not been marked as UP and has an address of ::/0,meaning an IPv6 interface is not configured.

```
# ifconfig net0 inet6
net0:
flags=120002000840<RUNNING,MULTICAST,IPv6,PHYSRUNNING> mtu 1500 index 2 inet6 ::/0
```

The in.ndpd daemon still runs on the system but does not operate on any IP interfaces that do not have an addrconf address configured.

Cannot Upgrade IPv4 Router to IPv6

If you cannot upgrade your existing equipment, you might need to purchase IPv6-ready equipment. Check the manufacturer's documentation for any equipment-specific procedures that you might be required to perform to support IPv6.

You cannot upgrade certain IPv4 routers for IPv6 support. If this situation applies to your topology, as an alternative, you can physically wire an IPv6 router next to the IPv4 router. Then, you can tunnel from the IPv6 router over the IPv4 router. For instructions on configuring IP tunnels, see Chapter 5, "Administering IP Tunnels," in "Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 ".

Problems Encountered When Upgrading Services to Support IPv6

You might encounter the following issues when preparing services for IPv6 support:

- Certain applications, even after being ported to IPv6, do not turn on IPv6 support by default. You might have to configure these applications to turn on IPv6.

- A server that runs multiple services, some of which are IPv4 only and others that are both IPv4 and IPv6, can experience problems. Some clients might need to use both types of services, which can lead to confusion on the server side.

Current ISP Does Not Support IPv6

If you want to deploy IPv6, but your current Internet Service Provider (ISP) does not offer IPv6 addressing, consider the following alternatives:

- Hire another ISP to provide a second line for IPv6 communications from your site. This solution is expensive.

- Get a virtual ISP. A *virtual ISP* provides your site with IPv6 connectivity but no link. Instead, you create a tunnel from your site, over your IPv4 ISP, to the virtual ISP.

- Use a 6to4 tunnel over your ISP to other IPv6 sites. For an address, you can use the registered IPv4 address of the 6to4 router as the public topology part of the IPv6 address. For more information, see "How to Configure a 6to4 Tunnel" in "Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 ".

Security Issues When Tunneling to a 6to4 Relay Router

By nature, a tunnel between a 6to4 router and a 6to4 relay router is insecure. The following types of security problems are inherent in such a tunnel:

- Though 6to4 relay routers do encapsulate and decapsulate packets, these routers do not check the data that is contained within the packets.

- Address spoofing is a major issue on tunnels to a 6to4 relay router. For incoming traffic, the 6to4 router is unable to match the IPv4 address of the relay router with the IPv6 address of the source. Therefore, the address of the IPv6 host can easily be spoofed. The address of the 6to4 relay router can also be spoofed.

- By default, no trusted mechanism exists between 6to4 routers and 6to4 relay routers. Thus, a 6to4 router cannot identify whether the 6to4 relay router is to be trusted or even if it is

a legitimate 6to4 relay router. A trusted relationship between the 6to4 site and the IPv6 destination must exist. Otherwise, both sites leave themselves open to possible attacks.

These problems and other security issues that are inherent with 6to4 relay routers are explained in RFC 3964, Security Considerations for 6to4 (http://www.rfc-editor.org/rfc/rfc3964.txt). See also RFC 6343, Advisory Guidelines for 6to4 Deployment (http://www.rfc-editor.org/rfc/rfc6343.txt) for updated information about using 6to4.

Generally, you should consider enabling support for 6to4 relay routers for the following reasons only:

- Your 6to4 site intends to communicate with a private, trusted IPv6 network. For example, you might enable 6to4 relay router support on a campus network that consists of isolated 6to4 sites and native IPv6 sites.
- Your 6to4 site has a compelling business reason to communicate with certain native IPv6 hosts.
- You have implemented the checks and trust models that are suggested in Security Considerations for 6to4 (http://www.ietf.org/rfc/rfc3964.txt) and Advisory Guidelines for 6to4 Deployment. (http://www.ietf.org/rfc/rfc6343.txt).

Troubleshooting IPMP Configuration

This section contains the following topics:

- "Failure Detection in IPMP" on page 20
- "Disabling Outbound Load Spreading in Link-Based IPMP Configuration" on page 21
- "Error Message: `*ipadm: cannot add net0 to ipmp0: Underlying interface has addresses managed by dhcpagent(1M)*` Displayed During IPMP Group Creation" on page 21

Note - The commands and tasks for configuring IP network multipathing (IPMP) have changed. The `ipadm` command is now used instead of the `ifconfig` command to configure and manage IPMP. To learn about how these two commands map to each other, see "Comparing the ifconfig Command to the ipadm Command" in "Transitioning From Oracle Solaris 10 to Oracle Solaris 11.2 ".

Failure Detection in IPMP

To ensure continuous availability of the network to send or receive traffic, IPMP performs failure detection on an IPMP group's underlying IP interfaces. Failed interfaces remain unusable

until they are repaired. Any remaining active interfaces continue to function while any existing standby interfaces are deployed, as needed.

The `in.mpathd` daemon handles the following types of failure detection:

- Probe-based failure detection:
  - No test addresses are configured (transitive probing)
  - Test addresses are configured
- Link-based failure detection, if supported by the NIC driver

For more details, see "Failure Detection in IPMP" in "Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 ".

Disabling Outbound Load Spreading in Link-Based IPMP Configuration

It is possible to disable outbound load spreading in link-based IPMP. If you mark an interface as standby, that interface will not be used until an active interface fails, regardless of whether link-based or probe-based failure detection is used. Link-based failure detection is always enabled by the `in.mpathd` daemon.

Use the `ipadm` command as follows:

```
# ipadm set-ifprop -m ip -p standby=on interface
```

For information about how inbound and outbound load spreading in link-based IPMP works, see "Benefits of Using IPMP" in "Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 ".

Error Message: `*ipadm: cannot add net0 to ipmp0: Underlying interface has addresses managed by dhcpagent(1M)*` Displayed During IPMP Group Creation

The following error message might be displayed when you attempt to add an IPMP group:

```
*ipadm: cannot add net0 to ipmp0: Underlying interface has addresses managed by dhcpagent(1M)*
```

This message is displayed because you cannot add an IP interface with an address that is managed by `dhcpagent` into an IPMP group. As a workaround, disable the DHCP and/or stateful address configuration on `net0` before adding it to the IPMP group.

Troubleshooting Issues With VRRP and the Oracle Solaris Bundled IP Filter

The Virtual Router Redundancy Protocol (VRRP) provides high availability of IP addresses, such as those that are used for routers and load balancers. Oracle Solaris supports both L2 and L3 VRRP. For more information, see Chapter 3, "Using Virtual Router Redundancy Protocol," in "Configuring an Oracle Solaris 11.2 System as a Router or a Load Balancer ".

The standard VRRP multicast address (224.0.0.18/32) is used to ensure that VRRP functions properly. See http://www.rfc-editor.org/rfc/rfc5798.txt for more information. When you use VRRP with the Oracle Solaris bundled IP Filter, you must explicitly check whether outgoing or incoming IP traffic is allowed for the multicast address.

Use the `ipfstat -io` command as follows to check for this information:

```
# ipfstat -io
empty list for ipfilter(out)
empty list for ipfilter(in)
```

If the output of the command indicates that traffic is not allowed for the standard multicast address, you must add the following rules to the IP Filter configuration for each VRRP router:

```
# echo "pass out quick on VRRP VIP Interface from VRRP VIP/32 to 224.0.0.18/32 \
pass in quick on VRRP VIP Interface from VRRP IP/32 to 224.0.0.18/32" | ipf -f
```

For more information about configuring an IP Filter rule set, see "How to Append Rules to the Active Packet Filtering Rule Set" in "Securing the Network in Oracle Solaris 11.2 ".

Resources for Monitoring and Detecting Problems on a TCP/IP Network

The following table describes tasks for monitoring and detecting problems on a TCP/IP network. For complete instructions, see "Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 ".

TABLE 1-1   Tasks for Monitoring TCP/IP Networks

Task	Command and/or Description	Task Information
Log the IP addresses of all incoming TCP connections.	Transport layer protocols typically need no intervention to run properly. However, in some circumstances, you might need to log or modify services that run over the transport layer protocols.	"Logging IP Addresses of All Incoming TCP Connections" in "Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 "
Determine whether a remote host is running.	Use the `ping` command to determine the status of a remote host.	"Determining if a Remote Host Is Reachable" in "Administering TCP/

Task	Command and/or Description	Task Information
		IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 ”
Detect whether a host is dropping packets	Use the `-s` option of the `ping` command to determine whether a remote host is running but losing packets.	“Determining if Packets Between Your Host and a Remote Host Are Being Dropped” in “Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 ”
Display network statistics on a per-protocol basis.	Use the `netstat` command to display statistics on a per-protocol basis for TCP, Stream Control Transmission Protocol (SCTP), and User Datagram Protocol (UDP) endpoints in table format.	“Monitoring Network Status With the netstat Command” in “Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 ”
Perform TCP and UDP management.	Use the `netcat` (or `nc`) utility to open TCP connections, send UDP packets, listen on arbitrary TCP and UDP ports, perform port scanning.	“Performing TCP and UDP Administration With the netcat Utility” in “Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 ”
Trace the actions of the IPv4 routing daemon, including all packet transfers.	If you suspect a malfunction of the `routed` daemon, you can start a log that traces the daemon's activity. The log includes all packet transfers when you start the `routed` daemon.	“Logging Actions of the IPv4 Routing Daemon” in “Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 ”
Discover the route to a remote host.	Use the `traceroute` command to discover the route to a remote system. The output displays the number of hops in the path a packet follows.	“Discovering the Route to a Remote Host” in “Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 ”
Check packets between an IPv4 server and a client.	Establish a snoop system off a hub that is connected to either a client or a server to check intervening traffic.	“How to Check Packets Between an IPv4 Server and a Client” in “Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 ”
Monitor the packet transfer process.	Use the `snoop` command to monitor the state of package (data) transfers.	“Monitoring Packet Transfers With the snoop Command” in “Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 ”
Analyze network traffic.	Use the TShark command line interface (CLI) or the Wireshark graphical user interface (GUI) to analyze network traffic.	“Analyzing Network Traffic With the TShark and Wireshark Analysers” in “Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 ”
Monitor network traffic on a server.	Use the `ipstat` and `tcpstat` commands to monitor network traffic on a server.	“Observing Network Traffic With the ipstat and tcpstat Commands” in “Administering TCP/IP Networks,

Task	Command and/or Description	Task Information
		IPMP, and IP Tunnels in Oracle Solaris 11.2 "
Monitor network traffic on an IPv6 network.	Use the `snoop ip6` command to display just IPv6 packets for a network node.	"Monitoring IPv6 Network Traffic" in "Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 "
Monitor the status of IPMP on your system.	Use the `ipmpstat` command to gather different types of information about the status of IPMP. You can also use the command to display information about the underlying IP interfaces for each IPMP group, as well as data and test addresses are configured for the group.	"Monitoring IPMP Information" in "Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 "
Control the output of the `ping`, `netstat`, and `traceroute` commands.	Create a file named `inet_type` that sets the `DEFAULT_IP` variable in the file that controls the display output of IPv6-related commands.	"How to Control the Display Output of IP-Related Commands" in "Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 "

Monitoring Network Traffic on Datalinks and Flows

You configure and administer datalinks and flows by using the `dladm` and `flowadm` commands, respectively. You use the newer `dlstat` and `flowstat` commands to obtain statistics on network traffic for datalinks and flows, respectively.

The following are some basic examples that show how to monitor network traffic on datalinks and flows. For more information, see the `dlstat`(1M) and `flowstat`(1M) man pages.

Display inbound and outbound traffic statistics per link as follows:

dlstat *link*

dlstat show-link *link*

Display inbound and outbound traffic statistics per physical network device as follows:

dlstat show-phys *link*

Display inbound and outbound traffic statistics per port and link aggregation as follows:

dlstat show-aggr *link*

For more examples, see Chapter 8, "Monitoring Network Traffic and Resource Usage," in "Managing Network Virtualization and Network Resources in Oracle Solaris 11.2 ".

2

Troubleshooting Profile-Based Network Administration Issues

This chapter provides information for troubleshooting problems that you might encounter when configuring and administering reactive profiles. The reactive mode is most commonly used for notebook PCs and in situations where network conditions change frequently.

For more information about profile-based network configuration, see Chapter 5, "About Administering Profile-Based Network Configuration in Oracle Solaris," in "Configuring and Administering Network Components in Oracle Solaris 11.2 ".

This chapter contains the following topics:

- "Answers to Common Profile-Based Network Configuration Questions" on page 25
- "Troubleshooting Profile Configuration Issues by Using the `netadm` Command" on page 28
- "Monitoring the Current State of All Network Connections" on page 30
- "Viewing and Setting Profile Properties by Using the `netcfg walkprop` Command" on page 30

Answers to Common Profile-Based Network Configuration Questions

Refer to the following troubleshooting information when using the *reactive mode* for network administration. For information about troubleshooting network administration issues when using the *fixed mode*, see "Answers to Common Network Administration Questions" on page 7. For further details, see "About Network Configuration Modes" in "Configuring and Administering Network Components in Oracle Solaris 11.2 ".

Question: How do I determine which networking mode my system is using after an installation?

Answer: The networking mode is determined by the profile that is activated during installation. If the `Automatic` profile is activated, you are in the reactive mode. If the `DefaultFixed` profile is activated, you are in the fixed mode. To determine which mode is currently active on your system, use the `netadm list` command as follows:

```
# netadm list
```

Question: My system defaulted to the fixed mode after an installation and the `DefaultFixed` profile is currently active. How do I switch to reactive mode?

Answer: To enable the reactive mode, you need to switch to either the `Automatic` profile or another reactive profile by using the `netadm enable` command. For example, you would enable the `Automatic` profile as follows:

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

Question: What profile must be referenced to not plumb IPv6 and how is this aspect of network configuration managed when using the Automated Installer (AI) or at installation time?

Answer: You can create any profile that does not have an IPv6 address configured. When the profile is enabled, IPv6 is not plumbed. You cannot create new reactive profiles at installation time from an AI manifest. If you want to create a reactive profile after an installation, use the `netcfg` command. See "Configuring Profiles" in "Configuring and Administering Network Components in Oracle Solaris 11.2 ". Note that an AI manifest does enable you to choose which profile to activate after the installation after the system reboots.

Question: The naming services settings on my system are not set correctly after installing Oracle Solaris. What should I do?

Answer: For the reactive mode, naming services information and other system-wide settings are specified in a `Location` profile, which is another primary profile type. See "Profile Type Descriptions" in "Configuring and Administering Network Components in Oracle Solaris 11.2 " for more details.

The following example shows how to display all of the profiles that are on a system and their states. Use this command to determine the currently active `Location` profile. The second part of the example shows how to initiative an interactive `netcfg` session, then select the currently active `Location` and list its configuration information:

```
# netadm list
TYPE        PROFILE       STATE
ncp         DefaultFixed  disabled
ncp         Automatic     online
ncu:phys    net0          offline
ncu:ip      net0          offline
loc         Automatic     online
loc         NoNet         offline
loc         DefaultFixed  offline

# netcfg
netcfg> select loc myloc
netcfg:loc:myloc> list
```

```
loc:myloc
      activation-mode            manual
      enabled                    false
      nameservices               dns
      nameservices-config-file   "/etc/nsswitch.dns"
      dns-nameservice-configsrc  dhcp
netcfg:loc:myloc>
```

In the previous example, DNS is used and the `/etc/nsswitch.dns` file is referenced.

The following example shows how you would modify the existing naming services configuration for a `Location` named `myloc`:

```
# netadm list
TYPE         PROFILE      STATE
ncp          DefaultFixed disabled
ncp          Automatic    online
ncu:phys     net0         offline
ncu:ip       net0         offline
loc          Automatic    offline
loc          NoNet        offline
loc          DefaultFixed offline
loc          myloc        online

# netcfg
netcfg> select loc myloc
netcfg:loc:myloc> list
loc:myloc
activation-mode            manual
enabled                    false
nameservices               nis
nameservices-config-file   "/etc/nsswitch.nis"
dns-nameservice-configsrc  dhcp
nfsv4-domain
netcfg:loc:myloc> set nameservices=dns
netcfg:loc:myloc> set nameservices-config-file="/etc/nsswitch.dns"
netcfg:loc:myloc> list
      activation-mode            system
      enabled                    false
      nameservices               dns
      nameservices-config-file   "/etc/nsswitch.dns"
netcfg:loc:myloc> commit
Committed changes
netcfg:loc:myloc> exit
```

For more information about configuring `Locations`, see "Creating Locations" in "Configuring and Administering Network Components in Oracle Solaris 11.2 ".

Question: I cannot start the network administration GUI (formerly NWAM) from the desktop. Can I start the GUI from the command line?

Answer: Use the following command to start the GUI from the command line:

```
% /usr/lib/nwam-manager
```

If the GUI still does not start, make sure that the network administration GUI icon is displayed in the GNOME notification area of your desktop panel. If the icon is not displayed, press the right mouse button to select the Add to Panel... option on the desktop panel, then add the Notification Area to the panel.

Question: I started the network administration GUI from the command line (`/usr/lib/nwam-manager`) as a regular user and received the message, "Another instance is running. This instance will exit now". The GUI seemed to start, but the icon is not displayed on the desktop. How can I access the GUI?

Answer: If the icon is not displayed in the desktop panel, press the right mouse button to select the Add to Panel... option on the desktop panel, then add the Notification Area to the panel.

Troubleshooting Profile Configuration Issues by Using the `netadm` Command

Use the `netadm list` command with the appropriate options and arguments to display information about the profiles on a system and troubleshoot profile-based network configuration. For complete details see the `netadm(1M)` man page.

When used with no additional options, the `netadm list` command displays all of the profiles that are on a system and their current state:

```
% netadm list
TYPE        PROFILE      STATE
ncp         DefaultFixed disabled
ncp         Automatic    online
ncu:phys    net0         online
ncu:ip      net0         online
loc         Automatic    online
loc         NoNet        offline
loc         DefaultFixed offline
```

To display information about a particular profile, specify the profile's name, as shown in the following example where the `Automatic` profile is specified:

```
% netadm list Automatic
TYPE        PROFILE      STATE
ncp         Automatic    online
ncu:ip      net1         offline
ncu:phys    net1         offline
ncu:ip      net0         online
ncu:phys    net0         online
loc         Automatic    online
```

To display information about all of the profiles on the system that are of a certain type, use the
`netadm list` command with the -p option. For example, you would display all of the `Location`
profiles on a system as follows:

```
% netadm list -p loc
TYPE          PROFILE        STATE
loc           NoNet          offline
loc           Automatic      online
loc           DefaultFixed   offline
```

In the following example, the `netadm list` command is used with the `-c` option to display
configuration details for the currently active profile.

```
% netadm list -c ip
TYPE          PROFILE        STATE
ncu:ip        net0           online
```

The `netadm list -x` command is useful for determining why a network interface might not be
configured correctly. Use this command to display the various profiles on a system, their current
state, and the reason for being in that state.

For example, if a cable is unplugged, use the `netadm list -x` command to determine if the link
state is offline and why, for example, "link is down." Similarly, for duplicate address detection,
the output of the `netadm list -x` command shows that the physical link is online (up), but the
IP interface is in a maintenance state. In this instance, the reason that is given is "Duplicate
address detected.".

The following example shows the type of information that you can obtain by using the `netadm`
`list -x` command:

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

After determining the reason a link or an interface is offline, you can then proceed to correct
the problem. In the case of a duplicate IP address, you must modify the static IP address that is
assigned to the specified interface by using the `netcfg` command. For instructions, see "Setting
Property Values for Profiles" in "Configuring and Administering Network Components in
Oracle Solaris 11.2 ". After you commit the changes, run the `netadm list -x` command again to
check that the interface is configured correctly and its state is showing as `online`.

Another example of why an interface might not be configured correctly is if no known wireless
local area networks (WLANs) are available. In this case, the WiFi link's state would be
displayed as "offline" and the reason would be indicated as "need WiFi network selection".
Alternatively, if a WiFi selection was initially made, but a key is required, the reason would be
indicated as "need WiFi key".

Monitoring the Current State of All Network Connections

Use the `netadm show-events` command to listen for and display events that are being monitored by the network management daemon, `nwamd`. This subcommand provides useful information about events that are related to the configuration process for network profiles.

```
% netadm show-events
EVENT              DESCRIPTION
OBJECT_ACTION      ncp Automatic -> action enable
OBJECT_STATE       ncp Automatic -> state online, active
OBJECT_STATE       ncu link:net0 -> state offline*, (re)initialized but not config
OBJECT_STATE       ncu link:net0 -> state online, interface/link is up
OBJECT_STATE       ncu interface:net0 -> state offline*, (re)initialized but not c
OBJECT_STATE       ncu interface:net0 -> state offline*, waiting for IP address to
PRIORITY_GROUP     priority-group: 0
LINK_STATE         net0 -> state up
OBJECT_STATE       loc NoNet -> state offline*, method/service executing
OBJECT_STATE       loc Automatic -> state offline, conditions for activation are u
OBJECT_STATE       loc NoNet -> state online, active
IF_STATE           net0 -> state flags 1004843 addr 10.153.125.198/24
OBJECT_STATE       ncu interface:net0 -> state offline*, interface/link is up
OBJECT_STATE       ncu interface:net0 -> state online, interface/link is up
IF_STATE           net0 -> state flags 2080841 addr 2002:a99:7df0:1:221:28ff:fe3c:
IF_STATE           net0 -> state flags 2004841 addr 2001:db8:1:2::4ee7/128
OBJECT_STATE       loc Automatic -> state offline*, method/service executing
OBJECT_STATE       loc NoNet -> state offline, conditions for activation are unmet
OBJECT_STATE       loc Automatic -> state online, active
```

Viewing and Setting Profile Properties by Using the `netcfg` `walkprop` Command

Use the `netcfg walkprop` command to view or change individual or multiple properties of a profile interactively. By using this command, you can display the various properties of a profile (one at a time) and make changes to each property, as needed. When you use the `walkprop` subcommand, you do not need to use the `set` subcommand to set property values.

Note that to view or change a profile's configuration by using the `walkprop` subcommand, you must be in the correct interactive *scope*. See "Configuring Profiles" in "Configuring and Administering Network Components in Oracle Solaris 11.2 ".

For instructions and examples, see "Setting Property Values for a Profile by Using the walkprop Subcommand" in "Configuring and Administering Network Components in Oracle Solaris 11.2 ".

3

Troubleshooting Naming Services Issues

This chapter describes basic naming services configuration in Oracle Solaris and how to manage and troubleshoot various related issues that could prevent your network from working properly.

This chapter contains the following topics:

About Naming Services Configuration

In this release, naming services configuration is managed by the Service Management Facility (SMF). What this change means is that the SMF repository is now the primary repository for all naming services configuration and you no longer modify a specific file to configure naming services. To make the configuration persistent, you must enable or refresh the appropriate SMF services.

If no network configuration exists after an installation, naming services default to `files only` behavior rather than `nis files`. To avoid potential configuration issues, make sure that the `svc:/system/name-service/cache` SMF service is enabled at all times. For more information, see Chapter 1, "About Naming and Directory Services," in "Working With Oracle Solaris 11.2 Directory and Naming Services: DNS and NIS ".

Troubleshooting DNS Issues

The following procedures are described:

▼ How to Troubleshoot DNS Client Issues

In Oracle Solaris 11, you no longer edit the /etc/resolv.conf file to make persistent changes to a DNS client. All DNS client configuration is managed by the dns/client SMF service. For information about how to enable a DNS client, see "How to Enable a DNS Client" in "Configuring and Administering Network Components in Oracle Solaris 11.2 ".

1. **Check the status of the DNS service.**

   ```
   # svcs -xv dns/client:default
   ```

2. **Check the DNS client service log.**

   ```
   # more /var/svc/log/network-dns-client:default.log
   ```

3. **Check the name server settings.**

   ```
   # svcprop -p config/nameserver dns/client
   ```

4. **Check the search settings.**

   ```
   # svcprop -p config/search dns/client
   ```

5. **Check all DNS settings.**

   ```
   # svcprop -p config dns/client
   ```

▼ How to Troubleshoot DNS Server Issues

1. **Check the status of the DNS service.**

   ```
   # svcs -xv dns/server:default
   ```

2. **Check the DNS service log.**

   ```
   # more /var/svc/log/network-dns-server:default.log
   ```

3. **Check for syslog messages.**

   ```
   # grep named /var/adm/messages
   ```

4. **Start the named daemon.**

   ```
   # named -g
   ```

5. **After resolving the issue, clear the DNS service.**

   ```
   # svcadm clear dns/server:default
   ```

6. **Verify that the DNS service is back online.**

   ```
   # svcs dns/server:default
   ```

Troubleshooting NFS Issues

The following procedures are described:

-
-
-

▼ How to Troubleshoot NFS Client Connectivity Issues

Troubleshooting issues with a client connecting to an NFS server can involve several steps, depending on the root cause. The following procedure follows the logical sequence that you might follow to resolve NFS client connectivity issues. If you cannot resolve the problem by performing a given step, proceed to the next step until you have identified and corrected the issue.

1. **Check that the NFS server is reachable from the client system.**

   ```
   # ping nfs-server
   ```

2. **If the server is not reachable from the client, check that the local naming service is running.**

3. **If the local naming service is running, check that the client has the correct host information.**

   ```
   # getent hosts nfs-server
   ```

4. **If the host information on the client is correct, try to reach the NSF server by running the ping command from another client.**

5. **If the NFS server is reachable from the second client, check whether the first client can connect to any other systems on the local network by using the ping command:**

   ```
   # ping other-client-system
   ```

6. **If other clients are unreachable, follow the steps that are described in "Running Basic Network Diagnostic Checks" on page 13.**

▼ How to Check the NFS Server Remotely

The following procedure describes the logical sequence that you would follow to check an NFS server remotely.

1. **Check that the NFS services have been started on the NFS server.**

    `# rpcinfo -s bee|egrep 'nfs|mountd`

2. **Check that the NFS server's `nfsd` processes are responding by running the following command on the client:**

    `# rpcinfo -u` *nfs-server* `nfs`

3. **Check that the `mountd` daemon is running on the NFS server.**

    `# rpcinfo -u` *nfs-server* `mountd`

4. **Check whether the local `autofs` service is being used.**

    `# cd /net/wasp`

5. **Verify that the file system is being shared as expected on the NFS server.**

    `# showmount -e` *nfs-server*

▼ How to Troubleshoot Issues With the NFS Service on the Server

The following procedure describes the logical sequence that you would follow to verify whether the NFS service is running on the server.

1. **Check that the server can reach the clients.**

    `# ping` *client*

2. **If the client is not reachable, check that the local naming service is running.**

3. **If the naming service is running, check the networking software configuration on the server, for example, the `/etc/netmasks` file and the properties that are set for the `svc:/system/name-service/switch` SMF service.**

4. **Check whether the `rpcbind` daemon is running.**

   ```
   # rpcinfo -u localhost rpcbind
   ```

5. **Check whether the `nfsd` daemon is running.**

   ```
   # rpcinfo -u localhost nfs
   # ps -ef | grep mountd
   ```

Troubleshooting Issues With the Name Service Switch File

Check the current configuration for the name service switch file (`/etc/nsswitch.conf`) as follows:

```
# svccfg -s name-service/switch listprop config
```

Troubleshooting NIS Issues

The following information describes how to debug issues with the Network Information Service (NIS) (pronounced "niss" in this guide). Before attempting to debug a NIS server or client problem, review Chapter 5, "About the Network Information Service," in "Working With Oracle Solaris 11.2 Directory and Naming Services: DNS and NIS ".

This section contains the following topics:

- "Troubleshooting NIS Binding Issues" on page 35
- "Troubleshooting NIS Issues That Affect a Single Client" on page 36
- "Troubleshooting NIS Issues That Affect Multiple Clients" on page 39

Troubleshooting NIS Binding Issues

The following are common symptoms of NIS binding problems:

- Messages saying that the `ypbind` daemon cannot find or communicate with a server.
- Messages that say server not responding.
- Messages that say NIS is unavailable.
- Commands on a client limp along in background mode or function much slower than normal.
- Commands on a client hang. Sometimes commands hang, even though the system as a whole seems fine and you can run new commands.
- Commands on a client crash with obscure messages or no messages at all.

Troubleshooting NIS Issues That Affect a Single Client

If only one or two clients are experiencing symptoms that indicate NIS binding difficulty, the problems probably are on those clients. However, if many NIS clients are failing to bind properly, the problem probably exists on one or more of the NIS servers. See .

The following are common NIS issues that affect a single client:

- **ypbind daemon not running on the client**

  One client has problems, but the other clients on the same subnet are operating normally. On the problem client, run the ls -l command on a directory that contains files that are owned by many users (such as /usr), including files that are not in the client /etc/passwd file. If the resulting display lists file owners who are not in the local /etc/passwd file as numbers rather than names, the NIS service is not working on the client.

  These symptoms usually indicate that the client's ypbind process is not running. Verify whether the NIS client services are running as follows:

  ```
  client# svcs \*nis\*
  STATE         STIME    FMRI
  disabled      Sep_01   svc:/network/nis/domain:default
  disabled      Sep_01   svc:/network/nis/client:default
  ```

  If the services are in a disabled state, log in and become the root role, then start the NIS client service as follows:

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

- **Missing or incorrect domain name**

  One client has problems and other clients are operating normally, but the ypbind daemon is running on that client. In this case, the client might have an incorrectly set domain.

  Run the domainname command on the client to determine which domain name is set:

  ```
  client# domainname
  example.com
  ```

  Compare the output with the actual domain name in the /var/yp directory on the NIS master server. As shown in the following example, the actual NIS domain is shown as a subdirectory in the /var/yp directory:

  ```
  client# ls -l /var/yp
  -rwxr-xr-x 1 root Makefile
  drwxr-xr-x 2 root binding
  ```

```
drwx------ 2 root example.com
```

If the domain name that is displayed in the output of the `domainname` command on the client is not the same as the server domain name that is listed as a subdirectory in the `/var/yp` directory, the domain name in the `config/domain` property of the `nis/domain` service is incorrect. Reset the NIS domain name. For instructions, see "How to Set a Machine's NIS Domain Name" in "Working With Oracle Solaris 11.2 Directory and Naming Services: DNS and NIS ".

Note - The NIS domain name is case-sensitive.

- **Client not bound to server**

  If your domain name is set correctly and the `ypbind` daemon is running, yet commands still hang, make sure that the client is bound to a server by running the `ypwhich` command. If you have just started the `ypbind` daemon, then run the `ypwhich` command. You might need to run the `ypwhich` command several times. Typically, the first time you run the command, it reports that the domain is not bound. The second time you run the command, it should proceed normally.

- **No server available**

  If your domain name is set correctly and the `ypbind` daemon is running, but you receive messages indicating the client cannot communicate with the server, check the following:

  - Does the client have a `/var/yp/binding/`*domainname*`/ypservers` file that contains a list of servers to bind to? To view the selected NIS servers, use the `svcprop -p config/ypservers nis/domain` command. If not, run the `ypinit -c` command to specify which servers this client should bind to, in order of preference.

  - If the client does have a `/var/yp/binding/`*domainname*`/ypservers` file, are there enough servers listed, in case one or two servers become unavailable? To view the selected NIS servers, use the `svcprop -p config/ypservers nis/domain` command. If not, add additional servers to the list by running the `ypinit -c` command.

  - Do the selected NIS servers have entries in the `/etc/inet/hosts` file? To view the selected NIS servers, use the `svcprop -p config/ypservers nis/domain` command. If these hosts are not in the local `/etc/inet/hosts` file, add the servers to the `hosts` NIS maps and rebuild your maps by running the `ypinit -c` or `ypinit -s` command. For information, see "Working With NIS Maps" in "Working With Oracle Solaris 11.2 Directory and Naming Services: DNS and NIS ".

  - Is the name service switch set up to check the system's local `hosts` file, in addition to NIS? For more information, see Chapter 2, "About the Name Service Switch," in "Working With Oracle Solaris 11.2 Directory and Naming Services: DNS and NIS ".

  - Is the name service switch set up to check `files` first for `services` and then `rpc`?

- `ypwhich` **displays are inconsistent**

If you run the `ypwhich` command several times on the same client, the resulting display varies because the NIS server changes. This behavior is normal. The binding of the NIS client to the NIS server changes over time when the network or the NIS servers are busy. Whenever possible, the network becomes stable at the point when all clients receive an acceptable response time from the NIS servers. As long as the client receives the NIS service, it does not matter where the service comes from. For example, one NIS server can receive NIS services from another NIS server on the network.

- **What to do when server binding is not possible**

  In extreme cases where local server binding is not possible, use the `ypset` option with the `ypbind` command to temporarily allow binding to another server on another network or subnet, if available. Note that to use the `-ypset` option, you must start the `ypbind` daemon by using either the `-ypset` or `-ypsetme` option. For more information, see the `ypbind`(1M) man page.

  ```
  # /usr/lib/netsvc/yp/ypbind -ypset
  ```

  For another method, see "Binding to a Specific NIS Server" in "Working With Oracle Solaris 11.2 Directory and Naming Services: DNS and NIS ".

> ⚠ **Caution -** For security reasons, using the `-ypset` or `-ypsetme` option is not recommended. Only use these options for debugging purposes under controlled circumstances. Using the `-ypset` or `-ypsetme` option can result in serious security breaches. While the daemons are running, anyone can alter server bindings, which can permit unauthorized access to sensitive data. If you must start the `ypbind` daemon by using either of these options, kill the `ypbind` process after you have corrected the problem, then restart it without specifying these options.

Restart the `ypbind` daemon as follows:

```
# svcadm enable -r svc:/network/nis/client:default
```

See the `ypset`(1M) man page.

- `ypbind` **daemon crashes**

  If the `ypbind` daemon crashes almost immediately each time you start it, look for a problem in the `svc:/network/nis/client:default` service log. Check for the presence of the `rpcbind` daemon as follows:

  ```
  % ps -e |grep rpcbind
  ```

  If the `rpcbind` daemon is not present, does not stay up, or behaves strangely, check the `svc:/network/rpc/bind:default` log file. For more information, see the `rpcbind`(1M) and `rpcinfo`(1M) man pages.

You might be able to communicate with the `rpcbind` daemon on the problematic client from a system that is functioning normally.

Run the following command from a functioning system:

```
% rpcinfo client
```

If the `rpcbind` daemon on the problematic system is fine, the following output is displayed:

```
program version netid address service owner
...
100007   3   udp6       ::.191.161        ypbind   1
100007   3   tcp6       ::.135.200        ypbind   1
100007   3   udp        0.0.0.0.240.221   ypbind   1
100007   2   udp        0.0.0.0.240.221   ypbind   1
100007   1   udp        0.0.0.0.240.221   ypbind   1
100007   3   tcp        0.0.0.0.250.107   ypbind   1
100007   2   tcp        0.0.0.0.250.107   ypbind   1
100007   1   tcp        0.0.0.0.250.107   ypbind   1
100007   3   ticlts     2\000\000\000     ypbind   1
100007   2   ticlts     2\000\000\000     ypbind   1
100007   3   ticotsord 9\000\000\000      ypbind   1
100007   2   ticotsord 9\000\000\000      ypbind   1
100007   3   ticots     @\000\000\000     ypbind   1
...
```

If no addresses are displayed (your system will have different addresses), the `ypbind` daemon was unable to register its services. Reboot the system and run the `rpcinfo` command again. If the `ypbind` processes are there and they change each time you attempt to restart the NIS service, reboot the system, even if the `rpcbind` daemon is running.

Troubleshooting NIS Issues That Affect Multiple Clients

If only one or two clients are experiencing symptoms that indicate NIS binding difficulty, the problems probably are on those clients. See . However, if several NIS clients are failing to bind properly, the problem most likely exists on one or more of the NIS servers.

The following are common NIS issues that can affect multiple clients:

- `rpc.yppasswdd` **considers a non-restricted shell that begins with** `r` **to be restricted**

  To resolve this problem, do the following:

1. Create a `/etc/default/yppasswdd` file that contains a special string:
   `"check_restricted_shell_name=1"`.

2. If the `"check_restricted_shell_name=1"` string is commented out, the `r` check does not occur.

- **Network or servers unreachable**

   NIS can hang if the network or NIS servers are so overloaded that the `ypserv` daemon cannot receive a response back from the client's `ypbind` process within the timeout period. NIS can also hang if the network is down.

   Under both of these circumstances, every client that is on the network experiences the same or similar problems. In most cases, the condition is temporary. The messages usually go away when the NIS server reboots and restarts the `ypserv` daemon, when the load on the NIS servers or the network itself decreases, or when the network resumes normal operations.

- **Server malfunction**

   Make sure the servers are up and running. If you are not physically near the servers, use the `ping` command to determine if the server is reachable.

- **NIS daemons not running**

   If the servers are up and running, try to find a client that is behaving normally and run the `ypwhich` command on it. If the `ypwhich` command does not respond, kill it. Then, become the `root` role on the NIS server and check whether the NIS process is running as follows:

   ```
   # ptree |grep ypbind
   100759 /usr/lib/netsvc/yp/ypbind -broadcast
   527360 grep yp
   ```

   If neither the `ypserv` daemon (NIS server) nor the `ypbind` daemon (NIS client) daemons are running, restart them as follows:

   Restart the NIS client service as follows:

   ```
   # svcadm restart network/nis/client
   ```

   If both the `ypserv` and `ypbind` processes are running on the NIS server, then run the `ypwhich` command. If the command does not respond, the `ypserv` daemon is probably hung and should be restarted.

   On the server, restart the NIS service as follows:

   ```
   # svcadm restart network/nis/server
   ```

- **Servers have different versions of a NIS map**

   Because NIS propagates maps among servers, occasionally you might find different versions of the same map on various NIS servers that are on the network. This version discrepancy is normal and acceptable if the differences do not last too long.

   The most common cause of map discrepancy is when normal map propagation is prevented. For example, a NIS server or router that is located between NIS servers

is down. When all NIS servers and the routers between them are running, the ypxfr command should succeed.

If the servers and routers are functioning properly, proceed as follows:

- Check the ypxfr log output. See Example 3-1.

- Check the svc:/network/nis/xfr:default log file for errors.

- Check the control files (crontab file and yupxfr shell script).

- Check the ypservers map on the master server.

- ypserv **process crashes**

  When the ypserv process crashes almost immediately and does not stay up even after repeated activations, the debugging process is virtually the same as the debugging process for ypbind crashes.

  First, run the following command to see if any errors are being reported:

  ```
  # svcs -vx nis/server
  ```

  Check for the existence of the rpcbind daemon as follows:

  ```
  # ptree |grep rpcbind
  ```

  Reboot the server if you do not find the daemon. Otherwise, if the daemon is running, run the following command and look for similar output:

  ```
  # rpcinfo -p ypserver
  ```

  ```
  program vers     proto  port  service
  100000 4     tcp 111 portmapper
  100000 3     tcp 111 portmapper
  100068 2     udp  32813 cmsd
  ...
  100007 1     tcp  34900 ypbind
  100004 2     udp 731 ypserv
  100004 1     udp 731 ypserv
  100004 1     tcp 732 ypserv
  100004 2     tcp  32772 ypserv
  ```

  In the previous example, the following four entries represent the ypserv process:

  ```
  100004     2     udp  731  ypserv
  100004  1  udp  731  ypserv
  100004  1  tcp  732  ypserv
  100004  2  tcp   32772  ypserv
  ```

If there are no entries, and `ypserv` is unable to register its services with `rpcbind`, reboot the system. If there are entries, deregister the service from `rpcbind` before restarting `ypserv`. For example, you would deregister the service from `rpcbind` as follows:

```
# rpcinfo -d number 1
# rpcinfo -d number 2
```

where *number* is the ID number that is reported by rpcinfo (`100004` in the preceding example).

EXAMPLE 3-1   Logging `ypxfr` Command Output

- If a particular slave server has problems updating the maps, log in to that server and run the `ypxfr` command interactively.

  If the command fails, a message about why it failed is displayed to enable you to fix the problem. If the command succeeds, but you suspect it has occasionally failed, create a log file on the slave server to enable the logging of messages as follows:

  ```
  ypslave# cd /var/yp
  ypslave# touch ypxfr.log
  ```

  The output of the log file resembles the output of the `ypxfr` command when you run it interactively, with the exception that each line in the log file is time stamped. If you notice unusual ordering in the timestamps that is because it shows each time that the `ypxfr` command was actually run. If copies of `ypxfr` ran simultaneously but took differing amounts of time to finish, each copy might write a summary status line to the log file in a different order than when the command was run. Any pattern of intermittent failure shows up in the log.

Note - When you have resolved the problem, turn off logging by removing the log file. If you forget to remove it, the file continues to grow without limit.

- Check the `crontab` file and `ypxfr` shell script.

  Inspect the `root crontab` file and check the `ypxfr` shell script that it invokes. Typographical errors in these files can cause propagation problems. Failures to refer to a shell script within the `/var/spool/cron/crontabs/root` file or failures to refer to a map within any shell script can also cause errors.

- Check the `ypservers` map.

  Also, make sure that the NIS slave server is listed in the `ypservers` map on the master server for the domain. If it is not listed, the slave server still operates perfectly as a server, but `yppush` does not propagate map changes to the slave server.

- Update the maps on a broken slave server.

If the NIS slave server problem is not obvious, you can perform a workaround while debugging the problem by using the `scp` or `ssh` command. These commands copy a recent version of the inconsistent map from any healthy NIS server.

The following example shows how to transfer the problem map:

```
ypslave# scp ypmaster:/var/yp/mydomain/map.\* /var/yp/mydomain
```

In the previous example, the * character has been escaped in the command line so that it will be expanded on `ypmaster` instead of locally on `ypslave`.

♦ ♦ ♦   **C H A P T E R   4**

4

Performing Network Diagnostics With the `network-monitor` Transport Module Utility

This chapter describes how to use the network diagnostics monitoring utility to detect misconfigured network resources and error conditions on your Oracle Solaris system.

This chapter contains the following topics:

- "Overview of the `network-monitor` Transport Module Utility" on page 45
- "Managing the `network-monitor` Module" on page 46
- "Retrieving Reports That Are Generated by the `network-monitor` Module" on page 47
- "Viewing Statistics for the `network-monitor` Module With the `fmstat` Command" on page 48
- "Controlling the Use of Probes Through the `svc:/network/diagnostics` SMF Service" on page 49

Overview of the `network-monitor` Transport Module Utility

The `network-monitor` (also referred to as the *monitor* in this chapter) is a fault manager daemon (`fmd`) transport module utility that you use to perform network diagnostics on your Oracle Solaris 11 system. The utility monitors network resources and reports conditions that might lead to limited or degraded network functionality. When the monitor detects an abnormal network condition, a report (called an *ireport*) is generated. You can retrieve ireports by using the `fmdump` command. See "Retrieving Reports That Are Generated by the `network-monitor` Module" on page 47. The monitor does not perform any further diagnosis of the error condition, nor does it perform any additional recovery actions. See the `network-diagnostics`(4) man page for more details.

The monitor is controlled by property values that are stored in the `svc:/network/diagnostics` Service Management Facility (SMF) service. See "Controlling the Use of Probes Through the `svc:/network/diagnostics` SMF Service" on page 49 for more details.

How Datalink MTU Mismatch Errors Are Detected

This error condition occurs when there is a mismatch in the maximum transmission unit (MTU) between two peer datalinks. This type of mismatch can result in dropped frames because one datalink might transmit frames that are larger than the peer datalink can receive. The monitor attempts to detect any datalinks on the local system with MTUs that are set too high. Datalinks are verified once upon system start-up and then again when an MTU change occurs.

MTU verification is performed by using either the Link-Layer Discovery Protocol (LLDP) or the Internet Control Message Protocol (ICMP) probe method. A peer host that has the LLDP service enabled can include MTU details in the information exchange. The utility performs MTU verification by extracting peer MTU information. When LLDP information is unavailable, the monitor attempts to verify the MTU by transmitting a series of ICMP probes of different sizes until the datalink MTU is reached. A mismatch is flagged if the utility consistently fails to reach a target by using maximum-sized probes.

How Datalink VLAN ID Mismatch Errors Are Detected

Virtual local area networks (VLANs) are used to group end-system hosts into the same broadcast domain. The hosts on a VLAN might not reside on the same LAN, but even if they do, each host can communicate with another host by using Layer 2 (L2) protocols. Conversely, hosts that reside on the same LAN but different VLANs cannot communicate by using L2 protocols. Each host that resides on a VLAN uses a network interface to communicate with other hosts on the VLAN. VLANs are identified by VLAN identifiers (VIDs) that are exported by LLDP daemons over the relevant network interfaces to their peers. These peers are typically network devices, for example, switches that use a VID to forward data packets to respective hosts.

Hosts might not receive the intended packets if the VIDs are not configured correctly on the relevant network interfaces. The VLAN ID mismatch monitor captures this type of misconfiguration because it verifies the VID information whenever the VLAN information is modified, at system boot time and periodically. If the VID for an interface changes, the appropriate ireport messages are generated. Because the VLAN information is verified by using LLDP packets, the peer host needs to have the LLDP service enabled. See Chapter 5, "Exchanging Network Connectivity Information With Link Layer Discovery Protocol," in "Managing Network Datalinks in Oracle Solaris 11.2 ".

Managing the `network-monitor` Module

The `fmadm` command reports the current status of the monitor, which is displayed as `active` when it is performing fault monitoring, as shown in the following example:

```
# fmadm config

 MODULE                VERSION STATUS  DESCRIPTION
 cpumem-retire         1.1     active  CPU/Memory Retire Agent
 disk-diagnosis        0.1     active  Disk Diagnosis engine
 ...
 network-monitor       1.0     active  Network monitor
```

The `/usr/lib/fm/fmd/plugins/network-monitor.conf` configuration file has an `enable`
property that controls the state of the `network-monitor`. To enable the monitor, set the `enable`
property to `true` as follows:

```
# enable
#
# Enable/disable the network-monitor.
#
setprop enable true
```

The monitor will be active upon reboot.

Retrieving Reports That Are Generated by the `network-monitor` Module

If a problem with your network occurs, or if you suspect degraded network performance,
you can retrieve the ireports that are generated by the `network-monitor` by using the `fmdump`
command. These reports include the name of the datalink for which a potential problem was
detected.

For example, you can retrieve an ireport by running the following command:

`# fmdump -IVp -c 'ireport.os.sunos.net.datalink.*'`

`-I`   Specifies to retrieve information reports.

`-V`   Specifies to dump the contents of the report.

`-p`   Specifies to print the report.

`-c` *class*   Specifies the type of event.

                       The -c *class* option can be used to output only those events that match
                       a specific class. Events that are generated by the monitor use the
                       `'ireport.os.sunos.net.datalink'` class prefix.

For more information, see the `fmdump`(1M) man page.

The following example shows the output of an ireport that is posted by the `network-monitor`.

```
nvlist version: 0
    class = ireport.os.sunos.net.datalink.mtu_mismatch
    version = 0x0
    uuid = f3832064-e83b-6ce8-9545-8588db76493d
    pri = high
    detector = fmd:///module/network-monitor
    attr = (embedded nvlist)
    nvlist version: 0
 linkname = net0
 linkid = 0x3
 mtu = 0x1b58
    (end attr)
 __ttl = 0x1
 __tod = 0x513a4f2e 0x279ba218
```

The output of this particular ireport includes the following information:

class   Specifies the type of error condition. The ireports that are
                      posted by the `network-monitor` module are prefixed by
                      `ireport.os.sunos.net.datalink`. This information is specified with the
                      `-c` option, as shown in the previous example.

linkname   Specifies the name of the datalink for which the condition was detected.

Viewing Statistics for the `network-monitor` Module With the `fmstat` Command

The `fmstat` command reports fault management module statistics. You can also use the
command to view statistics for diagnosis engines and agents that are currently participating in
fault management, which includes the `network-monitor` transport module utility.

To view statistics that are kept by a specific fault management module, use the following
command syntax:

fmstat -m *module*

where -m *module* specifies the fault management module.

For example, you would view statistics for the `network-monitor` as follows:

```
# fmstat -m network-monitor
              NAME VALUE          DESCRIPTION
mtu-mismatch.allocerr 0            memory allocation errors
mtu-mismatch.enabled true         operating status for mtu-mismatch
mtu-mismatch.nprobes 7            number of transmitted ICMP probes
mtu-mismatch.procerr 0            errors processing datalinks
        sysev_drop 0             number of dropped sysevents
vlan-mismatch.enabled true        operating status for vlan-mismatch
```

For more information about using the `fmstat` command, see the `fmstat(1M)` man page.

To obtain a list of the modules that participate in fault management, use the `fmadm` command. See the `fmadm(1M)` man page.

Controlling the Use of Probes Through the `svc:/network/diagnostics` SMF Service

The types of diagnostics that the monitor performs are controlled by values that are stored in the `policy/allow_probes` property of the `svc:/network/diagnostics` SMF service. This property determines whether probe packets can be transmitted by diagnostic agents for the purpose of monitoring and reporting network problems. To set or change values for this property, use the `svccfg` command. Valid values are `true` and `false`. By default, the property is set to `true`. See the `svccfg(1M)` and `network-diagnostics(4)` man pages for more information.

EXAMPLE 4-1 Disabling the Transmission of Diagnostic Probes

The following example shows how you would disable the transmission of diagnostic probes by setting the `policy/allow_probes` property of the `svc:/network/diagnostics` SMF service to `false`. You must refresh the SMF service after changing the default value for the changes to take effect.

```
# svccfg -s network/diagnostics setprop policy/allow_probes = boolean: false
# svccfg -s network/diagnostics refresh
```

Index

Numbers and Symbols

6to4 relay router
  security issues,   19

D

`domainname` command
  NIS and,   37

H

hosts
  troubleshooting general problems,   11
`hosts` database
  checking entries,   14

I

`in.ndpd` daemon
  checking the status,   14
`inetd` daemon
  checking the status,   14
IPv6
  checking the status of `in.ndpd`,   14
  troubleshooting common IPv6 problems,   18, 19

M

monitoring the current state of all network
connections,   30

N

`netstat` command

running software checks,   14
network databases
  `hosts` database
    checking entries,   14
NIS
  client problems,   36
  commands hang,   35
  not responding error messages,   35
  unavailable error messages,   35
  `ypbind` "can't" messages,   35
not responding error messages (NIS),   35

R

routers
  problems upgrading for IPv6,   18
routing tables
  displaying,   11

S

security considerations
  6to4 relay router issues,   19

T

TCP/IP networks
  troubleshooting
    general methods,   11, 13
    software checks,   13
    third-party diagnostic programs,   11
troubleshooting
  IPv6 problems,   18, 19
  TCP/IP networks
    general methods,   11, 13