TECHNICAL NOTE

EMC® Avamar®
Post-Installation Network Configuration
Avamar 5.x, 6.x, or 7.x Server Software

Technical Note

P/N 300-015-091
REV 03

December 23, 2014

This technical note describes how to change the networking configuration of an existing EMC Avamar system after installation and implementation is complete, including IPv6 considerations introduced in Avamar 7.0. Topics include:

Revision history

The following table presents the revision history of this document:

Revision	Date	Description
03	December 23, 2014	Made policy-based replication section universal for all Avamar 7.x installations, added firewall configuration, Brocade switch, and post-Change Network Settings workflow procedural sections.
02	May 13, 2014	Added procedure for changing the IP/hostname in a policy-based replication setup.
01	July 10, 2013	First release of this document

Purpose

This technical note provides the background information and steps to change the networking configuration of an existing Avamar system after installation and implementation is complete.

The procedures that follow include methods for changing the IP address and hostname of a system, setting up VLANs and NAT, and understanding how IPv4 and IPv6 address formatting affect Avamar configuration.

Avamar, IPv4, and IPv6

Internet Protocol (IP) is a set of communication rules for routing traffic across networks to addressable devices like Avamar system components. Beginning with Avamar 7.0, an Avamar system supports both Internet Protocol Version 4 (IPv4) and IPv6 address notation (SLES version only).

- IPv4 notation is displayed as four octets, that is 1- to 3-digit base 10 numbers in a range of 0 to 255. Each octet is separated by periods and represents 8 bits of data for a total address space of 32 bits.

  A subnet mask identifies a range (a subnet) of IP addresses on the same network. For Avamar purposes, the subnet mask is /24, representative of a 255.255.255.0 netmask.

  Example of an IPv4 address and subnet mask: 10.99.99.99/24

  IPv4 notation cannot be abbreviated. If an octet has zero (0) value, it is indicated by a 0.

- IPv6 notation is displayed as 16 octets, that is 2-digit hexadecimal (base 16) numbers in a range of 00 to FF. Octets are combined by pairs into eight groups separated by colons, each group representing 16 bits of data for a total address space of 128 bits.

  For Avamar purposes, the subnet mask (called prefix in IPv6) is /64.

Example of an IPv6 address and prefix:
2001:db8:85a3:0042:1000:8a2e:0370:7334/64

With respect to groups with zero (0) value, IPv6 notation is different from IPv4 in that it can be abbreviated. For example, the following is a valid IPv6 address and prefix: 2001:db8:abcd:12::/64.

For more reference information about applicable subnets in IPv4 and IPv6 notation, see *IPv4 and IPv6 Subnets* (page 5).

In the Avamar user interface, an IP address may be displayed in either IPv4 or IPv6 notation. The notation type you see is dependent on how that particular component was initially configured.

IPv4 and IPv6 are not interoperable. They operate in separate stacks (that is, parallel, independent networks).

Avamar can be set up in a dual stack configuration. In that case, an individual Avamar component may have an IPv4 address, an IPv6 address, or both (one primary and the other secondary). Any part of the Avamar user interface may display a component's primary address or both dual stack addresses. The following IP address for a particular device indicates it is configured as dual stack: 10.99.99.99/24,2001:db8:abcd:12::/64

Supported Avamar networking configurations

Pre-Avamar 7.0 software supports the use of VLANs (virtual local area networks) since 6.x, NAT (network address translation), and IPv4 address formatting. Beginning with Avamar 7.0, IPv6 is supported in a limited set of use cases; this is explained below in the context of installation/implementation and post-installation.

Determination of whether Avamar 7.x supports the use of particular network configuration features depends upon the timing of the configuration -- during the installation and implementation process or post-installation.

IMPORTANT

Assume that use cases not mentioned in the following sections are not supported.

Avamar 7.x supported configurations -- installation and implementation

During installation/implementation, Avamar 7.x software supports the following use cases.

1. Pure IPv4 environment either with VLANs or NAT (configured through the dpnnetutil utility) or without them
2. Pure IPv6 environment without VLANs or NAT
3. Dual stack (IPv4 and IPv6) without VLANs or NAT in either stack

4. Dual stack in a customer environment that uses an existing VLAN/NAT-enabled IPv4 stack

   This special case is a two-step process:

   a. Installation/implementation of an Avamar system as a pure IPv4 stack with VLANs or NAT (use case #1 above)

   b. Addition of a non-VLAN/NAT-enabled IPv6 stack (see use case #1 under supported post-installation configurations below)

Descriptions for configuring these use cases, except as noted in #4, are not included in this document, but rather are contained in Avamar Procedure Generator and workflow installation and implementation documentation.

Avamar 7.x supported configurations -- post-installation

During post-installation activities, some use cases are supported by Avamar 7.x software and other are not supported. The supported use cases are described in this document.

- Supported:
  - Adding IPv6 to an existing IPv4 configuration (VLAN/NAT-enabled or not), resulting in a dual stack environment
  - Adding VLAN or NAT features to an existing IPv4 system (no IPv6)
- Not supported:
  - Migrating a pure IPv4 system, with or without VLANs/NAT, to IPv6
  - Adding VLANs/NAT to an IPv6 configuration
  - Migration of IPv6 configuration to IPv4

The following tables place this same information in a matrix format for easy consideration.

Table 1 Supported Avamar 7.x installation scenarios

Installation Scenarios	IPv4	IPv6	Dual Stack
VLANs	Yes *	No	No ***
NAT	Yes *	No	No ***
No VLANs or NAT	Yes **	Yes **	Yes **

* Use the dpnnetutil utility and then run the installation workflow.

** Run only the installation workflow.

*** Special case: An IPv4-only installation with VLANs or NAT, followed by the addition of an IPv6 stack post-installation (no VLANs or NAT associated with IPv6).

Table 2 Supported Avamar 7.x post-installation scenarios

	Existing System			
Installation Scenarios	IPv4 (no VLANS or NAT)	IPv4 (with VLANS or NAT)	IPv6 (no VLANS or NAT)	Dual Stack
Add IPv6 (creating dual stack)	Yes	Yes	NA	NA
Add VLANs/NAT	Yes	Yes	No	X
Migrate to IPv6	No	No	NA	X
Migrate to IPv4	NA	NA	No	X

IPv4 and IPv6 Subnets

Following are different netmask combinations (short and long version) for IPv4 and IPv6 notation.

IPv4

/8	255.0.0.0
/9	255.128.0.0
/10	255.192.0.0
/11	255.224.0.0
/12	255.240.0.0
/13	255.248.0.0
/14	255.252.0.0
/15	255.254.0.0
/16	255.255.0.0
/17	255.255.128.0
/18	255.255.192.0
/19	255.255.224.0
/20	255.255.240.0
/21	255.255.248.0
/22	255.255.252.0
/23	255.255.254.0
/24	255.255.255.0
/25	255.255.255.128
/26	255.255.255.192
/27	255.255.255.224
/28	255.255.255.240
/29	255.255.255.248
/30	255.255.255.252
/31	255.255.255.254
/32	255.255.255.255

IPv6	
/16	ffff:0000:0000:0000:0000:0000:0000:0000
/32	ffff:ffff:0000:0000:0000:0000:0000:0000
/48	ffff:ffff:ffff:0000:0000:0000:0000:0000
/64	ffff:ffff:ffff:ffff:0000:0000:0000:0000
/80	ffff:ffff:ffff:ffff:ffff:0000:0000:0000
/96	ffff:ffff:ffff:ffff:ffff:ffff:0000:0000
/112	ffff:ffff:ffff:ffff:ffff:ffff:ffff:0000

Changing IP address, hostname of an Avamar System

Use the instructions in this section to change the IP address or hostname of an Avamar system. Some procedures in this section are for systems running RHEL or SLES operating systems only; for IPv4, IPv6, or dual stack configurations only; and for internal switch configuration only. Perform those procedures that apply to the system you are modifying. So, for instance, if you are configuring a SLES system, skip over sections that indicate RHEL-only instructions.

A note about changing the IP address or hostname of an Avamar Virtual Edition (AVE) system. For the most part, AVE is treated like an Avamar single node server. The major difference between the two is that AVE does not have bond[0,1,2,3] devices due to the fact that it has only one eth0 network interface.

Prerequisites

Your Avamar server must be running Avamar 5.x, 6.x, or 7.x server software.

IMPORTANT

Before starting this procedure, ensure the Avamar system you intend to reconfigure is a healthy one, that is, it is in a known good controlled state. This includes ensuring that a validated (hfschecked) checkpoint is present. If not, take a checkpoint and validate it before proceeding any further. Refer to EMC KnowledgeBase article 163733 for more details (search for it on http://support.emc.com).

Also ensure that neither backups nor replication is in progress.

If the hostnames for other network components, such as the smtp mail server or authentication servers, have changed, you must update the Avamar server with the correct information. Refer to *Configuring the Avamar Downloader Service* (page 63) for more information.

Server Preparation

This section applies only when you are using this technical note to manually change the networking configuration of an Avamar system. Do not perform the following if you initially used the Change Network Settings workflow on an Avamar 7.1.1 system.

1. Open a command shell.
2. Log into the server as user admin.
3. When prompted for a password, type the admin password and press **ENTER**.
4. Load the admin OpenSSH key by entering:

   ```
   ssh-agent bash
   ssh-add ~admin/.ssh/admin_key
   ```

   You are prompted to type a passphrase.
5. Type the admin user account passphrase.
6. If administering a single-node server, turn off the unattended shutdown/restart feature by entering:

   ```
   dpnctl disable
   ```
7. Ensure that neither backups nor replication is currently running.

How to proceed

The configuration procedure you must follow varies depending on several factors.

For Avamar 7.1.1, a limited number of use cases support the use of the Change Network Settings workflow for performing many reconfiguration changes on Gen4/Gen4S systems. Even after using the workflow, some manual steps might be required to complete the reconfiguration. For those instances in which you have used the workflow to modify IP and hostname settings, follow the procedure in *Post-Change Network Settings workflow configuration procedure (Avamar 7.1.1 only)* (page 32).

In all other cases in which the Change Network Settings workflow is not used, this technical note describe the manual steps required to reconfigure your system. Choose the configuration procedure you must follow depending on the server platform and whether the existing system is configured to use IPv4, IPv6, or dual stack addressing:

- For Avamar software running on an RHEL server (Gen1, Gen2, Gen3, and customer-provided hardware), follow the procedure in *Configuration procedure for RHEL platforms* (page 8).
- For Avamar 6.x/7.x software on Gen4/Gen4S hardware with IPv4 addressing, follow the procedure in *Configuration procedure for SLES platforms using only IPv4 addressing (Avamar 6.x/7.x)* (page 11).
- For Avamar 6.x/7.x software on Gen4/Gen4S hardware with IPv6 addressing, follow the procedure in *Configuration procedure for SLES platforms using only IPv6 addressing (Avamar 7.x only)* (page 19).

- For Avamar 6.x/7.x software on Gen4/Gen4S hardware with dual stack addressing, follow the procedure in *Configuration procedure for SLES platforms using dual stack addressing (Avamar 7.x only)* (page 25).
- For Avamar Virtual Edition, follow the RHEL or SLES procedure, as appropriate to which operating system is running on the AVE.

Configuration procedure for RHEL platforms

For Avamar software running on a RHEL server (Gen1, Gen2, Gen3, and customer-provided hardware):

1. Ensure that you are still logged into the Avamar server as user admin.
2. Shutdown the Avamar server by entering:

   ```
   dpnctl stop
   ```

3. Switch user to root by entering:

   ```
   su -
   ```

4. Using a Unix text editor, edit the /etc/resolv.conf, /etc/hosts, /etc/sysconfig/network and /etc/sysconfig/network-scripts/ifcfg-`ETH` network files

   where `ETH` can be `eth0`, `eth1`, `eth2`, `eth3`, or `bond0` (two ports combined for failover redundancy -- a Gen3-only feature). The specific `ifcfg` file depends on whether the hardware has more than one port and whether bonding has been configured.

IMPORTANT

When configuring multi-node systems, /etc/hosts and /etc/resolv.conf should be identical on all nodes in the system.

   a. /etc/resolv.conf

      ```
      domain        local.example.com
      search        local.example.com company.com
      nameserver    127.0.0.1
      ```

      Set a nameserver if given. Set the nameserver as itself if there is no other nameserver.

   b. /etc/hosts on single-node servers

      ```
      127.0.0.1 localhost.localdomain  localhost
      10.0.44.5 avamar1.example.com    avamar1   #single node server
      ```

      Set the Avamar server IP address. Change the corresponding name if it is being changed.

c. /etc/hosts on multi-node servers

```
127.0.0.1     localhost.localdomain   localhost
10.0.44.5     avamar1.example.com     avamar1   #utility
10.0.44.6     avamar2.example.com     avamar2   #data
10.0.44.7     avamar3.example.com     avamar3   #data
10.0.44.8     avamar4.example.com     avamar4   #data
10.0.44.9     avamar5.example.com     avamar5   #data
10.0.55.10    avamar6.example.com     avamar6   #spare
```

Set the Avamar server IP address. Change the corresponding name if it is being changed.

Note: The contents of the /etc/sysconfig/network and /etc/sysconfig/network-scripts/ifcfg-ETH contain server-specific information.

d. /etc/sysconfig/network

```
NETWORKING=yes
HOSTNAME=HOSTNAME
```

Set the proper hostname.

e. /etc/sysconfig/network-scripts/ifcfg-ETH

```
DEVICE=ETH
BOOTPROTO=static
IPADDR=IP-ADDR
GATEWAY=GATEWAY-IP
NETMASK=NETMASK-IP
ONBOOT=yes
```

Edit the IP of the system, and set the gateway (if present), netmask and so forth, if they are being changed.

IMPORTANT

If network interface bonding was previously configured, you must modify two separate ifcfg-ETH files as well as the ifcfg-bond0 file. The three files must have the same IP address and netmask information.

5. In order to properly apply the new network settings, you must reboot the server by entering:

```
touch /fastboot
reboot
```

6. Repeat steps 1 through 5 for each node on the Avamar server.
7. Open a command shell.
8. Log into the Avamar server as user admin with a new name.
9. When prompted for a password, type the admin password and press **ENTER**.
10. Load the admin OpenSSH key by entering:

```
ssh-agent bash
ssh-add ~admin/.ssh/dpnid
```

You are prompted to type a passphrase.

11. Type the admin user account passphrase.
12. Remove the known_hosts files by typing:

```
mapall --all+ --noerror --user=root \
'rm ~{root,admin,dpn}/.ssh/known_hosts'
```

13. Verify network connectivity with the new settings by entering the following for each node:

```
ping HOSTNAME
```

    where HOSTNAME is the hostname of each node rebooted in step 5 of this procedure.

14. Change the following Avamar files:

Note: /usr/local/avamar/var/probe.xml exists only on the single-node server and utility node.

    a. Modify the probe.xml file to update node IP addresses by typing the following command for each changed node:

```
nodedb update if --addr=OLD-IP --new-addr=NEW-IP
```

    where **OLD-IP** and **NEW-IP** are the old and new IP addresses for the node. Repeat for each node for which the IP address was changed.

    b. To verify these changes, type the following command:

```
nodedb print
```

    c. Modify the probe.xml file to update the hostname of the node by typing the following command:

```
nodedb update module --index=0 --new-name=NEWNAME
```

    where **NEWNAME** is the hostname for either the single node server or the utility node of a multi-node server.

    d. If the interface configured in step (a) was configured for NAT and NAT information has changed, go to step (e). Otherwise, go to step 15.

    e. Remove the existing INITIAL/TARGET IP address pair to be updated by typing the following command:

```
nodedb delete nat --nat=INITIAL=TARGET
```

    If you have more than one NAT rule for the interface, append as many "**INITIAL=TARGET"** options to this command as required.

    f. To verify these changes, type the following command:

```
nodedb print
```

    g. Update the specific interface with an updated pair by typing the following command:

```
nodedb update if --addr=IP --new-nat=INITIAL=TARGET
```

    where **IP** is the same as **NEW-IP** in step (a). If you have more than one NAT rule for the interface, append as many "**,INITIAL=TARGET"** options to this command as required.

h. To verify these changes, type the following command:

```
nodedb print
```

15. Modify the value for "--server" in /usr/local/avamar/etc/usersettings.cfg:

```
--server=SERVER-NAME-OR-IP-ADDR
--vardir=/usr/local/avamar/var
--bindir=/usr/local/avamar/bin
--id=root
--password=ENCRYPTPWD
```

where SERVER-NAME-OR-IP-ADDR is the name or IP address of the Avamar single node server or the utility node of a multi-node server, and `ENCRYPTPWD` is an encrypted string for the password.

16. Switch user to root by entering:

```
su -
```

17. Do one of the following:

a. If configuring an Avamar server version 4.x or 5.x, run the following commands to update the new IP address and server name:

```
website create-cfg
website init
website restart
```

These commands edit the avamar.cfg file at /usr/local/avamar/etc/.

b. If configuring an Avamar server version 6.x and 7.x, run the following command:

```
website restart
```

Skip to *Post Configuration Procedure* (page 52).

Configuration procedure for SLES platforms using only IPv4 addressing (Avamar 6.x/7.x)

For Avamar 6.x/7.x software running on Gen4/Gen4S hardware, there are several configuration options. Networking files that require modification will vary depending on the specific configuration. The networking files are located in /etc/sysconfig/network/. Configuration options include:

- Single node or AVE systems
- Multinode systems
  - eth0 and eth2 configured as slaves to bond0

    Reserved for backup.
  - eth1 and eth3 configured as slaves to bond1

    For internal traffic.
  - eth4 and eth6 configured as slaves to bond2

    Reserved for optional replication on the utility node.

- eth5 and eth7 configured as slaves to bond3

  Reserved for optional management on utility node.

◆ VLAN configuration

- bond0.VLAN-IDs

  Optional multiple-tagged backup networks.

Procedure for changing the hostname and IP address on SLES platforms

For Avamar 6.x/7.x software running on Gen4/Gen4S hardware:

1. Ensure that you are still logged into the Avamar server as user admin.
2. Shutdown the Avamar server by entering:

   ```
   dpnctl stop
   ```

3. Switch user to root by entering:

   ```
   su -
   ```

4. Using a Unix text editor, edit the /etc/resolv.conf, /etc/hosts, /etc/HOSTNAME, and /etc/sysconfig/network/ifcfg-ETH network files

   where ETH can be eth0, eth1, eth2, eth3, or bond0 (two ports bonded for failover redundancy). The specific ifcfg file depends on whether the hardware has more than one port and whether bonding has been configured.

IMPORTANT

When configuring multi-node systems, /etc/hosts and /etc/resolv.conf should be identical on all nodes in the system.

a. /etc/resolv.conf

```
domain        local.example.com
search        local.example.com company.com
nameserver    127.0.0.1
```

Set a nameserver if given. Set the nameserver as itself if there is no other nameserver.

b. /etc/hosts on single-node servers

```
127.0.0.1 localhost.localdomain      localhost
10.0.44.5 avamar1.example.com        avamar1      #single node
server
```

Set the Avamar server IP address. Change the corresponding name if it is being changed.

c. /etc/hosts on multi-node servers

```
127.0.0.1     localhost.localdomain   localhost
10.0.44.5     avamar1.example.com              avamar1
#utility
10.0.44.6     avamar2.example.com              avamar2
#data
10.0.44.7     avamar3.example.com              avamar3
#data
10.0.44.8     avamar4.example.com              avamar4
#data
10.0.44.9     avamar5.example.com              avamar5
#data
10.0.55.10    avamar6.example.com              avamar6
#spare
192.168.255.1 avamar1-internal.example.com     avamar1-internal
#utility
    internal
192.168.255.2 avamar2-internal.example.com     avamar2-internal
#data
    internal
192.168.255.3 avamar3-internal.example.com     avamar3-internal
#data
    internal
192.168.255.4 avamar4-internal.example.com     avamar4-internal
#data
    internal
192.168.255.5 avamar5-internal.example.com     avamar5-internal
#data
    internal
192.168.255.6 avamar6-internal.example.com     avamar6-internal
#spare
    internal
```

Set the Avamar server IP address. Change the corresponding name if it is being changed.

Note: Due to space constraints on this page, lines in the above example wrap to the next line. They should not wrap in the actual file.

Note: If the hostname of the server is changing, you must preserve the naming scheme of HOST_NAME-internal, where HOST_NAME is the new hostname.

Note: The contents of the /etc/HOSTNAME and /etc/sysconfig/networks/ifcfg-ETH contain server-specific information.

d. /etc/HOSTNAME

```
HOST_NAME
```

where HOST_NAME is the proper hostname of the server.

e. /etc/sysconfig/network/ifcfg-ETH

Note: The following example assumes that the interface ETH is a slave of bond0, as indicated with the MASTER entry. The value for this entry will vary depending on the bond-id. If bonding is configured and characteristics of the bond are not changing, the ifcfg-ETH files should not need modification.

where ETH can be eth0, eth1, eth2, eth3, or bond0 (two ports bonded for failover redundancy).

```
STARTMODE=onboot
BOOTPROTO=none
USERCONTROL=no
ONBOOT=yes
MASTER=bond0
SLAVE=yes
```

f. /etc/sysconfig/network/ifcfg-bond0

```
STARTMODE=onboot
BOOTPROTO=static
IPADDR=IP-ADDR
NETMASK=NETMASK-IP
BONDING_MASTER=yes
BONDING_SLAVE0=ETH0
BONDING_SLAVE1=ETH2
BONDING_MODULE_OPTS="mode=active-backup miimon=100 updelay=2000
primary=eth0"
```

where IP-ADDR is the IP address associated with this bond, NETMASK-IP is the netmask associated with the IP address, and ETH0 and ETH2 are the slaves of this bond.

Depending on Avamar software version, "BONDING_MODULE_OPTS=" may be either of the following in the above example:

```
BONDING_MODULE_OPTS="mode=active-backup miimon=100 updelay=2000
primary=eth0"

BONDING_MODULE_OPTS="primary=eth0"
```

Note: If you have additional bonds, as described in *Configuration procedure for SLES platforms using only IPv4 addressing (Avamar 6.x/7.x)* (page 11), you need to repeat steps e and f for their respective configuration files.

g. If you are changing the names or ip addresses on a VLAN, you must also configure the VLAN-specific ifcfg files in /etc/sysconfig/network/ifcfg-bond0.VLAN-ID, where VLAN-ID is the id of the VLAN, for example:

/etc/sysconfig/network/ifcfg-bond0.123
/etc/sysconfig/network/ifcfg-bond0.222
/etc/sysconfig/network/ifcfg-bond0.333

The following is example content of the ifcfg-bond0.123 file:

```
STARTMODE=onboot
BOOTPROTO=static
IPADDR=IP-ADDR
NETMASK=NETMASK-IP
ETHERDEVICE=bond0
VLAN_ID=123
```

where IP-ADDR is the IP address of bond0 for this particular VLAN, and NETMASK-IP is the netmask associated with the IP address.

5. Set the default gateway in the /etc/sysconfig/network/routes file:

```
default GATEWAY-IP - -
```

where GATEWAY-IP is the IP address of the default gateway associated with the primary bond or network interface.

Note: The above example only shows a default gateway. Besides the mandatory default, the /routes file may contain additional static destination network routes. For example:

2000::13 2620:0:170:588::1 - -
10.12.12.0/24 10.6.98.1 - -

---

6. In order to properly apply the new network settings, you must reboot the server by entering:

```
touch /fastboot
reboot
```

7. Repeat steps 1 through 6 for each node on the grid.
8. Open a command shell.
9. Log into the Avamar server as user admin.
10. When prompted for a password, type the admin password and press **ENTER**.
11. Load the dpnid and admin OpenSSH keys by entering:

```
ssh-agent bash
ssh-add ~admin/.ssh/dpnid
ssh-add ~admin/.ssh/admin_key
```

You are prompted to type a passphrase.

12. Type the admin user account passphrase.
13. Remove the known_hosts files:

```
mapall --all+ --noerror --user=root \
'rm ~{root,admin,dpn}/.ssh/known_hosts'
```

14. Verify network connectivity with the new settings by entering for each node:

```
ping HOSTNAME
```

    where HOSTNAME is the hostname of each node rebooted in step 6 of this procedure.

15. Change the following Avamar files:

Note: /usr/local/avamar/var/probe.xml exists only on the single-node server and utility node.

    a. Modify the probe.xml file to update node IP addresses by typing the following command for each changed node:

```
nodedb update if --addr=OLD-IP --new-addr=NEW-IP
```

       where **OLD-IP** and **NEW-IP** are the old and new IP addresses for the node. Repeat for each node for which the IP address was changed.

    b. To verify these changes, type the following command:

```
nodedb print
```

    c. Modify the probe.xml file to update node hostname by typing the following command:

```
nodedb update module --index=0 --new-name=NEWNAME
```

       where **NEWNAME** is the hostname for either the single node server or the utility node of a multi-node server.

    d. If the interface configured in step (a) was configured for NAT and NAT information has changed, go to step (e). Otherwise, go to step (h).

    e. Remove the existing INITIAL/TARGET IP address pair to be updated by typing the following command:

```
nodedb delete nat --nat=INITIAL=TARGET
```

       If you have more than one NAT rule for the interface, append as many "**,INITIAL=TARGET**" options to this command as required.

    f. To verify these changes, type the following command:

```
nodedb print
```

g. Update the specific interface with an updated pair by typing the following command:

```
nodedb update if --addr=IP --new-nat=INITIAL=TARGET
```

where **IP** is the same as **NEW-IP** in step (a). If you have more than one NAT rule for the interface, append as many "**,INITIAL=TARGET"** options to this command as required.

h. To verify these changes, type the following command:

```
nodedb print
```

IMPORTANT

Use steps 15i through 15k for Avamar 7.x systems only. Otherwise, skip to step 16.

i. As user admin, check if you have a probe.xml consistent with a new install of version 7 by running:

```
nodedb print | grep userinput | wc -l
```

If the value returned is greater than 0 attributes, you must further modify the probe.xml by continuing with step (j). Otherwise, skip to step 16.

j. With a text editor, modify module attributes in probe.xml (located in /usr/local/avamar/var/) for the entire Avamar system. Use the following as an example of an object with module attributes:

```
<dpn>
  <module name="a4ipn600" userinput_domain="example.com"
userinput_gateway="10.110.227.1" userinput_pns="10.110.195.11"
userinput_sns="10.110.188.5" userinput_summary="24 storages, 1
utility">
```

where:

Table 3 Attribute definitions

Attribute	Definition
userinput_domain	Default fully-qualified domain name for any interface if a custom one is not provided.
userinput_gateway	Default gateway.
userinput_pns	Primary name server.
userinput_sns	Secondary name server. If not defined, either leave blank or omit attribute completely.

k. Modify each additional node object defined in the probe.xml and its attributes. Use the following as an example of a node object and its attributes:

```
<node type="utility" userinput_hostname="a4ipn600">
  <network-interface id="1" userinput_bonded="eth0,eth2"
userinput_ifname="eth0">
    <address newuserinput_value="10.110.227.147"
userinput_customhostname="a4ipn600.example.com"
userinput_netmask="255.255.255.0" value="10.110.227.147"/>
    <uses allow="replication,management,backup"/>
  </network-interface>
  <network-interface id="2" userinput_bonded="eth1,eth3"
userinput_ifname="eth1">
    <address newuserinput_value="192.168.255.1"
userinput_customhostname="a4ipn600.example.com"
userinput_netmask="255.255.255.0" value="192.168.255.1"/>
    <uses allow="internal"/>
  </network-interface>
</node>
```

where:

Table 4 Attribute definitions

Attribute	Definition
userinput_hostname	Hostname of the node.
newuserinput_value	IP Address of the interface.
userinput_customhostname	Fully-qualified domain name for the IP address of the interface. For example: A node is configured for three VLANs, one untagged backup, one internal interface, and separate replication and management interfaces, each configured as dual stack. Each unique IP address must have a unique FQDN defined by this attribute.
userinput_netmask	Netmask associated with the interface.
value	Old IP address before modification.

16. On single node servers, modify the value for "--server" in /usr/local/avamar/etc/usersettings.cfg:

```
--server=SERVER-NAME-OR-IP-ADDR
--vardir=/usr/local/avamar/var
--bindir=/usr/local/avamar/bin
--id=root
--password=ENCRYPTPWD
```

where SERVER-NAME-OR-IP-ADDR is the name or IP address of the Avamar single node server. ENCRYPTPWD is an encrypted string for the password.

17. Switch user to root by entering:

```
su -
```

18. Run the following command:

```
website restart
```

Skip to *Post Configuration Procedure* (page 52). Afterwards, if the IP address of the internal network switches must be changed, go to *Procedure for changing the IP address on Allied Telesis internal network switches (Gen4 and Gen4S only)* (page 41) or *Procedure for changing the IP address on Brocade internal network switches (Gen4S only)* (page 46), depending on the switch type.

Configuration procedure for SLES platforms using only IPv6 addressing (Avamar 7.x only)

For Avamar 7.x software running on Gen4/Gen4S hardware, there are several configuration options. Networking files that require modification will vary depending on the specific configuration. The networking files are located in /etc/sysconfig/network/. Configuration options include:

- Single node or AVE systems
- Multinode systems
  - eth0 and eth2 configured as slaves to bond0

    Reserved for backup.
  - eth1 and eth3 configured as slaves to bond1

    For internal traffic.
  - eth4 and eth6 configured as slaves to bond2

    Reserved for optional replication on the utility node.
  - eth5 and eth7 configured as slaves to bond3

    Reserved for optional management on utility node.

Procedure for changing the hostname and IP address on SLES platforms

For Avamar 7.x software running on Gen4/Gen4S hardware:

1. Ensure that you are still logged into the Avamar server as user admin.
2. Shutdown the Avamar server by entering:

   ```
   dpnctl stop
   ```
3. Switch user to root by entering:

   ```
   su -
   ```
4. Using a Unix text editor, edit the /etc/resolv.conf, /etc/hosts, /etc/HOSTNAME, and /etc/sysconfig/network/ifcfg-`ETH` network files

   where `ETH` can be `eth0`, `eth1`, `eth2`, `eth3`, or `bond0` (two ports bonded for failover redundancy). The specific `ifcfg` file depends on whether the hardware has more than one port and whether bonding has been configured.

IMPORTANT

When configuring multi-node systems,
/etc/hosts and /etc/resolv.conf should be identical on all nodes in the system.

a. /etc/resolv.conf

```
domain       local.example.com
search       local.example.com company.com
nameserver   NAMESERVERIP
```

where NAMESERVERIP is a valid IPv6 address. Set a nameserver address, if given. Set the nameserver as itself if there is no other nameserver.

b. /etc/hosts on single-node servers

```
SINGLENODEIP avamar1.example.com    avamar1      #single node
server
127.0.0.1       localhost
# special IPv6 addresses
::1             localhost ipv6-localhost ipv6-loopback
fe00::0         ipv6-localnet
ff00::0         ipv6-mcastprefix
ff02::1         ipv6-allnodes
ff02::2         ipv6-allrouters
ff02::3         ipv6-allhosts
```

where SINGLENODEIP is a valid IPv6 address. Set the Avamar server IP address. Change the corresponding name if it is being changed.

c. /etc/hosts on multi-node servers

```
UTILITYIP    avamar1.example.com             avamar1
#utility
DATA1IP      avamar2.example.com             avamar2
#data
DATA2IP      avamar3.example.com             avamar3
#data
DATA3IP      avamar4.example.com             avamar4
#data
DATA4IP      avamar5.example.com             avamar5
#data
SPAREIP      avamar6.example.com             avamar6
#spare
UTILITYINTIP avamar1-internal.example.com    avamar1-internal
#utility
   internal
DATA1INTIP   avamar2-internal.example.com    avamar2-internal
#data
   internal
DATA2INTIP   avamar3-internal.example.com    avamar3-internal
#data
   internal
DATA3INTIP   avamar4-internal.example.com    avamar4-internal
#data
   internal
DATA4INTIP   avamar5-internal.example.com    avamar5-internal
#data
   internal
SPAREINTIP   avamar6-internal.example.com    avamar6-internal
#spare
   internal
127.0.0.1       localhost
# special IPv6 addresses
::1             localhost ipv6-localhost ipv6-loopback
```

```
fe00::0         ipv6-localnet
ff00::0         ipv6-mcastprefix
ff02::1         ipv6-allnodes
ff02::2         ipv6-allrouters
ff02::3         ipv6-allhosts
```

where each token IP is a valid IPv6 address for each component. Set the Avamar server IP address. Change the corresponding name if it is being changed.

Note: Due to space constraints on this page, lines in the above example wrap to the next line. They should not wrap in the actual file.

Note: If the hostname of the server is changing, you must preserve the naming scheme of HOST_NAME-internal, where HOST_NAME is the new hostname.

Note: The contents of /etc/sysconfig/networks/ifcfg-ETH and /etc/HOSTNAME contain server-specific information.

d. /etc/HOSTNAME

```
HOST_NAME
```

where HOST_NAME is the proper hostname of the server.

e. /etc/sysconfig/network/ifcfg-ETH

Note: The following example assumes that the interface ETH is a slave of bond0, as indicated with the MASTER entry. The value for this entry will vary depending on the bond-id. If bonding is configured and characteristics of the bond are not changing, the ifcfg-ETH files should not need modification.

where ETH can be eth0, eth1, eth2, eth3, or bond0 (two ports bonded for failover redundancy).

```
STARTMODE=onboot
BOOTPROTO=none
USERCONTROL=no
ONBOOT=yes
MASTER=bond0
SLAVE=yes
```

f. /etc/sysconfig/network/ifcfg-bond0

```
STARTMODE=onboot
BOOTPROTO=static
IPADDR=IP-ADDR
BONDING_MASTER=yes
BONDING_SLAVE0=ETH0
BONDING_SLAVE1=ETH2
BONDING_MODULE_OPTS="primary=eth0"
```

where IP-ADDR is the IPv6 address (including the /64 netmask suffix) associated with this bond, and ETH0 and ETH2 are the slaves of this bond.

Note: If you have additional bonds, as described in *Configuration procedure for SLES platforms using only IPv6 addressing (Avamar 7.x only)* (page 19), you need to repeat steps e and f for their respective configuration files.

5. Set the default gateway in the /etc/sysconfig/network/routes file:

   ```
   default GATEWAY-IP - -
   ```

   where GATEWAY-IP is the IP address of the default gateway associated with the primary bond or network interface.

Note: The above example only shows a default gateway. Besides the mandatory default, the /routes file may contain additional static destination network routes. For example:

2620:0:170:59a::/64 2620:0:170:58f::1 - -
2620:0:170:570::/60 2620:0:170:58f::1 - -

6. Ensure that you are still logged into the Avamar server as root.
7. Switch to admin user by typing the following:

   ```
   su - admin
   ```

8. Remove the known_hosts files:

   ```
   mapall --all+ --noerror --user=root \
   'rm ~{root,admin,dpn}/.ssh/known_hosts'
   ```

9. Switch to root user by typing the following:

   ```
   exit
   ```

10. In order to properly apply the new network settings, you must reboot the server by entering:

    ```
    touch /fastboot
    reboot
    ```

11. Repeat steps 1 through 10 for each node on the grid.
12. Open a command shell.
13. Log into the Avamar server as user admin.
14. When prompted for a password, type the admin password and press **ENTER**.
15. Load the admin OpenSSH key by entering:

    ```
    ssh-agent bash
    ssh-add ~admin/.ssh/admin_key
    ```

    You are prompted to type a passphrase.

16. Type the admin user account passphrase.

17. Verify network connectivity with the new settings by entering the following for each node:

    ```
    ping6 HOSTNAME
    ```

    where HOSTNAME is the hostname of each node rebooted in step 10 of this procedure.

18. Change the following Avamar files:

Note: /usr/local/avamar/var/probe.xml exists only on the single-node server and utility node.

    a. Modify the probe.xml file to update node IP addresses by typing the following command for each changed node:

    ```
    nodedb update if --addr=OLD-IP --new-addr=NEW-IP
    ```

    where **OLD-IP** and **NEW-IP** are the old and new IP addresses for the node. Repeat for each node for which the IP address was changed.

    b. To verify these changes, type the following command:

    ```
    nodedb print
    ```

    c. Modify the probe.xml file to update node hostname by typing the following command:

    ```
    nodedb update module --index=0 --new-name=NEWNAME
    ```

    where **NEWNAME** is the hostname for either the single node server or the utility node of a multi-node server.

    d. To verify these changes, type the following command:

    ```
    nodedb print
    ```

    e. With a text editor, modify module attributes in probe.xml (located in /usr/local/avamar/var/) for the entire Avamar system. Use the following as an example of an object with module attributes:

    ```
    <dpn>
      <module name="a4ipn600" userinput_domain="example.com"
    userinput_gateway="10.110.227.1" userinput_pns="10.110.195.11"
    userinput_sns="10.110.188.5" userinput_summary="24 storages, 1
    utility">
    ```

    where:

Table 5 Attribute definitions

Attribute	Definition
userinput_domain	Default fully-qualified domain name for any interface if a custom one is not provided.
userinput_gateway	Default gateway.
userinput_pns	Primary name server.
userinput_sns	Secondary name server. If not defined, either leave blank or omit attribute completely.

f. Modify each additional node object defined in the probe.xml and its attributes. Use the following as an example of a node object and its attributes:

```
<node type="utility" userinput_hostname="a4ipn600">
  <network-interface id="1" userinput_bonded="eth0,eth2"
userinput_ifname="eth0">
    <address newuserinput_value="10.110.227.147"
userinput_customhostname="a4ipn600.example.com"
userinput_netmask="255.255.255.0" value="10.110.227.147"/>
    <uses allow="replication,management,backup"/>
  </network-interface>
  <network-interface id="2" userinput_bonded="eth1,eth3"
userinput_ifname="eth1">
    <address newuserinput_value="192.168.255.1"
userinput_customhostname="a4ipn600.example.com"
userinput_netmask="255.255.255.0" value="192.168.255.1"/>
    <uses allow="internal"/>
  </network-interface>
</node>
```

where:

Table 6 Attribute definitions

Attribute	Definition
userinput_hostname	Hostname of the node.
newuserinput_value	IP Address of the interface.
userinput_customhostname	Fully-qualified domain name for the IP address of the interface. For example: A node is configured for three VLANs, one untagged backup, one internal interface, and separate replication and management interfaces, each configured as dual stack. Each unique IP address must have a unique FQDN defined by this attribute.
userinput_netmask	Netmask associated with the interface.
value	Old IP address before modification.

19. Switch user to root by entering:

```
su -
```

20. On single node servers, modify the value for "--server" in /usr/local/avamar/etc/usersettings.cfg:

```
--server=SERVER-NAME-OR-IP-ADDR
--vardir=/usr/local/avamar/var
--bindir=/usr/local/avamar/bin
--id=root
--password=ENCRYPTPWD
```

where SERVER-NAME-OR-IP-ADDR is the name or IP address of the Avamar single node server. ENCRYPTPWD is an encrypted string for the password.

21. Run the following command:

```
website restart
```

Skip to *Post Configuration Procedure* (page 52). Afterwards, if the IP address of the internal network switches must be changed, go to *Procedure for changing the IP address on Allied Telesis internal network switches (Gen4 and Gen4S only)* (page 41) or *Procedure for changing the IP address on Brocade internal network switches (Gen4S only)* (page 46), depending on the switch type.

Configuration procedure for SLES platforms using dual stack addressing (Avamar 7.x only)

For Avamar 7.x software running on Gen4/Gen4S hardware, there are several configuration options. Networking files that require modification will vary depending on the specific configuration. The networking files are located in /etc/sysconfig/network/. Configuration options include:

- Single node or AVE systems
- Multinode systems
  - eth0 and eth2 configured as slaves to bond0

    Reserved for backup.
  - eth1 and eth3 configured as slaves to bond1

    For internal traffic.
  - eth4 and eth6 configured as slaves to bond2

    Reserved for optional replication on the utility node.
  - eth5 and eth7 configured as slaves to bond3

    Reserved for optional management on utility node.
- VLAN configuration (for IPv4 stack only)
  - bond0.VLAN-IDs

    Optional multiple-tagged backup networks.

Procedure for changing the hostname and IP address on SLES platforms

For Avamar 7.x software running on Gen4/Gen4S hardware:

1. Ensure that you are still logged into the Avamar server as user admin.
2. Shutdown the Avamar server by entering:

   ```
   dpnctl stop
   ```
3. Switch user to root by entering:

   ```
   su -
   ```

4. Using a Unix text editor, edit the /etc/resolv.conf, /etc/hosts, /etc/HOSTNAME, and /etc/sysconfig/network/ifcfg-ETH network files

   where ETH can be eth0, eth1, eth2, eth3, or bond0 (two ports bonded for failover redundancy). The specific ifcfg file depends on whether the hardware has more than one port and whether bonding has been configured.

IMPORTANT

When configuring multi-node systems, /etc/hosts and /etc/resolv.conf should be identical on all nodes in the system.

a. /etc/resolv.conf

```
domain        local.example.com
search        local.example.com company.com
nameserver    NAMESERVERIP
```

   where NAMESERVERIP is either a valid IPv4 or IPv6 address. Set a nameserver address, if given. Set the nameserver as itself if there is no other nameserver.

b. /etc/hosts on single-node servers (for multi-node servers, skip to step c)

```
SINGLENODEIP avamar1.example.com     avamar1       #single node
server
127.0.0.1       localhost
# special IPv6 addresses
::1             localhost ipv6-localhost ipv6-loopback
fe00::0         ipv6-localnet
ff00::0         ipv6-mcastprefix
ff02::1         ipv6-allnodes
ff02::2         ipv6-allrouters
ff02::3         ipv6-allhosts
```

   where SINGLENODEIP is either a valid IPv4 or IPv6 address. Set the Avamar server IP address. Change the corresponding name if it is being changed.

IMPORTANT

When changing IP addresses in dual-stack configurations, EMC strongly recommends you also provide a unique hostname too for each changed IPv6 IP address. This can be accomplished by creating unique hostnames or subdomains like the following examples:

hostname.domain.local (IPv4)
hostname6.domain.local (IPv6)

hostname.domain.local (IPv4)
hostname.ipv6.domain.local (IPv6)

c. /etc/hosts on multi-node servers

```
UTILITYIP     avamar1.example.com              avamar1
#utility
DATA1IP       avamar2.example.com              avamar2
#data
DATA2IP       avamar3.example.com              avamar3
#data
DATA3IP       avamar4.example.com              avamar4
#data
DATA4IP       avamar5.example.com              avamar5
#data
SPAREIP       avamar6.example.com              avamar6
#spare
UTILITYINTIP  avamar1-internal.example.com     avamar1-internal
#utility
   internal
DATA1INTIP    avamar2-internal.example.com     avamar2-internal
#data
   internal
DATA2INTIP    avamar3-internal.example.com     avamar3-internal
#data
   internal

DATA3INTIP    avamar4-internal.example.com     avamar4-internal
#data
   internal
DATA4INTIP    avamar5-internal.example.com     avamar5-internal
#data
   internal
SPAREINTIP    avamar6-internal.example.com     avamar6-internal
#spare
   internal
127.0.0.1       localhost
# special IPv6 addresses
::1             localhost ipv6-localhost ipv6-loopback
fe00::0         ipv6-localnet
ff00::0         ipv6-mcastprefix
ff02::1         ipv6-allnodes
ff02::2         ipv6-allrouters
ff02::3         ipv6-allhosts
```

where each token IP is either a valid IPv4 or IPv6 address for each component. Set the Avamar server IP address. Change the corresponding name if it is being changed.

IMPORTANT

When changing IP addresses in dual-stack configurations, EMC strongly recommends you also provide a unique hostname too for each changed IPv6 IP address. This can be accomplished by creating unique hostnames or subdomains like the following examples:

hostname.domain.local (IPv4)
hostname6.domain.local (IPv6)

hostname.domain.local (IPv4)
hostname.ipv6.domain.local (IPv6)

Note: Due to space constraints on this page, lines in the above example wrap to the next line. They should not wrap in the actual file.

Note: If the hostname of the server is changing, you must preserve the naming scheme of HOST_NAME-internal, where HOST_NAME is the new hostname.

Note: The contents of the /etc/HOSTNAME and /etc/sysconfig/networks/ifcfg-ETH (next steps) contain server-specific information.

d. /etc/HOSTNAME

```
HOST_NAME
```

where HOST_NAME is the proper hostname of the server.

e. /etc/sysconfig/network/ifcfg-ETH

Note: The following example assumes that the interface ETH is a slave of bond0, as indicated with the MASTER entry. The value for this entry will vary depending on the bond-id. If bonding is configured and characteristics of the bond are not changing, the ifcfg-ETH files should not need modification.

where ETH can be eth0, eth1, eth2, eth3, or bond0 (two ports bonded for failover redundancy).

```
STARTMODE=onboot
BOOTPROTO=none
USERCONTROL=no
ONBOOT=yes
MASTER=bond0
SLAVE=yes
```

f. /etc/sysconfig/network/ifcfg-bond0

```
STARTMODE=onboot
BOOTPROTO=static
IPADDR=IP-ADDR
IPADDR_0=IP-ADDR_0
NETMASK=NETMASK-IP
BONDING_MASTER=yes
BONDING_SLAVE0=ETH0
BONDING_SLAVE1=ETH2
BONDING_MODULE_OPTS="primary=eth0"
```

where IP-ADDR is the IPv6 address (including /64 netmask suffix) associated with this bond, IP-ADDR_0 is the IPv4 address associated with this bond, NETMASK-IP is the IPv4 netmask associated with the IPv4 address, and ETH0 and ETH2 are the slaves of this bond.

Note: If you have additional bonds, as described in *Configuration procedure for SLES platforms using dual stack addressing (Avamar 7.x only)* (page 25), you need to repeat steps e and f for their respective configuration files.

g. If you are changing the names or IP addresses on a VLAN (in the IPv4 stack only), you must also configure the VLAN-specific ifcfg files in /etc/sysconfig/network/ifcfg-bond0.VLAN-ID, where VLAN-ID is the id of the VLAN, for example:

/etc/sysconfig/network/ifcfg-bond0.123
/etc/sysconfig/network/ifcfg-bond0.222
/etc/sysconfig/network/ifcfg-bond0.333

The following is example content of the ifcfg-bond0.123 file:

```
STARTMODE=onboot
BOOTPROTO=static
IPADDR=IP-ADDR
NETMASK=NETMASK-IP
ETHERDEVICE=bond0
VLAN_ID=123
```

where IP-ADDR is the IP address of bond0 for this particular VLAN, and NETMASK-IP is the netmask associated with the IP address.

5. Set the default gateway in the /etc/sysconfig/network/routes file:

```
default GATEWAY-IP - -
```

where GATEWAY-IP is the IP address (either IPv4 or IPv6) of the default gateway associated with the primary bond or network interface.

Note: The above example only shows a default gateway. Besides the mandatory default, the /routes file may contain additional static destination network routes. For example:

2000::13 2620:0:170:588::1 - -
10.12.12.0/24 10.6.98.1 - -

6. Ensure that you are still logged into the Avamar server as root.
7. Switch to admin user by typing the following:

```
su - admin
```

8. Remove the known_hosts files:

```
mapall --all+ --noerror --user=root \
'rm ~{root,admin,dpn}/.ssh/known_hosts'
```

9. Switch to root user by typing the following:

```
exit
```

10. In order to properly apply the new network settings, you must reboot the server by entering:

```
touch /fastboot
reboot
```

11. Repeat steps 1 through 9 for each node on the grid.

12. Open a command shell.
13. Log into the Avamar server as user admin.
14. When prompted for a password, type the admin password and press **ENTER**.
15. Load the admin OpenSSH key by entering:

    ```
    ssh-agent bash
    ssh-add ~admin/.ssh/admin_key
    ```

    You are prompted to type a passphrase.
16. Type the admin user account passphrase.
17. Verify network connectivity with the new settings by entering for each node:

    ```
    ping HOSTNAME
    ```

    where HOSTNAME is the hostname of each node rebooted in step 6 of this procedure.
18. Change the following Avamar files:

Note: /usr/local/avamar/var/probe.xml exists only on the single-node server and utility node.

a. Modify the probe.xml file to update node IP addresses by typing the following command for each changed node:

   ```
   nodedb update if --addr=OLD-IP --new-addr=NEW-IP
   ```

   where **OLD-IP** and **NEW-IP** are the old and new IP addresses for the node. Repeat for each node for which the IP address was changed.

b. To verify these changes, type the following command:

   ```
   nodedb print
   ```

c. Modify the probe.xml file to update node hostname by typing the following command:

   ```
   nodedb update module --index=0 --new-name=NEWNAME
   ```

   where **NEWNAME** is the hostname for either the single node server or the utility node of a multi-node server.

d. If the interface configured in step (a) was configured for NAT and NAT information has changed, go to step (e). Otherwise, go to step 16.

IMPORTANT

If applicable to the system you are reconfiguring, do steps e through g that follow only for the IPv4 stack.

e. Remove the existing INITIAL/TARGET IP address pair to be updated by typing the following command:

```
nodedb delete nat --nat=INITIAL=TARGET
```

If you have more than one NAT rule for the interface, append as many "**,INITIAL=TARGET"** options to this command as required.

f. To verify these changes, type the following command:

```
nodedb print
```

g. Update the specific interface with an updated pair by typing the following command:

```
nodedb update if --addr=IP --new-nat=INITIAL=TARGET
```

where **IP** is the same as **NEW-IP** in step (a). If you have more than one NAT rule for the interface, append as many "**,INITIAL=TARGET"** options to this command as required.

h. To verify these changes, type the following command:

```
nodedb print
```

19. On single node servers, modify the value for "--server" in /usr/local/avamar/etc/usersettings.cfg:

```
--server=SERVER-NAME-OR-IP-ADDR
--vardir=/usr/local/avamar/var
--bindir=/usr/local/avamar/bin
--id=root
--password=ENCRYPTPWD
```

where SERVER-NAME-OR-IP-ADDR is the name or IP address of the Avamar single node server. ENCRYPTPWD is an encrypted string for the password.

20. Switch user to root by entering:

```
su -
```

21. Run the following command:

```
website restart
```

Skip to *Post Configuration Procedure* (page 52). Afterwards, if the IP address of the internal network switches must be changed, go to *Procedure for changing the IP address on Allied Telesis internal network switches (Gen4 and Gen4S only)* (page 41) or *Procedure for changing the IP address on Brocade internal network switches (Gen4S only)* (page 46), depending on the switch type.

Post-Change Network Settings workflow configuration procedure (Avamar 7.1.1 only)

Use this procedure only on Gen4/Gen4S hardware running Avamar 7.1.1 software, and only after running the Change Network Settings workflow.

The Change Network Settings workflow supports a limited number of use cases related to network reconfiguration in three general scenarios:

- In-place without a physical move
- Prior to a physical move
- After a physical move

Supported use cases

Within the three scenarios outlined above, the following tables describe which use cases the workflow supports and which ones it does not.

Table 7 Workflow supports these use cases

Factor	Supported
IP address notation	IPv4 only
Interfaces that can be modified	• Backup interface on bond0 • Replicationinterface on bond0, bond2 (multi-node), or eth1 (single-node) • Management interface on bond0, bond3 (multi-node) or eth3 (single-node)
Global settings that can be modified	• Default domain • Default search • Default gateway • Primary, secondary, and tertiary DNS servers
Network interface settings that can be reconfigured on each node	• IP address • Netmask • Hostname • Domain
Interfaces that can be added on each node	• A dedicated source replication interface (multi-node server: on bond2, single node server: on eth1) • A dedicated management interface (multi-node server: on bond3, single node server: on eth3)
Interfaces that can be removed on each node (single node servers)	Management interface (eth3)
Interfaces that can be removed on each node (multi-node servers)	• Replication interface • Management interface

During its operation, the workflow does the following:

- Validates changed data
- Automates reconfiguration of firewall settings

- Ensures operability of the following:
  - Avamar subsystem (GSAN)
  - Avamar Administrator
  - Enterprise Manager
  - MCCLI
  - Avamar Installation Manager
- Verifies that all nodes are reachable after applying network configuration changes
- Logs changes in the EMC SYR system

In some circumstances, additional post-workflow configuration might be required for external systems connected to the server to operate properly. See *Potential manual steps* (page 33).

Table 8 Workflow does not support these use cases

Factor	Unsupported
IP address notation	IPv6 or dual stack
Systems with	VLANs, NAT, or both
Changes to interface	Internal server network IP configuration
Interfaces that cannot be modified	Internal interface on bond1 (multi-node)
Interfaces that cannot be removed (single node servers)	Replication interface (eth1)
Modifications to these Avamar products	• Avamar Virtual Edition (AVE) • Media Access nodes • Accelerator nodes

For each use case the workflow does not support, you must use the manual procedures described in this technical note. Begin with *How to proceed* (page 7).

Potential manual steps

After running the Change Network Settings workflow, you might have to reconfigure one or more of the following external subsystems:

Replication

If you reconfigured network settings on an Avamar server that is a replication source, you must reregister replication under /MC_SYSTEM. See *Replication-related post configuration procedure (Avamar 7.x only)* (page 60) for instructions.

If you reconfigured network settings on an Avamar server that is a replication target, you must update the IP address of the target on the source Avamar server. For instructions, see “Editing a replication destination” in the *EMC Avamar Adminstration Guide*.

Backup Clients

If you used the workflow to reconfigure only backup IP addresses (not hostnames) on an Avamar server and all clients backing up data to the server were originally registered via server hostname, no further steps are required. The same is true for the opposite scenario: You reconfigured only hostnames on the server and all clients were originally registered via server IP address.

Any scenario in which a) the method for originally registering clients (IP address or hostname of the Avamar server) and b) the aspect of the Avamar server you reconfigured with the workflow match, you must update all clients backing up data to the Avamar server with the new IP address or hostname of the Avamar server. For client-side registration instructions for each supported operating system, see the *EMC Avamar Backup Clients User Guide*.

Note: In large enterprises with many clients, the customer instead might consider doing an IP/hostname redirect function on the DNS resolver between clients and Avamar server. Since the topology of any given customer site is unknowable, the customer system administrator is responsible for knowing how to do this.

Avamar Enterprise Manager Server (EMS)

If the Avamar Enterprise Manager Server is associated with a single Avamar system, the Change Network Settings workflow reconfigures EMS automatically. No further steps are required.

If EMS is used to manage multiple Avamar systems, you must update the EMS with the changed IP address of the server. For instructions, see “Adding an Avamar system in Avamar Enterprise Manager” in the *EMC Avamar Adminstration Guide*.

Avamar Downloader Service

If Avamar Downloader Service was not originally installed, no further steps are required.

If ADS was originally installed, you must update the hosting Windows computer with the new IP address of the Avamar server. For instructions, see “Configuring the Avamar Downloader Service” in the *EMC Avamar Adminstration Guide*.

Mail Server

If the reconfigured Avamar system was not moved or was moved to a location that uses the same mail server, no futher steps are required.

If the reconfigured Avamar system was moved to a new location that requires a mail server hostname change, you must reconfigure ConnectEMC and EmailHome to use the new hostname of the mail server. For instructions, see “Automatic notifications to EMC Customer Support” in the *EMC Avamar Adminstration Guide*.

Apache SSL Certificate Regeneration

If the hostname of the reconfigured Avamar server has not changed (only a re-IP configuration), no futher steps are required.

If the hostname of the Avamar server has changed, then use /usr/local/avamar/bin/gen-ssl-cert to install a new self-signed SSL certificate for the Apache web server. For information about obtaining and installing an SSL certificate from a Certificate Authority (CA), see the *EMC Avamar Product Security Guide* .

Accelerator Node

If the Avamar server probe.xml file did not contain an accelerator node object before running the Change Network Settings workflow, no further steps are required.

If the probe.xml file did contain an accelerator node object pre-workflow, you must add that object back in the file with the following command (as root user):

```
nodedb add node --addr=IP_ADDRESS --type=accelerator --nwgrp=1
```

where `IP_ADDRESS` is the IP address of the accelerator node.

To confirm the addition in probe.xml, use the `nodedb print` command and search for information similar to the following sample screen output (should appear near the bottom of the file):

```
<node type="accelerator">
   <network-interface id="1">
      <address value="10.471.2.5"/>
   </network-interface>
</node>
```

Avamar System Internal Network

If reconfiguration of the Avamar server internal network or either internal switch is not needed, no futher steps are required.

If reconfiguration of the internal network is required, do this by following the manual steps in this technical note. Begin with *How to proceed* (page 7).

If reconfiguration of the internal switches is required, see either *Procedure for changing the IP address on Allied Telesis internal network switches (Gen4 and Gen4S only)* (page 41) or *Procedure for changing the IP address on Brocade internal network switches (Gen4S only)* (page 46), whichever is appropriate.

Interface Deletion

If you did not use the workflow to delete any interfaces, no further steps are required.

If you used the workflow to delete a dedicated replication or management interface on a multi-node Avamar system, some configuration cleanup is required.

If you used the workflow to delete a management interface on a single node Avamar server, some configuration cleanup is required.

Also, as noted above, the workflow does not support removal of a dedicated replication interface on a single node Avamar server. This requires a manual process.

See the appropriate subsection below for multi-node replication and management interface clean up, single node management interface clean up, or single node replication removal.

Multi-node system replication interface clean up

To clean up after running the workflow to remove a replication interface, you must perform the following manual steps:

Files to be modified

- /etc/sysconfig/network/ifcfg-bond2
- /etc/sysconfig/network/ifcfg-eth4
- /etc/sysconfig/network/ifcfg-eth6
- /etc/modprobe.conf.local

1. As root user, remove the /etc/sysconfig/network/ifcfg-bond2 file if it still exists.
   a. Confirm whether it still exists by typing:

      ```
      ls /etc/sysconfig/network/ifcfg-bond2
      ```

   b. If the result is the following, skip to step 2:

      ```
      ls: cannot access /etc/sysconfig/network/ifcfg-bond2: No such
      file or directory
      ```

   c. If not, remove the file by typing:

      ```
      rm /etc/sysconfig/network/ifcfg-bond2
      ```

2. Create both /etc/sysconfig/network/ifcfg-eth4 and /etc/sysconfig/network/ifcfg-eth6 files with a text editor.

   The contents of both files should be:

   ```
   STARTMODE=onboot
   ```

3. Save changes to both files.
4. Ensure the /etc/modprobe.conf.local file accurately reflect the correct number of bonds and the correct bonds, open the file with a text editor.

   The file contains content similar to the following:

   ```
   #
   # please add local extensions to this file
   #
   options bonding max_bonds=4 mode=active-backup miimon=100
   updelay=2000
   alias bond0 bonding
   alias bond1 bonding
   alias bond2 bonding
   alias bond3 bonding
   ```

5. Remove the alias line for bond2 bonding and change max_bonds (if it appears in the file) to reflect new number of bonds.

   The resulting file content would be similar to the following:

   ```
   #
   # please add local extensions to this file
   #
   options bonding max_bonds=3 mode=active-backup miimon=100
   updelay=2000
   alias bond0 bonding
   alias bond1 bonding
   alias bond3 bonding
   ```

6. Save changes.
7. Restart network by typing:

   ```
   service network restart
   ```

Multi-node system management interface clean up

To clean up after running the workflow to remove a management interface, you must perform the following manual steps:

Files to be modified

- /etc/sysconfig/network/ifcfg-bond3
- /etc/sysconfig/network/ifcfg-eth5
- /etc/sysconfig/network/ifcfg-eth7
- /etc/modprobe.conf.local

1. As root user, remove the /etc/sysconfig/network/ifcfg-bond3 file if it still exists.
   a. Confirm whether it still exists by typing:

      ```
      ls /etc/sysconfig/network/ifcfg-bond3
      ```

   b. If the result is the following, skip to step 2:

      ```
      ls: cannot access /etc/sysconfig/network/ifcfg-bond3: No such
      file or directory
      ```

   c. If not, remove the file by typing:

      ```
      rm /etc/sysconfig/network/ifcfg-bond3
      ```

2. Create an /etc/sysconfig/network-ifcfg-eth5 file with a text editor.

   The contents of the file should be:

   ```
   STARTMODE=onboot
   ```

3. Save changes.

4. For Avamar Gen4S systems only, create an /etc/sysconfig/network-ifcfg-eth7 file with a text editor.

   The contents of the file should be:

   ```
   STARTMODE=onboot
   ```

   If you want to return /etc/sysconfig/network/ifcfg-eth7 back to the maintenance port, do the following (otherwise save changes and then skip to step 5):

   a. Replace current content with:

      ```
      BOOTPROTO='static'
      IPADDR='10.99.99.5/24'
      PREFIXLEN='24'
      STARTMODE='auto'
      USERCONTROL='no'
      ```

   b. Save changes.

5. To ensure bond3 is removed, open the /etc/modprobe.conf.local file with a text editor.

   The file contains content similar to the following:

   ```
   #
   # please add local extensions to this file
   #
   options bonding max_bonds=4 mode=active-backup miimon=100
   updelay=2000
   alias bond0 bonding
   alias bond1 bonding
   alias bond2 bonding
   alias bond3 bonding
   ```

6. Remove the alias line for bond3 bonding and change max_bonds (if it appears in the file) to reflect new number of bonds.

   The resulting file content would be similar to the following:

   ```
   #
   # please add local extensions to this file
   #
   options bonding max_bonds=3 mode=active-backup miimon=100
   updelay=2000
   alias bond0 bonding
   alias bond1 bonding
   alias bond2 bonding
   ```

7. Save changes.

8. Restart network by typing:

   ```
   service network restart
   ```

Single node server management interface clean up

To clean up after running the workflow to remove a management interface on a single node server, you must perform the following manual steps:

1. Using a text editor, create the following file:

   - /etc/sysconfig/network/ifcfg-eth3

   Add the following entry: `STARTMODE=onboot`

2. Save changes.

3. Restart eth3 by typing:

```
ifdown eth3; ifup eth3
```

Single node server replication interface removal

To remove a replication interface on a single node server, you must perform the following manual steps:

Note: This section relies on the following sample data: hostname - replication-test, IP address - 1.2.3.4.

Files to be modified

- /etc/hosts
- /etc/ssh/sshd_config
- /etc/sysconfig/network/ifcfg-eth1
- /usr/local/avamar/var/probe.xml
- /usr/local/avamar/var/dpnnetutil.xml

1. As root user, stop the Avamar subsystem by typing:

```
dpnctl stop
```

2. Using a text editor, modify the following files:

   - /etc/hosts

   Remove the entire entry that corresponds to replication interface and then save changes.

```
1.2.3.4  replication-test.example.com  replication-test
```

   - /etc/ssh/sshd_config

   Remove the IP address that corresponds to the replication interface (1.2.3.4 in the example below) in a line similar to the following and then save changes:

```
Match Address ::1,10.20.5.4,1.2.3.4
```

   - /etc/sysconfig/network/ifcfg-eth1

   Remove all entries except "STARTMODE=onboot". If "STARTMODE=onboot" is not in the file, add it. Save changes.

   - /usr/local/avamar/var/probe.xml

   Example:

```
...
        <uses allow="management,internal,backup"/>
      </network-interface>
<network-interface id="2" userinput_bonded=""
userinput_ifname="eth1">
        <address value="1.2.3.4" userinput_netmask="255.255.255.0"
newuserinput_value="1.2.3.4"
userinput_customhostname="replication-test.example.com"/>
        <uses allow="replication"/>
      </network-interface>
```

Search for an instance of "network-interface" that contains a "uses allow" value set to "replication" and remove the entire entry (see below) from the file.

```
<network-interface id="2" userinput_bonded=""
userinput_ifname="eth1">
        <address value="1.2.3.4" userinput_netmask="255.255.255.0"
newuserinput_value="1.2.3.4"
userinput_customhostname="replication-test.example.com"/>
        <uses allow="replication"/>
      </network-interface>
```

Add "replication" to the “uses allow” element of bond0 (see example below).

```
<?xml version="1.0" encoding="UTF-8"?>
<dpn>
  <module name="example" userinput_gateway="10.20.5.4"
userinput_domain="example.com" userinput_search="example.com"
userinput_pns="10.10.10.10" userinput_sns="10.10.10.11">
    <node type="single-node server" userinput_gateway="10.20.5.1">
      <network-interface id="1" userinput_ifname="bond0"
userinput_bonded="eth0,eth2">
       <address value="10.20.5.4" userinput_netmask="255.255.255.0"
newuserinput_value="10.20.5.4"
userinput_customhostname="example.example.com"/>
        <uses allow="internal,backup,management,replication"/>
      </network-interface>
    </node>
  </module>
```

Save changes.

- /usr/local/avamar/var/dpnnetutil.xml

Example:

```
...
        <uses allow="management,internal,backup"/>
      </network-interface>
<network-interface id="2" userinput_bonded=""
userinput_ifname="eth1">
        <address value="1.2.3.4" userinput_netmask="255.255.255.0"
newuserinput_value="1.2.3.4"
userinput_customhostname="replication-test.example.com"/>
        <uses allow="replication"/>
      </network-interface>
...
```

Search for an instance of "network-interface" that contains a "uses allow" value set to "replication" and remove the entire entry (see below) from the file.

```
<network-interface id="2" userinput_bonded=""
userinput_ifname="eth1">
        <address value="1.2.3.4" userinput_netmask="255.255.255.0"
newuserinput_value="1.2.3.4"
userinput_customhostname="replication-test.example.com"/>
        <uses allow="replication"/>
      </network-interface>
```

Add "replication" to the “uses allow” element of bond0 (see example below).

```
<?xml version="1.0" encoding="UTF-8"?>
<dpn>
  <module name="example" userinput_gateway="10.20.5.4"
userinput_domain="example.com" userinput_search="example.com"
userinput_pns="10.10.10.10" userinput_sns="10.10.10.11">
    <node type="single-node server" userinput_gateway="10.20.5.1">
      <network-interface id="1" userinput_ifname="bond0"
userinput_bonded="eth0,eth2">
       <address value="10.20.5.4" userinput_netmask="255.255.255.0"
newuserinput_value="10.20.5.4"
userinput_customhostname="example.example.com"/>
        <uses allow="internal,backup,management,replication"/>
      </network-interface>
    </node>
  </module>
</dpn>
```

Save changes.

3. Restart eth1 by typing:

   ```
   ifdown eth1; ifup eth1
   ```

4. Restart sshd service by typing:

   ```
   service sshd restart
   ```

5. Restart the Avamar subsystem by typing:

   ```
   dpnctl start
   ```

Procedure for changing the IP address on Allied Telesis internal network switches (Gen4 and Gen4S only)

IMPORTANT

Use the steps in this section only if you wish to change the IP address on an internal Avamar network switch, and only if that switch is an Allied Telesis model. For Brocade switches, see *Procedure for changing the IP address on Brocade internal network switches (Gen4S only)* (page 46). This section is only applicable to Avamar systems running the SLES operating system on Avamar Data Store Gen4 or Gen4S hardware. Also, the Avamar internal switches accept only IPv4 addresses.

This section relies on the internal Avamar network having been configured by default during installation and using subnet 192.168.255.0/24.

Switch A IP address: 192.168.255.200

Switch B IP address: 192.168.255.201

Switches use default login: manager, password: friend

To change the IP address on the internal switches:

1. Back up the network configuration files on the utility node by typing the following command all on one command line:

   ```
   tar czvf /usr/local/avamar/src/HF_35564_undo.tgz /etc/hosts
   /usr/local/avamar/var/dpnnetutil.xml
   /usr/local/avamar/var/probe.xml
   /usr/local/avamar/var/mc/server_data/prefs/mcserver.xml
   /etc/ssh/sshd_config
   ```

2. Stop the GSAN by typing the following:

   ```
   dpnctl stop
   ```

   This is important because, during this procedure, the internal network is intermittent and goes down completely at times.

3. Consult with the customer network administrator for the new subnet for internal network.

   Example: 172.16.0.0/16 (netmask 255.255.0.0), new Switch A IP address 172.16.0.200/16, new Switch B IP address 172.16.0.201/16, utility node IP address 172.16.0.1/16.

4. Install atftpd on the utility node by typing the following:

   ```
   rpm -ivh /usr/local/avamar/src/atftp-0.7.0-135.6.x86_64.rpm

   Preparing...           ########################################### [100%]
           package atftp-0.7.0-135.6.x86_64 is already installed
   ```

5. Use scp to copy the default switch configuration files (avg4_swa.cfg, avg4_swb.cfg) to /usr/local/avamar/src/ on the utility node.

6. Modify both switch configuration files, updating the IP address and netmask in the files.

   Example for switch B:

   ```
   set system name=avg4_swb
   create switch trunk=1 port=21-22 speed=1000m
   enable ip
   add ip int=vlan1 ip=172.16.0.201 netmask=255.255.0.0
   ```

   Example for switch A:

   ```
   set system name=avg4_swa
   create switch trunk=1 port=21-22 speed=1000m
   enable ip
   add ip int=vlan1 ip=172.16.0.200 netmask=255.255.0.0
   ```

7. Start tftp server on the utility node by typing the following:

   ```
   /usr/sbin/in.tftpd --bind-address=192.168.255.1 --daemon
   /usr/local/avamar/src
   ```

8. Telnet to switch B by typing the following:

   ```
   telnet 192.168.255.201
   ```

   Type login and password.

9. Rename the switch configuration file currently on the switch.

   Example for switch B:

   ```
   rename avg4_swb.cfg avg4_swb_192_168_255_200.cfg
   ```

   Best practice, as shown in the example, is to include the current IP address in the filename.

10. Load the new switch config file.

    Example for switch B:

    ```
    load meth=tftp server=192.168.255.1 destfile=avg4_swb.cfg
    file=avg4_swb.cfg
    ```

11. Reboot the switch.

    Example for switches A and B:

    ```
    restart reboot
    ```

12. Telnet to switch A by typing the following:

    ```
    telnet 192.168.255.200
    ```

    Type login and password.

13. Delete the current switch configuration file.

    Example for switch A:

    ```
    delete file avg4_swa.cfg
    ```

14. Load the new switch configuration file.

    Example for switch A:

    ```
    load meth=tftp server=192.168.255.1 destfile=avg4_swa.cfg
    file=avg4_swa.cfg
    ```

15. Reboot the switch.

    Example for switches A and B:

    ```
    restart reboot
    ```

16. Modify file /etc/sysconfig/network/ifcfg-bond1, updating the IP address and netmask.

    Example for utility node:

    ```
    STARTMODE=onboot
    BOOTPROTO=static
    IPADDR=172.16.0.1
    NETMASK=255.255.0.0
    BONDING_MASTER=yes
    BONDING_SLAVE0=eth1
    BONDING_SLAVE1=eth3
    BONDING_MODULE_OPTS="mode=active-backup miimon=100 updelay=2000
    primary=eth1"
    ```

    Depending on Avamar software version, "BONDING_MODULE_OPTS=" may be either of the following in the above example:

    ```
    BONDING_MODULE_OPTS="mode=active-backup miimon=100 updelay=2000
    primary=eth0"

    BONDING_MODULE_OPTS="primary=eth0"
    ```

17. Restart bond1 by typing the following:

    ```
    ifdown bond1; ifup bond1
    ```

18. Follow instructions in *Reconfiguring the Avamar firewall (Avamar 7.x only)* (page 61) and then return to step 19.

19. Ensure you can ping internal switches again with the new IP addresses by typing the following:

    ```
    ping 172.16.0.201

    PING 172.16.0.201 (172.16.0.201) 56(84) bytes of data.
    64 bytes from 172.16.0.201: icmp_seq=1 ttl=64 time=1.42 ms
    64 bytes from 172.16.0.201: icmp_seq=2 ttl=64 time=1.36 ms
    ping 172.16.0.200
    PING 172.16.0.200 (172.16.0.200) 56(84) bytes of data.
    64 bytes from 172.16.0.200: icmp_seq=1 ttl=64 time=6.74 ms
    64 bytes from 172.16.0.200: icmp_seq=2 ttl=64 time=2.18 ms
    ```

20. Repeat steps 16 through 19 for all nodes of the system, sequentially incrementing IP addresses.

    ```
    IPADDR=172.16.0.2, IPADDR=172.16.0.3, IPADDR=172.16.0.4 ... etc.
    ```

    To ssh to the nodes of the Avamar system, use the corresponding backup IP addresses from /etc/hosts file because internal IP addresses are not available at this point.

21. Update /usr/local/avamar/var/dpnnetutil.xml file.

    Find "‹node " sections and "‹network-interface " (shown below), and update the IP address to the corresponding internal IP, netmask of the nodes in the attributes value, userinput_netmask, newuserinput_value:

```
    <node type="storage">
    ….
      <network-interface id='2' userinput_bonded='eth1,eth3'
userinput_ifname='bond1'>
       <address value='172.16.0.1' userinput_netmask='255.255.0.0'
newuserinput_value='172.16.0.1'
userinput_customhostname='sysdev02-internal'/>
        <uses allow='internal'/>
      </network-interface>
```

22. Repeat step 21 for all "‹node " sections in the /usr/local/avamar/var/dpnnetutil.xml file.

23. Update /usr/local/avamar/var/probe.xml file.

    Change the IP addresses ("<ip value>") to the new values for both Switch A and B in the following sample section:

```
<switch id="A">
      <ip value="192.168.255.200"/>
      <login value="manager"/>
      <password value="{OBFEX1}(86QL>3(T6#<`"/>
    </switch>
<switch id="B">
      <ip value="192.168.255.201"/>
      <login value="manager"/>
      <password value="{OBFEX1}(86QL>3(T6#<`"/>
    </switch>
```

24. Ensure the mapall command works with the new internal IP addresses by typing the following:

```
ssh-agent bash
ssh-add /home/dpn/.ssh/dpnid
mapall --all+ date

Using /usr/local/avamar/var/probe.xml
(0.s) ssh  -x  admin@172.16.0.1 'date'
Thu Feb 16 10:12:28 PST 2012
(0.0) ssh  -x  admin@172.16.0.2 'date'
Warning: Permanently added '172.16.0.2' (RSA) to the list of known
hosts.
Thu Feb 16 18:12:28 UTC 2012
(0.1) ssh  -x  admin@172.16.0.3 'date'
Warning: Permanently added '172.16.0.3' (RSA) to the list of known
hosts.
Thu Feb 16 18:12:28 UTC 2012
(0.2) ssh  -x  admin@172.16.0.4 'date'
Warning: Permanently added '172.16.0.4' (RSA) to the list of known
hosts.
Thu Feb 16 18:12:29 UTC 2012
```

25. Stop the tftp server by typing the following:

```
killall in.tftpd
```

26. Edit /etc/hosts file on the utility node.

    Update the IP addresses in the rows related to internal network hostnames.

    Example:

    ```
    172.16.0.1      avamarsrv-internal
    172.16.0.2      avamarsrvd1-internal
    172.16.0.3      avamarsrvd2-internal
    172.16.0.4      avamarsrvd3-internal
    ```

27. Copy the file /etc/hosts to the same location on all nodes of the Avamar system.
28. Type the following command with no wrapping or modifications:

    ```
    for a in `nodedb print --nodes=all+ --addr --internal`; do tar -cz
    /etc/hosts | ssh root@$a 'tar --overwrite -xzf - -C /'; done;
    ```

29. Modify the value for "--server" in /usr/local/avamar/etc/usersettings.cfg:

    ```
    --server=SERVER-NAME-OR-IP-ADDR
    --vardir=/usr/local/avamar/var
    --bindir=/usr/local/avamar/bin
    --id=root
    --password=ENCRYPTPWD
    ```

    where SERVER-NAME-OR-IP-ADDR is the name or IP address of the Avamar utility node. ENCRYPTPWD is an encrypted string for the password.

30. Follow instructions in *Reconfiguring the Avamar firewall (Avamar 7.x only)* (page 61) and then return to step 31.
31. Start the GSAN.

    ```
    dpnctl start
    ```

Procedure for changing the IP address on Brocade internal network switches (Gen4S only)

IMPORTANT

Use the steps in this section only if you wish to change the IP address on an internal Avamar network switch, and only if that switch is a Brocade model. For Allied Telesis switches, see *Procedure for changing the IP address on Allied Telesis internal network switches (Gen4 and Gen4S only)* (page 41). This section is only applicable to Avamar systems running the SLES operating system on Avamar Data Store Gen4S hardware. Also, the Avamar internal switches accept only IPv4 addresses.

This section relies on the internal Avamar network having been configured by default during installation and using subnet 192.168.255.0/24.

Switch A IP address: 192.168.255.200

Switch B IP address: 192.168.255.201

Switches use default login: manager, password: ally24X7

To change the IP address on the internal switches:

1. Back up the network configuration files on the utility node by typing the following command all on one command line:

   ```
   tar czvf /usr/local/avamar/src/switch_undo.tgz /etc/hosts
   /usr/local/avamar/var/dpnnetutil.xml
   /usr/local/avamar/var/probe.xml
   /usr/local/avamar/var/mc/server_data/prefs/mcserver.xml
   /etc/sysconfig/network/ifcfg-bond1
   ```

2. Stop the Avamar subsystem (GSAN) by typing the following:

   ```
   dpnctl stop
   ```

   This is important because, during this procedure, the internal network is intermittent and goes down completely at times.

3. Consult with the customer network administrator for the new subnet for internal network.

   Example:

   Switch A: 10.10.10.200/24

   Switch B: 10.10.10.201/24

   Utility node internal IP address: 10.10.10.1

4. Telnet to the switch A (192.168.255.200), password default is ally24X7.

   ```
   telnet 192.168.255.200
   ```

   a. Enable the ability to execute commands:

      ```
      enable
      ```

   b. Enter config mode

      ```
      conf t
      ```

   c. Set the ip address by typing the following (new IP address used is an example):

      ```
      ip address 10.10.10.200/24
      ```

      If this freezes, use "CTRL" + "}" then “CTRL” + “D” to exit the connection.

5. Telnet to switch B (192.168.255.201), password default is ally24X7

   ```
   telnet 192.168.255.201
   ```

   a. Enable the ability to execute commands:

      ```
      enable
      ```

   b. Enter config mode

      ```
      conf t
      ```

   c. Set the ip address by typing the following (new IP address used is an example):

      ```
      ip address 10.10.10.201/24
      ```

      If this freezes, "CTRL" + "}" then “CTRL” + “D” to exit the connection.

6. As root, modify /etc/sysconfig/network/ifcfg-bond1 to reflect new internal IP address.

   Example for utility node:

   ```
   STARTMODE=onboot
   BOOTPROTO=static
   IPADDR=10.10.10.1
   NETMASK=255.255.255.0
   BONDING_MASTER=yes
   BONDING_SLAVE0=eth1
   BONDING_SLAVE1=eth3
   BONDING_MODULE_OPTS="primary=eth1"
   ```

   Depending on Avamar software version, "BONDING_MODULE_OPTS=" may be either of the following in the above example:

   ```
   BONDING_MODULE_OPTS="mode=active-backup miimon=100 updelay=2000
   primary=eth1"

   BONDING_MODULE_OPTS="primary=eth1"
   ```

   a. Modify "IPADDR" to reflect new internal IP address of node.

   b. Modify "NETMASK" if netmask of new internal IP address is different from previous internal IP address' netmask.

   c. Save changes.

   d. As root, restart bond1.

      ```
      ifdown bond1; ifup bond1
      ```

   e. Run "ifconfig" to ensure that "bond1" is displaying the new IP.

   f. Make sure that switches' new IP addresses are pingable .

      Example:

      Switch A: ping 10.10.10.200

      Switch B : ping 10.10.10.201

   g. Make sure that switches' old IP addresses are not pingable

      Example:

      Switch A: ping 192.168.255.200

      Switch B: ping 192.168.255.201

   h. Repeat step 9 (all parts) for all nodes of the system, sequentially incrementing IP address

      IPADDR=10.10.10.2, IPADDR=10.10.10.3, IPADDR=10.10.10.4 … etc

      To ssh to the nodes of the Avamar system, use the corresponding backup IP addresses from /etc/hosts file because internal IP addresses are not available at this point.

7. Modify the entries in probe.xml to reflect new IP address' subnet:
   a. Make a copy of the original probe.xml by typing the following all on one command line:

      ```
      cp /usr/local/avamar/var/probe.xml
      /usr/local/avamar/var/ORIG.probe.xml
      ```

   b. Search for all occurrences of switch's previous IP's subnet

      Example: if previous IP was 192.168.255.200, then search for "192.168.255"

   c. Replace each occurrence with the switch's new IP's subnet

      Example: if found "192.168.255.3" then it should be changed to "10.10.10.3" since switch's new IP subnet is "10.10.10"

   d. Modify the corresponding switch IP addresses.

      Example: If Switch A was changed to 10.10.10.200 and Switch B to 10.10.10.201, then the “ip value” elements should be changed accordingly in the probe.xml (see example below).

      ```
      <switch id="A">
        <ip value="192.168.255.200"/>
        <login value="manager"/>
        <password value="{OBFEX1}(86QL>3(T6#<`"/>
      </switch>
      <switch id="B">
        <ip value="192.168.255.201"/>
        <login value="manager"/>
        <password value="{OBFEX1}(86QL>3(T6#<`"/>
      </switch>
      ```

   e. Save changes

   f. Run "nodedb print" to ensure no tags/brackets were accidentally removed

      If "nodedb print" fails, either debug and relook at probe.xml to locate any missing tags/brackets, or start with a fresh copy by copying /usr/local/avamar/var/ORIG.probe.xml back to /usr/local/avamar/var/probe.xml and repeat step 10.

8. Modify the entries in dpnnetutil.xml to reflect new IP address' subnet:
   a. Make a copy of the original dpnnetutil.xml by typing the following all on one command line:

      ```
      cp /usr/local/avamar/var/dpnnetutil.xml
      /usr/local/avamar/var/ORIG.dpnnetutil.xml
      ```

   b. Search for all occurrences of switch's previous IP's subnet.

      Example: If previous IP was 192.168.255.200, then search for "192.168.255"

   c. Replace each occurrence with the switch's new IP's subnet.

      Example: If found "192.168.255.3" then it should be changed to "10.10.10.3" since switch's new IP subnet is "10.10.10".

d. Modify the corresponding switch's IP as follows.

Example: If Switch A was changed to 10.10.10.200 and Switch B to 10.10.10.201, then the “ip value” elements should be changed accordingly in dpnnetutil.xml (see example below).

```
<switch id="A">
  <ip value="192.168.255.200"/>
  <login value="manager"/>
  <password value="{OBFEX1}(86QL>3(T6#<`"/>
</switch>
<switch id="B">
  <ip value="192.168.255.201"/>
  <login value="manager"/>
  <password value="{OBFEX1}(86QL>3(T6#<`"/>
</switch>
```

e. Save changes.

9. Ensure the mapall command works with the new internal IP addresses by typing the following:

```
ssh-agent bash
ssh-add /home/dpn/.ssh/dpnid
mapall --all+ date

Using /usr/local/avamar/var/probe.xml
(0.s) ssh -x admin@10.10.10.1 'date'
Mon Dec  1 08:29:24 PST 2014
(0.0) ssh -x admin@10.10.10.2 'date'
Warning: Permanently added '10.10.10.2' (RSA) to the list of known
hosts.
Mon Dec  1 16:29:24 UTC 2014
(0.1) ssh -x admin@10.10.10.3 'date'
Warning: Permanently added '10.10.10.3' (RSA) to the list of known
hosts.
Mon Dec  1 16:29:24 UTC 2014
(0.2) ssh -x admin@10.10.10.4 'date'
Warning: Permanently added '10.10.10.4' (RSA) to the list of known
hosts.
Mon Dec  1 16:29:24 UTC 2014
```

10. With a text editor, edit /etc/hosts file on the utility node.

Update the IP addresses in the rows related to internal network hostnames and save changes. Example:

```
10.10.10.1   avamarsrv-internal
10.10.10.2   avamarsrvd1-internal
10.10.10.3   avamarsrvd2-internal
10.10.10.4   avamarsrvd3-internal
```

11. Copy the /etc/hosts file from the utility node to the same location on all the nodes of the Avamar system by typing the following all on one line:

```
`nodedb print --nodes=all+ --addr --internal`; do tar -cz /etc/hosts
| ssh root@$a 'tar --overwrite -xzf - -C /'; done;
```

12. Telnet to Switch A using its new IP address (example: 10.10.10.200), password default is ally24X7.

    ```
    telnet 10.10.10.200
    ```

    a. Enable the ability to execute commands:

    ```
    enable
    ```

    b. Enter config mode:

    ```
    conf t
    ```

    c. To confirm the IP address that you set above, enter:

    ```
    show ip
    ```

    Example output:

    ```
    Switch IP address: 10.10.10.200
    Subnet mask: 255.255.255.0
    Default router address: None
    TFTP server address: None
    Configuration filename: None
    Image filename: None
    IP MTU: 1500
    ```

    d. Save the configuration:

    ```
    wr m
    ```

13. Telnet to Switch B using its new IP address (ex: 10.10.10.201), password default is ally24X7.

    ```
    telnet 10.10.10.201
    ```

    a. Enable the ability to execute commands:

    ```
    enable
    ```

    b. Enter config mode:

    ```
    conf t
    ```

    c. To confirm the IP address that you set above, enter:

    ```
    show ip
    ```

    Example output:

    ```
    Switch IP address: 10.10.10.201
    Subnet mask: 255.255.255.0
    Default router address: None
    TFTP server address: None
    Configuration filename: None
    Image filename: None
    IP MTU: 1500
    ```

    d. Save the configuration:

    ```
    wr m
    ```

14. Modify the value for "--server" in /usr/local/avamar/etc/usersettings.cfg:

```
--server=SERVER-NAME-OR-IP-ADDR
--vardir=/usr/local/avamar/var
--bindir=/usr/local/avamar/bin
--id=root
--password=ENCRYPTPWD
```

    where SERVER-NAME-OR-IP-ADDR is the name or IP address of the Avamar utility node. ENCRYPTPWD is an encrypted string for the password.

15. Follow instructions in *Reconfiguring the Avamar firewall (Avamar 7.x only)* (page 61) and then return to step 16.

16. Start the Avamar subsystem (GSAN).

```
dpnctl start
```

Post Configuration Procedure

Do the following procedure for all re-IP and re-hostname scenarios except when you initially used the Change Network Settings workflow on an Avamar 7.1.1 system.

1. Open a command shell.
2. Log into the Avamar server as user admin.
3. When prompted for a password, type the admin password and press **ENTER**.
4. Load the admin OpenSSH key by entering:

```
ssh-agent bash
ssh-add ~admin/.ssh/admin_key
```

    You are prompted to type a passphrase.

5. Type the admin user account passphrase.
6. Switch to the dpn user account by entering:

```
su - dpn
```

7. When prompted for a password, type the password and press **ENTER**.
8. Type the following command:

```
asktime
```

Note: The following example **asktime** prompts and user responses are the suggested ones for most sites. However, some customer configurations might require different responses. Contact EMC Technical Support for additional information. Also, for those **asktime** prompts calling for an IP address, either IPv4 or IPv6 addresses are valid.

If you are upgrading an existing Avamar server, **asktime** detects the previous NTP settings and prompts you as follows:

```
Do you want to make use of your previous answers?
(You will be given the chance to review and to change them.) y(es),
n(o), q(uit/exit):
```

a. Type **y** and press **ENTER** to accept the previous settings as default settings for the remainder of this **asktime** session.

   The following appears in your command shell:

   ```
   Are external time servers available?
   ```

b. Type **y** and press **ENTER**.

   The following appears in your command shell:

   ```
   Do you want to use U.S. public time servers out on the wider
   Internet, such as those offered by NIST or the U.S. Naval
   Observatory?
   ```

c. Type **n** and press **ENTER**.

   The following appears in your command shell:

   ```
   Do you have access to other external time servers either on-site
   or on the wider Internet?
   ```

d. Type **y** and press **ENTER**.

   The following appears in your command shell:

   ```
   Use these NTP servers (defined in DNS as ntp.example.com)?
   ```

e. Type **y** and press **ENTER**.

   The following appears in your command shell:

   ```
   Are there other external time servers that you would like to use?
   ```

f. Type **n** and press **ENTER**.

   The following appears in your command shell:

   ```
   Please enter the name of the local time zone, using one of the
   file names under /usr/share/zoneinfo/. Examples:
    US/Alaska
    US/Central
    US/Eastern
    US/Mountain
    US/Pacific
   Note: this is a case-sensitive file name that must exist in
   /usr/share/zoneinfo.
   ```

g. Type your time zone and press **ENTER**.

   The following appears in your command shell:

   ```
   Do you want to proceed with installation of these files on the
   selected node?
   ```

h. Type **y** and press **ENTER**.

   The following appears in your command shell:

   ```
   Is this approximately correct (within a minute or two)?
   ```

i. Type **y** and press **ENTER**.

   Older versions of **asktime** end at this point and return you to the command prompt. Newer versions of **asktime** continue to run.

9. Do one of the following:

Table 9 asktime steps

If	Do this
asktime ends and returns you to the command prompt.	Verify proper NTP configuration by entering: mapall --all '/usr/sbin/ntpq -pn' Current date and time are returned for each node in the server. If configuring a multi-node server, all times must be within one second of one another.
asktime continues to run.	The following appears in your command shell: Do you want to wait and watch for time synchronization? Type **y** and press **ENTER**. The following appears in your command shell: We appear to have time synchronization. Do you want to see results? Type **y** and press **ENTER**. NTP results appear in your command shell. If configuring a multi-node server, all times must be within one second of one another.

IMPORTANT

The Microsoft Windows W32Time service (the NTP service on domain controllers) does not meet the multi-node system requirement that all times must be within one second of one another. For that reason, W32Time cannot be used for NTP purposes. For more information, see the Microsoft knowledgebase article in the following location:
`http://support2.microsoft.com/kb/939322`

10. Switch back to admin user account by entering:

```
exit
```

11. Determine if you are using the Avamar lockbox tool:

```
ls -l /usr/local/avamar/var/avlock*
```

If this command returns a list of files such as:

```
-rw-r--r--  1 admin admin 2986 Jun 27 23:07
/usr/local/avamar/var/avlockbox.clb
-rw-r--r--  1 admin admin 2398 Jun 27 23:07
/usr/local/avamar/var/avlockbox.clb.bak
-rw-r--r--  1 admin admin    1 Jun 27 23:07
/usr/local/avamar/var/avlockbox.clb.bak.FCD
-rw-r--r--  1 admin admin    4 Jun 27 23:07
/usr/local/avamar/var/avlockbox.clb.FCD
```

then the lockbox is in use and you need to update your lockbox passphrase. Proceed to step 12.

IMPORTANT

If the command does not return any files, then the lockbox is not in use. Proceed to step 14.

12. Switch user to root by entering:

    ```
    su -
    ```

13. To update your lockbox passphrase:

    - If you know the passphrase of the lockbox, go to step (f).
    - If you do not know the passphrase for the lockbox, you must either locate the passphrase or delete and recreate the lockbox:

    a. Locate the password by searching for the text string `lockbox_password` in the file /usr/local/avamar/var/install.conf or the file /usr/local/avamar/var/upgrade.conf. If you locate the passphrase, go to step (f).

       If you cannot locate the passphrase, you must delete and recreate the lockbox:

    b. Change to the Avamar var directory:

       ```
       cd /usr/local/avamar/var
       ```

    c. Move all lockbox-related files to a temporary directory:

       ```
       mv avlockbox* /tmp
       ```

    d. Create a new lockbox by typing the following command:

       ```
       avlockboxcfg create --newpassphrase=NEW-LOCKBOX-PASSPHRASE
       ```

       The new passphrase must meet the following complexity requirements:

       – Contains 8 or more characters
       – Contains at least one numeric character
       – Contains at least one uppercase and one lowercase character
       – Contains at least one non-alphanumeric character (for example, !@#$%, and so forth)

    e. Store user credentials on the utility node:

       ```
       avlockboxcfg setcredentials
       ```

    f. Type the following command:

       ```
       avlockboxcfg rekey --passphrase=LOCKBOX-PASSPHRASE
       ```

       where **`LOCKBOX-PASSPHRASE`** is either the existing passphrase or the passphrase created in step (d).

14. Switch to the admin user account by entering:

    ```
    exit
    ```

    The admin OpenSSH key should still be loaded.

15. Restart the Avamar server by entering:

    ```
    restart.dpn
    ```

16. Modify the systemname parameter after starting the server by entering:

    ```
    avmaint config --avamaronly systemname="NEWNAME"
    ```

    where **NEWNAME** is the fully qualified hostname as defined in corporate DNS or IP address of the Avamar server.

    Confirm the new systemname took effect by reviewing the output of the following command:

    ```
    status.dpn
    ```

17. Configure the Avamar server login manager by entering:

    ```
    avmaint config lmaddr=SERVER-IP-ADDR --avamaronly
    ```

    where SERVER-IP-ADDR is the Avamar utility node or server IP address for multi-node and single-node servers, respectively. If an internal network has been configured, then SERVER-IP-ADDR is the internal IP address of the utility node.

    For single node Avamar servers, the SERVER-IP-ADDR is always 127.0.0.1 no matter whether the server is configured for IPv4 or IPv6 use.

IMPORTANT

If the Avamar system is configured as dual stack (IPv4 and IPv6 addressing), SERVER-IP-ADDR must be the IPv4 address.

18. Verify the new login manager address by entering:

    ```
    avmaint config --avamaronly | grep lmaddr
    ```

19. Initiate an administrator server rename by entering:

    ```
    mcserver.sh --restore --restoretype=rename-system --norestart
    ```

    The following appears in your command shell:

    ```
    === BEGIN === check.mcs (prerestore)
    check.mcs                          passed
    === PASS === check.mcs PASSED OVERALL (prerestore)
    --restore will modify your Administrator Server database and
    preferences.
    Do you want to proceed with the restore Y/N? [Y]:
    ```

    After entering Y, the following appears in your command shell:

    ```
    Enter the Avamar Server IP address or fully qualified domain name to
    restore from (i.e. dpn.your_company.com):
    ```

    This step asks for the most recent hostname of the Avamar server.

    This step also updates the system address in the mcserver.xml file.

20. Type the Avamar server hostname or IP address and press **ENTER**.

    The following appears in your command shell:

    ```
    Enter the Avamar Server IP port to restore from [27000]:
    ```

21. Press **ENTER** to accept the default Avamar Server IP port.

    The following appears in your command shell:

    ```
    Enter password for MCUser:
    ```

Note: In the steps that follow, replace `NODENAME` with the name of the Avamar server (for single-node servers) or the utility node (for multi-node servers).

22. Type the correct MCUser account password (MCUser1 on new systems) for the original Avamar server and press **ENTER**.

    The following appears in your command shell:

    ```
    Select the Avamar server IP address or fully qualified host name to
    be used by backup clients:
    1) NODENAME.example.com
    2) Enter another address

    Enter your selection (1-2):
    ```

Note: Steps 23 through 24 ask for the hostname and IP address of the new Avamar server.

23. Do one of the following:

Table 10 Hostname choices

If	Do this
The Avamar server will use a specified hostname.	Type the number of the specified hostname (for example, **1**) and press **ENTER**.
The Avamar server will use a hostname that is not specified as a pre-defined choice.	Type the number that corresponds to the option "Enter another address" (for example, **2**) and press **ENTER**.

    If you selected option 1, the following appears in your command shell:

    ```
    'NODENAME.example.com' resolvable and pingable from MCS node.

    Enter the Avamar server IP address or fully qualified host name to
    be used for Avamar Administrator server to Avamar server
    communication. Use 'NODENAME.local.example.com' or its IP address to
    rely on internal Avamar server name resolution and network or use
    'NODENAME.example.com', its IP address, or another name to rely on
    external name resolution and network [NODENAME.local.example.com]:
    ```

24. If you are configuring an Avamar system that uses an internal network, type the utility node's internal network hostname and press **ENTER**. Otherwise, type the Avamar server IP address or fully qualified hostname to be used for Avamar Administrator server and press **ENTER**.

    The following appears in your command shell:

    ```
    'NODENAME.example.com' resolvable and pingable from MCS node.

    Enter the IP port used to communicate with the Avamar Server [27000]:
    ```

25. Press **ENTER** to accept the default Avamar Server IP port.

    The following appears in your command shell:

    ```
    Using port '27000'.

    Select the RMI machine IP address or fully qualified host name to be
    used by management clients (Avamar Administrator and MCCLI):
    1) Default value.
    2) Enter another address

    Enter your selection (1-2): 1
    ```

26. Press **ENTER** to accept the default IP address.

    The following appears in your command shell:

    ```
    'Default value.' resolvable and pingable from MCS node.

    Enter the Avamar server accounting system root user password for
    Avamar server NODENAME.example.com:
    ```

27. Type the Avamar server accounting system root user password (8RttoTriz on new systems) and press **ENTER**.

    The following appears in your command shell:

    ```
    mcserver.xml file updated.
    Performing restore administrative tasks...

    Enter the Avamar server accounting system MCUser user password for
    Avamar server NODENAME.example.com:
    ```

28. Type the Avamar server accounting system MCUser user password (ask the system administrator for this) and press **ENTER**.

    Output similar to the following appears in your command shell:

    ```
    mcserver.xml file updated.
    encrypting passwords for (MCUSERAP|rootAP) in file
    /usr/local/avamar/var/mc/server_data/prefs/mcserver.xml ...
    encrypting preferences (MCUSERAP|rootAP) for mask
    /usr/local/avamar/var/mc/server_data/prefs ...
    see MCCipher log for details:
    /usr/local/avamar/var/mc/server_log/mccipher.log.0
    Performing restore administrative tasks...
    restore : Closing DB Connections
    Restore administrative tasks complete.
    Timestamp of flush restored: 2014-12-05 17:27:40 PST

    Encrypting all registered passwords...
    see MCCipher log for details:
    /usr/local/avamar/var/mc/server_log/mccipher.log.0
    ```

29. Restart the administrator server by entering:

    ```
    cd ~
    mcserver.sh --start
    ```

30. Wait for administrator server startup to complete.

31. For Gen4/Gen4S systems, if optional management over a separate network is configured, reset the Remote Method Invocation (RMI) address:

    ```
    avsetup_mcs --noprompt --prefs --rmi_address=ADDRESS
    ```

    where ADDRESS is the fully-qualified domain name or IP address of the alternate administration.

32. Switch user to root by entering:

    ```
    su -
    ```

33. Reinitialize the Avamar Administrator web start function by entering:

    ```
    avsetup_webstart
    ```

34. Reinitialize the Avamar Administrator command line interface tool by entering:

    ```
    avsetup_mccli
    ```

35. Press **ENTER** each time the user is prompted for a response.

36. For Avamar 4.x or 5.0.x, remove the following files by entering:

    ```
    rm /root/.avamardata/var/mc/cli_data/prefs/mcclimcs.xml
    rm /data01/home/dpn/.avamardata/var/mc/cli_data/prefs/mcclimcs.xml
    rm
    /data01/home/admin/.avamardata/var/mc/cli_data/prefs/mcclimcs.xml
    ```

37. Switch to the admin user account by entering:

    ```
    exit
    ```

    The admin OpenSSH key should still be loaded.

38. Type the following command:

    ```
    emserver.sh --renameserver --uselocalmcs
    ```

    Respond to the prompts.

39. Type the following command:

    ```
    dpnctl start ems
    ```

40. If administering a single-node server, re-enable the unattended shutdown/restart feature by entering:

    ```
    dpnctl enable
    ```

41. All clients must be reactivated with the new Avamar server name.

    If possible, connect to the network and run the Avamar Administrator in order to locate and verify the active (or not active) status of these clients. If they show active, de-activate them.

42. Start desktop/laptop by entering:

    ```
    dpnctl start dtlt
    ```

43. Start the scheduler by entering:

    ```
    dpnctl start sched
    ```

44. Resume the maintenance cron jobs by entering:

    ```
    dpnctl start maint
    ```

45. Create and validate a checkpoint.
46. For Avamar 7.x systems:
    - That has replication configured, go to *Replication-related post configuration procedure (Avamar 7.x only)* (page 60).
    - That might have firewall hardening configured, go to *Reconfiguring the Avamar firewall (Avamar 7.x only)* (page 61).

    If both of these bullets apply, you should do one and then return to do the second.

Replication-related post configuration procedure (Avamar 7.x only)

IMPORTANT

Do the following procedure on all Avamar systems running Avamar 7.x for all re-IP and re-hostname scenarios, and for systems using either cron-based or policy-based replication.

This procedure ensures replication continues to work properly in a policy-based environment. It also ensures that the policy-based environment of the Avamar system is re-enabled after an IP/hostname change should the Avamar system be changed from cron-based to policy-based at a later time. See “Configuring policy-based replication” in the Replication chapter of the EMC Avamar Administration Guide for additional information.

To re-enable replication:

1. In Avamar Administrator, click the **Administration** launcher button.

   The **Administration** window appears.
2. Click the **Account Management** tab.
3. Select the MC_SYSTEM domain in the top left panel.
4. Select the replication client in the bottom left panel.
5. Select **Actions › Account Management › Edit Client**.

   The **Edit Client** dialog box appears.
6. Change the client name to the new fully-qualified domain name.
7. Click **OK**.
8. Click the **Policy** launcher button at bottom left.

   The **Policy** window appears.
9. Click the **Clients** tab.
10. Select the MC_SYSTEM domain in the left panel.
11. Select the replication client in the right pane.

12. Click the **Edit** button.

    The **Edit Client** dialog box appears.

13. Clear the **Activated** checkbox.
14. In the **Paging** section, select Manual.
15. Change the **Address (IP or hostname)** field to the new replication client name.
16. Click **OK**.
17. In a command shell, as root user, type the following command:

    ```
    service avagent register NEWHOSTNAME /MC_SYSTEM
    ```

Reconfiguring the Avamar firewall (Avamar 7.x only)

This procedure applies only to Avamar systems in which firewall hardening was implemented and only when you are following manual re-IP instructions in this technical note (this section is not required if you used the Change Network Settings workflow).

Firewall hardening was introduced as an optional feature in Avamar 7.0, and it was made mandatory for Avamar 7.1 installations. Use the following procedure as appropriate for the version of Avamar software on the system whose IP addressing has been reconfigured.

1. Open a command shell on the utility node or single node Avamar server.
2. Log into the server as user admin.
3. When prompted for a password, type the admin password and press **ENTER**.
4. Load the admin OpenSSH key by entering:

   ```
   ssh-agent bash
   ssh-add ~admin/.ssh/admin_key
   ```

   You are prompted to type a passphrase.

5. Type the admin user account passphrase.
6. Switch to the root user account by entering:

   ```
   su -
   ```

7. On Avamar 7.0.x systems (otherwise, continue with step 8), enter the following:

   ```
   rpm –q avfwb
   ```

   If avfwb is installed, continue with step 8 (otherwise, skip to *Configuring the Avamar Downloader Service* (page 63)):

8. Update the firewall IP tables on each node (utility and storage, or single node server) by entering the following:

   ```
   sh /usr/local/avamar/lib/admin/security/sec_create_nodeips.sh
   ```

   If the script fails to operate, edit the /etc/firewall-ips file manually in a text editor. The following is an example of the file contents.

   ```
   UTILITY=10.25.92.30
   OTHER_IPS="10.25.92.31 10.25.92.32 10.25.92.33"
   INTERNAL_IPS="192.168.255.1 192.168.255.2 192.168.255.3
   192.168.255.4"
   ```

   Use the following information as a guide to determine which IP addresses need to be changed:

   - UTILITY - If you changed the utility node IP address
   - OTHER_IPS - External IP addresses of storage nodes on a multi-node server
   - INTERNAL_IPS - If you changed the IP addresses of the internal network switches

   After you are done, save the changes.

9. Using a Unix editor such as vi, open /etc/ssh/sshd_config.
10. Look for a line of text (near the bottom) similar to the following example:

    For single node Avamar servers:

    ```
    Match Address ::1,127.0.0.1,10.241.238.240
    ```

    For multi-node Avamar servers:

    ```
    Match Address ::1,127.0.0.1,10.241.238.240,192.44.16.1
    ```

11. Replace the IP address (10.241.238.240 in the example) with the server's new IP address.
12. Save the file and close the editor.
13. Load the admin OpenSSH key by entering:

    ```
    ssh-agent bash
    ssh-add ~admin/.ssh/dpnid
    ```

    You are prompted to type a passphrase.

14. Type the dpn user account passphrase.
15. Copy the sshd_config file to all data nodes by entering the following three commands:

    ```
    mapall copy /etc/ssh/sshd_config

    mapall --noerror --user=root cp /home/admin/etc/ssh/sshd_config
    /etc/ssh/sshd_config

    mapall --noerror --user=root rm –rf etc
    ```

16. Restart the sshd service on all nodes by entering the following:

    ```
    mapall --all --noerror --user=root service sshd restart
    ```

Configuring the Avamar Downloader Service

For Avamar servers running version 6.0, you must configure the Avamar Downloader Service to connect to the renamed Avamar server.

On the host server of the Avamar Downloader Service:

1. Click the Avamar Downloader Service task tray icon.

   The Welcome! page appears.

2. Click **Next**.

   The EMC FTP Credentials page appears.

IMPORTANT

Do not make any changes to the FTP username and password credentials. If you make changes to the FTP username or password, you must reinstall the Avamar Downloader Service to recover these credentials.

3. Accept the default FTP username and password, and then click **Next**.

   The Avamar Systems page appears.

4. Click **Add**.

   The Avamar Downloader Service - Add Known System dialog box appears.

5. Complete the settings as described in the following table:

Table 11 Setting descriptions

Setting	Description
Hostname	Type the IP address or hostname.
Username	Type `root`.
Password	Type the root password.
Confirm password	Retype the root password.

6. Click **OK**.

   Click **Next**.

   Click **Finish**.

Other Considerations

Mail server

If the hostname of your mail server has changed, you need to reconfigure ConnectEMC and Email Home to use the new hostname of the mail server. Refer to the EMC Avamar Administration Guide for procedures to change the hostname of the mail server for ConnectEMC and Email Home.

Desktop/Laptop and Client Manager authentication

Both Avamar Desktop/Laptop and Avamar Client Manager require configuration to authenticate users through LDAP-compliant directory servers. LDAP authentication configuration is simplified by using avldap, an LDAP configuration tool provided with Avamar Desktop/Laptop. Refer to the EMC Avamar Administration Guide for details about using the avldap tool.

Data Domain integration

When using Avamar integration with Data Domain, an Avamar server is a client of a Data Domain server, because it runs programs such as ddrmaint that use DDBOOST to connect to the Data Domain server. For a Data Domain server to permit DDBOOST connections from a client, the client must have NFS access configured to the share /backup/ost.

Client access for DDBOOST is configured in the Data Domain Command Line Interface, via ssh, or from the Data Domain Enterprise Manager, via HTTP. A resolvable client name or IP address must be provided. Best practice is to use the fully-qualified domain name for the client, or an asterisk can be used to allow any client to access the /backup/ost path. If the name of the Avamar server has changed and the previous name is configured as a client on the Data Domain server, you will need to change this information using either the Data Domain Command Line Interface or the Data Domain Enterprise Manager. The Data Domain documentation provides details.

Adding, updating VLANs or NAT on an existing Avamar system

IMPORTANT

The instructions in this section are for only SLES systems using only IPv4 addresses.

Adding, updating VLANS

The purpose of this section is to provide instructions for configuring VLAN traffic for backup networks for Avamar Data Store Gen4/Gen4S advanced network configurations.

- An IP Address must be associated with the default device (for example bond0).
- VLANs can only be associated with the backup interface (bond0).
- Each node in an Avamar grid must have an interface defined per VLAN ID each with a unique IP address and hostname.

Procedure

First, you must prepare the server by shutting down the Avamar software as follows:

1. As user admin, open a command shell.
2. Log into the server as user admin.
3. When prompted for a password, type the admin password and press ENTER.

4. Load the admin OpenSSH key by entering:

   ```
   ssh-agent bash
   ssh-add ~admin/.ssh/admin_key
   ```

   You are prompted to type a passphrase.

5. Type the admin user account passphrase.

6. If administering a single-node server, turn off the unattended shutdown/restart feature by entering:

   ```
   dpnctl disable
   ```

7. Shutdown the Avamar server by entering:

   ```
   dpnctl stop
   ```

Next, on each storage node, add VLANs as follows:

1. As user root, using a Unix text editor, create an ifcfg file with the following name:

   /etc/sysconfig/network/ifcfg-bond0.VLAN_ID

   where VLAN_ID is the ID of the VLAN, for example:

   /etc/sysconfig/network/ifcfg-bond0.123

   /etc/sysconfig/network/ifcfg-bond0.222

   /etc/sysconfig/network/ifcfg-bond0.333

2. Create a file with the following format (the following is an example of the content of the ifcfg-bond0.123 file):

   ```
   STARTMODE=onboot
   BOOTPROTO=static
   IPADDR=IP-ADDR
   NETMASK=NETMASK-IP
   ETHERDEVICE=bond0
   VLAN_ID=123
   ```

   where IP-ADDR is the IP address of bond0 for this particular VLAN, and NETMASK-IP is the netmask associated with the IP address.

3. If adding more than one VLAN interface, repeat steps 1 and 2 adjusting the VLAN_ID, IPADDR, and NETMASK fields accordingly.

4. Restart network services:

   ```
   service network restart
   ```

Finally, configure the utility node.

1. As user root, using a Unix text editor, create an ifcfg file with the following name:

   /etc/sysconfig/network/ifcfg-bond0.VLAN_ID

   where VLAN_ID is the ID of the VLAN, for example:

   /etc/sysconfig/network/ifcfg-bond0.123

   /etc/sysconfig/network/ifcfg-bond0.222

   /etc/sysconfig/network/ifcfg-bond0.333

2. Create a file with the following format, the following is example content of the ifcfg-bond0.123 file:

   ```
   STARTMODE=onboot
   BOOTPROTO=static
   IPADDR=IP-ADDR
   NETMASK=NETMASK-IP
   ETHERDEVICE=bond0
   VLAN_ID=123
   ```

   where IP-ADDR is the IP address of bond0 for this particular VLAN, and NETMASK-IP is the netmask associated with the IP address.

3. If adding more than one VLAN interface, repeat steps 1 and 2 adjusting the VLAN_ID, IPADDR, and NETMASK fields accordingly.

4. Add an entry to the /etc/hosts file for any new VLAN interfaces defined:

   ```
   127.0.0.1 localhost.localdomain localhost
   10.0.44.5 avamar1.example.com avamar1 #utility
   10.0.44.6 avamar2.example.com avamar2 #data
   10.0.44.7 avamar3.example.com avamar3 #data
   VLAN-IP-ADDR1  avamar1-4000.example.com avamar1-4000 # utility VLAN
   4000
   VLAN-IP-ADDR2  avamar2-4000.example.com avamar2-4000 # data VLAN
   4000
   VLAN-IP-ADDR3  avamar3-4000.example.com avamar3-4000 # data VLAN
   4000
   10.0.55.10 avamar6.example.com avamar6 #spare
   ```

   where VLAN-IP-ADDR 1-3 is the unique IP address associated with the specific data node for the VLAN.

Note: Depending on the environment, hostname schemes may differ. The above example uses the suffix "4000" to indicate a hostname associated with a VLAN with ID 4000.

IMPORTANT

When configuring multi-node systems, the /etc/hosts file should be identical on all nodes in the system.

5. Load ssh keys and copy the modified /etc/hosts file to all other nodes in the Avamar server.

   For instance, use the following command:

   ```
   scp /etc/hosts root@STORAGENODE1:/etc/hosts
   ```

   where STORAGENODE1 is the hostname for each storage node.

6. Obtain the current highest network interface ID by running the following:

   ```
   nodedb print
   ```

   The output should look similar to:

   ```
   <dpn>
     <module name="AVAMAR">
       <node type="single-node server">
         <network-interface id="1">
           <address value="10.6.240.191"/>
           <uses allow="replication,management,backup,internal"/>
         </network-interface>
       </node>
    </module>
   </dpn>
   ```

Note: On a multi-node grid, the output will contain entries for each node and the defined interfaces therein.

7. For each node, add a new interface:

   ```
   nodedb add if --addr=IP_ADDRESS --node=NODE_ID
   --nwgrp=INTERFACE_ID --allow=backup
   ```

   where:

Table 12 Token descriptions

Token	Description
IP_ADDRESS	Is the IP address associated with the VLAN interface being added.
NODE_ID	Is the physical node ID associated with the node being added.
INTERFACE_ID	This is the highest network interface ID you found in step 6 plus one (highest ID number + 1). In the example above, the interface ID would be 2. Use the same interface ID for each node entry.

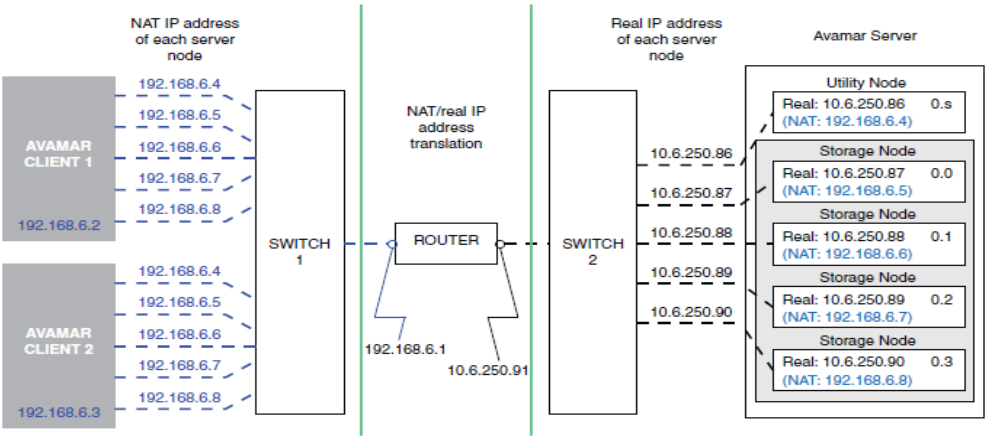
Configuring Avamar to use NAT

This topic applies only to Avamar configurations that use Network Address Translation (NAT).

Starting with version 5.0, some or all Avamar clients can access Avamar storage nodes by using a set of addresses that undergo NAT. To make NAT information known to the Avamar server, the probe.xml file must contain nat-address elements for storage nodes. After a client makes initial contact with the Avamar server's utility node, the Avamar server

provides a set of routable addresses for the storage nodes to each client. In the absence of a nat-address element, a client uses a pre-configured “real” (untranslated) network-interface address.

The following figure illustrates an example of a 1x4 multi-node server configuration in which Avamar uses NAT.



The following instructions explain how to set up the probe.xml file (node resource database) to enable the Avamar server to use NAT. These instructions assume that each Avamar node has a unique address (from the Avamar clients’ perspective), and that you configure a router on the network to apply transparent one-to-one network address translation. You can also use these instructions to enable NAT for use in a single-node server configuration.

Note: Setting up the hardware for NAT is beyond the scope of this guide.

Procedure

To configure Avamar to use NAT:

1. Add NAT addresses to probe.xml with the nodedb command.

   An example command using nodedb:

   ```
   nodedb update if --addr=10.6.250.87 --new-nat=INITIAL=TARGET
   ```

   where INITIAL is the NAT utility node IP address and TARGET is the NAT storage node IP address.

   - For multi-node Avamar servers, add an entry for each storage node (for instance, for a 1x4 server, you must add four entries to probe.xml).
   - For single node Avamar servers, add only one entry. Also, INITIAL and TARGET IP addresses will be the same.

   The nodedb command updates an existing network interface element in the probe.xml file with NAT information that corresponds to the example diagram shown on the previous page.

2. If the Avamar storage subsystem is currently stopped, restart it by typing:

   ```
   dpnctl start gsan
   ```

3. If the Avamar storage subsystem is currently running, re-read the probe.xml file by typing:

   ```
   avmaint networkconfig /usr/local/avamar/var/probe.xml --avamaronly
   ```

4. Register clients by using the avregister (UNIX) or avregister.bat (Windows) command, or by using Avamar Administrator.

   - An example command to register a UNIX client using avregister:

     ```
     /usr/local/avamar/bin/avregister
     ```

     Respond to the interactive prompts displayed by avregister.

     To determine whether NAT is in use, the client and Avamar server must have a network connection.

   - "Client registration" in the *Avamar Administration Guide* provides more information about registering clients.

Resolving NAT connection and configuration problems

The following table provides solutions for common NAT connection and configuration problems.

Table 13 NAT problem resolutions

Problem	Solution
Avamar server terminates with a FATAL ERROR message.	Ensure that the probe.xml file: Exists in the /usr/local/avamar/var/ directory. Is a valid XML file and adheres to the node resource database format. Lists NAT IP addresses correctly. Use the nodedb print --say command to view the contents of probe.xml. The --say option shows the path and name of the current node resource database.
Server/client connection fails.	Use network diagnostic tools such as ping, traceroute, tracert, or iperf to verify network connectivity.

Configuring an Avamar System as Dual Stack

This section describes how to configure the following scenarios:

- Adding an IPv6 stack to an existing non-VLAN/NAT-enabled IPv4 configuration.
- Adding an IPv6 stack to an existing VLAN/NAT-enabled IPv4 configuration.

  This would be a situation in which a customer is transitioning to IPv6 due to IPv4 address space crunch that is currently being ameliorated with VLAN or NAT usage. To integrate an Avamar server into an environment that already uses VLANs or NAT, you must configure it as a dual stack system post-installation.

For Avamar 7.x software running on Gen4/Gen4S hardware, there are several configuration options. Networking files that require modification will vary depending on the specific configuration. The networking files are located in /etc/sysconfig/network/. Configuration options include:

- Single node or AVE systems
- Multinode systems
  - eth0 and eth2 configured as slaves to bond0

    Reserved for backup.
  - eth1 and eth3 configured as slaves to bond1

    For internal traffic.
  - eth4 and eth6 configured as slaves to bond2

    Reserved for optional replication on the utility node.
  - eth5 and eth7 configured as slaves to bond3

    Reserved for optional management on utility node.
- VLAN configuration (for IPv4 stack only)
  - bond0.VLAN-IDs

    Optional multiple-tagged backup networks.

Procedure

1. Shut down the Avamar software.
   a. As user admin, open a command shell.
   b. Log into the server as user admin.
   c. When prompted for a password, type the admin password and press ENTER.
   d. Load the admin OpenSSH key by entering:

      ```
      ssh-agent bash
      ssh-add ~admin/.ssh/admin_key
      ```

      You are prompted to type a passphrase.
   e. Type the admin user account passphrase.

f. If administering a single-node server, turn off the unattended shutdown/restart feature by entering:

```
dpnctl disable
```

g. Shutdown the Avamar server by entering:

```
dpnctl stop
```

2. Switch user to root by entering:

```
su -
```

3. Using a Unix text editor, edit the /etc/resolv.conf, /etc/hosts, /etc/sysconfig/network/routes, and /etc/sysconfig/network/ifcfg-bond0 network files.

IMPORTANT

When configuring multi-node systems,
/etc/hosts and /etc/resolv.conf should be identical on all nodes in the system.

a. /etc/resolv.conf

```
domain        local.example.com
search        local.example.com company.com
nameserver    NAMESERVERIP
```

where NAMESERVERIP is a valid IPv6 address. Edit or add a nameserver address. Set the nameserver as itself if there is no other nameserver.

Note: The above example only shows a single nameserver. The /resolv.conf file may contain additional nameservers. For example:

```
domain     local.example.com
search     local.example.com company.com
nameserver 10.6.254.4
nameserver 10.6.254.5
nameserver 2620:0:170:58e::4
```

The first two are IPv4 addresses, the last one is IPv6.

IMPORTANT

Regarding step 3(b) and 3(c) below, when changing IP addresses in dual-stack configurations, EMC strongly recommends you also provide a unique hostname for each changed IPv6 IP address. This can be accomplished by creating unique hostnames or subdomains like the following examples:

hostname.domain.local (IPv4)
hostname6.domain.local (IPv6)

hostname.domain.local (IPv4)
hostname.ipv6.domain.local (IPv6)

b. /etc/hosts on single-node servers

```
SINGLENODEIPV4 avamar1v4.example.com avamar1v4 #single node
server
SINGLENODEIPV6 avamar1v6.example.com avamar1v6 #single node
server
127.0.0.1 localhost
# special IPv6 addresses
::1 localhost ipv6-localhost ipv6-loopback
fe00::0 ipv6-localnet
ff00::0 ipv6-mcastprefix
ff02::1 ipv6-allnodes
ff02::2 ipv6-allrouters
ff02::3 ipv6-allhosts
```

where each token IP is either a valid IPv4 or IPv6 address for each component.

c. /etc/hosts on multi-node servers

```
UTILITYIPV4 avamar1v4.example.com avamar1v4 #utility
DATA1IPV4 avamar2v4.example.com avamar2v4 #data
DATA2IPV4 avamar3v4.example.com avamar3v4 #data
DATA3IPV4 avamar4v4.example.com avamar4v4 #data
DATA4IPV4 avamar5v4.example.com avamar5v4 #data
SPAREIPV4 avamar6v4.example.com avamar6v4 #spare
UTILITYIPV6 avamar1v6.example.com avamar1v6 #utility
DATA1IPV6 avamar2v6.example.com avamar2v6 #data
DATA2IPV6 avamar3v6.example.com avamar3v6 #data
DATA3IPV6 avamar4v6.example.com avamar4v6 #data
DATA4IPV6 avamar5v6.example.com avamar5v6 #data
SPAREIPV6 avamar6v6.example.com avamar6v6 #spare
UTILITYINTIP avamar1-internal #utility internal
DATA1INTIP avamar2-internal #data internal
DATA2INTIP avamar3-internal #data internal
DATA3INTIP avamar4-internal #data internal
DATA4INTIP avamar5-internal #data internal
SPAREINTIP avamar6-internal #spare internal
127.0.0.1 localhost
# special IPv6 addresses
::1 localhost ipv6-localhost ipv6-loopback
fe00::0 ipv6-localnet
ff00::0 ipv6-mcastprefix
ff02::1 ipv6-allnodes
ff02::2 ipv6-allrouters
ff02::3 ipv6-allhosts
```

where each token IP is either a valid IPv4 or IPv6 address for each component.

Note: Due to space constraints on this page, lines in the above example wrap to the next line. They should not wrap in the actual file.

Note: If the hostname of the server is changing, you must preserve the naming scheme of HOST_NAME-internal, where HOST_NAME is the new hostname.

d. /etc/sysconfig/network/routes

```
cat /etc/sysconfig/network/routes
default GATEWAY-IP - -
```

where `GATEWAY-IP` is a valid IPv4 address for the default gateway associated with the primary bond or network interface.

Note: The example above calls for an IPv4 GATEWAY-IP because that is the most likely common case. Best practice is: do not change the default gateway protocol; that is, if it is an IPv4 address before conversion, leave it as as the IPv4 address. This makes IPv4 the "primary" protocol.

Note: The above example only shows a default gateway. Besides the mandatory default, the /routes file may contain additional static destination network routes. For example:

2000::13 2620:0:170:588::1 - -
10.12.12.0/24 10.6.98.1 - -

Note: The contents of /etc/sysconfig/networks/ifcfg-bond0 (next step) contains server-specific information.

e. /etc/sysconfig/network/ifcfg-bond0

```
cat /etc/sysconfig/network/ifcfg-bond0
STARTMODE='auto'
BOOTPROTO='static'
IPADDR='192.168.222.44/24'
IPADDR_0='2620:0:170:580::3:2/64'
```

The IPADDR and IPADDR_0 fields are IPv4 and IPv6 addresses with subnet suffixes (/24 and /64, respectively). If the current ifcfg-bond0 file displays netmask in NETMASK=xxx.xxx.xxx.xxx format, convert it to /24 and /64 format. See examples above.

4. Repeat steps 1a through 1e and steps 2 and 3 for each node on the grid.
5. Using a Unix text editor, edit the probe.xml file.

   The following is an example of the probe.xml before changing to dual stack:

```
<node type="utility" userinput_gateway="10.25.113.1">
   <network-interface id="0" userinput_bonded="eth0,eth2"
userinput_ifname="bond0">
      <address value="123.25.113.184"
userinput_netmask="255.255.255.0"
newuserinput_value="123.25.113.184"
userinput_customhostname="a4dpn62.default"/>
      <uses allow="replication,management,backup"/>
      </network-interface>
   <network-interface id="1" userinput_bonded="eth1,eth3"
userinput_ifname="bond1">
      <address value="192.168.255.1"
userinput_netmask="255.255.255.0"
newuserinput_value="192.168.255.1"
userinput_customhostname="a4dpn62-internal"/>
      <uses allow="internal"/>
   </network-interface>
</node>
   <node type="storage" userinput_gateway="10.25.113.1">
   <network-interface id="0" userinput_bonded="eth0,eth2"
userinput_ifname="bond0">
     <address value="123.25.113.185"
userinput_netmask="255.255.255.0"
newuserinput_value="123.25.113.185"
userinput_customhostname="a4dpn62d1.default"/>
     <uses allow="replication,backup"/>
```

```
   </network-interface>
   <network-interface id="1" userinput_bonded="eth1,eth3"
userinput_ifname="bond1">
     <address value="192.168.255.2"
userinput_netmask="255.255.255.0"
newuserinput_value="192.168.255.2"
userinput_customhostname="a4dpn62d1-internal"/>
     <uses allow="internal"/>
   </network-interface>
</node>
```

The following is an example of the probe.xml after adding an IPv6 stack, thus creating a dual stack:

```
<node type="utility" userinput_gateway="10.25.113.1">
  <network-interface id="0" userinput_bonded="eth0,eth2"
userinput_ifname="bond0">
     <address value="123.25.113.184"
userinput_netmask="255.255.255.0"
newuserinput_value="123.25.113.184"
userinput_customhostname="a4dpn62.default"/>
     <address value="aa:bb:cc:123" userinput_netmask="/64"
newuserinput_value="aa:bb:cc:123"
userinput_customhostname="a4dpn62.v6.default"/>
     <uses allow="replication,management,backup"/>
  </network-interface>
  <network-interface id="1" userinput_bonded="eth1,eth3"
userinput_ifname="bond1">
   <address value="192.168.255.1" userinput_netmask="255.255.255.0"
newuserinput_value="192.168.255.1"
userinput_customhostname="a4dpn62-internal"/>
    <uses allow="internal"/>
  </network-interface>
</node>
<node type="storage" userinput_gateway="10.25.113.1">
  <network-interface id="0" userinput_bonded="eth0,eth2"
userinput_ifname="bond0">
    <address value="123.25.113.185"
userinput_netmask="255.255.255.0"
newuserinput_value="123.25.113.185"
userinput_customhostname="a4dpn62d1.default"/>
    <address value="aa:bb:cc:124" userinput_netmask="/64"
newuserinput_value="aa:bb:cc:124"
userinput_customhostname="a4dpn62.v6.default"/>
    <uses allow="replication,backup"/>
  </network-interface>
  <network-interface id="1" userinput_bonded="eth1,eth3"
userinput_ifname="bond1">
   <address value="192.168.255.2" userinput_netmask="255.255.255.0"
newuserinput_value="192.168.255.2"
userinput_customhostname="a4dpn62d1-internal"/>
    <uses allow="internal"/>
  </network-interface>
</node>
```

Edit the probe.xml as follows:

a. With a text editor, modify module attributes in probe.xml (located in /usr/local/avamar/var/) for the entire Avamar system. Use the following as an example of an object with module attributes:

```
<dpn>
  <module name="a4ipn600" userinput_domain="example.com"
userinput_gateway="10.110.227.1" userinput_pns="10.110.195.11"
userinput_sns="10.110.188.5" userinput_summary="24 storages, 1
utility">
```

where:

Table 14 Attribute definitions

Attribute	Definition
userinput_domain	Default fully-qualified domain name for any interface if a custom one is not provided.
userinput_gateway	Default gateway.
userinput_pns	Primary name server.
userinput_sns	Secondary name server. If not defined, either leave blank or omit attribute completely.

b. Modify each additional node object defined in the probe.xml and its attributes. Use the following as an example of a node object and its attributes:

```
<node type="utility" userinput_hostname="a4ipn600">
  <network-interface id="1" userinput_bonded="eth0,eth2"
userinput_ifname="eth0">
    <address newuserinput_value="10.110.227.147"
userinput_customhostname="a4ipn600.example.com"
userinput_netmask="255.255.255.0" value="10.110.227.147"/>
    <uses allow="replication,management,backup"/>
  </network-interface>
  <network-interface id="2" userinput_bonded="eth1,eth3"
userinput_ifname="eth1">
    <address newuserinput_value="192.168.255.1"
userinput_customhostname="a4ipn600.example.com"
userinput_netmask="255.255.255.0" value="192.168.255.1"/>
    <uses allow="internal"/>
  </network-interface>
</node>
```

where:

Table 15 Attribute definitions

Attribute	Definition
userinput_hostname	Hostname of the node.
newuserinput_value	IP Address of the interface.

Table 15 Attribute definitions

Attribute	Definition
userinput_customhostname	Fully-qualified domain name for the IP address of the interface. For example: A node is configured for three VLANs, one untagged backup, one internal interface, and separate replication and management interfaces, each configured as a dual stack. Each unique IP address must have a unique FQDN defined by this attribute.
userinput_netmask	Netmask associated with the interface.
value	Old IP address before modification.

IMPORTANT

Ensure all IPv6 addresses added to the probe.xml file are written in canonic and not full notation. See examples below for the proper format:

Full: 2620:0000:0170:0588:0000:0000:0001:0024
Canonic: 2620:0:170:588::1:24

6. Run the following command on all nodes in the Avamar system:

   ```
   touch /fastboot
   ```

7. Reboot all nodes.

   ```
   reboot
   ```

8. Start the Avamar software.

   ```
   dpnctl start
   ```

9. On single node Avamar servers only, type:

   ```
   dpnctl enable
   ```


Published December, 2014