

Starting the NMC-RX Provisioning Service

This chapter presents information on starting the NMC-RX Provisioning service software.

Topic	Page
Overview	3-1
Starting the NMC-RX Provisioning Service	3-4
Testing the NMC-RX Provisioning Service	3-4

Overview

Juniper Networks uses CORBA freeware Distributed Object Group (DOG) for development and testing. We ship the DOG CORBA freeware package version 2.6 with our NMC-RX Provisioning service (PVS) installer. We offer sample scripts to start NMC-RX Provisioning service based on this CORBA package. You can use a different CORBA package; however, you must recompile CorbaApi.java, which is under the PVS directory.

Sample Scripts for Solaris Users



NOTE: If a service is already running, it will not start again.

The `pvsSampleScript` directory contains the following sample scripts:

- `naming` – Use to start the DOG CORBA Naming service. It uses port 2001. When this service starts, it generates an `ins.ior` file and copies it to `../pvs/JuniperERXNameService.ior` file for use in starting the CORBA Bridge service. Because this relative path is used, be sure to start the sample script from this directory, unless you choose to modify the script.
- `nmcPvsRmiService` – Use to start the remote method invocation (RMI) service.
- `nmcPvsCorbaBridgeService` – Use to start the CORBA Bridge service. If you do not designate a port number, the system assigns a port number. If you want to designate a specific port number (for example, port 12345), modify the `nmcPvsCorbaBridgeService` script by adding option `-OAPort=12345`.
- `nmcPvsTestUtility` – Use to start the NMC-RX Provisioning service test utility. For additional information about this utility, see “Testing the NMC-RX Provisioning Service” on page 3-5.
- `allpvs` (Solaris only) – Use to launch all services at one time in the correct order (NMC-RX Database, Naming service, `remiService`, and CORBA Bridge service).

Use the help option to obtain usage instructions for the scripts.



NOTE: The help feature and all of the options (for example, start, start console, and so on) exist only on Solaris.

The following is a sample of allpvs help:

```
# allpvs help

Usage: allpvs {[start [console]]|stop|restart [console]|status|help}

start - starts NMC-RX database, Naming Service, RMI Service,
       and Corba Bridge Service as background processes.

start console - starts NMC-RX database, Naming Service, RMI Service,
               and Corba Bridge Service in new windows.

stop - stops NMC-RX database, Naming Service, RMI Service,
       and Corba Bridge Service.

restart - stops the NMC-RX database, Naming Service, RMI Service,
          and Corba Bridge Service if they're running, and starts them
          again.

restart console - stops the NMC-RX database, Naming Service, RMI Service,
                 and Corba Bridge Service if they're running, and starts
                 them in new windows.

status - states whether or not the NMC-RX database, Naming Service, RMI
        Service, and Corba Bridge Service are currently running.

help - displays this message.
```

Starting the NMC-RX Provisioning Service

You must run the following four services in order to start the Provisioning service:

- NMC-RX Database service
- CORBA Naming service
- NMC-RX Provisioning RMI service
- NMC-RX Provisioning CORBA Bridge service

To use the NMC-RX Provisioning Service, all of these services must be running.

Three of the services—Database, Naming, and Provisioning RMI—are located in the `pvsSampleScripts` directory. For information about starting the NMC-RX Database service, see *NMC-RX User Guide, Vol. 1*.

To start the NMC-RX Provisioning service:

- 1 (Optional – This step is an alternative to performing steps 2 through 7 and available only on Solaris)
Run the `allpvs` with or without console windows.
 - a Run `allpvs start`.
The four services (NMC-RX Database, Naming, RMI, and CORBA Bridge) start automatically without individual console windows.
 - b Run `allpvs start console`.
The four services (NMC-RX Database, Naming, RMI, and CORBA Bridge) start automatically. Each service has its own console window, which allows you to monitor each process.
- 2 Start the NMC-RX database.
- 3 Start the CORBA Naming service by running either `naming` or `naming start console`.
An `ins.ior` file is generated and is copied to the filename `JuniperERXNameService.ior` located under the `pvs` directory.
- 4 Start the NMC-RX Provisioning Service RMI Service by running the `nmcPvsRmiService` script.

When you see the following message, the RMI service is ready:

```
EmsSessionsFactory binded to JUNIPER_NMCRX_TMF814_EMSSSESSIONFACTORY_8284
```



NOTE: The name "JUNIPER_NMCRX_TMF814_EMSSSESSIONFACTORY_8384" is configurable by using file PVS.rc under NMCRX_HOME. There are four lines in this file:

```
provisioning_server_port=8384
provisioning_server_emsname=JUNIPER_NMCRX
provisioning_server_log_enabled=true
provisioning_server_session_checking_time_interval=180000
```



NOTE: The name "JUNIPER_NMCRX_TMF814_EMSSSESSIONFACTORY_8384" is used for NMC-RX Provisioning Service CORBA service to find RMI Service's entry reference. This is the name that the CORBA client (NMS client) program uses to find the CORBA reference from the CORBA Naming Service.

- 5 Start the NMC-RX Provisioning service CORBA Bridge service by running the nmcPvsCorbaBridgeService script.

The NMC-RX Provisioning service software is ready to be used by network management system (NMS) clients.

Testing the NMC-RX Provisioning Service

You can use the PVS test utility to test the NMC-RX Provisioning service. This utility attempts to connect to the database, the RMI server, and the CORBA server. After connecting, the utility attempts to exchange data over the CORBA bridge by viewing a specified node or all of the nodes.

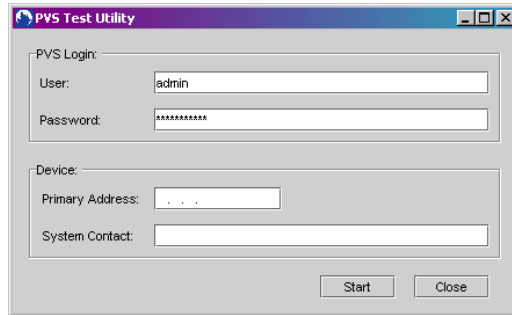
Running the PVS Test Utility

To run the PVS test utility:

- 1 Launch the nmcPvsTestUtility program from the following location:

<NMC-RX-HOME>/pvsSampleScripts/nmcPvsTestUtility

The PVS Test Utility window opens.



- 2 Specify your NMC-RX username and password in the PVS Login area.
- 3 (Optional) Specify a primary IP address in the Device area for the device that you want to test.



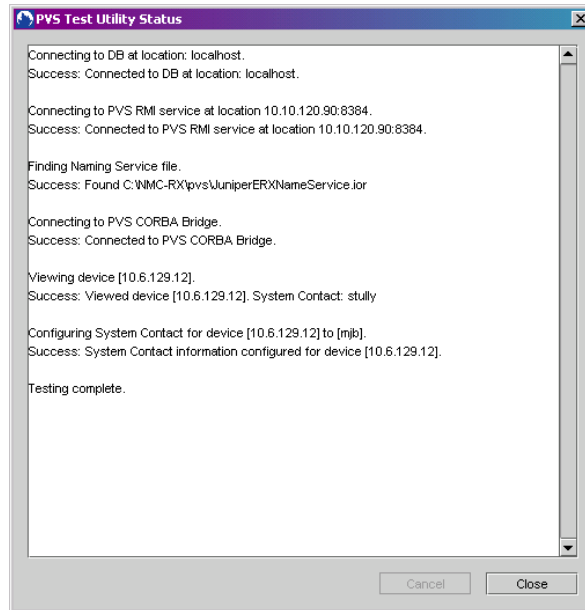
NOTE: If you do not specify a primary IP address, the PVS test utility lists all devices on the system.

- 4 (Optional) Specify a system contact in the Device area to change the system contact.



NOTE: If you do not specify a system contact, the system contact on the device does not change.

- 5 Click Start.
The PVS Test Utility Status dialog box opens.



Once the PVS test utility starts, you can cancel the test or close the dialog box at any time. After the PVS test utility finishes its testing, you can close the dialog box.

Understanding PVS Test Utility Output

The PVS test utility performs a sequence of tests. These tests provide output based on test results. The following table lists each test performed, failure criteria for each test, and potential output text for each test.

Test	Procedure	Failure Criteria	Messages
Database	Obtains database location and user information from local NMC-RX resource file; attempts to contact database	<ul style="list-style-type: none"> Database is not running at the assigned location Database settings in the NMC-RX resources are not correct Various other reasons 	<p>Initial status text:</p> <p>Connecting to DB at location: localhost.</p> <p>Final status text:</p> <p>Success: Connected to DB at location: localhost.</p> <p>Success: Connected to DB at location: <i>ipAddress</i>.</p> <p>Failure: Could not connect to DB at location: <i>ipAddress</i>. [Followed by exception message]</p>
RMI Server	Obtains PVS port number from Provisioning service resources file; contacts RMI server at local IP address	<ul style="list-style-type: none"> RMI server is not running Various other reasons 	<p>Initial status text:</p> <p>Connecting to PVS RMI service at location <i>ipAddress:portNumber</i>.</p> <p>Final status text:</p> <p>Success: Connected to PVS RMI service at location <i>ipAddress:portNumber</i>.</p> <p>Failure: Could not connect to PVS RMI service at location <i>ipAddress:portNumber</i>. [Followed by exception message]</p>
IOR File	Searches for JuniperERXNameService.ior file in PVS subdirectory	<ul style="list-style-type: none"> Unable to find IOR file (indicates naming service has not been run) 	<p>Initial status text:</p> <p>Finding Naming Service file.</p> <p>Final status text:</p> <p>Success: Found d:<i>pathfilename.ior</i>.</p> <p>Failure: Could not find d:<i>pathfilename.ior</i>. [Followed by exception message]</p>
CORBA Bridge	Uses the Naming service to contact the PVS	<ul style="list-style-type: none"> The PVS CORBA bridge is not functioning Naming service is not running Various other reasons 	<p>Initial status text:</p> <p>Connecting to PVS CORBA Bridge.</p> <p>Final status text:</p> <p>Success: Connected to PVS CORBA Bridge.</p> <p>Failure: Could not connect to PVS CORBA Bridge. [Followed by exception message]</p>

Test	Procedure	Failure Criteria	Messages
Nonempty IP Address field	Requests information about the device associated with the IP address you provide	<ul style="list-style-type: none"> Unable to find the specified device Naming service is not running CORBA bridge is not running 	Initial status text: Viewing device [<i>ipAddress</i>]. Final status text: Success: Viewed device [<i>ipAddress</i>]. Failure: Could not view information for device [<i>ipAddress</i>]. [Followed by exception message]
Empty IP address field	Requests information about all devices on the system	<ul style="list-style-type: none"> Unable to list devices Naming service is not running CORBA bridge is not running 	Initial status text: Listing devices. Successful status text: Success: No devices found. Successful status text: Success: <i>ipAddress1</i> <i>ipAddress2</i> <i>ipAddress3</i> <i>ipAddress4</i> Status text for failure: Failure: Could not list devices. [Followed by exception message]
Nonempty IP address field and nonempty System Contact field	Sets the system contact for the device to the specified value	<ul style="list-style-type: none"> Unable to configure system contact value 	Initial status text: Configuring System Contact for device [<i>ipAddress</i>] to [<i>systemContactText</i>]. Final status text: Success: System Contact information configured for device [<i>ipAddress</i>]. Failure: Could not configure System Contact Information.
Status	Indicates that all tests are complete	None	Status text: Testing complete.

