

Chapter 12

Obtaining Interface Configuration for OnePopStaticRouteIp

This chapter describes how to obtain configuration information for a JUNOS interface for use with the OnePopStaticRouteIp configuration scenario in NIC. Topics include:

- JUNOS Interface Information for OnePopStaticRouteIp on page 223
- Information Collection for OnePopStaticRouteIp from the Network Publisher on page 224
- Before You Run the Network Publisher on page 224
- Configuring the Network Publisher on page 224
- Running the Network Publisher on page 229
- Troubleshooting Router Connections and Configuration for the Network Publisher on page 229
- Changing the Location of an Input Directory for the Network Publisher on page 230
- Reviewing the Information Collected from a JUNOS Routing Platform on page 230
- Reviewing and Editing Interface Information from SDX Admin on page 231

JUNOS Interface Information for OnePopStaticRouteIp

The OnePopStaticRouteIp configuration scenario for NIC resolves an IP address for a subscriber whose traffic enters the network through a JUNOS interface to a reference for the SAE that manages the interface. To perform this resolution, the NIC needs information about the JUNOS interfaces. The Threat Mitigation Application Portal relies on the OnePopStaticRouteIp configuration scenario.

The interface information is stored in the directory in an XML document. You can add the interface information to the directory and update the information as needed from the network publisher. The network publisher is a NIC component that collects interface information from the routing tables on specified JUNOS routing platforms. You can view and update the interface configuration from SDX Admin.

Information Collection for OnePopStaticRoutelp from the Network Publisher

The network publisher gathers information about interfaces on specified JUNOS routers and then stores that information in the directory. You run the network publisher whenever you want to get interface information from one or more routers; NIC does not automatically update configuration information in the directory.

The network publisher uses a configuration file on the system on which the NIC host is configured. When you run the network publisher, the information in the configuration file determines the types of information to be collected.

Before You Run the Network Publisher

When you run the network publisher, it connects to a number of JUNOS routing platforms through Telnet.

Before you run the network publisher:

- Verify the version of the JUNOS software that is running on each JUNOS routing platform.

Typically, all of the JUNOS routing platforms should run the same version of the JUNOS software.

- Make sure that a Telnet service is enabled on each router from which the network publisher is to collect interface information.

Configuring the Network Publisher

To configure the network publisher:

- In a text editor, edit the file `/opt/UMC/nic/etc/networkPublisher/config.properties`.

For information about the fields in this file, see *Network Publisher Configuration File Fields* on page 224.

Network Publisher Configuration File Fields

The `/opt/UMC/nic/etc/networkPublisher/config.properties` file contains the following types of configuration fields:

- *Logging Configuration Fields* on page 225
- *Router Configuration Fields* on page 225
- *Filter Configuration Fields* on page 226
- *Directory Configuration Fields* on page 227
- *Troubleshooting Configuration Fields* on page 228

Entries in the file have the format `< field > = < value > .`

The network publisher identifies routers by a number; for example, r1, r2.

Logging Configuration Fields

The network publisher uses the same logging properties as other SRC components.

For information about logging properties and about managing log files, see:

- *SRC-PE Monitoring and Troubleshooting Guide, Chapter 4, Configuring Logging for SRC Components on a Solaris Platform*

Router Configuration Fields

The router configuration fields provide information about the JUNOS routing platforms on which interfaces reside for which the network publisher collects information.

`/routers/tags.release`

- Release number of the JUNOS software installed on the JUNOS routing platform.
- Value—Release number in the format:
`< major release > . < minor release > R < revision >`
- Guidelines—If a value is also specified under `/routers/r < number > tags.release`, the value for the specified router is used for that router.
- Default—No value
- Example—`/routers/tags.release = 7.6R1`

`/routers/tags.hostname`

- Hostname of the machine on which the network publisher runs.
- Value—`< hostname >`
- Default—No value
- Example—`/routers/tags.hostname = myhost`

`/routers/junoscript-authentication.username`

- Username to log in to the JUNOScript server.
- Value—`< username >`
- Default—No value
- Example—`/routers/junoscript-authentication.username = root`

/routers/junoscript-authentication.challenge_response

- Password to log in to the JUNOScript server
- Value— < password >
- Default—No value
- Example—/routers/junoscript-authentication.challenge_response = secret

/routers/r<number>/hostname

- Hostname of a JUNOS routing platform.
- Value— < hostname >
- Default—No value
- Example—/routers/r1/hostname = RouterExternal

/routers/r<number>/address

- IP address of the JUNOS routing platform for which you specified a hostname.
- Value— < IP address in dotted decimal notation >
- Default—No value
- Example—/routers/r1/address = 10.10.10.10

/routers/r<number>/tags.release

- Release number of the JUNOS software installed on a specific JUNOS routing platform.
- Value—Release number in the format:
 < major release > . < minor release > R < release number >
- Guidelines—If a value is also specified under */routers/tags.release*, the value for the specified router is used for that router.
- Default—No value
- Example—/routers/r1/tags.release = 7.6R1

Filter Configuration Fields

The filter configuration fields specify filters that the network publisher uses to collect information from JUNOS routing platforms. You can specify two filters.

/transform/route_table_filter

- Routing table from which the network publisher collects interface information.
- Value—Element name in the format:
 (element-name = < value >)
- Default—(table-name = inet.0)
- Example—/transform/route_table_filter = (table-name = inet.0)

/transform/route_entry_filter

- Element(s) in a specified router table from which the network publisher collects interface information.
- Value—Element name in the format:
(< element-name > = < value >)
- Default—(protocol-name = *)
- Example—/transform/route_entry_filter = (protocol-name = *)

Directory Configuration Fields

The directory fields specify information used to connect to the directory.

/dir/java.naming.provider.url

- URL of the primary directory.
- Value—URL in the format ldap:// < host > :389
< host > —IP address or name of directory host
- Default—No value
- Example—/dir/java.naming.provider.url = ldap://127.0.0.1:389/

/dir/java.naming.security.principal

- Distinguished name (DN) of the directory entry that defines the username with which the network publisher accesses the directory.
- Value— < DN >
- Default—No value
- Example—/dir/java.naming.security.principal = cn = umcadmin, o = umc

/dir/java.naming.security.credentials

- Password with which the network publisher accesses the directory.
- Value— < password >
- Guidelines—The password can be encoded in base64 and not visible in plain text. To use an encoded value, use the format {BASE64} < encoded-value > .
- Default—No value
- Example
 - /dir/java.naming.security.credentials = admin123
 - /dir/java.naming.security.credentials = {BASE64}c3Nw

/dir/baseDN

- Subtree in the directory that stores router data.
- Value— < DN >
- Default—o = Network, o = umc
- Example—/dir/baseDN = o = Network, o = umc

Troubleshooting Configuration Fields

The troubleshooting configuration fields let you specify file information that you can use to troubleshoot configuration for the network publisher.

/routers/r<number>/session_type

- File that contains properties for the routers.
- Value— < filename >
- Guidelines—By default, the network publisher obtains router information through a Telnet session. You can specify an input file for one or more routers to troubleshoot the configuration for the network publisher. The values in an input file for a specified router take precedence over values obtained from the router through a Telnet session.
- Default—LocalFile
- Example—/routers/r1/session_type = LocalFile

/routers/r<number>/input_dir

- Directory that contains the < router_name >_1.xml document (where router_name is the hostname of a router), which contains router properties for the network publisher.
- Value— < directory-name >
- Guidelines—Use a file in the input directory if you do not want to connect to the router to obtain the interface configuration information. Use a file defined by routers/r < number > /session_type in this directory to troubleshoot the configuration for the network publisher.
- Default—/opt/UMC/nic/sample/junos/rt
- Example—/routers/r1/input_dir = /opt/UMC/nic/myconfig

/routers/r<number>/output_dir

- Directory that contains the < router_name >_1.xml document (where router_name is the hostname of a router), which contains interface configuration information collected from the routing table on a JUNOS routing platform.
- Value— < directory-name >
- Guidelines—You must specify an output directory for information to be written to an output file. You can read the information stored in files in this directory to determine whether they contain the expected information from the routing table on the specified JUNOS routing platform.
- Default—opt/UMC/nic/var/junos/rt
- Example—/routers/r1/output_dir = /var/junos/mydir

Running the Network Publisher

You run the network publisher each time you want to collect information about interfaces on JUNOS routing platforms.

To run the network publisher:

1. Move to the *etc* directory that is under the NIC installation directory, typically */opt/UMC/nic/etc*.
2. Run the **networkPublisher appl** command:

```
./networkPublisher appl
```

Troubleshooting Router Connections and Configuration for the Network Publisher

You can troubleshoot the connection between the network publisher and one or more routers, and the configuration on the routers by providing configuration information to the publisher from a file rather than from JUNOS routing platforms.

To specify that the network publisher obtain router configuration information from a file:

1. Edit the */opt/UMC/nic/etc/networkPublisher/config.properties* file.
2. Specify an input directory for the routers by configuring the following property in the file:

```
/routers/r<number>/input_dir =<directory>
```

For example:

```
/routers/r1/input_dir =/opt/UMC/nic/myconfig
```

3. Specify that the network publisher obtain properties for a router from a file on the local system, instead of from the directory, by configuring the following property in the file:

```
/routers/r<number>/session_type = LocalFile
```

For example:

```
/routers/r1/session_type=LocalFile
```

By default, the input file is */opt/UMC/nic/sample/junos/rt/<router_name>_1.xml* where *router_name* is the hostname of a JUNOS routing platform.

You can change the location of the input directory.

See *Changing the Location of an Input Directory for the Network Publisher* on page 230.

Changing the Location of an Input Directory for the Network Publisher

By default, an input file is located in the directory `/opt/UMC/nic/sample/junos/rt`. You can use an input file to troubleshoot the configuration of the network publisher.

You can define a different location for this file in the network publisher configuration in the `/opt/UMC/nic/etc/networkPublisher/config.properties` file.

To change the location of an input file that contains router configuration:

- Edit the `config.properties` file. Specify that the network publisher locate an input file by configuring the following property in the file:

```
/routers/r<number>/input_dir =
```

See *Network Publisher Configuration File Fields* on page 224.

Reviewing the Information Collected from a JUNOS Routing Platform

To review information that the network publisher collects from a JUNOS routing platform:

1. Locate the directory that contains the output file(s) by reviewing the value of the following properties in the `/opt/UMC/nic/etc/networkPublisher/config.properties` configuration file:

- For one router:

```
/routers/r1/output_dir =
```

- For all routers:

```
/routers/output_dir =
```

2. In the directory specified `/routers/r1/output_dir`, open the `<router_name>_1.xml` document (where `router_name` is the hostname of a JUNOS routing platform), and review the file content.

If the information in the file is different from the information expected, there may be a problem with the configuration on the router.

Reviewing and Editing Interface Information from SDX Admin

You can use SDX Admin to edit the XML document that contains interface information used by the OnePopStaticRouteIp configuration scenario for NIC. You use the network publisher to collect the interface information and populate the XML document in the directory for you.

To use SDX Admin to edit the XML document that provides interface information for JUNOS routing platforms:

1. In the navigation pane, select a router under *o = network, o = umc*.
2. Click the **Interface Configuration** tab, and edit values in the content pane.

For information about file syntax and a sample file, see *NIC Document That Maps Subscriber IP Addresses to a JUNOS Interface* on page 231.

NIC Document That Maps Subscriber IP Addresses to a JUNOS Interface

NIC stores information about subscriber IP addresses that map to JUNOS interfaces on which the associated traffic enters the network. This XML document is stored in the specified directory. These files comply with the syntax in the file */opt/UMC/nic/etc/networkConfig.xsd*. An example file */opt/UMC/nic/networkConfig.xml* shows the type of information generated by the network publisher.

For an example configuration in SDX Admin, see the Interface Configuration tab for *o = OrderedCimKeys = THMA1, o = network, o = umc*.

