Juniper Networks | Engineering Simplicity

NorthStar Controller/Planner Getting Started Guide

Published
2024-12-05

RELEASE
6.2.7

Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

NorthStar Controller/Planner Getting Started Guide
6.2.7

The information in this document is current as of the date on the title page.

YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. Junos OS has no known time-related limitations through the year 2038. However, the NTP application is known to have some difficulty in the year 2036.

END USER LICENSE AGREEMENT

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement ("EULA") posted at https://support.juniper.net/support/eula/. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

Table of Contents

About This Guide

Use this guide to install the NorthStar Controller application, perform initial configuration tasks, install optional features, establish connectivity to the network, and access the NorthStar UI. System requirements and deployment scenario server requirements are included.

CHAPTER 1

Installation and Configuration Overview

Platform and Software Compatibility

IN THIS SECTION

The NorthStar Controller 6.2.7 release is qualified to work with Junos OS Release 18.3R2.4. We recommend contacting JTAC for information about the compatibility of other Junos OS releases.Table 1 on page 2 lists feature-specific Junos OS requirements. The NorthStar features listed have been qualified with the specified Junos OS release and are intended to work with that release.

Table 1: Feature-Specific Minimum Junos OS Requirements

NorthStar Feature	Junos OS Release
Analytics	15.1F6
Segment Routing (SPRING), MD5 authentication for PCEP, P2MP, Admin groups	17.2R1
PCEP-Provisioned P2MP Groups	18.3R2
PCEP-Provisioned P2MP Groups with MVPN (S,G) Service Mapping via Flowspec	19.4R1
EPE	19.2R1.8
Bandwidth sizing and container LSPs for SR-TE LSPs	19.2R1.2
PCC Delegated LSP Support for SR LSPs	19.4R3, 20.1R1

NOTE: The Path Computation Element Protocol (PCEP) configuration on the PCC routers does not persist across upgrades when the SDN package is not part of the installation binary. Before upgrading the Junos OS image to this release, save the existing

configuration to a file by using the `save` command. After you upgrade the Junos OS image on each PCC router, use the `load override` command to restore the PCEP configuration.

The NorthStar Controller is supported on the following Juniper platforms: M Series, T Series, MX Series, PTX Series, ACX Series, and QFX10008. As of Junos OS Release 17.4R1, NorthStar Controller is also supported on QFX5110, QFX5100, and QFX5200.

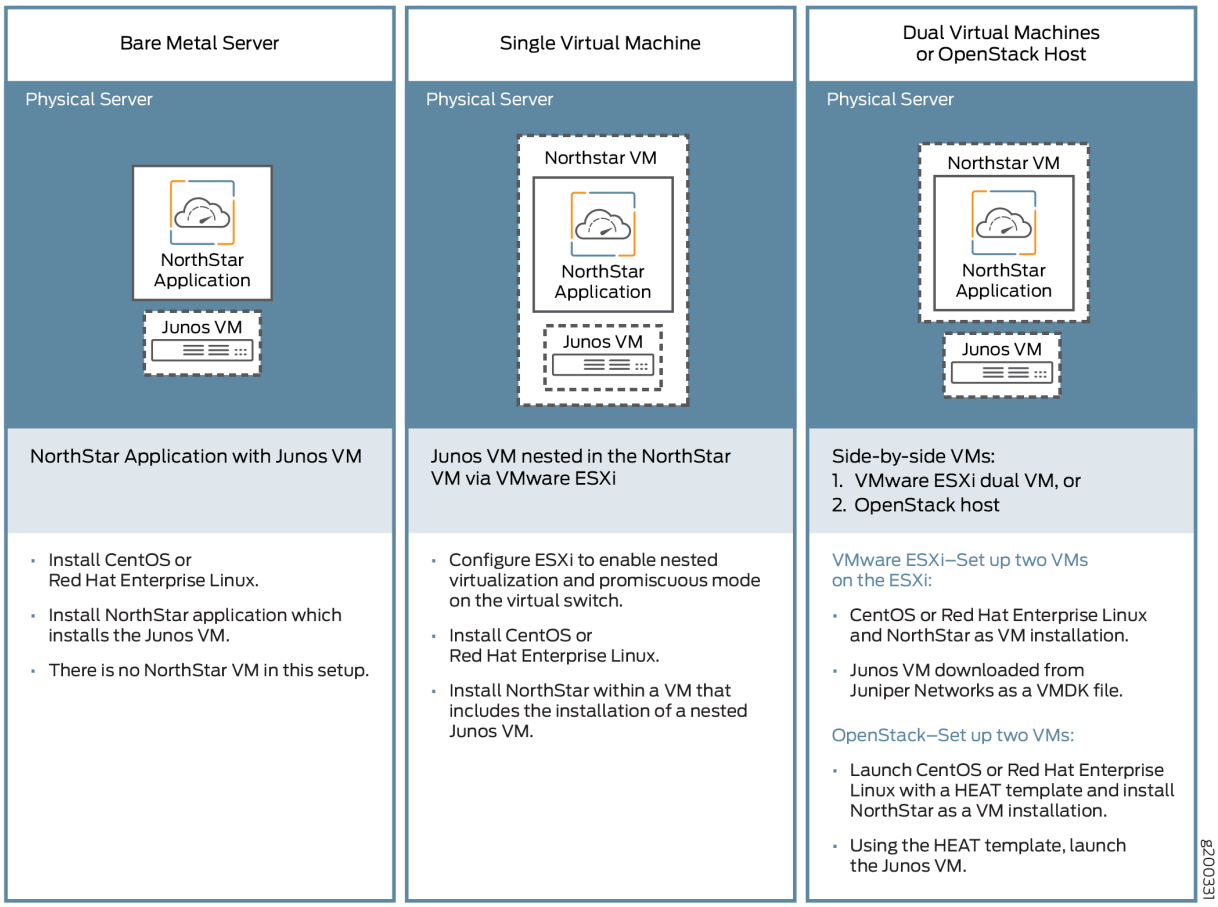
With NorthStar Release 6.2.7, PTX10000 line of routers and ACX7000 line of routers are also supported. For more information on device support, please contact Juniper Networks Technical Assistance Center (JTAC).

Junos OS supports Internet draft draft-crabbe-pce-pce-initiated-lsp-03 for the stateful PCE-initiated LSP implementation (M Series, MX Series, PTX Series, T Series, ACX Series, and QFX Series).

Installation Options

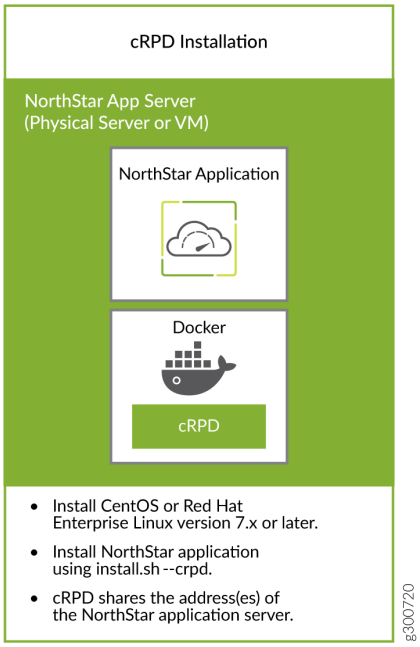
There are three NorthStar Controller installation options for use with Junos VM as summarized in Figure 1 on page 4.

Figure 1: NorthStar/Junos VM Installation Options



You can also install NorthStar Controller using cRPD as summarized in Figure 2 on page 5.

Figure 2: NorthStar/cRPD Installation



NOTE: If you require multiple BGP-LS peering on different subnets for different AS domains at the same time, you should use a JunosVM installation rather than cRPD. That configuration for cRPD is not supported.

For installation procedures, see:

- "Installing the NorthStar Controller" on page 47

  This topic also includes information about installing with NorthStar cRPD.

- "Overview of NorthStar Controller Installation in an OpenStack Environment" on page 90

RELATED DOCUMENTATION

NorthStar Controller System Requirements | 6

Installing the NorthStar Controller | 47

NorthStar Controller System Requirements

IN THIS SECTION

The NorthStar Controller runs on Linux systems running CentOS or Red Hat Enterprise Linux (RHEL).

Before you install, ensure that:

- You use a Juniper-validated version of CentOS Linux or Red Hat Enterprise Linux (RHEL). These are our Linux recommendations:
  - CentOS Linux 7.x or RHEL 7.x (7.6, 7.7, 7.9), or 8.x (8.4, 8.10) image. Earlier versions are not supported, and CentOS Stream has not been validated.

    NOTE: If you use RHEL 8.x (8.4, 8.10), ensure that you:
    - Install legacy network scripts and tools by running the following command.

      ```
      yum -y install net-tools bridge-utils ntp wget ksh telnet java-1.8.0-openjdk-headless network-scripts
      ```

    - Remove the following bundles:
      - `rpm -e buildah cockpit-podman podman-catatonit podman`
      - `yum -y remove runc-1:1.1.12-5.module+el8.10.0+22346+28c02849.x86_64`
    - If you use RHEL 8.10, ensure that you disable RPF on the interface where JTI packets are received. See "Analytics Requirements" on page 16.
  - Install your choice of supported Linux version using the minimal ISO.

- You use RAM, number of virtual CPUs, and hard disk specified in "Server Sizing Guidance" on page 7 for your installation.
- You open the ports listed in "Firewall Port Guidance" on page 12.

NOTE: When upgrading NorthStar Controller, files are backed up to the **/opt** directory.

Server Sizing Guidance

The guidance in this section should help you to configure your servers with sufficient resources to efficiently and effectively support the NorthStar Controller functions. The recommendations in this section are the result of internal testing combined with field data.

A typical NorthStar deployment contains the following systems:

- An application system

  The application system contains the path computation element (PCE), the path computation server (PCS), the components for Web access, topology acquisition, CLI or SNMP message collection and, a configuration database.

- An analytics system

  The analytics system is used for telemetry and collecting NetFlow data, and contains the analytics database. The analytics system is used in deployments tracking traffic levels of a network.

- (Optional) A dedicated or secondary collector

  A secondary collector is used for collecting CLI and SNMP messages from large nodes and is needed when there is a need for a heavy collection of data; see Table 3 on page 8.

- (Optional) A dedicated planner node

  A planner node is required for running offline network simulation on a system other than the application system; see Table 3 on page 8.

For high availability deployments, described in "Configuring a NorthStar Cluster for High Availability" on page 161, a cluster would have 3 or more application and analytics systems, but they would be sized similarly to a deployment with a single application system and a single analytics system.

Table 2 on page 8 outlines the estimated server requirements of the application and analytics systems by network size.

Table 2: Server Requirements for Application and Analytics Systems by Network Size

Instance Type	POC/LAB (RAM / vCPU / HDD)	Medium (<75 nodes) (RAM / vCPU / HDD)	Large (<300 nodes) (RAM / vCPU / HDD)	XL (300+ nodes)* (RAM / vCPU / HDD)
Application	16G / 4vCPU / 500G	64G / 8vCPU / 1T	96G / 8vCPU / 1.5T	128G / 8vCPU / 2T
			For collecting a large number of SNMP and CLI messages on a single, non-high availability (HA) system, you may require additional 16GB RAM and 8 vCPUs or a secondary collector; see Table 3 on page 8.	
Analytics	16G/ 4vCPU/ 500G	48G / 6vCPU/ 1T	64G / 8vCPU/ 2T	64G / 12vCPU / 3T
		NetFlow deployments may require additional 16G to 32G RAM and doubling of the virtual CPUs on the analytics system.		

NOTE: Based on the number of devices in your network, check with your Juniper Networks representative to confirm your specific requirements for networks in the XL category.

Table 3 on page 8 outlines the estimated server requirements for the secondary collectors and dedicated planner.

Table 3: Server Requirements for Secondary Collector and Dedicated Planner

Instance Type	POC/LAB (RAM / vCPU / HDD)	Medium (<75 nodes) (RAM / vCPU / HDD)	Large (<300 nodes) (RAM / vCPU / HDD)	XL (300+ nodes) (RAM / vCPU / HDD)
All-In-One	24G / 8vCPU/ 1T	Not applicable	Not applicable	Not applicable
Secondary Collectors	8G / 4vCPU / 200G	16G / 8vCPU / 500G	16G / 8vCPU / 500G	16G / 8vCPU / 500G

Table 3: Server Requirements for Secondary Collector and Dedicated Planner *(Continued)*

Instance Type	POC/LAB (RAM / vCPU / HDD)	Medium (<75 nodes) (RAM / vCPU / HDD)	Large (<300 nodes) (RAM / vCPU / HDD)	XL (300+ nodes) (RAM / vCPU / HDD)
Dedicated Planner	8G / 4vCPU / 200G	16G / 8vCPU / 1T	16G / 8vCPU / 1T	16G / 8vCPU / 1T
		Additional RAM may be necessary based on the number of active planner sessions and complexity of models.		

NOTE: All-In-One is a configuration, where all the components are installed in a single virtual machine, is intended only for demonstration purposes. It is not recommended for production deployments or lab configurations intended to model production deployments.

When installing the minimal installation CentOS or RHEL Linux, the filesystems can be collapsed to a single root (/) filesystem or separate filesystems. If you are using separate filesystems, you can assign space for each customer according to the size mentioned in Table 4 on page 9 for the different directories.

Table 4: Recommended Space for Filesystem

Filesystem	Space Requirement	Purpose
/boot	1G	Linux kernel and necessary files for boot
swap	0 to 4G	Not needed, but can have minimal configuration
/	10G	Operating system (including **/usr**)
/var/lib/docker	20G	Containerized processes (application system only)
/tmp	24G	NorthStar debug files in case of process error (application system only)
/opt	Remaining space in the filesystem	NorthStar components

Additional Disk Space for JTI Analytics in ElasticSearch

Considerable storage space is needed to support JTI analytics in ElasticSearch. Each JTI record event requires approximately 330 bytes of disk space. A reasonable estimate of the number of events generated is (<num-of-interfaces> + <number-of-LSPs>) ÷ reporting-interval-in-seconds = events per second.

So for a network with 500 routers, 50K interfaces, and 60K LSPs, with a configured five-minute reporting interval (300 seconds), you can expect something in the neighborhood of 366 events per second to be generated. At 330 bytes per event, it comes out to 366 events x 330 bytes x 86,400 seconds in a day = over 10G of disk space per day or 3.65T per year. For the same size network, but with a one-minute reporting interval (60 seconds), you would have a much larger disk space requirement —over 50G per day or 18T per year.

There is an additional roll-up event created per hour per element for data aggregation. In a network with 50K interfaces and 60K LSPs (total of 110K elements), you would have 110K roll-up events per hour. In terms of disk space, that would be 110K events per hour x 330 bytes per event x 24 hours per day = almost 1G of disk space required per day.

For a typical network of about 100K elements (interfaces + LSPs), we recommend that you allow for an additional 11G of disk space per day if you have a five-minute reporting interval, or 51G per day if you have a one-minute reporting interval.

See NorthStar Analytics Raw and Aggregated Data Retention in the *NorthStar Controller User Guide* for information about customizing data aggregation and retention parameters to reduce the amount of disk space required by ElasticSearch.

Additional Disk Space for Network Events in Cassandra

The Cassandra database is another component that requires additional disk space for storage of network events.

Using that same example of 50K interfaces and 60K LSPs (110 elements) and estimating one event every 15 minutes (900 seconds) per element, there would be 122 events per second. The storage needed would then be 122 events per second x 300 bytes per event x 86,400 seconds per day = about 3.2 G per day, or 1.2T per year.

Using one event every 5 minutes per element as an estimate instead of every 15 minutes, the additional storage requirement is more like 9.6G per day or 3.6T per year.

For a typical network of about 100K elements (interfaces + LSPs), we recommend that you allow for an additional 3-10G of disk space per day, depending on the rate of event generation in your network.

By default, NorthStar keeps event history for 35 days. To customize the number of days event data is retained:

1. Modify the dbCapacity parameter in `/opt/northstar/data/web_config.json`
2. Restart the pruneDB process using the `supervisorctl restart infra:prunedb` command.

Collector (Celery) Memory Requirements

When you use the collector.sh script to install secondary collectors on a server separate from the NorthStar application (for distributed collection), the script installs the default number of collector workers described in Table 5 on page 11. The number of celery processes started by each worker is the number of cores in the CPU plus one. So in a 32-core server (for example), the one installed default worker would start 33 celery processes. Each celery process uses about 50M of RAM.

Table 5: Default Workers, Processes, and Memory by Number of CPU Cores

CPU Cores	Workers Installed	Total Worker Processes	Minimum RAM Required
1-4	4	20 (CPUs +1) x 4 = 20	1 GB
5-8	3	18 (CPUs +1) x 2 = 18	1 GB
16	1	17 (CPUs +1) x 1 = 17	1 GB
32	1	33 (CPUs +1) x 1 = 33	2 GB

See "Secondary Collector Installation for Distributed Data Collection" on page 158 for more information about distributed data collection and secondary workers.

The default number of workers installed is intended to optimize server resources, but you can change the number by using the provided config_celery_workers.sh script. See "Collector Worker Installation Customization" on page 156 for more information. You can use this script to balance the number of workers installed with the amount of memory available on the server.

NOTE: This script is also available to change the number of workers installed on the NorthStar application server from the default, which also follows the formulas shown in Table 5 on page 11.

Firewall Port Guidance

The ports listed in Table 6 on page 12 must be allowed by any external firewall being used. The ports with the word **cluster** in their purpose descriptions are associated with high availability (HA) functionality. If you are not planning to configure an HA environment, you can ignore those ports. The ports with the word **Analytics** in their purpose descriptions are associated with the Analytics feature. If you are not planning to use Analytics, you can ignore those ports. The remaining ports listed must be kept open in all configurations.

Table 6: Ports That Must Be Allowed by External Firewalls

Port	Purpose
179	BGP: JunosVM or cRPD for router BGP-LS—not needed if IGP is used for topology acquisition. In a cRPD installation, the router connects port 179/TCP (BGP) directly to the NorthStar application server. cRPD runs as a process inside the NorthStar application server. Junos VM and cRPD are mutually exclusive.
161	SNMP
450	NTAD
830	NETCONF communication between NorthStar Controller and routers. This is the default port for NETCONFD, but in some installations, port 22 is preferred. To change to port 22, access the NorthStar CLI as described in "Configuring NorthStar Settings Using the NorthStar CLI" on page 78, and modify the value of the port setting. Use the **set northstar netconfd device-connection-pool netconf port** command.
1514	Syslog: Default Junos Telemetry Interface reports for RPM probe statistics (supports Analytics)
1812	RADIUS authentication

Table 6: Ports That Must Be Allowed by External Firewalls *(Continued)*

Port	Purpose
2222	Containerized Management Daemon (cMGD). Used to access NorthStar CLI.
2888	Zookeeper cluster
3000	JTI: Default Junos Telemetry Interface reports for IFD, IFL, and LSP (supports NorthStar Analytics). In previous NorthStar releases, three JTI ports were required (2000, 2001, 2002). Starting with Release 4.3.0, this single port is used instead.
3001	Model Driven Telemetry (MDT)
3100	Streaming telemetry for Ericsson.
3888	Zookeeper cluster
4000	MDT (pipeline) - Logstash communication
4189	PCEP: PCC (router) to NorthStar PCE server
5000	cMGD-REST
5672	RabbitMQ
6379	Redis
7000	Communications port to NorthStar Planner
7001	Cassandra database cluster
8123	Health Monitor notification when the NorthStar controller and analytics are installed on different servers;

Table 6: Ports That Must Be Allowed by External Firewalls *(Continued)*

Port	Purpose
8124	Health Monitor
8443	Web: Web client/REST to secure web server (https)
9000	Netflow
9042	Remote Planner Server
9200	Elasticsearch API calls (monitoring, search, and aggregation) over HTTP.
9201	Elasticsearch when NorthStar controller and analytics server are installed on separate servers.
9300	Elasticsearch cluster
10001	BMP passive mode: By default, the monitor listens on this port for incoming connections from the network.
17000	Cassandra database cluster
50051	PRPD: NorthStar application to router network

Figure 3 on page 15 details the direction of data flow through the ports, when node clusters are not being used. Figure 4 on page 16 and Figure 5 on page 16 detail the additional flows for NorthStar application HA clusters and analytics HA clusters, respectively.

Figure 3: NorthStar Main Port Map

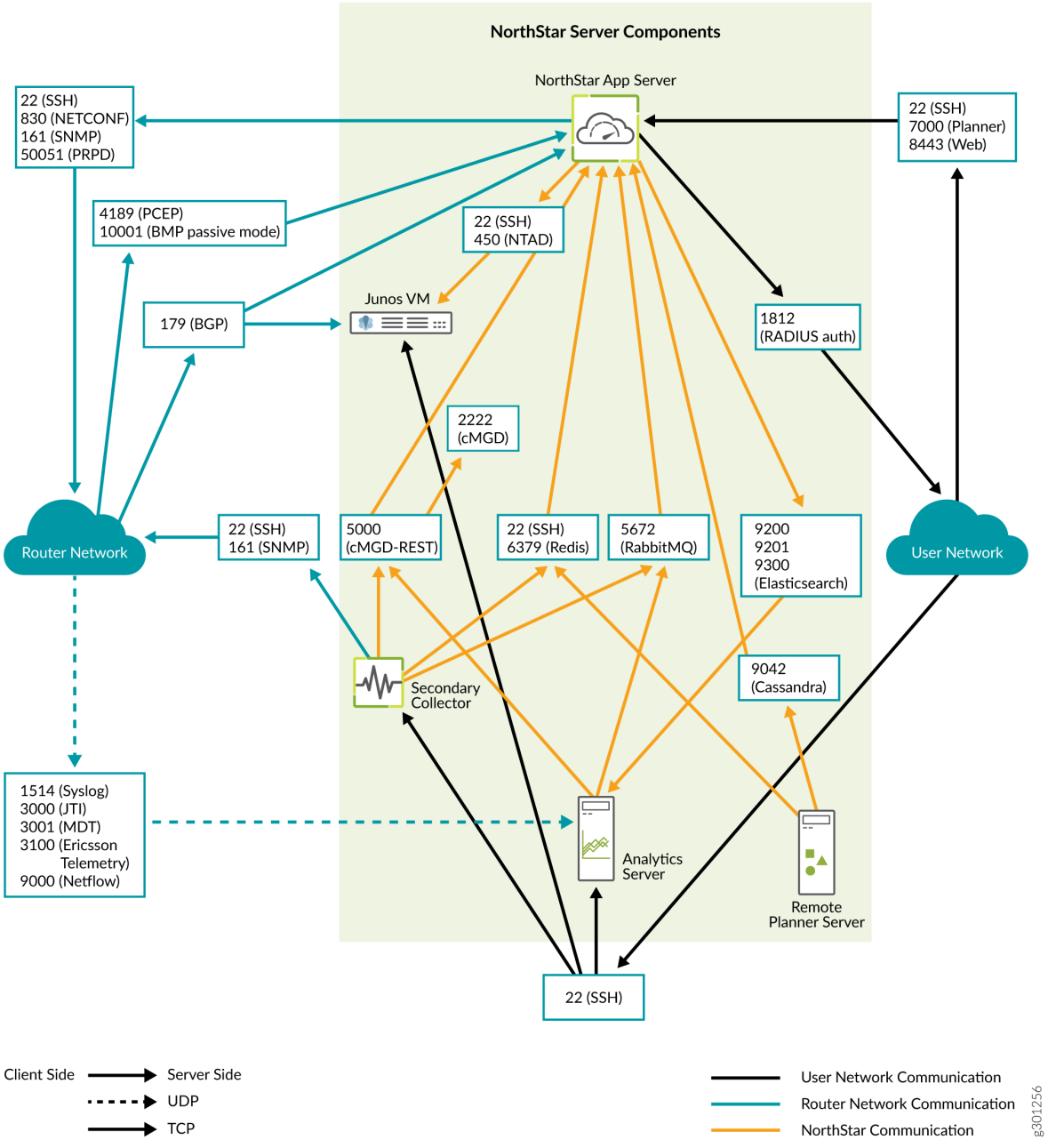


Figure 4: NorthStar Application HA Port Map

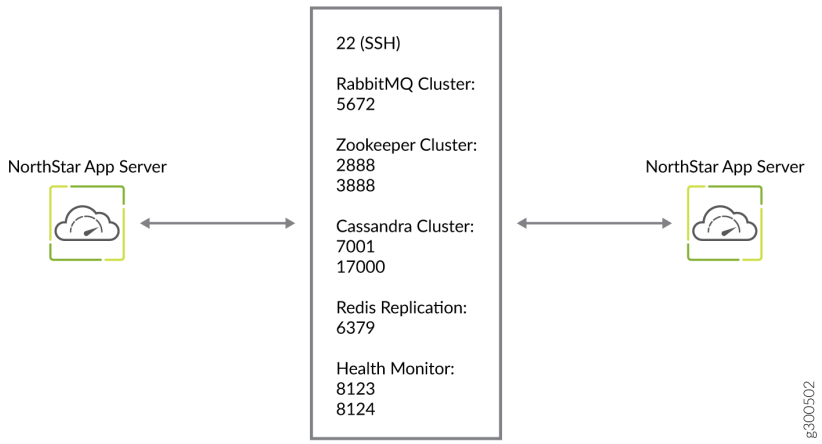
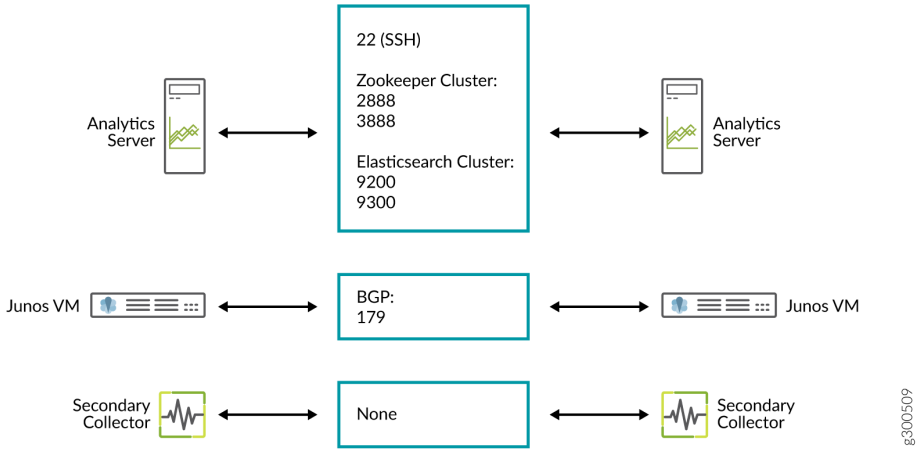


Figure 5: Analytics HA Port Map



Analytics Requirements

In addition to ensuring that ports 3000 and 1514 are kept open, using the NorthStar analytics features requires that you counter the effects of Reverse Path Filtering (RPF) if necessary. If your kernel does RPF by default, you must do **one** of the following to counter the effects:

- Disable RPF.

- Ensure there is a route to the source IP address of the probes pointing to the interface where those probes are received.
- Specify loose mode reverse filtering (if the source address is routable with any of the routes on any of the interfaces).

NOTE: If you use RHEL 8.10, ensure that you disable RPF on the interface where JTI packets are received.

The following commands are examples of how you can disable RPF:

```
sysctl -w net.ipv4.conf.all.rp_filter=0
sysctl -w net.ipv4.conf.ens3f0.rp_filter=0
sysctl -w net.ipv4.conf.ens3f2.rp_filter=0
```

Two-VM Installation Requirements

A two-VM installation is one in which the JunosVM is not bundled with the NorthStar Controller software.

VM Image Requirements

- The NorthStar Controller application VM is installed on top of a Linux VM, so Linux VM is required. You can obtain a Linux VM image in either of the following ways:
  - Use the generic version provided by most Linux distributors. Typically, these are cloud-based images for use in a cloud-init-enabled environment, and do not require a password. These images are fully compatible with OpenStack.
  - Create your own VM image. Some hypervisors, such as generic DVM, allow you to create your own VM image. We recommend this approach if you are not using OpenStack and your hypervisor does not natively support cloud-init.
- The JunosVM is provided in Qcow2 format when inside the NorthStar Controller bundle. If you download the JunosVM separately (not bundled with NorthStar) from the NorthStar download site, it is provided in VMDK format.

- The JunosVM image is only compatible with IDE disk controllers. You must configure the hypervisor to use IDE rather than SATA controller type for the JunosVM disk image.

```
glance image-update --property
hw_disk_bus=ide --property
hw_cdrom_bus=ide
```

JunosVM Version Requirements

If you have, and want to continue using a version of JunosVM older than Release 17.2R1, you can change the NorthStar configuration to support it, but segment routing support would not be available. See "Installing the NorthStar Controller" on page 47 for the configuration steps.

VM Networking Requirements

The following networking requirements must be met for the two-VM installation approach to be successful:

- Each VM requires the following virtual NICs:
  - One connected to the external network
  - One for the internal connection between the NorthStar application and the JunosVM
  - One connected to the management network if a different interface is required between the router facing and client facing interfaces
- We recommend a flat or routed network without any NAT for full compatibility.
- A virtual network with one-to-one NAT (usually referenced as a floating IP) can be used as long as BGP-LS is used as the topology acquisition mechanism. If IS-IS or OSPF adjacency is required, it should be established over a GRE tunnel.

NOTE: A virtual network with n-to-one NAT is not supported.

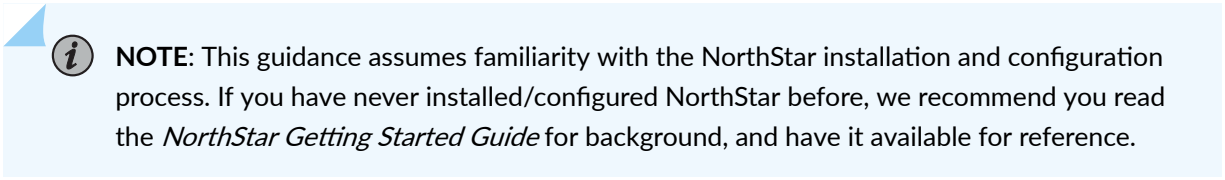
Guidance for Migrating to CentOS 7.x for NorthStar 6.0.0 and Later

IN THIS SECTION

Introduction

NOTE: If you are already using Juniper-validated Linux recommendations such as CentOS 7.x (7.6, 7.7, 7.9) or RHEL 7.x (7.6, 7.7, 7.9) or 8.x (8.4, 8.10), you do not need these instructions. Instead, follow the installation procedures in the *NorthStar Controller/Planner Getting Started Guide* to install or upgrade your NorthStar application.

These instructions are intended to assist you in migrating a working NorthStar 5.1.0 three-node cluster running on CentOS or RHEL 6.10 to a NorthStar 5.1.0 three-node cluster on CentOS 7.x or RHEL 7.x or 8.x. This creates an upgrade path for NorthStar 5.1.0 to NorthStar 6.0.0 or later as CentOS and RHEL 6.x are no longer supported. If you are running a VM or if you have a current backup plan in production, we recommend you take a snapshot or create a backup before proceeding, as the instructions will involve wiping out your HDD/SDD and removing all data on those drives.



NOTE: This guidance assumes familiarity with the NorthStar installation and configuration process. If you have never installed/configured NorthStar before, we recommend you read the *NorthStar Getting Started Guide* for background, and have it available for reference.

You must upgrade the operating system first because NorthStar 6.0.0 or later installation requires CentOS 7.x or RHEL 7.x or 8.x. The order of these procedures is important:

1. Back up your data.

   The following files should be backed up:

   - /opt/northstar/data/*.json
   - /opt/northstar/data/northstar.cfg*
   - /opt/northstar/data/crpd/juniper.conf*
   - /opt/pcs/db/sys/npatpw
   - Output from the /opt/northstar/utils/cmgd_cli -c "show config" command.

2. Upgrade the operating system to CentOS 7.x or RHEL 7.x or 8.x.
3. Install NorthStar 5.1.0 on the upgraded operating system.
4. When all nodes are running CentOS 7.x or RHEL 7.x or 8.x and NorthStar 5.1.0, upgrade NorthStar to 6.0.0 or later.

Example Scenario

For example purposes, these instructions assume you are migrating from CentOS 6.10 to CentOS 7.x, and your network configuration includes:

- Three NorthStar application servers in a cluster
- Three analytics servers in a cluster
- Three collector nodes

Your actual operating system version and network topology might be different, but the principles still apply.

We recommend backing up your operating system files and directories so you have a reference since some of the files differ between CentOS 6.x and CentOS 7.x. Back up these operating system files and directories, and save them to an external or network drive:

1. **/etc/selinux/config**
2. **/etc/sysconfig/**
3. **/etc/hosts**
4. **/etc/ntp.conf**
5. **/etc/resolv.conf**
6. **/etc/ssh/**
7. **/root/.ssh/**

Back up these NorthStar files and directories, and save them to an external or network drive:

1. **/opt/pcs/db/sys/npatpw**
2. **/opt/northstar/data/northstar.cfg**
3. **/opt/northstar/data/*.json**
4. **/opt/northstar/data/junosvm.conf**
5. **/opt/northstar/northstar.env**
6. **/opt/northstar/thirdparty/netconfd/templates**
7. **/opt/northstar/saved_models** (if used for saving NorthStar Planner projects)

The Basic Work Flow

For any node, whether it is a NorthStar application node, an analytics node, or a collector node, the work flow to upgrade your operating system while preserving your clusters and data is essentially the same:

1. Power down one standby node in the cluster setup.
2. Boot that node from the operating system minimal ISO.
3. Install the operating system on the node.
4. Run `yum -y update` to address any critical or security updates.

5. Install recommended packages:

```
yum -y install net-tools bridge-utils ntp wget ksh telnet java-1.8.0-openjdk-headless network-scripts
```

6. Install the NorthStar 5.1.0 application on this same node, setting it up as a standalone host.

> **NOTE**: For NorthStar application nodes, you will need a new license because the interface names change from **eth*x*** to **ens*x*** when you upgrade the operating system. You will not need a new license for analytics or collector nodes.

7. For NorthStar application nodes, launch the web UI on the host **https://*northstar_ip_address*:8443** to ensure the license is working and you can log in successfully.
8. You can check the status of the NorthStar processes by running the `supervisorctl status` command.

In this procedure, we have you start with upgrading the operating system on your analytics cluster, then your NorthStar application cluster, and your collector cluster last. However, this order is not a strict requirement. When all nodes in all clusters are running the upgraded operating system and NorthStar 5.1.0, you then upgrade to NorthStar 6.0.0 or later.

Upgrade the Operating System on Your Analytics Nodes

For analytics nodes, Elasticsearch will self-form the cluster and distribute the data per the replication policy. Therefore, there is no need to first delete the node from Elasticsearch history. To migrate your analytics cluster, use the following procedure:

1. Install CentOS 7.7, 7.9 or 8.6 on a standby analytics node, including the previously stated recommended packages.
2. Install NorthStar-Bundle-5.1.0-20191210_220522_bb37a329b_64.x86_64.rpm on the node where you have the freshly installed operating system.
3. Copy the SSH keys from the existing active node in the analytics cluster and all application nodes to the new analytics node:

```
ssh-copy-id
root@new_analytics_node_ip_address
```

4. Working from an existing node in the cluster, add the new analytics node into the cluster:
   a. From net_setup.py, select **Analytics Data Collector Setting** (G) for external standalone/cluster analytics server setup.

b. Select **Add new Collector node to existing cluster** (E).

   You can use the previous node's ID and other setup information.

Once this process is completed for the first node, repeat the steps for the remaining analytics cluster nodes. Once the process is complete on all three nodes, your analytics cluster will be up and running with CentOS 7.7 and NorthStar 5.1.0.

The following are useful Elasticsearch (REST API) commands you can use before, during and after upgrading your operating system. Run these from an existing node in the analytics cluster.

1. curl -X GET "localhost:9200/_cluster/health?pretty"
2. curl -X GET "localhost:9200/_cat/nodes?v"
3. curl -X GET "localhost:9200/_cat/indices"
4. curl -X GET "localhost:9200/_cat/shards"

Use the following command to check that all nodes in your analytics cluster are up:

```
[root@centos-610-analytics1 root]# /opt/northstar/utils/cluster_status.py -u admin -p %password%
| grep -v Connection | grep -v OAuth2
ZooKeeper cluster status:

Host Name           IPv4                Mode                    Version
centOS-610-analytics1172.25.153.167      follower                 3.5.4-
beta-7f51e5b68cf2f80176ff944a9ebd2abbc65e7327, built on 05/11/2018 16

centOS-610-analytics3172.25.153.70       leader                   3.5.4-
beta-7f51e5b68cf2f80176ff944a9ebd2abbc65e7327, built on 05/11/2018 16

centOS-610-analytics2172.25.153.62       follower                 3.5.4-
beta-7f51e5b68cf2f80176ff944a9ebd2abbc65e7327, built on 05/11/2018 16
```

Upgrade the Operating System on Your NorthStar Application Nodes

Use the following procedure to upgrade your operating system on the NorthStar application nodes:

NOTE: You can refer to the *NorthStar Getting Started Guide*, *Replace a Failed Node if Necessary* section for reference.

1. Install CentOS 7.7 on one of the NorthStar application standby nodes (server or VM), including the recommended packages listed previously.
2. Install the NorthStar 5.1.0 application software (NorthStar-Bundle-5.1.0-20191210_220522_bb37a329b_64.x86_64.rpm). It is important to provide the installation script with the same database password that is on the existing nodes. If necessary, you can reset the database passwords on the existing nodes for consistency before adding the node into the cluster.

   a. Install /opt/pcs/db/sys/nptapw and chown pcs.pcs /opt/pcs/db/sys/npatpw

      Copy your **npatpw** file to the location **/opt/pcs/db/sys/npatpw**. Then run the `chown pcs:pcs /opt/pcs/db/sys/npatpw` command.

   b. Update **/opt/northstar/netconfd/templates**.
3. Copy the SSH keys from the existing active node in the NorthStar cluster and all application nodes.

```
ssh-copy-id
root@new_northstar_node_ip_address
```

4. From an existing node in the cluster, delete the knowledge of the CentOS 6.x node from the cluster, then add it back as a new node:

   a. The example below shows identifying the node that needs to be deleted (the one that is down), removing the node from Cassandra, and then observing the output of status commands as the new node is added back into the cluster. UN = up normal, DN = down normal, UJ = up joining. The goal is to replace all nodes and see them return to UN status.

```
[root@node-1 ~]# . /opt/northstar/northstar.env

[root@node-1 ~]# nodetool status

[root@node1 northstar]# nodetool status

Datacenter: datacenter1
=======================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
--  Address       Load       Tokens       Owns (effective)  Host
ID                              Rack
UN  172.16.18.11  1.28 MB    256          100.0%
56ae8cb0-8ee6-4d3a-9cc0-9499faf60a5f  rack1
UN  172.16.18.12  1.3 MB     256          100.0%            c4566fc1-3b31-40ce-
adcc-729bbabc174e  rack1
```

```
DN  172.16.18.13  2.4 MB     256          100.0%            1cd5aa2f-b8c9-40bb-8aa0-
a7c211842c62  rack1


# identify which node needs to be deleted… it will be in Down (D) state

[root@GNAQP13B1 northstar]# nodetool removenode 1cd5aa2f-b8c9-40bb-8aa0-a7c211842c62
[root@GNAQP13B1 northstar]# nodetool status

Datacenter: datacenter1
=======================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
--  Address       Load       Tokens       Owns (effective)  Host
ID                               Rack
UN  172.16.18.11  1.28 MB    256          100.0%
56ae8cb0-8ee6-4d3a-9cc0-9499faf60a5f  rack1
UN  172.16.18.12  1.31 MB    256          100.0%            c4566fc1-3b31-40ce-
adcc-729bbabc174e  rack1

# later when the node is being added back (track in Cassandra log on new node)
[root@GNAQP13B1 northstar]# nodetool status

Datacenter: datacenter1
=======================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
--  Address       Load       Tokens       Owns (effective)  Host
ID                               Rack
UN  172.16.18.11  1.28 MB    256          100.0%
56ae8cb0-8ee6-4d3a-9cc0-9499faf60a5f  rack1
UN  172.16.18.12  1.95 MB    256          100.0%            c4566fc1-3b31-40ce-
adcc-729bbabc174e  rack1
UJ  172.16.18.13  265.45 KB  256          ?
d068ca2f-9fd4-438f-9df6-6d9c7fa5bdd9  rack1

[root@GNAQP13B1 northstar]# nodetool status

Datacenter: datacenter1
=======================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
--  Address       Load       Tokens       Owns (effective)  Host
```

```
ID                                   Rack
UN  172.16.18.11  1.28 MB    256          100.0%
56ae8cb0-8ee6-4d3a-9cc0-9499faf60a5f  rack1
UN  172.16.18.12  1.95 MB    256          100.0%            c4566fc1-3b31-40ce-
adcc-729bbabc174e  rack1
UN  172.16.18.13  265.45 KB  256          100.0%
d068ca2f-9fd4-438f-9df6-6d9c7fa5bdd9  rack1
```

b. It is important that you resynchronize all your SSH keys once you have rebuilt each node, which includes updating the SSH key on your JunosVM.

c. After the SSH keys are updated on each JunosVM, back up any changes made to the JunosVM by using the net_setup.py script and selecting Option **D** > Option **1**.

d. From the net_setup.py main menu, select **HA Setup** (E).

   Select **Add a new node to existing cluster** (J), using the existing node data in the script, and allow HA deployment to complete.

e. Monitor failover to ensure that it completes properly:

   i. Check the output of the `supervisorctl status` command on the current active node to ensure all processes come up.

   ii. Check the cluster status using the following command:

   ```
   /opt/northstar/utils/cluster_status.py -u admin -p %password%
   ```

   iii. On the node with the VIP (the active node), test failover using the following command:

   ```
   supervisorctl restart infra:ha_agent
   ```

   iv. On the restored node promoting to VIP, use the following command to observe the failover process:

   ```
   tail -f /opt/northstar/logs/ha_agent.msg
   ```

   v. Test the failover process between the three nodes. Optionally, you can add host priority using the net_setup.py script option E (HA Settings).

vi. Run the following command to determine which nodes are currently standby nodes. They should be the two with the higher priority numbers:

```
priority/opt/northstar/utils/cluster_status.py -u admin -p %password%
```

vii. Check the NorthStar web UI again for each node while it is the active node, to make sure the data is synchronized properly between the three nodes.

viii. At this point, you should have a fully-functioning NorthStar 5.1.0 three-node cluster running on the CentOS 7.7 operating system.

Upgrade the Operating System on Your Collector Nodes

Collector nodes operate independently, but are tied to the application VIP. They can be deleted or installed back in independently. Proceed one node at a time with reinstallation.

All three collectors are currently running CentOS 6.10 with NorthStar 5.1.0 (NorthStar-Bundle-5.1.0-20191210_220522_bb37a329b_64.x86_64.rpm).

If you have not already done so, back up the NorthStar files and directories listed previously, and save them to an external or network drive.

1. Install the CentOS 7.7 operating system minimal installation on any one of the collector nodes.
2. Install the following recommended packages: net-tools, bridge-utils, wget, ntp, telnet, ksh, java-1.8.0-openjdk-headless.
3. Bring the system back online with the same IP address. Download the NorthStar 5.1.0 package and install it.

```
rpm -Uvh  NorthStar-Bundle-5.1.0-20191210_220522_bb37a329b_64.x86_64.rpm
```

4. Run the collector install script.

```
cd /opt/northstar/northstar_bundle_5.1.0/ && ./collector.sh install
Config file /opt/northstar/data/northstar.cfg does not exist copying it from Northstar APP
server,  please enter below info:
-------------------------------------------------------------------------------
Please enter application server IP address or host name: 172.25.153.89 (IP of APP Server or
VIP)
Please enter Admin Web UI username: admin
Please enter Admin Web UI password:
```

```
retrieving config file from application server...
Saving to /opt/northstar/data/northstar.cfg
Collector installed....
```

5. Repeat this process on the remaining collector nodes, one at a time.

Special Notes for Nested JunosVM Nodes

The following additional procedure applies to migrating a nested JunosVM setup:

1. Copy the configuration here: **/opt/northstar/data/junosvm/junosvm.conf**.
2. Use the net_setup.py script to assign the JunosVM IP address back to the JunosVM.
3. Copy your backup of **junosvm.conf** into **/opt/northstar/data/junosvm/junosvm.conf**.
4. Restart the JunosVM:

```
supervisorctl restart junos:junosvm
```

5. Observe the JunosVM boot process using this command:

```
#tail -f /opt/northstar/logs/junosvm_telnet.log
```

Upgrade all Nodes to NorthStar 6.0.0 or Later

Now that your network and configuration are upgraded to CentOS 7.7, you can proceed with upgrading NorthStar to 6.0.0 or later.

Analytics Node Upgrade to NorthStar 6.0.0 or Later

Upgrade the nodes in the analytics cluster using the following procedure:

1. Determine which nodes are standby versus active using this command:

```
/opt/northstar/utils/cluster_status.py -u admin -p %password% | grep -v Connection | grep -v
OAuth2
```

2. Back up any NorthStar files to an external or network directory.
3. Download the official NorthStar 6.0.0 or later RPM.
4. Install NorthStar using this command:

```
yum -y install NorthStar-Bundle-6.x.x-20200427_213714_5096f11f3_41.x86_64.rpm
```

5. Install the analytics application using this command:

```
cd /opt/northstar/northstar_bundle_6.x.x/ && ./install-analytics.sh
```

6. Netflowd will be in a FATAL state until the NorthStar application nodes are upgraded and the analytics data collector settings are redeployed as netflowd cannot communicate with cMGD until then. This is an expected error.

```
[root@centos-7-analytics3 northstar_bundle_6.x.x]# supervisorctl status
analytics:elasticsearch          RUNNING   pid 14595, uptime 0:19:10
analytics:esauthproxy            RUNNING   pid 14592, uptime 0:19:10
analytics:logstash               RUNNING   pid 14809, uptime 0:18:08
analytics:netflowd               FATAL     Exited too quickly (process log may have details)
analytics:pipeline               RUNNING   pid 14593, uptime 0:19:10
bmp:bmpMonitor                   RUNNING   pid 13016, uptime 0:30:57
infra:ha_agent                   RUNNING   pid 12656, uptime 0:31:41
infra:healthmonitor              RUNNING   pid 15317, uptime 0:12:50
infra:zookeeper                  RUNNING   pid 12653, uptime 0:31:41
listener1:listener1_00           RUNNING   pid 13113, uptime 0:30:26
```

7. Repeat this process on the remaining standby nodes, then do the same on the active node.
8. Check the Zookeeper status of the analytics cluster:

```
/opt/northstar/utils/cluster_status.py -u admin -p %password% | grep -v Connection | grep -v
OAuth2

ZooKeeper cluster status:
```

```
Host Name           IPv4                Mode
Version
centOS-610-analytics1172.25.153.167      follower                3.5.4-
beta-7f51e5b68cf2f80176ff944a9ebd2abbc65e7327, built on 05/11/2018 16
centOS-610-analytics3172.25.153.70       leader                  3.5.4-
beta-7f51e5b68cf2f80176ff944a9ebd2abbc65e7327, built on 05/11/2018 16
centOS-610-analytics2172.25.153.62       follower                3.5.4-
beta-7f51e5b68cf2f80176ff944a9ebd2abbc65e7327, built on 05/11/2018 16
```

NorthStar Application Node Upgrade to NorthStar 6.0.0 or Later

Upgrade the NorthStar application nodes using the following procedure:

1. Back up any NorthStar files on all nodes.
2. Determine which nodes are standby versus active using this command:

```
/opt/northstar/utils/cluster_status.py -u admin -p %password%
```

3. Start the upgrade procedure on standby nodes first.
4. Download the official NorthStar 6.0.0 or later RPM.
5. Install NorthStar using these commands:

```
yum -y install NorthStar-Bundle-6.x.x-20200427_213714_5096f11f3_41.x86_64.rpm
cd /opt/northstar/northstar_bundle_6.x.x/ && ./install.sh --skip-bridge --yes
```

6. Once installation is complete, set the cMGD root password. If this is not done, the cMGD-rest service will continually loop. The requirement to set a cMGD-rest password is due to the addition of the cMGD service in NorthStar 6.0.0.
   a. In net_setup.py, select **Maintenance & Troubleshooting** (D).
   b. Select **Change cMGD Root Password** (8).
7. Redeploy the analytics data collector configuration settings so netflowd can communication with cMGD.
   a. In net_setup.py, select **Analytics Data Collector Setting** (G) for external standalone/cluster analytics server setup.

b. Select **Prepare and Deploy SINGLE Data Collector Setting** (A), **Prepare and Deploy HA Analytics Data Collector Setting** (B), or **Prepare and Deploy GEO-HA Analytics Data Collector Setting** (C) whichever you had set up before the upgrade.

8. Upgrading a standby node should not trigger a failover. Failover should only occur when the active node is upgraded. At that time, the active node should fail over to an already upgraded standby node.
9. After all standby nodes are upgraded, upgrade the active node to NorthStar 6.0.0 or later.
10. Once all nodes are upgraded and one of the standby nodes has assumed the active role and VIP, monitor the cluster using the following procedure:

    a. Check the status of the NorthStar processes on the current active node using this command:

    ```
    supervisorctl status
    ```

    b. Check the cluster status using this command:

    ```
    /opt/northstar/utils/cluster_status.py -u admin -p %password%
    ```

    c. On the node with the VIP, test the failover using this command:

    ```
    supervisorctl restart infra:ha_agent
    ```

    d. Use the following command to monitor the progress of the failover on the restored node being promoted to active node (with the VIP):

    ```
    tail -f /opt/northstar/logs/ha_agent.msg
    ```

    e. Optionally, add priority to the nodes using the net_setup.py script, Option E (HA Settings). Test the failover process between the three nodes to ensure the priorities are working properly.

    f. Run the following command to find which nodes are currently standby nodes and ensure that failover is proceeding. The standby nodes should be the two with the higher number priority.

    ```
    /opt/northstar/utils/cluster_status.py -u admin -p %password%
    ```

    g. Check the NorthStar web UI again for each node while it is the active node to make sure the data is synchronized properly between the three nodes. Check your nodes, links, LSPs, device profiles, and so on.

h. At this point you should have a fully functioning 6.0.0 (or later) three-node NorthStar application cluster running on the CentOS 7.7 operating system.

Collector Node Upgrade to NorthStar 6.0.0 or Later

Upgrade your collector nodes using the following procedure.

1. Backup any NorthStar files to an external or network drive.
2. Download the official NorthStar RPM.
3. Install NorthStar.

```
yum -y install NorthStar-Bundle-6.x.x-20200427_213714_5096f11f3_41.x86_64.rpm
```

4. Install the NorthStar Collector Application.

```
cd /opt/northstar/northstar_bundle_6.x.x/ && ./collector.sh install
Adding config file /opt/northstar/data/northstar.cfg from Northstar APP server, Please enter
below info:
------------------------------------------------------------------------------------------
----------------------------
Please enter application server IP address or host name: 172.25.153.119
Please enter Admin Web UI username: admin
Please enter Admin Web UI password:

Error sending request to: 172.25.153.119
Collector installed....
collector_main: stopped
collector_main: removed process group
collector:worker1: stopped
collector:worker3: stopped
collector:worker2: stopped
collector:worker4: stopped
collector:worker1: started
collector:worker3: started
collector:worker2: started
collector:worker4: started
```

5. Repeat this process on all remaining collector nodes. When complete, your collector nodes are running NorthStar 6.0.0 or later on CentOS 7.x.

Changing Control Packet Classification Using the Mangle Table

The NorthStar application uses default classification for control packets. To support a different packet classification, you can use Linux firewall iptables to reclassify packets to a different priority.

The following sample configuration snippets show how to modify the ToS bits using the mangle table, changing DSCP values to cs6.

Zookeeper:

```
iptables -t mangle -A POSTROUTING -p tcp -sport 3888 -j DSCP -set-dscp-class cs6
iptables -t mangle -A POSTROUTING -p tcp -dport 3888 -j DSCP -set-dscp-class cs6
iptables -t mangle -A POSTROUTING -p tcp -sport 2888 -j DSCP -set-dscp-class cs6
iptables -t mangle -A POSTROUTING -p tcp -dport 2888 -j DSCP -set-dscp-class cs6
```

Cassandra database:

```
iptables -t mangle -A POSTROUTING -p tcp -sport 7001 -j DSCP -set-dscp-class cs6
iptables -t mangle -A POSTROUTING -p tcp -dport 7001 -j DSCP -set-dscp-class cs6

iptables -t mangle -A POSTROUTING -p tcp -sport 17000 -j DSCP -set-dscp-class cs6
iptables -t mangle -A POSTROUTING -p tcp -dport 17000 -j DSCP -set-dscp-class cs6
iptables -t mangle -A POSTROUTING -p tcp -sport 7199 -j DSCP -set-dscp-class cs6
iptables -t mangle -A POSTROUTING -p tcp -dport 7199 -j DSCP -set-dscp-class cs6
```

RabbitMQ:

```
iptables -t mangle -A POSTROUTING -p tcp -sport 25672 -j DSCP -set-dscp-class cs6
iptables -t mangle -A POSTROUTING -p tcp -dport 25672 -j DSCP -set-dscp-class cs6
iptables -t mangle -A POSTROUTING -p tcp -sport 15672 -j DSCP -set-dscp-class cs6
iptables -t mangle -A POSTROUTING -p tcp -dport 15672 -j DSCP -set-dscp-class cs6
iptables -t mangle -A POSTROUTING -p tcp -sport 4369 -j DSCP -set-dscp-class cs6
iptables -t mangle -A POSTROUTING -p tcp -dport 4369 -j DSCP -set-dscp-class cs6
```

NTAD:

```
iptables -t mangle -A POSTROUTING -p tcp -dport 450 -j DSCP -set-dscp-class cs6
```

PCEP protocol:

```
iptables -t mangle -A POSTROUTING -p tcp -sport 4189 -j DSCP -set-dscp-class cs6
```

ICMP packets used by ha_agent (replace the variable *NET-SUBNET* with your configured network subnet):

```
iptables -t mangle -A POSTROUTING -p icmp -s NET-SUBNET -d NET-SUBNET -j DSCP -set-dscp-class cs6
```

To verify that the class of service setting matches best effort, use the following command on the NorthStar server:

```
tcpdump -i interface-name -v -n -s 1500 “(src host host-IP ) && (ip[1]==0)”
```

To verify that the class of service setting matches cs6, use the following command on the NorthStar server:

```
tcpdump -i interface-name -v -n -s 1500 “(src host host-IP ) && (ip[1]==192)”
```

Renew SSL Certificates for NorthStar Web UI

NorthStar generates SSL certificates during installation. You can renew or replace these SSL certificates generated during installation with the trusted certificates issued or approved by the information technology department in your organization. This topic describes how to replace the SSL certificates for web processes.

The SSL certificate files **cert.pem** and **key.pem** are located at **/opt/northstar/web/certs/**. Both these certificates are in X.509 format and you must restart the web process after you replace the files.

For internal server communications to happen seamlessly, the servers must have valid security certificates installed. However, these certificates do not affect the web processes, and needs to be replaced or renewed only if your security team needs you to do so.

SSL certificates for individual servers are located in these locations:

- Health Monitor—**/opt/northstar/healthMonitor/certs**
- ES Proxy—**/opt/northstar/esauthproxy/certs**
- Web Health—**/opt/northstar/web/routes/v1/health/certs**
- SNMP Collection—**/opt/northstar/snmp-collector/conf**

To replace the SSL certificates for NorthStar web UI:

1. Establish an SSH connection to device on which NorthStar is installed.
2. Navigate to **/opt/northstar/web/**.

```
user@host:~$ cd /opt/northstar/web/
user@host:~/web$ ls -l
total 264
-rwx------.   1 pcs pcs    166 Dec  4  2020 appGlobals.js*
-rwx------.   1 pcs pcs  46457 Jun  3 11:56 app.js*
drwx------.   2 pcs pcs     37 Dec  4  2020 certs/
drwx------.  11 pcs pcs    153 Mar 15 20:07 client/
...
drwx------.   7 pcs pcs   4096 May  7 11:21 test/
drwx------.   6 pcs pcs   4096 May  7 11:22 thirdparty/
drwx------.   2 pcs pcs     55 May  7 11:22 util/
drwx------.   3 pcs pcs     17 Mar 15 20:08 webstart/
```

3. Locate the folder named **certs**. The trusted SSL certificates are stored in this folder.

```
user@host:~/web$ cd certs/
user@host:~/web/certs$ ls -l
total 8
-rwx------. 1 pcs pcs 1294 Feb 17 07:14 cert.pem*
-rwx------. 1 pcs pcs 1679 Feb 17 07:14 key.pem*
```

   - cert.pem—Certificate file
   - key.pem—Key used to generate the certificate.

4. Verify expiration date of the current SSL certificates.

```
user@host:~/web/certs$ openssl x509 -enddate -noout -in cert.pem
notAfter=Apr 28 12:14:11 2023 GMT
```

5. Run the following command to view the contents of the certificate file:

```
user@host:~/web/certs$ openssl x509  -in cert.pem
```

6. Copy the new certificate files and back up the existing certificate files. You can use the backed up certificate files to restore them later in case you face any issue.

```
user@host:~/web/certs$ cp cert.pem cert.pem.bak
user@host:~/web/certs$ cp key.pem key.pem.bak

user@host:~/web/certs$ ls -l
total 16
-rwx------. 1 pcs pcs 1294 Feb 17 07:14 cert.pem*
-rwx------. 1 pcs pcs 1294 Jul  9 11:55 cert.pem.bak*
-rwx------. 1 pcs pcs 1679 Feb 17 07:14 key.pem*
-rwx------. 1 pcs pcs 1679 Jul  9 11:55 key.pem.bak*
```

NOTE: The names of the certificate files must be **cert.pem** and **key.pem**, respectively.

7. (Optional) Verify the status of the severs and web processes.

```
user@host:~/web/certs$ supervisorctl status
bmp:bmpMonitor                   RUNNING   pid 2492, uptime 42 days, 22:05:18
collector:worker1                RUNNING   pid 9737, uptime 42 days, 22:02:59
collector:worker2                RUNNING   pid 9739, uptime 42 days, 22:02:59
collector:worker3                RUNNING   pid 9738, uptime 42 days, 22:02:59
collector:worker4                RUNNING   pid 9740, uptime 42 days, 22:02:59
...
web:app                          RUNNING   pid 7769, uptime 29 days, 0:47:11
web:gui                          RUNNING   pid 6536, uptime 29 days, 1:01:44
web:notification                 RUNNING   pid 6530, uptime 29 days, 1:01:44
web:planner                      RUNNING   pid 6529, uptime 29 days, 1:01:44
web:proxy                        RUNNING   pid 6533, uptime 29 days, 1:01:44
```

```
web:restconf                     RUNNING   pid 6535, uptime 29 days, 1:01:44
web:resthandler                  RUNNING   pid 6532, uptime 29 days, 1:01:44
```

8. Restart the web processes for the changes to take effect.

```
user@host:~/web/certs$ supervisorctl restart web:*
web:proxy: stopped
web:planner: stopped
web:notification: stopped
web:resthandler: stopped
web:gui: stopped
web:app: stopped
web:restconf: stopped
web:planner: started
web:notification: started
web:app: started
web:resthandler: started
web:proxy: started
web:restconf: started
web:gui: started
user@host:~/web/certs$
```

9. Verify that the severs and web processes are running after the restart.

```
user@host:~/web/certs$ supervisorctl status
bmp:bmpMonitor                   RUNNING   pid 2492, uptime 42 days, 22:06:10
collector:worker1                RUNNING   pid 9737, uptime 42 days, 22:03:51
collector:worker2                RUNNING   pid 9739, uptime 42 days, 22:03:51
collector:worker3                RUNNING   pid 9738, uptime 42 days, 22:03:51
collector:worker4                RUNNING   pid 9740, uptime 42 days, 22:03:51
...
web:app                          RUNNING   pid 14383, uptime 0:00:15
web:gui                          RUNNING   pid 14387, uptime 0:00:15
web:notification                 RUNNING   pid 14382, uptime 0:00:15
web:planner                      RUNNING   pid 14381, uptime 0:00:15
web:proxy                        RUNNING   pid 14385, uptime 0:00:15
web:restconf                     RUNNING   pid 14386, uptime 0:00:15
web:resthandler                  RUNNING   pid 14384, uptime 0:00:15
user@host:~/web/certs$
```

The certificates have been successfully renewed and web services restarted. You can now verify the certificate information from your web browser.

NOTE: NorthStar overwrites any user-defined certificates during an upgrade. You need to replace the certificates again after an upgrade.

CHAPTER 2

Installation on a Physical Server

Using an Ansible Playbook to Automate NorthStar Installation

IN THIS SECTION

An Ansible playbook is bundled with the NorthStar download package. You can download the NorthStar download package from the NorthStar download page. The playbook enables automation of NorthStar software installation, and is appropriate for both lab and production systems. If you are not familiar with the Ansible open-source automation tool, information is readily available online. Sample resources include:

- For the Ansible User Guide:

  https://docs.ansible.com/ansible/latest/user_guide/index.html

- For an introduction to the inventory file format:

  https://docs.ansible.com/ansible/latest/user_guide/intro_inventory.html

- For information about “become” (privilege escalation):

  https://docs.ansible.com/ansible/latest/user_guide/become.html

- For information on encrypting secret variables:

  https://docs.ansible.com/ansible/latest/user_guide/vault.html

- For a complete list of arguments that can be included when executing the playbook:

  https://docs.ansible.com/ansible/latest/cli/ansible-playbook.html

The Ansible playbook installs NorthStar with cRPD. There is no playbook available for a Junos VM installation.

See "Installing the NorthStar Controller" on page 47 for installation procedures not using the Ansible playbook.

The Ansible playbook requires a host machine (VM or a laptop/desktop) from which the installation is initiated. This host is called the “control node”. The status of the installation is maintained on the control node to facilitate future configuration changes. This is also a good place to save the inventory file (hosts.yml) and license files for a future reinstallation or update. You will install the public SSH keys existing in the control node on the hosts targeted for installation (the “managed nodes”) so Ansible can communicate with those nodes.

Before You Begin

To prepare for executing the Ansible playbook:

1. Install the following on your control node:

   - Linux operating system
   - Python
   - Python-pip (for installing Ansible)
   - SSH
   - Ansible

   We recommend using virtualenv to create an isolated Python environment in which to install Ansible. It creates a folder with all the necessary executables. You can install Ansible using the `pip` command within the virtualenv. You could also use `pip` to install Ansible in the system environment.

   Here is an example of Ansible installation using virtualenv:

   ```
   $ virtualenv ansible
   $ source ansible/bin/activate
   $ pip install ansible
   ```

2. Identify all the managed nodes where NorthStar software is to be installed. Ensure that each node has the following:

   - Basic operating system. See "NorthStar Controller System Requirements" on page 6.
   - Network connectivity
   - SSH server

3. Ensure that you can SSH from the control node to all of the managed nodes.

   To execute the playbook, you must be able to connect to each managed node and become root. You can set this up by adding your SSH key to the ~/.ssh/authorized_keys file on each managed node either for root directly, or for another account that can become root by executing `sudo` without a password.

   An alternate method is to use variables to specify the username/password to connect to the managed nodes and the sudo password in the inventory file. These are the variables:

   - *ansible_user*

     The user to connect to, such as root. The default is the current user.

   - *ansible_password*

     Password that authenticates the *ansible_user*.

   - *ansible_become*

     Set this to true if *ansible_user* is not root.

   - *ansible_sudo_password*

     Password provided to sudo.

4. Copy the Ansible playbook for NorthStar and the NorthStar-Bundle.rpm to the control node and change to that directory.

Creating the Ansible Inventory File

Create a custom inventory file for your NorthStar installation. The inventory is a group of lists that define the managed nodes in your planned NorthStar installation. The Ansible playbook for NorthStar contains a sample inventory file named **hosts.yml.sample** that you can use as a template to start a custom inventory file. The default name for the inventory file is **hosts.yml**. Use a text editor to customize the inventory file.

The template inventory file is organized into several groups:

- **all**

  Contains the subsection **vars** to define variables that apply to all managed nodes. For example, *ansible_user* defines the account name used to connect to the managed nodes.

- **northstar**

Defines nodes and variables for managed nodes that will run NorthStar services such as PCS, TopoServer, and web front end. Nodes in the **northstar** group should define a *northstar_license* variable that contains the license information for that node

- **northstar_ha**

  Contains nodes or subgroups of nodes that are configured for NorthStar high availability.

- **northstar_analytics**

  Contains nodes and variables for analytics.

- **northstar_collector**

  Contains nodes and variables for analytics secondary collectors.

This example shows a portion of an inventory file including some of these groups:

```
all:
  vars:                             # define common variables
    ansible_user: root              # connect directly to root@target
    northstar_ha_vip: 10.10.10.1       # define HA virtual ip address
    northstar_password: PASSSWORD   # PASSWORD can be encrypted
    northstar_crpd_asn: "100"       # configure CRPD
northstar:
  hosts:
    10.10.10.4:                        # target-host by ip-address or hostname
      northstar_license_file: npatpw.host1 # file containing license
    10.10.10.5:
      northstar_license: |          # (alt:) inline license information:
        expire_date=...             #   each line is indented by 2 space
        card=...
        MAC=ab:cd:ef:00:01:02
        usercount=...
        . . .
    10.10.10.6:
        . . .
northstar_ha:
  children:                         # define subgroup (optional)
    site1:
      vars:
        northstar_ha_site: site1    # common variable for hosts in this child
      hosts:
        10.10.10.4:                    # re-use host defined in northstar group
          northstar_ha_priority: 10 # override priority
```

```
        10.10.10.5:                 # these two hosts use default values
        10.10.10.6:                 #
```

You can encrypt secret variables, such as northstar_password or ansible_password, using the `ansible-vault encrypt_string` command. More information is available here: https://docs.ansible.com/ansible/latest/user_guide/vault.html.

Executing the Playbook

After defining the inventory file, execute the `ansible -m ping all` command to verify that all managed nodes are defined correctly, are reachable, and that SSH login was successful.

Execute the `./install.yml` (or `ansible-playbook install.yml`) command to execute the installation playbook and install all managed nodes as defined in the inventory file. You can add optional arguments to the `install.yml` command. Some useful examples include:

- -e *key=value*

  Extra variables. For example, `-e northstar_bundle_rpm=NorthStar-Bundle-6.2.0.xxx.rpm`

- -i *inventory-file*

  Use a different inventory file. You might utilize this, for example, if you use the control node to install software for independent clusters.

- -l *limit*

  Limit execution to a subset of managed nodes. For example, `-l 10.10.10.4, 10.10.10.5` would only install on those two managed nodes.

- -t *taglist*

  Limit execution to a set of tagged tasks. For example, `-t northstar` would only install the NorthStar application.

- --ask-vault-pass

  Ask for the decryption key for embedded secrets.

Installing Data Collectors and Secondary Collectors for Analytics

You can install NorthStar data collectors to support either of two analytics configurations:

- Analytics co-hosted with the NorthStar application

  For this configuration, add the same managed nodes to the **northstar_analytics** inventory group that are in the **northstar** inventory group.

- External analytics node or cluster

  For this configuration, add one or more managed nodes to the **northstar_analytics** inventory group.

Install analytics secondary collectors by adding managed nodes to the **northstar_collector** inventory group. In order to successfully install secondary collectors, the installation script needs access to a node running the NorthStar application. The primary node must either be installed together with analytics/collector nodes, or it must be running before the analytics/collector nodes are installed. The script takes the required information from the **northstar** inventory group, but you can override that by using the variable *northstar_primary*.

Variables

The variables provided specifically for use with the NorthStar playbook are listed in Table 7 on page 45.

Table 7: NorthStar Ansible Playbook Variables

Variable Name	Description
northstar_bundle_rpm	Name of bundle RPM to install.
chrony_config_server	List of NTP servers. If you do not specify any NTP servers, the managed nodes are configured to synchronize with the following four NTP servers: • 0.pool.ntp.org • 1.pool.ntp.org • 2.pool.ntp.org • 3.pool.ntp.org
northstar_license_file	Per-node NorthStar license file.

Table 7: NorthStar Ansible Playbook Variables *(Continued)*

Variable Name	Description
northstar_license	Per-node NorthStar license (inline).
northstar_crpd_asn	ASN for cRPD route reflector. The default behavior is that the default ASN in the NorthStar configuration script is not modified. The default ASN in the NorthStar configuration script is 64512.
northstar_crpd_license_file	cRPD license file.
northstar_crpd_license	cRPD license (inline).
northstar_ha_group	Managed nodes that are part of HA. By default, all members of the **northstar_ha** inventory group are included.
northstar_ha_vip	Virtual IP address for HA.
northstar_ha_site	Name of a geo-HA site. The default is site1.
northstar_ha_priority	Per-node HA priority. The default is 100.
northstar_primary	The NorthStar application node used to configure remote analytics and collector nodes. The primary node must either be installed together with analytics/collector nodes, or it must be running before the analytics/collector nodes are installed. The default is the first member of the **northstar** inventory group .
northstar_app_nodes	Managed nodes running the NorthStar application. By default, all members of the **northstar** inventory group are included.

Table 7: NorthStar Ansible Playbook Variables *(Continued)*

Variable Name	Description
northstar_analytics_priority	Per-node HA priority. The default is 100.
northstar_analytics_vip	Virtual IP address for the analytics cluster.

Table2 lists some other useful variables.

Variable	Description
ansible_user	User to connect to the managed node.
ansible_password	Password to connect to the control node.
ansible_become	Set this to true if *ansible_user* is not root.
ansible_sudo_password	Password provided to sudo.

RELATED DOCUMENTATION

Installing the NorthStar Controller | 47

Installing the NorthStar Controller

IN THIS SECTION

You can use the procedures described in the following sections if you are performing a fresh install of NorthStar Controller or upgrading from an earlier release, *unless you are using NorthStar analytics and are upgrading from a release older than NorthStar 4.3*. Steps that are not required if upgrading are noted. Before performing a fresh install of NorthStar, you must first use the ./uninstall_all.sh script to uninstall any older versions of NorthStar on the device. See "Uninstalling the NorthStar Controller Application" on page 87.

If you are upgrading from a release earlier than NorthStar 4.3 and you *are* using NorthStar analytics, you must upgrade NorthStar manually using the procedure described in "Upgrading from Pre-4.3 NorthStar with Analytics" on page 227.

If you are upgrading NorthStar from a release earlier than NorthStar 6.0.0, you must redeploy the analytics settings after you upgrade the NorthStar application nodes. This is done from the Analytics Data Collector Configuration Settings menu described in "Installing Data Collectors for Analytics" on page 113. This is to ensure that netflowd can communicate with cMGD (necessary for the NorthStar CLI available starting in NorthStar 6.1.0).

We also recommend that you uninstall any pre-existing older versions of Docker before you install NorthStar. Installing NorthStar will install a current version of Docker.

NOTE: If you use RHEL 8.x (8.4, 8.10), ensure that you:

- Install legacy network scripts and tools by running the following command.

  ```
  yum -y install net-tools bridge-utils ntp wget ksh telnet java-1.8.0-openjdk-
  headless network-scripts
  ```

- Remove the following bundles:

  - `rpm -e buildah cockpit-podman podman-catatonit podman`

  - `yum -y remove runc-1:1.1.12-5.module+el8.10.0+22346+28c02849.x86_64`

- If you use RHEL 8.10, ensure that you disable Reverse Path Filtering (RPF) on the interface where JTI packets are received.

  The following commands are examples of how you can disable RPF:

  ```
  sysctl -w net.ipv4.conf.all.rp_filter=0
  sysctl -w net.ipv4.conf.ens3f0.rp_filter=0
  sysctl -w net.ipv4.conf.ens3f2.rp_filter=0
  ```

The NorthStar software and data are installed in the /opt directory. Be sure to allocate sufficient disk space. See "NorthStar Controller System Requirements" on page 6 for our memory recommendations.

NOTE: When upgrading NorthStar Controller, ensure that the /tmp directory has enough free space to save the contents of the /opt/pcs/data directory because the /opt/pcs/data directory contents are backed up to /tmp during the upgrade process.

If you are installing NorthStar for a high availability (HA) cluster, ensure that:

- You configure each server individually using these instructions before proceeding to HA setup.
- The database and rabbitmq passwords are the same for all servers that will be in the cluster.
- All server time is synchronized by NTP using the following procedure:

  1. Install NTP.

     ```
     yum -y install ntp
     ```

2. Specify the preferred NTP server in **ntp.conf**.

3. Verify the configuration.

```
ntpq -p
```

NOTE: All cluster nodes must have the same time zone and system time settings. This is important to prevent inconsistencies in the database storage of SNMP and LDP task collection delta values.

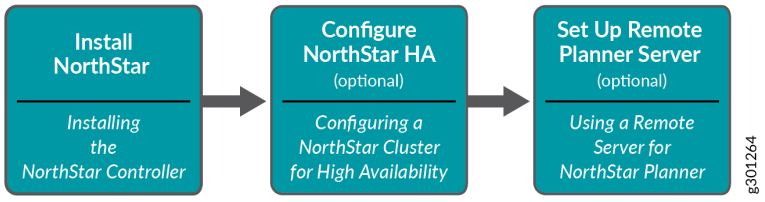
NOTE: To upgrade NorthStar Controller in an HA cluster environment, see "Upgrade the NorthStar Controller Software in an HA Environment" on page 74.

For HA setup after all the servers that will be in the cluster have been configured, see "Configuring a NorthStar Cluster for High Availability" on page 161.

To set up a remote server for NorthStar Planner, see "Using a Remote Server for NorthStar Planner" on page 182.

The high-level order of tasks is shown in Figure 6 on page 50. Installing and configuring NorthStar comes first. If you want a NorthStar HA cluster, you would set that up next. Finally, if you want to use a remote server for NorthStar Planner, you would install and configure that. The text in italics indicates the topics in the *NorthStar Getting Started Guide* that cover the steps.

Figure 6: High Level Process Flow for Installing NorthStar



The following sections describe the download, installation, and initial configuration of NorthStar.

NOTE: The NorthStar software includes a number of third-party packages. To avoid possible conflict, we recommend that you only install these packages as part of the NorthStar Controller RPM bundle installation rather than installing them manually.

Activate Your NorthStar Software

To obtain your serial number certificate and license key, see Obtain Your License Keys and Software for the NorthStar Controller.

Download the Software

The NorthStar Controller software download page is available at https://www.juniper.net/support/downloads/?p=northstar#sw.

1. From the Version drop-down list, select the version number.
2. Click the NorthStar Application (which includes the RPM bundle and the Ansible playbook) and the NorthStar JunosVM to download them.

If Upgrading, Back Up Your JunosVM Configuration and iptables

If you are doing an upgrade from a previous NorthStar release, and you previously installed NorthStar and Junos VM together, back up your JunosVM configuration before installing the new software. Restoration of the JunosVM configuration is performed automatically after the upgrade is complete as long as you use the *net_setup.py* utility to save your backup.

1. Launch the *net_setup.py* script:

```
[root@hostname~]# /opt/northstar/utils/net_setup.py
```

2. Type **D** and press **Enter** to select Maintenance and Troubleshooting.
3. Type **1** and press **Enter** to select Backup JunosVM Configuration.
4. Confirm the backup JunosVM configuration is stored at `'/opt/northstar/data/junosvm/junosvm.conf'`.
5. Save the iptables.

```
iptables-save > /opt/northstar/data/iptables.conf
```

If Upgrading from an Earlier Service Pack Installation

You cannot directly upgrade to NorthStar Release 6.2.7 from an earlier NorthStar Release with service pack installation; for example, you cannot upgrade to NorthStar Release 6.2.7 directly from a NorthStar 6.2.0 SP1 or 6.1.0 SP5 installation. So, to upgrade to NorthStar Release 6.2.7 from an earlier NorthStar Release with service pack installation, you must rollback the service packs or run the **upgrade_NS_with_patches.sh** script to allow installation of a newer NorthStar version over the service packs.

To upgrade to NorthStar Release 6.2.7, before proceeding with the installation:

1. Navigate to the service pack deployment directory. For example:

```
[root@host]# cd NorthStar_6.2.x-Patch-All-20210715
```

2. Do one of the following:

   - Rollback the service packs by running the **batch-uninstall.sh** script.

```
[root@host]# ./batch-uninstall.sh 1
```

   - Upgrade the installation by executing **upgrade_NS_with_patches.sh.**

```
[root@host]# ./upgrade_NS_with_patches.sh
```

     The **upgrade_NS_with_patches.sh** script removes the entries from the package database so that the NorthStar Release 6.2.7 packages can be installed without any dependency conflict.

Install NorthStar Controller

You can either install the RPM bundle on a physical server or use a two-VM installation method in an OpenStack environment, in which the JunosVM is not bundled with the NorthStar Controller software.

The following optional parameters are available for use with the *install.sh* command:

--vm Same as ./install-vm.sh, creates a two-VM installation.

--crpd Creates a cRPD installation.

--skip-bridge For a physical server installation, skips checking if the external0 and mgmt0 bridges exist.

The default bridges are external0 and mgmt0. If you have two interfaces such as eth0 and eth1 in the physical setup, you must configure the bridges to those interfaces. However, you can also define any bridge names relevant to your deployment.

NOTE: We recommend that you configure the bridges before running *install.sh*.

NOTE: Bridges are not used with cRPD installations.

- For a physical server installation, execute the following commands to install NorthStar Controller:

```
[root@hostname~]# yum install <rpm-filename>
[root@hostname~]# cd /opt/northstar/northstar_bundle_x.x.x/
[root@hostname~]# ./install.sh
```

NOTE: yum install works for both upgrade and fresh installation.

- For a two-VM installation, execute the following commands to install NorthStar Controller:

```
[root@hostname~]# yum install <rpm-filename>
[root@hostname~]# cd /opt/northstar/northstar_bundle_x.x.x/
[root@hostname~]# ./install-vm.sh
```

NOTE: yum install works for both upgrade and fresh installation.

The script offers the opportunity to change the JunosVM IP address from the system default of 172.16.16.2.

```
Checking current disk space
INFO: Current available disk space for /opt/northstar is 34G. Will proceed with installation.
System currently using 172.16.16.2 as NTAD/junosvm ip
Do you wish to change NTAD/junosvm ip (Y/N)? y
Please specify junosvm ip:
```

- For a cRPD installation, you must have:
  - CentOS or Red Hat Enterprise Linux 7.x. Earlier versions are not supported.
  - A Junos cRPD license.

    The license is installed during NorthStar installation. Verify that the cRPD license is installed by running the `show system license` command in the cRPD container.

> **NOTE**: If you require multiple BGP-LS peering on different subnets for different AS domains at the same time, you should choose the default JunosVM approach. This configuration for cRPD is not supported.

For a cRPD installation, execute the following commands to install NorthStar Controller:

```
[root@hostname~]# yum install <rpm-filename>
[root@hostname~]# cd /opt/northstar/northstar_bundle_x.x.x/
[root@hostname~]# ./install.sh --crpd
```

> **NOTE**: yum install works for both upgrade and fresh installation.

Configure Support for Different JunosVM Versions

> **NOTE**: This procedure is not applicable to cRPD installations.

If you are using a two-VM installation, in which the JunosVM is not bundled with the NorthStar Controller, you might need to edit the northstar.cfg file to make the NorthStar Controller compatible with the external VM by changing the version of NTAD used. **For a NorthStar cluster configuration, you must change the NTAD version in the northstar.cfg file for every node in the cluster**. NTAD is a 32-bit process which requires that the JunosVM device running NTAD be configured accordingly. You can copy the default JunosVM configuration from what is provided with the NorthStar release (for use in a nested installation). You must at least ensure that the force-32-bit flag is set:

```
[northstar@jvm1]#set system processes routing force-32-bit
```

To change the NTAD version in the northstar.cfg file:

1. SSH to the NorthStar application server.
2. Using a text editor such as vi, edit the ntad_version statement in the **opt/northstar/data/northstar.cfg** file to the appropriate NTAD version according to Table 8 on page 55:

```
[root@ns]# vi /opt/northstar/data/northstar.cfg
...
# NTAD versions(1=No SR; 2=SR, no local addr; 3=V2+local addr 18.2; *4=V3+BGP peer SID
18.3R2, 18.4R2; 5=V4+OSPF SR 19.1+)
ntad_version=version-number
```

Table 8: NTAD Versions by Junos OS Release

NTAD Version	Junos OS Release	Change
1	Earlier than Release 17.2	Initial version
2	17.2	Segment routing
3	18.2	NTAD version 2 + local address "Local address" refers to multiple secondary IP addresses on interfaces. This is especially relevant in certain use cases such as loopback interface for VPN-LSP binding.
4	18.3R2, 18.4R2	NTAD version 3 + BGP peer SID
5	19.1 and later	NTAD version 4 + OSPF SR

3. Manually restart the toposerver process:

```
[root@ns]# supervisorctl restart northstar:toposerver
```

4. Log into the Junos VM and restart NTAD:

```
[northstar@jvm1]#restart network-topology-export
```

5. Set up the SSH key for the external VM by selecting option **H** from the Setup Main Menu when you run the net_setup.py script, and entering the requested information.

Create Passwords

NOTE: This step is not required if you are doing an upgrade rather than a fresh installation.

When prompted, enter new database/rabbitmq, web UI Admin, and cMGD root passwords.

1. Create an initial database/rabbitmq password by typing the password at the following prompts:

```
Please enter new DB and MQ password (at least one digit, one lowercase, one uppercase and no
space):
Please confirm new DB and MQ password:
```

2. Create an initial Admin password for the web UI by typing the password at the following prompts:

```
Please enter new UI Admin password:
Please confirm new UI Admin password:
```

3. Create a cMGD root password (for access to the NorthStar CLI) by typing the password at the following prompts:

```
Please enter new cMGD root password:
Please confirm new cMGD root password:
```

Enable the NorthStar License

NOTE: This step is not required if you are doing an upgrade rather than a fresh installation.

You must enable the NorthStar license as follows, unless you are performing an upgrade and you have an activated license.

NOTE: For HA cluster, the license file must be updated on each application server.

1. Copy or move the license file.

```
[root@northstar]# cp /path-to-license-file/npatpw /opt/pcs/db/sys/npatpw
```

2. Set the license file owner to the PCS user.

```
[root@northstar]# chown pcs:pcs /opt/pcs/db/sys/npatpw
```

3. Restart the following services on the standalone or primary application server:

```
[root@northstar]# supervisorctl restart northstar_pcs:* && supervisorctl restart web:*
```

4. Wait a few minutes and then check the status of the NorthStar Controller processes until they are all up and running.

```
[root@northstar]# supervisorctl status
```

Adjust Firewall Policies

The iptables default rules could interfere with NorthStar-related traffic. If necessary, adjust the firewall policies.

Refer to "NorthStar Controller System Requirements" on page 6 for a list of ports that must be allowed by iptables and firewalls.

Launch the Net Setup Utility

NOTE: This step is not required if you are doing an upgrade rather than a fresh installation.

NOTE: For installations that include a remote Planner server, the Net Setup utility is used only on the Controller server and not on the Remote Planner Server. Instead, the install-remote_planner.sh installation script launches a different setup utility, called setup_remote_planner.py. See "Using a Remote Server for NorthStar Planner" on page 182.

Launch the *Net Setup* utility to perform host server configuration.

```
[root@northstar]# /opt/northstar/utils/net_setup.py
```

The main menu that appears is slightly different depending on whether your installation uses Junos VM or is a cRPD installation.

For Junos VM installations (installation on a physical server or a two-server installation), the main menu looks like this:

```
Main Menu:
    ........................................
    A.) Host Setting
    ........................................
    B.) JunosVM Setting
    ........................................
    C.) Check Network Setting
    ........................................
    D.) Maintenance & Troubleshooting
    ........................................
    E.) HA Setting
    ........................................
    F.) Collect Trace/Log
    ........................................
    G.) Analytics Data Collector Setting
      (External standalone/cluster analytics server)
    ........................................
    H.) Setup SSH Key for external JunosVM  setup
    ........................................
    I.) Internal Analytics Setting (HA)
    ........................................
    X.) Exit
    ........................................
Please select a letter to execute.
```

For cRPD installations, the main menu looks like this:

```
Main Menu:
    ........................................
    A.) Host Setting
    ........................................
```

```
    B.) Junos CRPD Setting
    ........................................
    C.) Check Network Setting
    ........................................
    D.) Maintenance & Troubleshooting
    ........................................
    E.) HA Setting
    ........................................
    F.) Collect Trace/Log
    ........................................
    G.) Analytics Data Collector Setting
      (External standalone/cluster analytics server)
    ........................................
    I.) Internal Analytics Setting (HA)
    ........................................
    X.) Exit
    ........................................
Please select a letter to execute.
```

Notice that option B is specific to cRPD and option H is not available as it is not relevant to cRPD.

Configure the Host Server

NOTE: This step is not required if you are doing an upgrade rather than a fresh installation.

1. From the NorthStar Controller setup Main Menu, type **A** and press **Enter** to display the Host Configuration menu:

```
 Host Configuration:
  ******************************************************
  In order to commit your changes you must select option Z
  ******************************************************
  ........................................
  1. ) Hostname                                : northstar
  2. ) Host default gateway                    :
  3A.) Host Interface #1 (external_interface)
           Name                                : external0
           IPv4                                :
           Netmask                             :
```

```
          Type (network/management)              : network
   3B.) Delete Host Interface #1 (external_interface) data
   4A.) Host Interface #2 (mgmt_interface)
          Name                                   : mgmt0
          IPv4                                   :
          Netmask                                :
          Type (network/management)              : management
   4B.) Delete Host Interface #2 (mgmt_interface) data
   5A.) Host Interface #3
          Name                                   :
          IPv4                                   :
          Netmask                                :
          Type (network/management)              : network
   5B.) Delete Host Interface #3 data
   6A.) Host Interface #4
          Name                                   :
          IPv4                                   :
          Netmask                                :
          Type (network/management)              : network
   6B.) Delete Host Interface #4 data
   7A.) Host Interface #5
          Name                                   :
          IPv4                                   :
          Netmask                                :
          Type (network/management)              : network
   7B.) Delete Host Interface #5 data
   8. ) Show Host current static route
   9. ) Show Host candidate static route
   A. ) Add Host candidate static route
   B. ) Remove Host candidate static route

   ........................................
   X. ) Host current setting
   Y. ) Apply Host static route only
   Z. ) Apply Host setting and static route
   ........................................
   ........................................

Please select a number to modify.
[<CR>=return to main menu]:
```

To interact with this menu, type the number or letter corresponding to the item you want to add or change, and press **Enter**.

2. Type **1** and press **Enter** to configure the hostname. The existing hostname is displayed. Type the new hostname and press **Enter**.

```
Please select a number to modify.
[<CR>=return to main menu]:
1
current host hostname : northstar
new host hostname : node1
```

3. Type **2** and press **Enter** to configure the host default gateway. The existing host default gateway IP address (if any) is displayed. Type the new gateway IP address and press **Enter**.

```
Please select a number to modify.
[<CR>=return to main menu]:
2
current host default_gateway :
new host default_gateway : 10.25.152.1
```

4. Type **3A** and press **Enter** to configure the host interface #1 (external_interface). The first item of existing host interface #1 information is displayed. Type each item of new information (interface name, IPv4 address, netmask, type), and press **Enter** to proceed to the next.

NOTE: The designation of network or management for the type of interface is a label only, for your convenience. NorthStar Controller does not use this information.

```
Please select a number to modify.
[<CR>=return to main menu]:
3A
current host interface1 name : external0
new host interface1 name : external0

current host interface1 ipv4 :
new host interface1 ipv4 : 10.25.153.6

current host interface1 netmask :
new host interface1 netmask : 255.255.254.0

current host interface1 type (network/management) : network
new host interface1 type (network/management) : network
```

5. Type **A** and press **Enter** to add a host candidate static route. The existing route, if any, is displayed. Type the new route and press **Enter**.

```
Please select a number to modify.
[<CR>=return to main menu]:
A
Candidate static route:
new static route (format: x.x.x.x/xy via a.b.c.d dev <interface_name>):
10.25.158.0/24 via 10.25.152.2 dev external0
```

6. If you have more than one static route, type **A** and press **Enter** again to add each additional route.

```
Please select a number to modify.
[<CR>=return to main menu]:
A
Candidate static route:
[0] 10.25.158.0/24 via 10.25.152.2 dev external0
new static route (format: x.x.x.x/xy via a.b.c.d dev <interface_name>):
10.25.159.0/24 via 10.25.152.2 dev external0
```

7. Type **Z** and press **Enter** to save your changes to the host configuration.

NOTE: If the host has been configured using the CLI, the Z option is not required.

The following example shows saving the host configuration.

```
Host Configuration:
   *********************************************************
   In order to commit your changes you must select option Z
   *********************************************************
   .........................................
   1. ) Hostname                                : node1
   2. ) Host default gateway                    : 10.25.152.1
   3A.) Host Interface #1 (external_interface)
            Name                                : external0
            IPv4                                : 10.25.153.6
            Netmask                             : 255.255.254.0
            Type (network/management)           : network
   3B.) Delete Host Interface #1 (external_interface) data
   4A.) Host Interface #2 (mgmt_interface)
```

```
            Name                                    : mgmt0
            IPv4                                    :
            Netmask                                 :
            Type (network/management)               : management
   4B.) Delete Host Interface #2 (mgmt_interface) data
   5A.) Host Interface #3
            Name                                    :
            IPv4                                    :
            Netmask                                 :
            Type (network/management)               : network
   5B.) Delete Host Interface #3 data
   6A.) Host Interface #4
            Name                                    :
            IPv4                                    :
            Netmask                                 :
            Type (network/management)               : network
   6B.) Delete Host Interface #4 data
   7A.) Host Interface #5
            Name                                    :
            IPv4                                    :
            Netmask                                 :
            Type (network/management)               : network
   7B.) Delete Host Interface #5 data
   8. ) Show Host current static route
   9. ) Show Host candidate static route
   A. ) Add Host candidate static route
   B. ) Remove Host candidate static route
   ........................................
   X.) Host current setting
   Y.) Apply Host static route only
   Z.) Apply Host setting and static route
   ........................................
   ........................................
Please select a number to modify.
[<CR>=return to main menu]:
z
Are you sure you want to setup host and static route configuration? This option will restart
network services/interfaces (Y/N) y
Current host/PCS network configuration:
host current interface external0 IP: 10.25.153.6/255.255.254.0
host current interface internal0 IP: 172.16.16.1/255.255.255.0
host current default gateway: 10.25.152.1
Current host static route:
```

```
[0] 10.25.158.0/24 via 10.25.152.2 dev external0
[1] 10.25.159.0/24 via 10.25.152.2 dev external0

Applying host configuration: /opt/northstar/data/net_setup.json
Please wait ...
Restart Networking ...
Current host static route:
[0] 10.25.158.0/24 via 10.25.152.2 dev external0
[1] 10.25.159.0/24 via 10.25.152.2 dev external0
Deleting current static routes ...
Applying candidate static routes
Static route has been added successfully for cmd 'ip route add 10.25.158.0/24 via 10.25.152.2'
Static route has been added successfully for cmd 'ip route add 10.25.159.0/24 via 10.25.152.2'
Host has been configured successfully
```

8. Press **Enter** to return to the Main Menu.

Configure the JunosVM and its Interfaces

This section applies to physical server or two-VM installations that use Junos VM. If you are installing NorthStar using cRPD, skip this section and proceed to "Configure Junos cRPD Settings" on page 70.

NOTE: This step is not required if you are doing an upgrade rather than a fresh installation.

From the Setup Main Menu, configure the JunosVM and its interfaces. Ping the JunosVM to ensure that it is up before attempting to configure it. The net_setup script uses IP 172.16.16.2 to access the JunosVM using the login name **northstar**.

1. From the Main Menu, type **B** and press **Enter** to display the JunosVM Configuration menu:

```
Junos VM  Configuration Settings:
   ******************************************************
   In order to commit your changes you must select option Z
   ******************************************************
   ................................................
   1. ) JunosVM hostname                          : northstar_junosvm
   2. ) JunosVM default gateway                   :
   3. ) BGP AS number                             : 100
   4A.) JunosVM Interface #1 (external_interface)
            Name                                  : em1
```

```
            IPv4                                    :
            Netmask                                 :
            Type(network/management)                : network
            Bridge name                             : external0
   4B.) Delete JunosVM Interface #1 (external_interface) data
   5A.) JunosVM Interface #2 (mgmt_interface)
            Name                                    : em2
            IPv4                                    :
            Netmask                                 :
            Type(network/management)                : management
            Bridge name                             : mgmt0
5B.) Delete JunosVM Interface #2 (mgmt_interface) data
   6A.) JunosVM Interface #3
            Name                                    :
            IPv4                                    :
            Netmask                                 :
            Type(network/management)                : network
            Bridge name                             :
   6B.) Delete JunosVM Interface #3 data
   7A.) JunosVM Interface #4
            Name                                    :
            IPv4                                    :
            Netmask                                 :
            Type(network/management)                : network
            Bridge name                             :
7B.) Delete JunosVM Interface #4 data
   8A.) JunosVM Interface #5
            Name                                    :
            IPv4                                    :
            Netmask                                 :
            Type(network/management)                : network
            Bridge name                             :
   8B.) Delete JunosVM Interface #5 data
   9. ) Show JunosVM current static route
   A. ) Show JunosVM candidate static route
   B. ) Add JunosVM candidate static route
   C. ) Remove JunosVM candidate static route

   ................................................
   X. ) JunosVM current setting
   Y. ) Apply JunosVM static route only
   Z. ) Apply JunosVM Setting and static route
   ................................................
```

```
Please select a number to modify.
[<CR>=return to main menu]:
```

To interact with this menu, type the number or letter corresponding to the item you want to add or change, and press **Enter**.

2. Type **1** and press **Enter** to configure the JunosVM hostname. The existing JunosVM hostname is displayed. Type the new hostname and press **Enter**.

```
Please select a number to modify.
[<CR>=return to main menu]:
1
current junosvm hostname : northstar_junosvm
new junosvm hostname : junosvm_node1
```

3. Type **2** and press **Enter** to configure the JunosVM default gateway. The existing JunosVM default gateway IP address is displayed. Type the new IP address and press **Enter**.

```
Please select a number to modify.
[<CR>=return to main menu]:
2
current junosvm default_gateway :
new junosvm default_gateway : 10.25.152.1
```

4. Type **3** and press **Enter** to configure the JunosVM BGP AS number. The existing JunosVM BGP AS number is displayed. Type the new BGP AS number and press **Enter**.

```
Please select a number to modify.
[<CR>=return to main menu]:
3
current junosvm AS Number : 100
new junosvm AS Number: 100
```

5. Type **4A** and press **Enter** to configure the JunosVM interface #1 (external_interface). The first item of existing JunosVM interface #1 information is displayed. Type each item of new information (interface name, IPv4 address, netmask, type), and press **Enter** to proceed to the next.

NOTE: The designation of network or management for the type of interface is a label only, for your convenience. NorthStar Controller does not use this information.

```
Please select a number to modify.
[<CR>=return to main menu]:
4A
current junosvm interface1 name : em1
new junosvm interface1 name: em1

current junosvm interface1 ipv4 :
new junosvm interface1 ipv4 : 10.25.153.144

current junosvm interface1 netmask :
new junosvm interface1 netmask : 255.255.254.0

current junosvm interface1 type (network/management) : network
new junosvm interface1 type (network/management) : network

current junosvm interface1 bridge name : external0
new junosvm interface1 bridge name : external0
```

6. Type **B** and press **Enter** to add a JunosVM candidate static route. The existing JunosVM candidate static route (if any) is displayed. Type the new candidate static route and press **Enter**.

```
Please select a number to modify.
[<CR>=return to main menu]:
B
Candidate static route:
new static route (format: x.x.x.x/xy via a.b.c.d):
10.25.158.0/24 via 10.25.152.2
```

7. If you have more than one static route, type **B** and press **Enter** again to add each additional route.

```
Please select a number to modify.
[<CR>=return to main menu]:
B
Candidate static route:
[0] 10.25.158.0/24 via 10.25.152.2 dev any
```

```
new static route (format: x.x.x.x/xy via a.b.c.d):
10.25.159.0/24 via 10.25.152.2
```

NOTE: If you are adding a route and not making any other additional configuration changes, you can use option Y on the menu to apply the JunosVM static route only, without restarting the NorthStar services.

8. Type **Z** and press **Enter** to save your changes to the JunosVM configuration.

   The following example shows saving the JunosVM configuration.

```
Junos VM  Configuration Settings:
   ******************************************************
   In order to commit your changes you must select option Z
   ******************************************************
   ................................................
   1. ) JunosVM hostname                          : northstar_junosvm
   2. ) JunosVM default gateway                   :
   3. ) BGP AS number                             : 100
   4A.) JunosVM Interface #1 (external_interface)
            Name                                  : em1
            IPv4                                  :
            Netmask                               :
            Type(network/management)              : network
            Bridge name                           : external0
   4B.) Delete JunosVM Interface #1 (external_interface) data
   5A.) JunosVM Interface #2 (mgmt_interface)
            Name                                  : em2
            IPv4                                  :
            Netmask                               :
            Type(network/management)              : management
            Bridge name                           : mgmt0
   5B.) Delete JunosVM Interface #2 (mgmt_interface) data
   6A.) JunosVM Interface #3
            Name                                  :
            IPv4                                  :
            Netmask                               :
            Type(network/management)              : network
            Bridge name                           :
   6B.) Delete JunosVM Interface #3 data
   7A.) JunosVM Interface #4
            Name                                  :
```

```
            IPv4                                    :
            Netmask                                 :
            Type(network/management)                : network
            Bridge name                             :
   7B.) Delete JunosVM Interface #4 data
   8A.) JunosVM Interface #5
            Name                                    :
            IPv4                                    :
            Netmask                                 :
            Type(network/management)                : network
            Bridge name                             :
   8B.) Delete JunosVM Interface #5 data
   9. ) Show JunosVM current static route
   A. ) Show JunosVM candidate static route
   B. ) Add JunosVM candidate static route
   C. ) Remove JunosVM candidate static route
   ................................................
   X.) JunosVM current setting
   Y.) Apply JunosVM static route only
   Z.) Apply JunosVM Setting and static route
   ................................................

Please select a number to modify.
[<CR>=return to main menu]:
z
Are you sure you want to setup junosvm and static route configuration? (Y/N) y

Current junosvm network configuration:
junosvm current interface em0 IP: 10.16.16.2/255.255.255.0
junosvm current interface em1 IP: 10.25.153.144/255.255.254.0
junosvm current default gateway: 10.25.152.1
junosvm current asn: 100
Current junosvm static route:
[0] 10.25.158.0/24 via 10.25.152.2 dev any
[1] 10.25.159.0/24 via 10.25.152.2 dev any
Applying junosvm configuration ...
Please wait ...
Commit Success.
JunosVM has been configured successfully.
Please wait ... Backup Current JunosVM config ...

Connecting to JunosVM to backup the config ...
```

```
Please check the result at /opt/northstar/data/junosvm/junosvm.conf
JunosVm configuration has been successfully backed up
```

9. Press **Enter** to return to the Main Menu.

Configure Junos cRPD Settings

From the Setup Main Menu, configure the Junos cRPD settings. This section applies only to cRPD installations (not to installations that use Junos VM).

1. From the Main Menu, type **B** and press **Enter** to display the Junos cRPD Configuration menu:

```
Junos CRPD  Configuration Settings:
   ******************************************************
   In order to commit your changes you must select option Z
   ******************************************************
   ................................................
   1. ) BGP AS number                              : 65412
   2. ) BGP Monitor IPv4 Address                   : 172.25.153.154
   3. ) BGP Monitor Port                           : 10001
   ................................................
   X. ) Junos CRPD current setting
   Z. ) Apply Junos CRPD Setting
   ................................................

Please select a number to modify.
[<CR>=return to main menu]:
```

   To interact with this menu, type the number or letter corresponding to the item you want to add or change, and press **Enter**. Notice that option Y in the lower section is omitted from this menu as it is not relevant to cRPD.

2. Type **1** and press **Enter** to configure the BGP AS number. The existing AS number is displayed. Type the new number and press **Enter**.

```
Please select a number to modify.
[<CR>=return to main menu]:
1
current BGP AS Number : 65412
new BGP AS Number : 64525
```

3. Type **2** and press **Enter** if you need to change the default BGP Monitor IPv4 Address. By default, BMP monitor runs on the same host as cRPD, and the address is configured based on the local address of the host. We therefore recommend **not** changing this address.
4. Type **3** and press **Enter** if you need to change the default BGP Monitor Port. We recommend **not** changing this port from the default of 10001. The BMP monitor listens on port 10001 for incoming BMP connections from the network. The connection is opened from cRPD, which runs on the same host as the BMP monitor.
5. Type Z and press Enter to save your configuration changes. The following example show saving the Junos cRPD configuration.

```
Junos CRPD  Configuration Settings:
   ********************************************************
   In order to commit your changes you must select option Z
   ********************************************************
   ..............................................
   1. ) BGP AS number                           : 64525
   2. ) BGP Monitor IPv4 Address                : 172.17.153.154
   3. ) BGP Monitor Port                        : 10001
   ..............................................
   X. ) Junos CRPD current setting
   Z. ) Apply Junos CRPD Setting
   ..............................................

Please select a number to modify.
[<CR>=return to main menu]:
z
Are you sure you want to setup junos crpd configuration? (Y/N) y

Current junos crpd  configuration:
junos crpd current bgp asn: 64525
junos crpd current bmp_host: 172.17.153.154
junos crpd bgp_port: 10001
Please wait ...
Commit Success.
Junos CRPD has been configured successfully.
```

Set Up the SSH Key for External JunosVM

This section only applies to two-VM installations. Skip this section if you are installing NorthStar using cRPD.

NOTE: This step is not required if you are doing an upgrade rather than a fresh installation.

For a two-VM installation, you must set up the SSH key for the external JunosVM.

From the Main Menu, type **H** and press **Enter**.

```
Please select a number to modify.
[<CR>=return to main menu]:
H
```

Follow the prompts to provide your JunosVM username and router login class (super-user, for example). The script verifies your login credentials, downloads the JunosVM SSH key file, and returns you to the main menu.

For example:

```
Main Menu:
    ........................................
    A.) Host Setting
    ........................................
    B.) JunosVM Setting
    ........................................
    C.) Check Network Setting
    ........................................
    D.) Maintenance & Troubleshooting
    ........................................
    E.) HA Setting
    ........................................
    F.) Collect Trace/Log
    ........................................
    G.) Analytics Data Collector Setting
      (External standalone/cluster analytics server)
    ........................................
    H.) Setup SSH Key for external JunosVM  setup
    ........................................
    I.) Internal Analytics Setting (HA)
```

```
    ........................................
    X.) Exit
    ........................................
Please select a letter to execute.
H
Please provide JunosVM login:
admin

2 VMs Setup is detected

Script will create user: northstar. Please provide user northstar router login class e.g super-
user, operator:
super-user

The authenticity of host '10.49.118.181 (10.49.118.181)' can't be established.
RSA key fingerprint is xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx.
Are you sure you want to continue connecting (yes/no)? yes

Applying user northstar login configuration
Downloading JunosVM ssh key file. Login to JunosVM
Checking md5 sum. Login to JunosVM
SSH key has been sucessfully updated

Main Menu:
    ........................................
    A.) Host Setting
    ........................................
    B.) JunosVM Setting
    ........................................
    C.) Check Network Setting
    ........................................
    D.) Maintenance & Troubleshooting
    ........................................
    E.) HA Setting
    ........................................
    F.) Collect Trace/Log
    ........................................
    G.) Analytics Data Collector Setting
      (External standalone/cluster analytics server)
    ........................................
    H.) Setup SSH Key for external JunosVM  setup
    ........................................
    I.) Internal Analytics Setting (HA)
```

```
    ........................................
    X.) Exit
    ........................................
Please select a letter to execute.
    ........................................

Please select a letter to execute.
```

Upgrade the NorthStar Controller Software in an HA Environment

There are some special considerations for upgrading NorthStar Controller when you have an HA cluster configured. Use the following procedure:

1. Before installing the new release of the NorthStar software, ensure that all individual cluster members are working. On each node, execute the **supervisorctl status** script:

   ```
   [root@node-1]# supervisorctl status
   ```

   For an active node, all processes should be listed as RUNNING as shown in this example:

   This is just an example. The actual list of processes varies according to the version of NorthStar on the node, your deployment setup, and the optional features installed.

   ```
   [root@node-1 ~]# supervisorctl status
   bmp:bmpMonitor                   RUNNING   pid 2957, uptime 0:58:02
   collector:worker1                RUNNING   pid 19921, uptime 0:01:42
   collector:worker2                RUNNING   pid 19923, uptime 0:01:42
   collector:worker3                RUNNING   pid 19922, uptime 0:01:42
   collector:worker4                RUNNING   pid 19924, uptime 0:01:42
   collector:main:beat_scheduler    RUNNING   pid 19925, uptime 0:01:42
   collector_main:es_publisher      RUNNING   pid 19771, uptime 0:01:53
   collector_main:task_scheduler    RUNNING   pid 19772, uptime 0:01:53
   config:cmgd                      RUNNING   pid 22087, uptime 0:01:53
   config:cmgd-rest                 RUNNING   pid 22088, uptime 0:01:53
   docker:dockerd                   RUNNING   pid 4368, uptime 0:57:34
   epe:epeplanner                   RUNNING   pid 9047, uptime 0:50:34
   infra:cassandra                  RUNNING   pid 2971, uptime 0:58:02
   infra:ha_agent                   RUNNING   pid 9009, uptime 0:50:45
   infra:healthmonitor              RUNNING   pid 9172, uptime 0:49:40
   ```

```
infra:license_monitor            RUNNING   pid 2968, uptime 0:58:02
infra:prunedb                    RUNNING   pid 19770, uptime 0:01:53
infra:rabbitmq                   RUNNING   pid 7712, uptime 0:52:03
infra:redis_server               RUNNING   pid 2970, uptime 0:58:02
infra:zookeeper                  RUNNING   pid 2965, uptime 0:58:02
ipe:ipe_app                      RUNNING   pid 2956, uptime 0:58:02l
istener1:listener1_00            RUNNING   pid 9212, uptime 0:49:29
netconf:netconfd_00              RUNNING   pid 19768, uptime 0:01:53
northstar:anycastGrouper         RUNNING   pid 19762, uptime 0:01:53
northstar:configServer           RUNNING   pid 19767, uptime 0:01:53
northstar:mladapter              RUNNING   pid 19765, uptime 0:01:53
northstar:npat                   RUNNING   pid 19766, uptime 0:01:53
northstar:pceserver              RUNNING   pid 19441, uptime 0:02:59
northstar:privatet1vproxy        RUNNING   pid 19432, uptime 0:02:59
northstar:prpdclient             RUNNING   pid 19763, uptime 0:01:53
northstar:scheduler              RUNNING   pid 19764, uptime 0:01:53
northstar:topologyfilter         RUNNING   pid 19760, uptime 0:01:53
northstar:toposerver             RUNNING   pid 19762, uptime 0:01:53
northstar_pcs:PCServer           RUNNING   pid 19487, uptime 0:02:49
northstar_pcs:PCViewer           RUNNING   pid 19486, uptime 0:02:49
northstar_pcs:SRPCServer         RUNNING   pid 19490, uptime 0:02:49
web:app                          RUNNING   pid 19273, uptime 0:03:18
web:gui                          RUNNING   pid 19280, uptime 0:03:18
web:notification                 RUNNING   pid 19272, uptime 0:03:18
web:proxy                        RUNNING   pid 19275, uptime 0:03:18
web:restconf                     RUNNING   pid 19271, uptime 0:03:18
web:resthandler                  RUNNING   pid 19275, uptime 0:03:18
```

For a standby node, processes beginning with “northstar” and “northstar_pcs”should be listed as STOPPED. Also, if you have analytics installed, some of the processes beginning with “collector” are STOPPED. Other processes, including those needed to preserve connectivity, remain RUNNING. An example is shown here.

NOTE: This is just an example; the actual list of processes varies according to the version of NorthStar on the node, your deployment setup, and the optional features installed.

```
[root@node-1 ~]# supervisorctl status
bmp:bmpMonitor              RUNNING   pid 2957, uptime 0:58:02
collector:worker1                RUNNING   pid 19921, uptime 0:01:42
collector:worker2                RUNNING   pid 19923, uptime 0:01:42
collector:worker3                RUNNING   pid 19922, uptime 0:01:42
```

```
collector:worker4                RUNNING   pid 19924, uptime 0:01:42
collector:main:beat_scheduler    STOPPED   Dec 24, 05:12 AM
collector_main:es_publisher      STOPPED   Dec 24, 05:12 AM
collector_main:task_scheduler    STOPPED   Dec 24, 05:12 AM
config:cmgd                      STOPPED   Dec 24, 05:12 AM
config:cmgd-rest                 STOPPED   Dec 24, 05:12 AM
docker:dockerd                   RUNNING   pid 4368, uptime 0:57:34
epe:epeplanner                   RUNNING   pid 9047, uptime 0:50:34
infra:cassandra                  RUNNING   pid 2971, uptime 0:58:02
infra:ha_agent                   RUNNING   pid 9009, uptime 0:50:45
infra:healthmonitor              RUNNING   pid 9172, uptime 0:49:40
infra:license_monitor            RUNNING   pid 2968, uptime 0:58:02
infra:prunedb                    STOPPED   Dec 24, 05:12 AM
infra:rabbitmq                   RUNNING   pid 7712, uptime 0:52:03
infra:redis_server               RUNNING   pid 2970, uptime 0:58:02
infra:zookeeper                  RUNNING   pid 2965, uptime 0:58:02
ipe:ipe_app                      STOPPED   Dec 24, 05:12 AM
listener1:listener1_00           RUNNING   pid 9212, uptime 0:49:29
netconf:netconfd_00              RUNNING   pid 19768, uptime 0:01:53
northstar:anycastGrouper         STOPPED   Dec 24, 05:12 AM
northstar:configServer           STOPPED   Dec 24, 05:12 AM
northstar:mladapter              STOPPED   Dec 24, 05:12 AM
northstar:npat                   STOPPED   Dec 24, 05:12 AM
northstar:pceserver              STOPPED   Dec 24, 05:12 AM
northstar:privatet1vproxy        STOPPED   Dec 24, 05:12 AM
northstar:prpdclient             STOPPED   Dec 24, 05:12 AM
northstar:scheduler              STOPPED   Dec 24, 05:12 AM
northstar:topologyfilter         STOPPED   Dec 24, 05:12 AM
northstar:toposerver             STOPPED   Dec 24, 05:12 AM
northstar_pcs:PCServer           STOPPED   Dec 24, 05:12 AM
northstar_pcs:PCViewer           STOPPED   Dec 24, 05:12 AM
northstar_pcs:SRPCServer         STOPPED   Dec 24, 05:12 AM
web:app                          STOPPED   Dec 24, 05:12 AM
web:gui                          STOPPED   Dec 24, 05:12 AM
web:notification                 STOPPED   Dec 24, 05:12 AM
web:proxy                        STOPPED   Dec 24, 05:12 AM
web:restconf                     STOPPED   Dec 24, 05:12 AM
web:resthandler                  STOPPED   Dec 24, 05:12 AM
```

2. Ensure that the SSH keys for HA are set up. To test this, try to SSH from each node to every other node in the cluster using user “root”. If the SSH keys for HA are set up, you will not be prompted for a password. If you are prompted for a password, see "Configuring a NorthStar Cluster for High Availability" on page 161 for the procedure to set up the SSH keys.

3. On one of the standby nodes, install the new release of the NorthStar software according to the instructions at the beginning of this topic. Check the processes on this node before proceeding to the other standby node(s) by executing the **supervisorctl status** script.

```
[root@node-1]# supervisorctl status
```

   Since the node comes up as a standby node, some processes will be STOPPED, but the “infra” group of processes, the “listener1” process, the “collector:worker” group of processes (if you have them), and the “junos:junosvm” process (if you have it) should be RUNNING. Wait until those processes are running before proceeding to the next node.

4. Repeat this process on each of the remaining standby nodes, one by one, until all *standby* nodes have been upgraded.
5. On the active node, restart the ha-agent process to trigger a switchover to a standby node.

```
[root@node-2]# supervisorctl restart infra:ha_agent
```

   One of the standby nodes becomes active and the previously active node switches to standby mode.

6. On the previously active node, install the new release of the NorthStar software according to the instructions at the beginning of this section. Check the processes in this node using **supervisorctl status**; their status (RUNNING or STOPPED) should be consistent with the node’s new standby role.

> **NOTE**: The newly upgraded software automatically inherits the net_setup settings, HA configurations, and all credentials from the previous installation. Therefore, it is not necessary to re-run net_setup unless you want to change settings, HA configurations, or password credentials.

RELATED DOCUMENTATION

- NorthStar Controller System Requirements | 6
- Installing Data Collectors for Analytics | 113
- Configuring a NorthStar Cluster for High Availability | 161
- Uninstalling the NorthStar Controller Application | 87
- Using a Remote Server for NorthStar Planner | 182

Configuring NorthStar Settings Using the NorthStar CLI

IN THIS SECTION

Beginning with NorthStar Release 6.1.0, component-specific and service-specific NorthStar settings previously maintained in the **northstar.cfg** file are maintained in an internal cache and are configurable using the NorthStar CLI. The NorthStar CLI is very similar to the Junos OS CLI.

NOTE: Certain bootstrap and infrastructure configuration settings continue to be maintained in the northstar.cfg file.

If you are not already familiar with the Junos OS CLI, see Junos OS CLI User Guide which covers:

- Accessing operational and configuration command modes, and switching between modes
- The concept of command hierarchies
- Navigation among hierarchy levels
- Getting help on command syntax including how to display all possible completions for a partial command
- Helpful keyboard sequences including command completion shortcuts
- Committing or backing out of configuration changes

Accessing the NorthStar CLI

To access the NorthStar CLI:

1. After you successfully install NorthStar, log in to the NorthStar application server.

2. Launch the NorthStar CLI:

```
[root@ns]# /opt/northstar/utils/cmgd_cli
root@ns>
```

3. The > prompt indicates you are in operational mode. In this mode, you can display the NorthStar configuration, but you cannot change it. For example:

```
root@ns> show configuration northstar

config-server {
    health-monitor {
        heartbeat-interval 5s;
        poll-interval 10m;
        history-ttl 2d;
        heartbeat-holdown-timer 15s;
    }
}
path-computation-server {
    health-monitor {
        heartbeat-interval 5s;
        poll-interval 10m;
        history-ttl 2d;
        heartbeat-holdown-timer 15s;
    }
}
netconfd {
    in-memory-datastore {
        reconnect-delay 1000;
        reconnect-retries 10000;
    }
}
programmable-rpd-client {
    health-monitor {
        heartbeat-interval 5s;
        poll-interval 10m;
        history-ttl 2d;
        heartbeat-holdown-timer 15s;
    }
    enable-top-prefix-filter;
        publish-top-prefix-only;
```

```
}

root@ns1>
```

4. Use the **edit** command to enter configuration mode (prompt changes to #):

```
root@ns> edit
Entering configuration mode
Users currently editing the configuration:
  root terminal pts/1 (pid 246) on since 2020-08-25 17:26:47 UTC, idle 1d 23:54
      [edit]
  root terminal pts/2 (pid 262) on since 2020-08-26 23:05:31 UTC, idle 17:13:12
      [edit]

[edit]
root@ns1#
```

   [edit] indicates the top of the command hierarchy.

5. All NorthStar configuration commands begin with **set northstar**. Enter **set northstar** with a question mark to display the NorthStar configuration command top level categories:

```
root@ns1# set northstar ?

Possible completions:
> analytics            General configuration parameters related to analytics
> config-server        Config Server run time parameters
> mladapter            General configuration parameters related to ML Adapter. Common
                       configuration parameters like amqp or database are taken from
                       amqpSettings, but can be overridden for MLAdapter.
> netconfd             General configuration parameters related to netconfd
> path-computation-server  Path computation server run time parameters
> peer-engineering     General configuration parameters for IPE
> programmable-rpd-client  General configuration parameters related to the PRPD client
> system
> topology-server      General configuration parameters related to the Topology Server. Common
                       configuration parameters like amqp or database are taken from
                       amqpSettings, but can be overridden for the Topology Server.

[edit]
root@ns1# set northstar
```

6. Continue with any category and a question mark to see the next level breakdown. For example:

```
root@ns1# set northstar config-server ?

Possible completions:
> health-monitor          Configuration parameters related to Health Monitor
+ include-interface-type  The interfaces to be published by Config Server
                          Space-separated list enclosed in [ ] or single interface type with
no brackets. Indicates
                          discovered interface types to be added to NorthStar.
                          The following interface types are supported:

                          - physical     Physical interfaces : interface name without dot (.)
in it
                          - loopback-mgmt Loopback and management interfaces :  interface
name starting with lo, fxp, me and em
                          - vrf-if        Interfaces associated with VRF
                          - links-if      Interfaces on links
                          - all          all interfaces
> log-destination         List of logging configuration
  publish-aslink          Enable ConfigServer to publish aslink created by getipconf to the
Northstar model

[edit]
root@ns1# set northstar config-server
```

7. Continue in this fashion to reach a configuration setting and, if the command requires it, specify the value you wish to change. For example:

```
root@ns1# set northstar config-server include-interface-type ?
Possible completions:
  [                    Open a set of values
  all
  links-if
  loopback-mgmt
  physical
  vrf-if
[edit]

root@ns1# set northstar config-server include-interface-type [links-if physical]
```

```
[edit]
root@ns1#
```

8. Once you are familiar with the command hierarchy, you can navigate directly to a different level once you are in configuration mode. For example:

```
root@ns> edit
Entering configuration mode
Users currently editing the configuration:
  root terminal pts/0 (pid 162) on since 2020-09-05 17:08:10 UTC, idle 1d 07:59
      [edit]

[edit]
root@ns# edit northstar system health-monitor

[edit northstar system health-monitor]
root@ns# set ?
Possible completions:
  heartbeat-holdown-timer  Health monitor holdown timer. Can be expressed as seconds ('s' or
                       'seconds'). Examples: 30s, 30seconds. (default=15s)
  heartbeat-interval   Health monitoring heartbeat interval. Can be expressed as seconds ('s'
                       or 'seconds'). Examples: 10s, 10seconds. (0 or -ve value = disabled,
                       default=5s)
  history-ttl          Health monitor history retention. Can be expressed as days ('d' or
                       'days'). Examples: 7d, 7days. (default=2d)
  poll-interval        Health monitor poll interval. Can be expressed as minutes ('m' or
                       'minutes'). Examples: 4m, 4minutes. (default=10m)

[edit northstar system health-monitor]
```

NorthStar Configuration Settings

Table 9 on page 83 lists the NorthStar configuration settings most likely to require modification. It is not a complete listing of all available settings.

Table 9: Frequently Used NorthStar Configuration Settings

Setting	Command	Description
port (NETCONF)	**set northstar netconfd device-connection-pool netconf port**	Use this command to change the default port for NETCONFD from 830. In some installations, port 22 is preferred.
aggregate-by-prefix (Netflow collector)	**set northstar analytics netflowd aggregate-by-prefix**	By default, Netflow aggregates traffic by PE, but for some applications (such IPE), you would want the traffic aggregated by prefix. If the parameter is enabled, Netflow aggregates all traffic from a specific ingress PE router to a specific destination (prefix) within the defined period of time.
enable-ssl (Netflow collector)	**set northstar analytics netflowd enable-ssl**	Enables Secure Socket Layer (SSL). See Netflow Collector in the *NorthStar Controller User Guide* for more information.
level (Netflow collector)	**set northstar analytics netflowd log-destination *destination-name* level**	Level of information that is captured in the log file. See Netflow Collector in the *NorthStar Controller User Guide* for more information.
default-sampling-interval (Netflow collector)	**set northstar analytics netflowd default-sampling-interval**	The default sampling interval, if the router does not provide the interval in the Template FlowSet. See Netflow Collector in the *NorthStar Controller User Guide* for more information.

Table 9: Frequently Used NorthStar Configuration Settings *(Continued)*

Setting	Command	Description
publish-interval (Netflow collector)	**set northstar analytics netflowd publish-interval**	Publishing interval to both Elasticsearch and the PCS. Traffic is aggregated per publishing interval. See Netflow Collector in the *NorthStar Controller User Guide* for more information.
workers (Netflow collector)	**set northstar analytics netflowd workers**	Number of worker processes to start. See Netflow Collector in the *NorthStar Controller User Guide* for more information. See "Secondary Collector Installation for Distributed Data Collection" on page 158 for more information about distributed data collection.
notify-final-bandwidth- on-inactive-flow (Netflow collector)	**set northstar analytics netflowd notify-final-bandwidth-on-inactive-flow**	If enabled, netflowd sends one final update after a flow is no longer active, reporting the bandwidth as 0. See Netflow Collector in the *NorthStar Controller User Guide* for more information.
stats-interval (Netflow collector)	**set northstar analytics netflowd stats-interval**	Interval at which statistics are printed to the log file. See Netflow Collector in the *NorthStar Controller User Guide* for more information.
generate-as-demands (Netflow collector)	**set northstar analytics netflowd generate-as-demands**	Enable the generation of AS demands by netflowd. See Netflow Collector in the *NorthStar Controller User Guide* for more information.

Table 9: Frequently Used NorthStar Configuration Settings *(Continued)*

Setting	Command	Description
polling-interval (Multilayer)	**set northstar mladapter polling-interval**	Controls the polling interval for interfaces without notification support. See *Configuring the Multilayer Feature* in the *NorthStar Controller User Guide* for more information.
interval (Collection-cleanup)	**set northstar system scheduler tasks collection-cleanup interval**	Controls how often the collection-cleanup system task is run, in number of days. See NorthStar Analytics Raw and Aggregated Data Retention in the *NorthStar Controller User Guide* for more information.
raw-data-retention-duration (Collection-cleanup)	**set northstar system scheduler tasks collection-cleanup raw-data-retention-duration**	Defines what is considered an “old” log of raw data in number of days. See NorthStar Analytics Raw and Aggregated Data Retention in the *NorthStar Controller User Guide* for more information.
rollup-data-retention-duration (Collection-cleanup)	**set northstar system scheduler tasks collection-cleanup rollup-data-retention-duration**	Defines what is considered “old” aggregated data in number of days. See NorthStar Analytics Raw and Aggregated Data Retention in the *NorthStar Controller User Guide* for more information.
interval (Rollup)	**set northstar system scheduler tasks rollup interval**	Sets how often the ESRollup system task is run in number of hours. See NorthStar Analytics Raw and Aggregated Data Retention in the *NorthStar Controller User Guide* for more information.

Table 9: Frequently Used NorthStar Configuration Settings *(Continued)*

Setting	Command	Description
lsp-latency-interval (Analytics)	**set northstar path-computation-server lsp-latency-interval**	Enables PCViewer to calculate LSP delay and display the data in the web UI. See Viewing Analytics Data in the Web UI in the *NorthStar Controller User Guide* for more information.
include-interface-type (all tasks related to interfaces)	**set northstar config-server include-interface-type**	Interface types that can be discovered on devices and that are to be used by traffic collection. The supported interface types are: • physical: Physical interfaces, expressed as the interface name without a dot (.) in it • loopback-mgmt: Loopback and management interfaces expressed as the interface name starting with lo, fxp, me, or em • vrf-if: Interfaces associated with a VRF • links-if: Interfaces on links • all: All interfaces **NOTE**: configServer publishes to all components only the interface types that you specify. The web UI and data collection only receive information about interfaces representing those interface types. If you modify this setting, deselecting interface types that are already represented by interfaces in the NorthStar model, those existing interfaces remain in the model.

Uninstalling the NorthStar Controller Application

IN THIS SECTION

You can uninstall the NorthStar Controller application using the supplied uninstall script. One use case for uninstalling is to revert back to a previous version of NorthStar after testing a new version.

The following sections provide the steps to follow.

Uninstall the NorthStar Software

Use the following procedure to uninstall NorthStar:

1. Preserve your license file by copying it to the root directory:

```
cp -prv/u/wandl/db/sys/npatpw /root/
```

> **NOTE**: You can also preserve any other important user or configuration data you have on the server using the same method.

2. Navigate to the NorthStar bundle directory:

```
cd /opt/northstar/northstar_bundle_x_x_x
```

3. Run the uninstall script:

```
./uninstall_all.sh
```

4. When prompted, confirm that you want to uninstall NorthStar.

Reinstate the License File

After you have reinstalled the NorthStar application, use the following procedure to reinstate the license file that you copied to the root directory:

1. Copy the license file from the root directory back to its original directory:

```
cp -prv/root/npatpw /u/wandl/db/sys/
```

> **NOTE**: You can also restore any other data preserved in the root directory by copying it back to its original directory.

2. Change the user and group ownership to pcs. This is likely unnecessary if you used -prv (preserve) in the copy command.

```
chown pcs:pcs /u/wandl/db/sys/npatpw
```

CHAPTER 3

Installation in an OpenStack Environment

Overview of NorthStar Controller Installation in an OpenStack Environment

IN THIS SECTION

The NorthStar Controller can be installed in an OpenStack environment in either standalone or cluster mode. Figure 7 on page 91 illustrates standalone mode. Figure 8 on page 92 illustrates cluster mode. Note that in both cases, each node has one NorthStar Controller application VM and one JunosVM.

Figure 7: OpenStack Environment, Standalone Mode

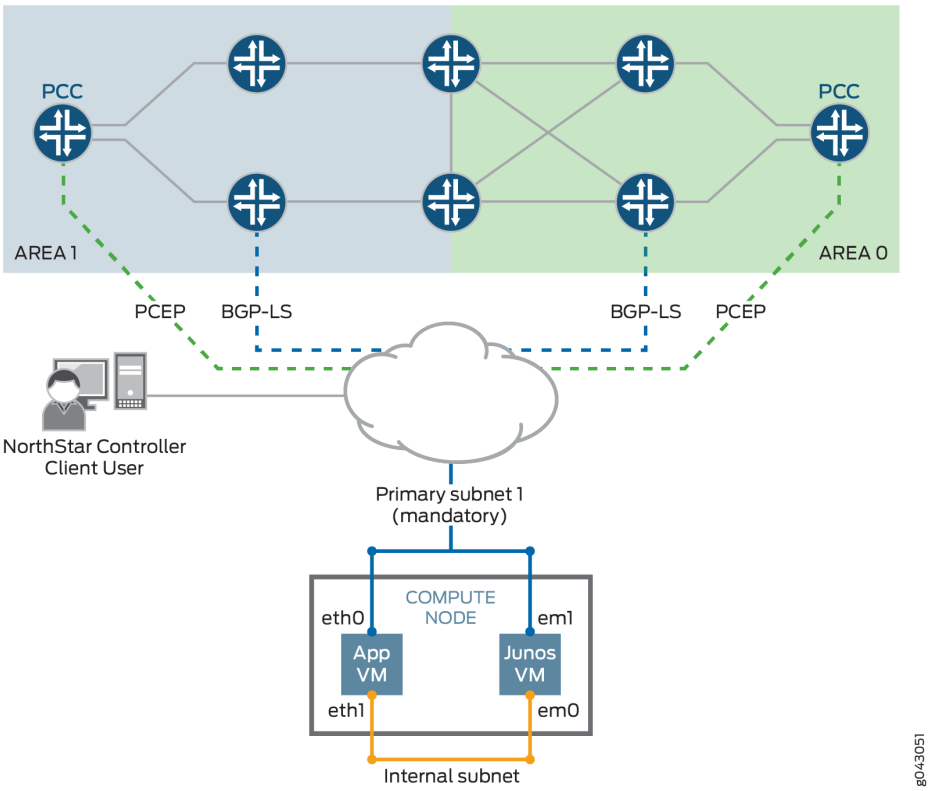
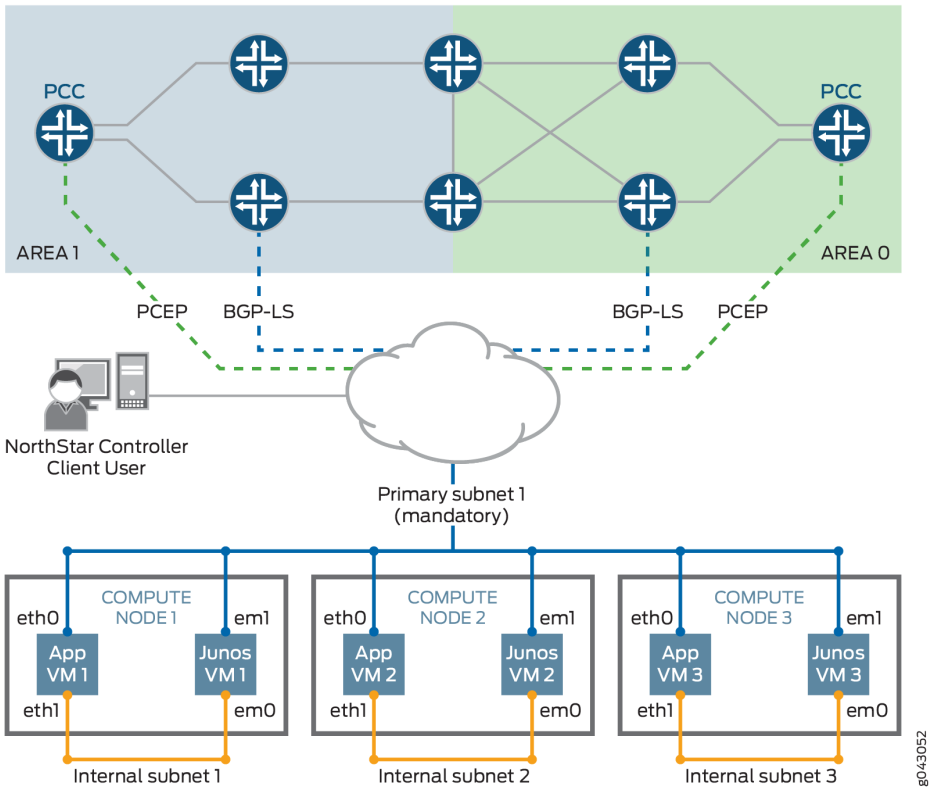


Figure 8: OpenStack Environment, Cluster Mode



Networking Scenarios

There are two common networking scenarios for using VMs on OpenStack:

- The VM is connected to a private network, and it uses a floating IP address to communicate with the external network.

  A limitation to this scenario is that direct OSPF or IS-IS adjacency does not work behind NAT. You should, therefore, use BGP-LS between the JunosVM and the network devices for topology acquisition.

- The VM is connected or bridged directly to the provider network (flat networking).

  In some deployments, a VM with flat networking is not able to access OpenStack metadata services. In that case, the official CentOS cloud image used for the NorthStar Controller application VM cannot install the SSH key or post-launch script, and you might not be able to access the VM.

  One workaround is to access metadata services from outside the DHCP namespace using the following procedure:

CAUTION: This procedure interrupts traffic on the OpenStack system. We recommend that you consult with your OpenStack administrator before proceeding.

1. Edit the /etc/neutron/dhcp_agent.ini file to change “enable_isolated_metadata = False” to “enable_isolated_metadata = True”.
2. Stop all neutron agents on the network node.
3. Stop any dnsmasq processes on network node or on the node that serves the flat network subnet.
4. Restart all neutron agents on the network node.

HEAT Templates

The following HEAT templates are provided with the NorthStar Controller software:

- northstar310.heat (standalone installation) and northstar310.3instances.heat (cluster installation)

  These templates can be appropriate when the NorthStar Controller application VM and the JunosVM are to be connected to a virtual network that is directly accessible from outside OpenStack, without requiring NAT. Typical scenarios include a VM that uses flat networking, or an existing OpenStack system that uses Contrail as the Neutron plugin, advertising the VM subnet to the MX Series Gateway device.

- northstar310.floating.heat (standalone installation) and northstar310.3instances.floating.heat (cluster installation)

  These templates can be appropriate if the NorthStar Controller application VM and the JunosVM are to be connected to a private network behind NAT, and require a floating IP address for one-to-one NAT.

We recommend that you begin with a HEAT template rather than manually creating and configuring all of your resources from scratch. You might still need to modify the template to suit your individual environment.

HEAT Template Input Values

The provided HEAT templates require the input values described in Table 10 on page 94.

Table 10: HEAT Template Input Values

Parameter	Default	Notes
customer_name	(empty)	User-selected name to identify the NorthStar stack
app_image	CentOS-6-x86_64-GenericCloud.qcow2	Modify this variable with the Centos 6 cloud image name that is available in Glance
junosvm_image	northstar-junosvm	Modify this variable with the JunosVM image name that is available in Glance
app_flavor	m1.large	Instance flavor for the NorthStar Controller VM with a minimum 40 GB disk and 8 GB RAM
junosvm_flavor	m1.small	Instance flavor for the JunosVM with a minimum of a 20 GB disk and 2GB of RAM
public_network	(empty)	UUID of the public-facing network, mainly for managing the server
asn	11	AS number of the backbone routers for BGP-LS peering
rootpassword	northstar	Root password
availability_zone	nova	Availability zone for spawning the VMs
key_name	(empty)	Your ssh-key must be uploaded in advance

Known Limitations

The following limitations apply to installing and using the NorthStar Controller in a virtualized environment.

Virtual IP Limitations from ARP Proxy Being Enabled

In some OpenStack implementations, ARP proxy is enabled, so virtual switch forwarding tables are not able to learn packet destinations (no ARP snooping). Instead, ARP learning is based on the hypervisor configuration.

This can prevent the virtual switch from learning that the virtual IP address has been moved to a new active node as a result of a high availability (HA) switchover.

There is currently no workaround for this issue other than disabling ARP proxy on the network where the NorthStar VM is connected. This is not always possible or allowed.

Hostname Changes if DHCP is Used Rather than a Static IP Address

If you are using DHCP to assign IP addresses for the NorthStar application VM (or NorthStar on a physical server), you should never change the hostname manually.

Also if you are using DHCP, you should not use net_setup.py for host configuration.

Disk Resizing Limitations

OpenStack with cloud-init support is supposed to resize the VM disk image according to the version you select. Unfortunately, the CentOS 6 official cloud image does not auto-resize due to an issue within the cloud-init agent inside the VM.

The only known workaround at this time is to manually resize the partition to match the allocated disk size after the VM is booted for the first time. A helper script for resizing the disk (**/opt/northstar/utils/resize_vm.sh**) is included as part of the NorthStar Controller RPM bundle.

RELATED DOCUMENTATION

- OpenStack Resources for NorthStar Controller Installation | **96**
- NorthStar Controller in an OpenStack Environment Pre-Installation Steps | **97**
- Installing the NorthStar Controller in Standalone Mode Using a HEAT Template | **98**
- Installing a NorthStar Cluster Using a HEAT Template | **104**

OpenStack Resources for NorthStar Controller Installation

Table 11 on page 96 and Table 12 on page 96 describe the required and optional OpenStack resources for running the NorthStar Controller in an OpenStack environment.

Table 11: Required OpenStack Resources

Resource	Description
OS::Nova::Server	Two of these resources are required: one for the NorthStar Controller application VM and one for the JunosVM.
OS::Neutron::Port	At least two of these resources are required for the Ethernet connections of each OS:Nova::Server resource.
OS::Neutron::Net	Each NorthStar installation requires one of this resource for internal communication between the NorthStar Controller application VM and the JunosVM. Connection to an existing OS::Neutron::Net resource for public network connectivity is also required.
OS::Neutron::Subnet	A fixed 172.16.16.0/24 subnet is required for internal communication between the NorthStar Controller application VM and the JunosVM.

Table 12: Optional OpenStack Resources

Resource	Description
OS::Neutron::SecurityGroup	Use this resource (either new or existing) to access the NorthStar Controller application VM and JunosVM from outside OpenStack.
OS::Neutron::FloatingIP	Use this resource if the NorthStar Controller application VM and JunosVM are connected to a virtual private network behind NAT. This resource is not usually necessary in a flat networking scenario or a private network using Contrail.
OS::Nova::ServerGroup	Use this resource with an anti-affinity rule to ensure that no more than one NorthStar Controller application VM, or no more than one JunosVM are spawned in the same compute node. This is for additional redundancy purposes.

Table 12: Optional OpenStack Resources *(Continued)*

Resource	Description
OS::Neutron::Port for VIP	Use an additional OS::Neutron::Port for cluster setup, to provide a virtual IP address for the client facing connection.

RELATED DOCUMENTATION

Overview of NorthStar Controller Installation in an OpenStack Environment | 90

NorthStar Controller in an OpenStack Environment Pre-Installation Steps

Before you install the NorthStar Controller in an OpenStack environment, prepare your system by performing the following pre-installation steps.

1. (Optional) Upload an SSH keypair.

   ```
   # nova keypair-add --pub-key ssh-public-key-file keypair-name
   ```

   Alternatively, you can use any existing keypair that is available in your OpenStack system. You can also use Horizon UI to upload the image. Consult your OpenStack user guide for more information about creating, importing, and using keypairs.

2. Upload an official CentOS 6 Cloud image.

   ```
   # glance image-create --name glance-centos-image-name  --disk-format qcow2 --container-format bare --file image-location-and-filename-to-upload
   ```

   For example:

   ```
   # glance image-create --name northstar_junosvm_17.2R1.openstack.qcow2 --disk-format qcow2 --container-format bare --file images/northstar_junosvm_17.2R1.openstack.qcow2
   ```

3. Change the JunosVM disk bus type to IDE and the Ethernet driver to e1000.

```
# glance image-update --property hw_disk_bus=ide --property hw_cdrom_bus=ide --property
hw_vif_model=e1000 junosvm-image-id
```

NOTE: The variable *junosvm-image-id* is the UUID of the JunosVM image. You can find this ID in the output of the following command:

```
# glance image-list
```

RELATED DOCUMENTATION

Overview of NorthStar Controller Installation in an OpenStack Environment | 90

OpenStack Resources for NorthStar Controller Installation | 96

Installing the NorthStar Controller in Standalone Mode Using a HEAT Template

IN THIS SECTION

This topic describes installing a standalone NorthStar Controller in an OpenStack environment using a HEAT template. These instructions assume you are using one of the provided HEAT templates.

Launch the Stack

Perform the following steps to launch the stack.

1. Create a stack from the HEAT template file using the **heat stack-create** command.

```
# heat stack-create stack-name -f heat-template-name --parameters
customer_name=instance-name;app_image=centos6-image-name;junosvm_image=
junosvm-image-name;public_network=public-network-uuid;key_name=
keypair-name;app_flavor=app-vm-flavor;junosvm_flavor=junosvm-flavor
```

Obtain the Stack Attributes

1. Ensure that the stack creation is complete by examining the output of the **heat stack-show** command.

```
# heat stack-show stack-name | grep stack_status
```

2. Obtain the UUID of the NorthStar Controller VM and the JunosVM instances by executing the **resource-list** command.

```
# heat resource-list stack-name | grep ::Server
```

3. Using the UUIDs obtained from the **resource-list** command output, obtain the associated IP addresses by executing the **interface-list** command for each UUID.

```
# nova interface-list uuid
```

4. Once the NorthStar Controller VM finishes its booting process, you should be able to ping its public IP address.

NOTE: You can use the **nova console-log** command to monitor the booting status.

At this point, the NorthStar Controller VM is remotely accessible, but the JunosVM is not because it does not support DHCP. Once the NorthStar Controller RPM bundle installation is completed, the JunosVM can be remotely accessed.

5. Connect to the NorthStar Controller VM using SSH.

   If you are using a different SSH key from the one that is defined in the HEAT template, the default credentials are root/northstar and centos/northstar.

Resize the Image

The CentOS 6 official cloud image does not resize correctly for the selected OpenStack flavor. This results in the NorthStar Controller VM filesystem size being set at 8G instead of the size that is actually specified by the flavor. Using the following procedure, you can adjust your filesystem to be in sync with the allocated disk size. Alternatively, you can hold off on the resizing procedure until after you complete the NorthStar Controller RPM bundle installation. There is a resize-vm script inside /opt/northstar/utils/.

CAUTION: The **fdisk** command can have undesirable effects if used inappropriately. We recommend that you consult with your system administrator before proceeding with this workaround, especially if you are unfamiliar with the **fdisk** command.

1. Determine whether the size of the VM is correct. If it is correct, you do not need to proceed with resizing.

```
# ssh centos@App_Public_IPv4
Warning: Permanently added '172.25.158.161' (RSA) to the list of known hosts.

[centos@app_instance ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/vda1       7.8G  646M  6.8G   9% /
tmpfs           1.9G     0  1.9G   0% /dev/shm
```

2. Use the **fdisk** command to recreate the partition.

```
# ssh centos@App_Public_IPv4
Warning: Permanently added '172.25.158.161' (RSA) to the list of known hosts.

[user@demo-northstar-app centos]# fdisk /dev/vda
```

```
WARNING: DOS-compatible mode is deprecated. It's strongly recommended to
        switch off the mode (command 'c') and change display units to
        sectors (command 'u').

Command (m for help): c
DOS Compatibility flag is not set

Command (m for help): u
Changing display/entry units to sectors

Command (m for help): p

Disk /dev/vda: 85.9 GB, 85899345920 bytes
255 heads, 63 sectors/track, 10443 cylinders, total 167772160 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00050c05

Device Boot      Start         End      Blocks   Id  System
/dev/vda1   *        2048    16777215     8387584   83  Linux

Command (m for help): d
Selected partition 1

Command (m for help): n
Command action
e   extended
p   primary partition (1-4)
p
Partition number (1-4): 1
First sector (2048-167772159, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-167772159, default 167772159):
Using default value 167772159

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.

WARNING: Re-reading the partition table failed with error 16: Device or resource busy.
The kernel still uses the old table. The new table will be used at
```

```
the next reboot or after you run partprobe(8) or kpartx(8)
Syncing disks.
[user@demo-northstar-app centos]#
```

3. Reboot the VM to apply the partition changes.

```
[user@app_instance centos]# reboot

Broadcast message from centos@app_instance
    (/dev/pts/0) at 14:54 ...

The system is going down for reboot NOW!
```

4. Wait until the NorthStar Controller VM has returned to an up state.
5. Reconnect to the VM using SSH.
6. Check the partition size again to verify that the partition was resized.
7. If the partition size is still incorrect, use the **resize2fs** command to adjust the filesystem.

```
# resize2fs /dev/vda1
```

Install the NorthStar Controller RPM Bundle

Install the NorthStar Controller RPM bundle for an OpenStack environment as described in *Installing the NorthStar Controller*. The procedure uses the **rpm** and **install-vm.sh** commands.

Configure the JunosVM

For security reasons, the JunosVM does not come with a default configuration. Use the following procedure to manually configure the JunosVM using the OpenStack novnc client.

1. Obtain the novnc client URL.

```
# nova get-vnc-console JunosVM-ID novnc
```

2. Configure the JunosVM as you would in a fresh install of the Junos OS.
3. Copy the root user of the NorthStar Controller VM SSH public key to the JunosVM. This allows configuration from the NorthStar Controller VM to the JunosVM using an ssh-key based connection.
4. On the NorthStar Controller VM, run the net_setup.py script, and select option B to complete the configuration of the JunosVM. Once complete, you should be able to remotely ping the JunosVM IP address.

Configure SSH Key Exchange

Use the following procedure to configure SSH key exchange between the NorthStar Controller VM and the JunosVM.

1. Log in to the NorthStar Controller server and display the contents of the id_rsa.pub file by executing the **concatenate** command.

```
$cat /opt/pcs/.ssh/id_rsa.pub
```

   You will need the ssh-rsa string from the output.

2. Log in to the JunosVM and replace the ssh-rsa string with the one from the id_rsa.pub file by executing the following commands.

```
ssh northstar@JunosVM-ip
configure
set system login user northstar authenication ssh-rsa replacement-string
commit
exit
```

3. On the NorthStar Controller server, update the known hosts file by executing the following commands.

```
$su - pcs
$ssh -o UserKnownHostsFile=/opt/pcs/.ssh/known_hosts -i /opt/pcs/.ssh/id_rsa
northstar@JunosVM-ip
exit
exit
```

RELATED DOCUMENTATION

Installing a NorthStar Cluster Using a HEAT Template | 104

Installing a NorthStar Cluster Using a HEAT Template

IN THIS SECTION

This topic describes installing a NorthStar cluster in an OpenStack environment using a HEAT template. These instructions assume that you are using one of the provided HEAT templates.

System Requirements

In addition to the system requirements for installing the NorthStar Controller in a two-VM environment, a cluster installation also requires that:

- An individual compute node is hosting only one NorthStar Controller VM and one JunosVM. You can ensure this by launching the NorthStar Controller VM into a specific availability zone and compute node, or by using a host affinity such as OS::Nova::ServerGroup with an anti-affinity rule.

- The cluster has a single virtual IP address for the client facing connection. If promiscuous mode is disabled in OpenStack (blocking the virtual IP address), you can use the Neutron::Port allowed-address-pair attribute to permit the additional address.

Launch the Stack

Create a stack from the HEAT template file using the **heat stack-create** command.

```
# heat stack-create stack-name -f heat-template-name --parameters
customer_name=instance-name;app_image=centos6-image-name;junosvm_image=
junosvm-image-name;public_network=public-network-uuid;key_name=
keypair-name;app_flavor=app-vm-flavor;junosvm_flavor=junosvm-flavor
```

Obtain the Stack Attributes

1. Ensure that the stack creation is complete by examining the output of the **heat stack-show** command.

   ```
   # heat stack-show stack-name | grep stack_status
   ```

2. Obtain the UUID of the NorthStar Controller VM and the JunosVM instances for each node in the cluster by executing the **resource-list** command.

   ```
   # heat resource-list stack-name | grep ::Server
   ```

3. Using the UUIDs obtained from the **resource-list** command output, obtain the associated IP addresses by executing the **interface-list** command for each UUID.

   ```
   # nova interface-list uuid
   ```

4. Verify that each compute node in the cluster has only one NorthStar Controller VM and only one JunosVM by executing the following command for each UUID:

```
# nova show uuid | grep hypervisor
```

Configure the Virtual IP Address

1. Find the UUID of the virtual IP port that is defined in the HEAT template by examining the output of the **heat resource-list** command.

```
# heat resource-list stack-name | grep vip_port
```

2. Find the assigned virtual IP address for that UUID by examining the output of the **neutron port-show** command.

```
# neutron port-show vip-port-uuid
```

3. Find the UUID of each public-facing NorthStar Controller port by examining the output of the **neutron port-list** command.

```
# neutron port-list | grep stack-name-app_port_eth0
```

For example:

```
# neutron port-list | grep northstarHAexample-app_port_eth0
```

4. Update each public-facing NorthStar Controller port to accept the virtual IP address by executing the **neutron port-update** command for each port.

```
# neutron port-update vip-port-uuid --allowed_address_pairs list=true type=dict
ip_address=vip-ip
```

For example:

```
# neutron port-update a15578e2-b9fb-405c-b4c4-1792f5207003  --allowed_address_pairs list=true
type=dict ip_address=172.25.158.139
```

5. Wait until each NorthStar Controller VM finishes its booting process, at which time, you should be able to ping its public IP address. You can also use the **nova console-log** command to monitor the booting status of the NorthStar Controller VM.

Resize the Image

The CentOS 6 official cloud image does not resize correctly for the selected OpenStack flavor. This results in the NorthStar Controller VM filesystem size being set at 8G instead of the size that is actually specified by the flavor. Using the following procedure, you can adjust your filesystem to be in sync with the allocated disk size. Alternatively, you can hold off on the resizing procedure until after you complete the NorthStar RPM bundle installation. There is a resize-vm script inside /opt/northstar/utils/.

CAUTION: The **fdisk** command can have undesirable effects if used inappropriately. We recommend that you consult with your system administrator before proceeding with this workaround, especially if you are unfamiliar with the **fdisk** command.

Use the following procedure for each NorthStar Controller VM. Replace *XX* in the commands with the number of the VM (01, 02, 03, and so on).

1. Determine whether the size of the VM is correct. If it is correct, you do not need to proceed with the resizing.

```
# ssh centos@App_XX_Public_IPv4
Warning: Permanently added '172.25.158.161' (RSA) to the list of known hosts.

[centos@app_instance_XX ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/vda1       7.8G  646M  6.8G   9% /
tmpfs           1.9G     0  1.9G   0% /dev/shm
```

2. Use the **fdisk** command to recreate the partition.

```
# ssh centos@App_XX_Public_IPv4
Warning: Permanently added '172.25.158.161' (RSA) to the list of known hosts.

[user@demo-northstar-app centos]# fdisk /dev/vda

WARNING: DOS-compatible mode is deprecated. It's strongly recommended to
        switch off the mode (command 'c') and change display units to
        sectors (command 'u').

Command (m for help): c
DOS Compatibility flag is not set

Command (m for help): u
Changing display/entry units to sectors

Command (m for help): p

Disk /dev/vda: 85.9 GB, 85899345920 bytes
255 heads, 63 sectors/track, 10443 cylinders, total 167772160 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00050c05

Device Boot      Start         End      Blocks   Id  System
/dev/vda1   *        2048    16777215     8387584   83  Linux

Command (m for help): d
Selected partition 1

Command (m for help): n
Command action
e   extended
p   primary partition (1-4)
p
Partition number (1-4): 1
First sector (2048-167772159, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-167772159, default 167772159):
Using default value 167772159
```

```
Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.

WARNING: Re-reading the partition table failed with error 16: Device or resource busy.
The kernel still uses the old table. The new table will be used at
the next reboot or after you run partprobe(8) or kpartx(8)
Syncing disks.
[user@demo-northstar-app centos]#
```

3. Reboot the VM to apply the partition changes.

```
[user@app_instance_XX centos]# reboot

Broadcast message from centos@app_instance_XX
    (/dev/pts/0) at 14:54 ...

The system is going down for reboot NOW!
```

4. Wait until the NorthStar Controller VM has returned to an up state.
5. Reconnect to the VM using SSH.
6. Check the partition size again to verify that the partition was resized.
7. If the partition size is still incorrect, use the **resize2fs** command to adjust the filesystem.

```
# resize2fs /dev/vda1
```

Install the NorthStar Controller RPM Bundle

Install the NorthStar Controller RPM bundle for an OpenStack environment. The procedure uses the **rpm** and **install-vm.sh** commands.

Configure the JunosVM

For security reasons, the JunosVM does not come with a default configuration. Use the following procedure to manually configure the JunosVM using the OpenStack novnc client.

1. Obtain the novnc client URL.

```
# nova get-vnc-console JunosVM-ID novnc
```

2. Configure the JunosVM as you would in a fresh install of the Junos OS.
3. Copy the root user of the NorthStar Controller VM SSH public key to the JunosVM. This allows configuration from the NorthStar Controller VM to the JunosVM using an ssh-key based connection.
4. On the NorthStar Controller VM, run the net_setup.py script, and select option **B** to complete the configuration of the JunosVM. Once complete, you should be able to remotely ping the JunosVM IP address.

Configure SSH Key Exchange

Use the following procedure to configure SSH key exchange between the NorthStar Controller VM and the JunosVM. For High Availability (HA) in a cluster, this must be done for every pair of VMs.

1. Log in to the NorthStar Controller server and display the contents of the id_rsa.pub file by executing the **concatenate** command.

```
$cat /opt/pcs/.ssh/id_rsa.pub
```

   You will need the ssh-rsa string from the output.

2. Log in to the JunosVM and replace the ssh-rsa string with the one from the id_rsa.pub file by executing the following commands.

```
ssh northstar@JunosVM-ip
configure
set system login user northstar authenication ssh-rsa replacement-string
commit
exit
```

3. On the NorthStar Controller server, update the known hosts file by executing the following commands.

```
$su - pcs
$ssh -o UserKnownHostsFile=/opt/pcs/.ssh/known_hosts -i /opt/pcs/.ssh/id_rsa
northstar@JunosVM-ip
exit
exit
```

Configure the HA Cluster

HA on the NorthStar Controller is an active/standby solution. That means that there is only one active node at a time, with all other nodes in the cluster serving as standby nodes. All of the nodes in a cluster must be on the same local subnet for HA to function. On the active node, all processes are running. On the standby nodes, those processes required to maintain connectivity are running, but NorthStar processes are in a stopped state.

If the active node experiences a hardware- or software-related connectivity failure, the NorthStar HA_agent process elects a new active node from amongst the standby nodes. Complete failover is achieved within five minutes. One of the factors in the selection of the new active node is the user-configured priorities of the candidate nodes.

All processes are started on the new active node, and the node acquires the virtual IP address that is required for the client-facing interface. This address is always associated with the active node, even if failover causes the active node to change.

See the *NorthStar Controller User Guide* for further information on configuring and using the HA feature.

RELATED DOCUMENTATION

Installing the NorthStar Controller in Standalone Mode Using a HEAT Template | 98

CHAPTER 4

Installing and Configuring Optional Features

Installing Data Collectors for Analytics

IN THIS SECTION

Overview

The Analytics functionality streams data from the network devices, via data collectors, to the NorthStar Controller where it is processed, stored, and made available for viewing in the web UI.

NOTE: See the *NorthStar Controller User Guide* for information about collecting and viewing telemetry data.

NOTE: Junos OS Release 15.1F6 or later is required to use Analytics. For hardware requirements for analytics nodes, see "NorthStar Controller System Requirements" on page 6. For supported deployment scenarios, see "Platform and Software Compatibility" on page 2.

If you are not using NorthStar application high availability (HA), you can install a data collector either in the same node where the NorthStar Controller application is installed (single-server deployment) or in one or more external nodes that are dedicated to log collection and storage. In both cases, the supplied install scripts take care of installing the required packages and dependencies.

In a NorthStar application HA environment, you have three options:

- Configure an external analytics node.
- Configure an external analytics cluster. An analytics cluster provides backup nodes in the event of an analytics node failure. This cluster can be local (within the same data center) or geo-diverse (analytics geo-HA).
- Install data collectors in the same nodes that make up the NorthStar cluster. In this scenario, the NorthStar application cluster nodes are also analytics cluster nodes.

The configuration options from the analytics processes are read from the /opt/northstar/data/northstar.cfg file. In a single-server deployment, no special changes are required because the parameters needed to start up the collector are part of the default configuration. For your reference, Table 13 on page 114 lists some of the settings that the analytics processes read from the file.

Table 13: Some of the Settings Read by Collector Processes

Setting	Description
mq_host	Points to the IP address or virtual IP (VIP) (for multiple NorthStar node deployments) of hosts running the messaging bus service (the NorthStar application node). Defaults to **localhost** if not present.
mq_username	Username used to connect to the messaging bus. Defaults to **northstar**.
mq_password_enc	Password used to connect to the messaging bus. There is no default; the service fails to start if this is not configured. On single-server deployments, the password is set during the normal application install process.
mq_port	TCP port number used by the messaging bus. Defaults to **5672**.
es_port	TCP port used by Elasticsearch. Defaults to **9200**.
es_cluster_name	Used by Elasticsearch in HA scenarios to form a cluster. Nodes in the same cluster must be configured with the same cluster name. Defaults to **NorthStar**.

Two additional settings are relevant to collector processes, but are not part of northstar.cfg. These parameters configure analytics port numbers and are configurable using the NorthStar CLI:

NOTE: If you make port number changes, you must restart logstash using **supervisorctl restart analytics:logstash** for those changes to take effect.

- rpm-statistics-port

  Port used to read the syslog messages that are generated from the device, containing the results of the RPM stats. The default is 1514. To modify the port, use the NorthStar CLI command **set northstar analytics log-collector rpm-statistics-port** ***port-number***.

- jti-port

  UDP port number to which the collector listens for telemetry packets from the devices. The default is **3000**. To modify the port, use the NorthStar CLI command **set northstar analytics log-collector jti-port** ***port-number***.

> **NOTE**: If you are upgrading NorthStar from a release earlier than NorthStar 4.3.0, and you are using NorthStar analytics, you must upgrade NorthStar manually using the procedure described in "Upgrading from Pre-4.3 NorthStar with Analytics" on page 227.

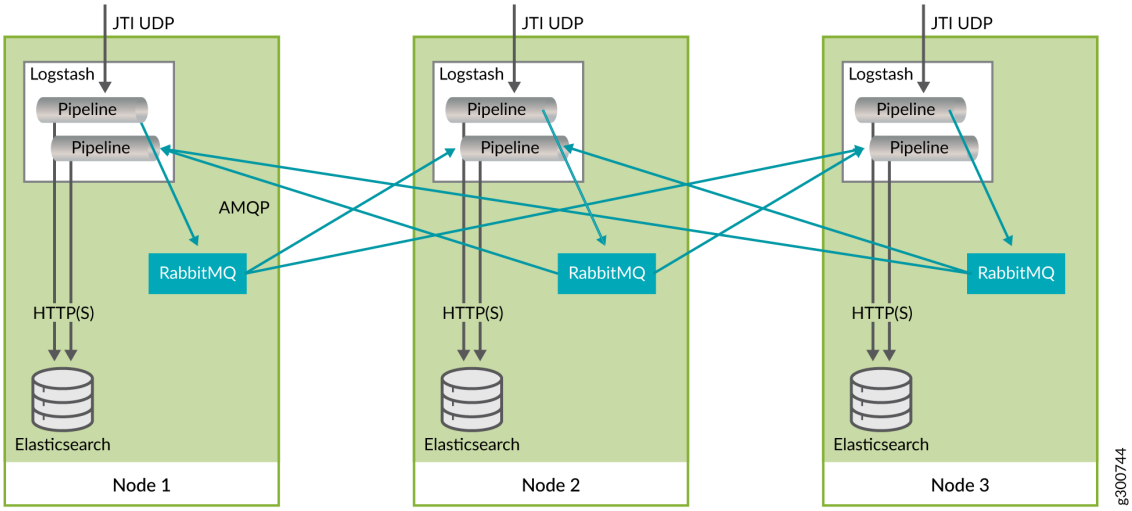
> **NOTE**: If you are upgrading NorthStar from a release earlier than NorthStar 6.0.0, you must redeploy the analytics settings after you upgrade the NorthStar application nodes. This is done from the Analytics Data Collector Configuration Settings menu described later in this topic. This is to ensure that netflowd can communicate with cMGD (necessary for the NorthStar CLI).

Analytics Geo-HA

NorthStar Controller supports analytics geo-HA as of release 5.1.0. While original analytics HA was designed for local clusters (same data center), geo-HA makes all data available on all nodes, to better serve networks where the nodes are geographically remote from one another. To achieve this, a local RabbitMQ (messaging bus) is installed on each analytics (ElasticSearch) node. This improves the tolerance for latency and helps compensate for the tendency of remote nodes to become out of sync.

The remote ElasticSearch nodes use JTI logstash to retrieve and process the data from the other ElasticSearch nodes. The replication pipeline creates a named queue on each remote server. The queues are persistent so that if any ElasticSearch node goes down, it can recover by resuming the processing of data pushed onto the remote queue. Figure 9 on page 116 shows the interactions within a node and between nodes.

Figure 9: Analytics Geo-HA Interactions



The Analytics Collector Configuration Settings menu within the net_setup.py script has an option to prepare and deploy Geo-HA.

Single-Server Deployment–No NorthStar HA

To install the data collector together with the NorthStar application in a single-server deployment (without NorthStar HA), use the following procedure:

1. On the NorthStar application node, install the NorthStar Controller bundle, using the install.sh script. See the "Installing the NorthStar Controller" on page 47.

2. On the same node, run the install-analytics.sh script.

```
[root@ns ~]# cd /opt/northstar/northstar_bundle_x.x.x/
[root@ns northstar_bundle_x.x.x]# ./install-analytics.sh
groupadd: group 'pcs' already exists
package NorthStar-libUtils is not installed
Loaded plugins: fastestmirror
Setting up Install Process
Loading mirror speeds from cached hostfile
northstar_bundle                | 2.9 kB     00:00 ...
Resolving Dependencies
--> Running transaction check
---> Package NorthStar-libUtils.x86_64 0:3.1.0-20161127_68470_213 will be installed
```

```
--> Finished Dependency Resolution

Dependencies Resolved
.
.
.
```

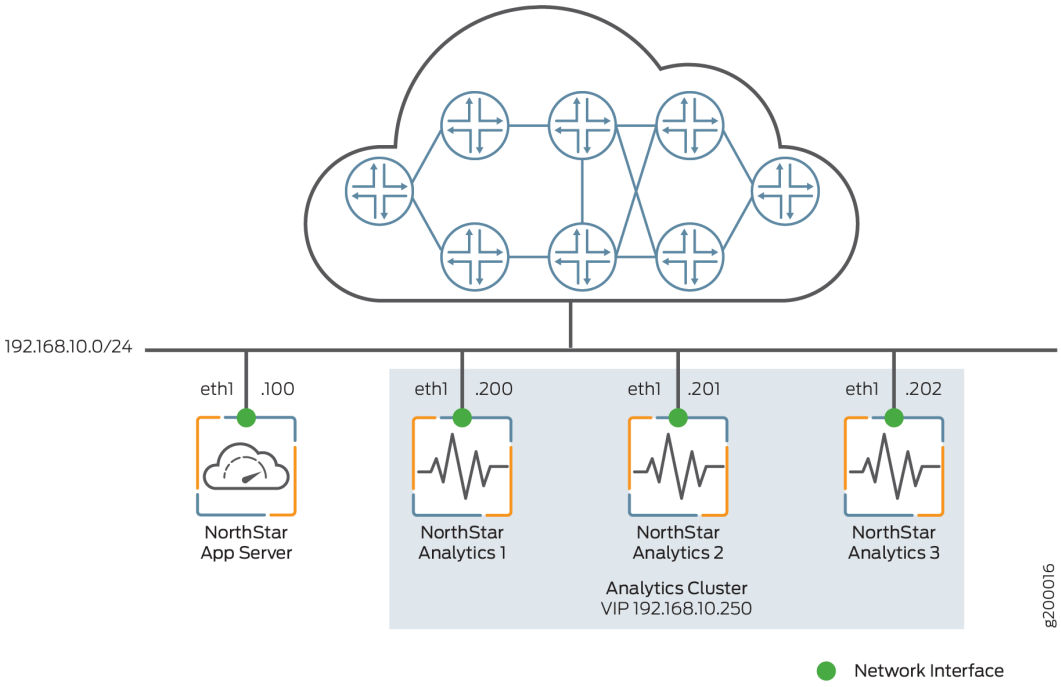
3. Verify that the three analytics processes are installed and running by executing **supervisorctl status** on the PC Server:

```
[root@ns ~]# supervisorctl status
analytics:elasticsearch          RUNNING   pid 7073, uptime 21:57:29
analytics:esauthproxy            RUNNING   pid 7072, uptime 21:57:29
analytics:logstash               RUNNING   pid 7231, uptime 21:57:26
```

External Analytics Node(s)–No NorthStar HA

Figure 10 on page 118 shows a sample configuration with a single NorthStar application node and three analytics nodes comprising an analytics cluster. All the nodes connect to the same Ethernet network, through the eth1 interface. Optionally, you could have a single analytics node rather than creating an analytics cluster. The instructions in this section cover both a single external analytics node and an external analytics cluster.

Figure 10: Analytics Cluster Deployment (No NorthStar HA)



To install one or a cluster of external analytics nodes, use the following procedure:

1. On the NorthStar application node, install the NorthStar Controller application, using the install.sh script. See "Installing the NorthStar Controller" on page 47.

2. On each analytics node, install northstar_bundle.rpm, but do not run the install.sh script. Instead, run the install-analytics.sh script. The script installs all required dependencies such as NorthStar-JDK, NorthStar-Python, and so on. For NorthStar Analytics1, it would look like this:

```
[root@NorthStarAnalytics1]# rpm -Uvh <rpm-filename>
[root@NorthStarAnalytics1]# cd /opt/northstar/northstar_bundle_x.x.x/
[root@NorthStarAnalytics1 northstar_bundle_x.x.x]# ./install-analytics.sh
groupadd: group 'pcs' already exists
package NorthStar-PCS is not installed
Loaded plugins: fastestmirror
Setting up Update Process
Loading mirror speeds from cached hostfile
northstar_bundle            | 2.9 kB     00:00 ...
No Packages marked for Update
Loaded plugins: fastestmirror
Setting up Update Process
```

```
Loading mirror speeds from cached hostfile
No Packages marked for Update
Loaded plugins: fastestmirror
Setting up Update Process
.
.
.
```

3. The next configuration steps require you to run the net_setup.py script to configure the NorthStar node and the analytics nodes(s) so they can connect to each other. But before you do that, we recommend that you copy the public SSH key of the node where the net_setup.py script is to be executed to all other nodes. The net_setup.py script can be run on either the NorthStar application node or one of the analytics nodes to configure all the nodes. This is not a required step, but it saves typing the passwords of all the systems later when the script is deploying the configurations or testing the connectivity to the different nodes.

```
[root@NorthStarAnalytics1 network-scripts]# ssh-copy-id root@192.168.10.200
root@192.168.10.200's password:
```

Try logging into the machine using `ssh root@192.168.10.200` and check in with `.ssh/authorized_keys`.

Repeat this process for all nodes (192.168.10.100, 192.168.10.200, 192.168.10.201, and 192.168.10.202 in our example).

4. Run net_setup.py on the NorthStar application node or on one of the analytics nodes. The Main Menu is displayed:

```
Main Menu:
    ..........................................
    A.) Host Setting
    ..........................................
    B.) JunosVM Setting
    ..........................................
    C.) Check Network Setting
    ..........................................
    D.) Maintenance & Troubleshooting
    ..........................................
    E.) HA Setting
    ..........................................
    F.) Collect Trace/Log
    ..........................................
    G.) Analytics Data Collector Setting
```

```
      (External standalone/cluster analytics server)
    ........................................
    H.) Setup SSH Key for external JunosVM  setup
    ........................................
    I.) Internal Analytics Setting (HA)
    ........................................
    X.) Exit
    ........................................
Please select a letter to execute.
```

5. Select **G** Analytics Data Collector Setting. The Data Collector Configuration Settings menu is displayed.

```
    Analytics Data Collector Configuration Settings:
        (External standalone/cluster analytics server)
    ******************************************************
    Note: This configuration only applicable for data collector
    installation in separate server
    ******************************************************
    ....................................................

        NorthStar App #1
             Hostname                              :
             Interface
                  Name                             : external0
                  IPv4                             :
    ....................................................
        Analytics Collector #1
             Hostname                              :
             Priority                              : 0
             Interface
                  Name                             : external0
                  IPv4                             :

    1. ) Add NorthStar App
    2. ) Add analytics data collector
    3. ) Modify NorthStar App
    4. ) Modify analytics data collector
    5A.) Remove NorthStar App
    5B.) Delete NorthStar App data
    6A.) Remove analytics data collector
    6B.) Delete analytics data collector data
```

```
.....................................................
7A.) Virtual IP for Northstar App                 :
7B.) Delete Virtual IP for Northstar App
8A.) Virtual IP for Analytics Collector                     :
8B.) Delete Virtual IP for Analytics Collector
.....................................................
9. ) Test Analytics Data Collector Connectivity
A. ) Prepare and Deploy SINGLE Data Collector Setting
B. ) Prepare and Deploy HA Analytics Data Collector Setting
C. ) Prepare and Deploy GEO-HA Analytics Data Collector Setting
D. ) Copy Collector setting to other nodes
E. ) Add a new Collector node to existing cluster
F. ) Sync Config with NorthStar App
.....................................................

Please select a number to modify.
[<CR>=return to main menu]:
```

6. Select options from the Data Collector Configuration Settings menu to make the following configuration changes:

   - Select **3** to modify the NorthStar application node settings, and configure the NorthStar server name and IP address. For example:

```
Please select a number to modify.
[CR=return to main menu]:
3
NorthStar App ID : 1

current NorthStar App #1 hostname (without domain name) :
new NorthStar App #1 hostname (without domain name) : NorthStarAppServer

current NorthStar App #1 interface name : external0
new NorthStar App #1 interface name : eth1

current NorthStar App #1 interface IPv4 address :
new NorthStar App #1 interface IPv4 address : 192.168.10.100


Press any key to return to menu
```

- Select **4** to modify the analytics node IP address. For example:

```
Please select a number to modify.
[CR=return to main menu]:
4
Collector ID : 1

current collector #1 hostname (without domain name) :
new collector #1 hostname (without domain name) : NorthStarAnalytics1

current collector #1 node priority : 0
new collector #1 node priority : 10

current collector #1 interface name : external0
new collector #1 interface name : eth1

current collector #1 interface IPv4 address :
new collector #1 interface IPv4 address : 192.168.10.200


Press any key to return to menu
```

- Select **2** to add additional analytics nodes as needed. In our analytics cluster example, two additional analytics nodes would be added:

```
Please select a number to modify.
[CR=return to main menu]:
2
New collector ID : 2

current collector #2 hostname (without domain name) :
new collector #2 hostname (without domain name) : NorthStarAnalytics2

current collector #2 node priority : 0
new collector #2 node priority : 20

current collector #2 interface name : external0
new collector #2 interface name : eth1

current collector #2 interface IPv4 address :
new collector #2 interface IPv4 address : 192.168.10.201
```

```
Press any key to return to menu

Please select a number to modify.
[CR=return to main menu]:
2
New collector ID : 3

current collector #3 hostname (without domain name) :
new collector #3 hostname (without domain name) : NorthStarAnalytics3

current collector #3 node priority : 0
new collector #3 node priority : 30

current collector #3 interface name : external0
new collector #3 interface name : eth1

current collector #3 interface IPv4 address :
new collector #3 interface IPv4 address : 192.168.10.202


Press any key to return to menu
```

- Select **8A** to configure a VIP address for the cluster of analytics nodes. This is required if you have an analytics cluster. If you have a single external analytics node only (not a cluster), you can skip this step.

```
Please select a number to modify.
[CR=return to main menu]:
8A
current Virtual IP for Collector :
new Virtual IP for Collector : 192.168.10.250

Press any key to return to menu
```

This VIP serves two purposes:

- It allows the NorthStar server to send queries to a single endpoint. The VIP will be active on one of the analytics nodes, and will switch over in the event of a failure (a full node failure or failure of any of the processes running on the analytics node).

- Devices can send telemetry data to the VIP, ensuring that if an analytics node fails, the telemetry data can still be processed by whichever non-failing node takes ownership of the VIP.

The configuration for our analytics cluster example should now look like this:

```
Analytics Data Collector Configuration Settings:
(External standalone/cluster analytics server)
*******************************************************
Note: This configuration only applicable for analytics
data collector installation in separate server
*******************************************************
.....................................................
    NorthStar App #1
         Hostname                              : NorthStarAppServer
         Interface
              Name                             : eth1
              IPv4                             : 192.168.10.100
.....................................................
    Analytics Collector #1
         Hostname                              : NorthStarAnalytics1
         Priority                              : 10
         Interface
              Name                             : eth1
              IPv4                             : 192.168.10.200
    Analytics Collector #2
         Hostname                              : NorthStarAnalytics2
         Priority                              : 20
         Interface
              Name                             : eth1
              IPv4                             : 192.168.10.201
    Analytics Collector #3
         Hostname                              : NorthStarAnalytics3
         Priority                              : 30
         Interface
              Name                             : eth1
              IPv4                             : 192.168.10.202

1. ) Add NorthStar App
2. ) Add analytics data collector
3. ) Modify NorthStar App
4. ) Modify analytics data collector
```

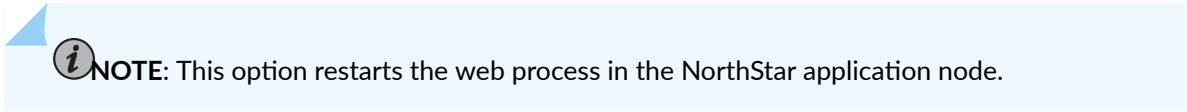
```
    5A.) Remove NorthStar App
    5B.) Delete NorthStar App data
    6A.) Remove analytics data collector
    6B.) Delete analytics data collector data
    ....................................................
    7A.) Virtual IP for Northstar App                 :
    7B.) Delete Virtual IP for Northstar App
    8A.) Virtual IP for Analytics Collector           : 192.168.10.250
    8B.) Delete Virtual IP for Analytics Collector
    ....................................................
    9. ) Test Analytics Data Collector Connectivity
    A. ) Prepare and Deploy SINGLE Data Collector Setting
    B. ) Prepare and Deploy HA Analytics Data Collector Setting
    C. ) Prepare and Deploy GEO-HA Analytics Data Collector Setting
    D. ) Copy Collector setting to other nodes
    E. ) Add a new Collector node to existing cluster
    F. ) Sync Config with NorthStar App
    ....................................................

    Please select a number to modify.
    [<CR>=return to main menu]:
```

7. Select **9** to test connectivity between nodes. This is applicable whenever you have external analytics nodes, whether just one or a cluster of them. For example:

```
Please select a number to modify.
[CR=return to main menu]:
9
Validate NorthStar App configuration interface
Validate Collector configuration interface

Verifying the NorthStar version on each NorthStar App node:
NorthStar App #1 NorthStarAppServer: NorthStar-Bundle-3.1.0-20170517_195239_70090_547.x86_64

Collector #1 NorthStarAnalytics1 :
NorthStar-Bundle-3.1.0-20170517_195239_70090_547.x86_64
Collector #2 NorthStarAnalytics2 :
NorthStar-Bundle-3.1.0-20170517_195239_70090_547.x86_64
Collector #3 NorthStarAnalytics3 :
NorthStar-Bundle-3.1.0-20170517_195239_70090_547.x86_64

Checking NorthStar App connectivity...
```

```
NorthStar App #1 interface name eth1 ip 192.168.10.100: OK

Checking collector connectivity...
Collector #1 interface name eth1 ip 192.168.10.200: OK
Collector #2 interface name eth1 ip 192.168.10.201: OK
Collector #3 interface name eth1 ip 192.168.10.202: OK
Press any key to return to menu
```

8. Select **A** (for a single analytics node), **B** (for an analytics cluster), or **C** for analytics geo-HA to configure the node(s) for deployment.



NOTE: This option restarts the web process in the NorthStar application node.

For our example, select **B**:

```
Please select a number to modify.
[CR=return to main menu]:
B
Setup mode set to "cluster"

Validate NorthStar App configuration interface
Validate Collector configuration interface

Verifying the NorthStar version on each NorthStar App node:
NorthStar App #1 NorthStarAppServer: NorthStar-Bundle-3.1.0-20170517_195239_70090_547.x86_64

Verifying the NorthStar version on each Collector node:
Collector #1 NorthStarCollector1 :
NorthStar-Bundle-3.1.0-20170517_195239_70090_547.x86_64
Collector #2 NorthStarCollector2 :
NorthStar-Bundle-3.1.0-20170517_195239_70090_547.x86_64
Collector #3 NorthStarCollector3 :
NorthStar-Bundle-3.1.0-20170517_195239_70090_547.x86_64

WARNING !
The selected menu will restart nodejs process in Northstar App node
Type YES to continue...

YES
Checking NorthStar App connectivity...
NorthStar App #1 interface name eth1 ip 192.168.10.100: OK
```

```
Checking collector connectivity...
Collector #1 interface name eth1 ip 192.168.10.200: OK
Collector #2 interface name eth1 ip 192.168.10.201: OK
Collector #3 interface name eth1 ip 192.168.10.202: OK

Checking analytics process in NorthStar App node ...
Detected analytics is not in NorthStar App node #1: OK

Checking analytics process in collector node ...
Detected analytics in collector node #1: OK
Detected analytics in collector node #2: OK
Detected analytics in collector node #3: OK

External data collector set to "yes"

Sync configuration for NorthStar App #1: OK

Sync configuration for Collector #1: OK

Sync configuration for Collector #2: OK

Sync configuration for Collector #3: OK

Preparing collector #1 basic configuration ..

Uploading config files to collector01

Preparing collector #2 basic configuration ..

Uploading config files to collector02

Preparing collector #3 basic configuration ..

Uploading config files to collector03

Applying data collector config files

Applying data collector config files at NorthStar App
Deploying NorthStar App #1 collector configuration ...

Applying data collector config files at collector
Deploying collector #1 collector configuration ...
```

```
Deploying collector #2 collector configuration ...
Deploying collector #3 collector configuration ...

Deploying collector #1 zookeeper configuration ...
Wait 2 minutes before adding new node
...10 seconds
...20 seconds
...30 seconds
...40 seconds
...50 seconds
...60 seconds
...70 seconds
...80 seconds
...90 seconds
...100 seconds
...110 seconds

Deploying collector #2 zookeeper configuration ...
Wait 2 minutes before adding new node
...10 seconds
...20 seconds
...30 seconds
...40 seconds
...50 seconds
...60 seconds
...70 seconds
...80 seconds
...90 seconds
...100 seconds
...110 seconds

Deploying collector #3 zookeeper configuration ...


Restart ZooKeeper at collector #1 collector01

Restart ZooKeeper at collector #2 collector02

Restart ZooKeeper at collector #3 collector03



Restart Analytics at collector #1 collector01
```

```
Restart Analytics at collector #2 collector02

Restart Analytics at collector #3 collector03



Restart HA Agent at collector #1 collector01
Please wait for HA Agent process initialization
...10 seconds
...20 seconds

Restart HA Agent at collector #2 collector02
Please wait for HA Agent process initialization
...10 seconds
...20 seconds

Restart HA Agent at collector #3 collector03
Please wait for HA Agent process initialization
...10 seconds
...20 seconds



Restart Nodejs at Northstar App #1 pcs


Collector configurations has been applied successfully

Press any key to return to menu
```

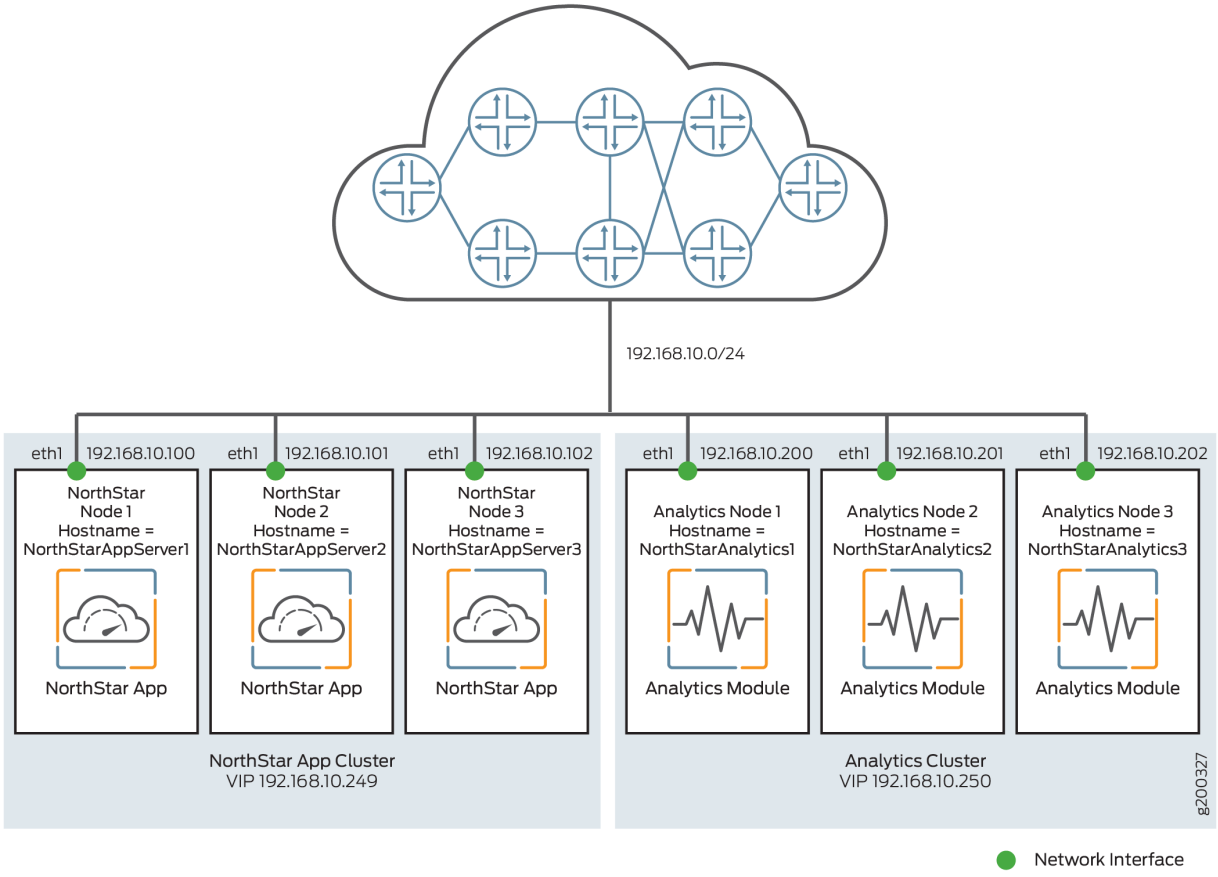
This completes the installation, and telemetry data can now be sent to the analytics nodes via the analytics VIP.

NOTE: If you opt to send telemetry data to an individual node instead of using the VIP of the analytics cluster, and that node goes down, the streams to the node are lost. If you opt to install only one analytics node instead of an analytics cluster that uses a VIP, you run the same risk.

External Analytics Node(s)–With NorthStar HA

Figure 11 on page 130 shows a sample configuration with a NorthStar HA cluster of three nodes and three analytics nodes comprising an analytics cluster, for a total of six nodes. All the nodes connect to the same Ethernet network, through the eth1 interface. In a NorthStar HA environment, you could also opt to have a single analytics node, for a total of four nodes, but analytics collection would not be protected in the event of analytics node failure.

Figure 11: Analytics Cluster Deployment (With NorthStar HA)



For this scenario, you first configure the NorthStar application HA cluster according to the instructions in "Configuring a NorthStar Cluster for High Availability" on page 161.

Once the NorthStar HA cluster is configured, set up the external analytics cluster. The setup steps for the external analytics cluster are exactly the same as in the previous section, *External Analytics Node(s)–No NorthStar HA*. Once you complete them, the configuration should look like this:

```
Analytics Data Collector Configuration Settings:
(External standalone/cluster analytics server)
*****************************************************
Note: This configuration only applicable for analytics
data collector installation in separate server
*****************************************************
.....................................................
    NorthStar App #1
         Hostname                              : NorthStarAppServer1
         Interface
              Name                             : eth1
              IPv4                             : 192.168.10.100
    NorthStar App #2
         Hostname                              : NorthStarAppServer2
         Interface
              Name                             : eth1
              IPv4                             : 192.168.10.101
    NorthStar App #3
         Hostname                              : NorthStarAppServer3
         Interface
              Name                             : eth1
              IPv4                             : 192.168.10.102
.....................................................
    Analytics Collector #1
         Hostname                              : NorthStarAnalytics1
         Priority                              : 10
         Interface
              Name                             : eth1
              IPv4                             : 192.168.10.200
    Analytics Collector #2
         Hostname                              : NorthStarAnalytics2
         Priority                              : 20
         Interface
              Name                             : eth1
              IPv4                             : 192.168.10.201
    Analytics Collector #3
         Hostname                              : NorthStarAnalytics3
         Priority                              : 30
```

```
          Interface
               Name                                : eth1
               IPv4                                : 192.168.10.202

    1. ) Add NorthStar App
    2. ) Add analytics data collector
    3. ) Modify NorthStar App
    4. ) Modify analytics data collector
    5A.) Remove NorthStar App
    5B.) Delete NorthStar App data
    6A.) Remove analytics data collector
    6B.) Delete analytics data collector data
    ....................................................
    7A.) Virtual IP for Northstar App                 : 192.168.10.249
    7B.) Delete Virtual IP for Northstar App
    8A.) Virtual IP for Analytics Collector           : 192.168.10.250
    8B.) Delete Virtual IP for Analytics Collector
    ....................................................
    9. ) Test Analytics Data Collector Connectivity
    A. ) Prepare and Deploy SINGLE Data Collector Setting
    B. ) Prepare and Deploy HA Analytics Data Collector Setting
    C. ) Prepare and Deploy GEO-HA Analytics Data Collector Setting
    D. ) Copy Collector setting to other nodes
    E. ) Add a new Collector node to existing cluster
    F. ) Sync Config with NorthStar App
    ....................................................

    Please select a number to modify.
    [<CR>=return to main menu]:
```

Test connectivity between nodes by selecting **9** from the menu.

Configure the nodes for deployment by selecting **B** for HA analytics or **C** for Geo-HA analytics. This restarts the web process in the NorthStar application node.

Verifying Data Collection When You Have External Analytics Nodes

Verify that data collection is working by checking that all services are running. Only the relevant processes are shown below.

```
[root@NorthStarAnalytics1 ~]# supervisorctl status
analytics:elasticsearch          RUNNING   pid 4406, uptime 0:02:06
analytics:esauthproxy            RUNNING   pid 4405, uptime 0:02:06
analytics:logstash               RUNNING   pid 4407, uptime 0:02:06
infra:ha_agent                   RUNNING   pid 4583, uptime 0:00:19
infra:healthmonitor              RUNNING   pid 3491, uptime 1:01:09
infra:zookeeper                  RUNNING   pid 4324, uptime 0:03:16
listener1:listener1_00           RUNNING   pid 4325, uptime 0:03:16
```

The analytics node(s) should start processing all records from the network, and pushing statistics to the NorthStar node through RabbitMQ. Check the pcs.log in the NorthStar node to see the statistics being pushed to the PC server. For example:

```
11-28T13:18:02.174126 30749 PCServer  [NorthStar][PCServer][<-AMQP] msg=0x00004018 routing_key =
ns_tunnel_traffic
11-28T13:18:02.174280 30749 PCServer  [NorthStar][PCServer][Traffic] msg=0x00005004  EF1-PE1-
PE2@PE1 111094
11-28T13:18:02.174429 30749 PCServer  [NorthStar][PCServer][Traffic] msg=0x00005004  EF1-PE1-
PE3@PE1 824
11-28T13:18:02.174764 30749 PCServer  [NorthStar][PCServer][Traffic] msg=0x00005004  CS1-PE3-
PE3@PE3 0
11-28T13:18:02.174930 30749 PCServer  [NorthStar][PCServer][Traffic] msg=0x00005004  CS2-PE3-
PE2@PE3 0
11-28T13:18:02.175067 30749 PCServer  [NorthStar][PCServer][Traffic] msg=0x00005004  EF2-PE3-
PE3@PE3 0
11-28T13:18:02.175434 30749 PCServer  [NorthStar][PCServer][Traffic] msg=0x00005004  EF2-PE3-
PE1@PE3 0
11-28T13:18:02.175614 30749 PCServer  [NorthStar][PCServer][Traffic] msg=0x00005004  EF1-PE3-
PE1@PE3 0
11-28T13:18:02.175749 30749 PCServer  [NorthStar][PCServer][Traffic] msg=0x00005004  CS2-PE3-
PE3@PE3 0
11-28T13:18:02.175873 30749 PCServer  [NorthStar][PCServer][Traffic] msg=0x00005004  CS1-PE3-
PE1@PE3 0
11-28T13:18:02.175989 30749 PCServer  [NorthStar][PCServer][Traffic] msg=0x00005004  CS1-PE3-
PE2@PE3 0
```

```
11-28T13:18:02.176128 30749 PCServer  [NorthStar][PCServer][Traffic] msg=0x00005004  CS2-PE3-
PE1@PE3 824
11-28T13:18:02.176256 30749 PCServer  [NorthStar][PCServer][Traffic] msg=0x00005004  EF1-PE3-
PE3@PE3 0
11-28T13:18:02.176393 30749 PCServer  [NorthStar][PCServer][Traffic] msg=0x00005004  EF1-PE2-
PE1@PE2 112552
11-28T13:18:02.176650 30749 PCServer  [NorthStar][PCServer][Traffic] msg=0x00005004  AF1-PE2-
PE1@PE2 0
11-28T13:18:02.176894 30749 PCServer  [NorthStar][PCServer][Traffic] msg=0x00005004  AF2-PE2-
PE1@PE2 0
11-28T13:18:02.177059 30749 PCServer  [NorthStar][PCServer][Traffic] msg=0x00005004  EF12-PE2-
PE1@PE2 0
```

You can also use the REST APIs to get some aggregated statistics. This tests the path from client to nodejs to Elasticsearch.

```
 curl --insecure -X POST -H "Authorization: Bearer 7IEvYhvABrae6m1AgI+zi4V0n7UiJNA2HqliK7PfGhY="
-H "Content-Type: application/json"  -d '{
  "endTime": "now",
  "startTime": "now-1h",
  "aggregation": "avg",
  "counter": "interface_stats.egress_stats.if_bps"
}' "https://localhost:8443/NorthStar/API/v2/tenant/1/statistics/device/top"
[

  {
    "id": {
      "statisticType": "device",
      "name": "vmx105",
      "node": {
        "topoObjectType": "node",
        "hostName": "vmx105"
      }
    },
    "interface_stats.egress_stats.if_bps": 525088
  },
  {
    "id": {
      "statisticType": "device",
      "name": "PE1",
      "node": {
        "topoObjectType": "node",
```

```
        "hostName": "PE1"
      }
    },
    "interface_stats.egress_stats.if_bps": 228114
  },
  {
    "id": {
      "statisticType": "device",
      "name": "PE2",
      "node": {
        "topoObjectType": "node",
        "hostName": "PE2"
      }
    },
    "interface_stats.egress_stats.if_bps": 227747
  },
  {
    "id": {
      "statisticType": "device",
      "name": "PE3",
      "node": {
        "topoObjectType": "node",
        "hostName": "PE3"
      }
    },
    "interface_stats.egress_stats.if_bps": 6641
  },
  {
    "id": {
      "statisticType": "device",
      "name": "PE4",
      "node": {
        "topoObjectType": "node",
        "hostName": "PE4"
      }
    },
    "interface_stats.egress_stats.if_bps": 5930
  }
]
```

Replacing a Failed Node in an External Analytics Cluster

On the Data Collector Configuration Settings menu, options D and E can be used when physically replacing a failed node. They allow you to replace a node without having to redeploy the entire cluster.

> ⚠ **CAUTION**: While a node is being replaced in a three-node cluster, HA for analytics data is not guaranteed.

1. Replace the physical node in the network and install northstar_bundle.rpm on the replacement node. In our example, the replacement node is NorthStarAnalytics3.

2. Run the install-analytics.sh script to install all required dependencies such as NorthStar-JDK, NorthStar-Python, and so on. For NorthStarAnalytics3, it would look like this:

```
[root@NorthStarAnalytics3]# rpm -Uvh <rpm-filename>
[root@NorthStarAnalytics3]# cd /opt/northstar/northstar_bundle_x.x.x/
[root@NorthStarAnalytics3 northstar_bundle_x.x.x]# ./install-analytics.sh
groupadd: group 'pcs' already exists
package NorthStar-PCS is not installed
Loaded plugins: fastestmirror
Setting up Update Process
Loading mirror speeds from cached hostfile
northstar_bundle            | 2.9 kB     00:00 ...
No Packages marked for Update
Loaded plugins: fastestmirror
Setting up Update Process
Loading mirror speeds from cached hostfile
No Packages marked for Update
Loaded plugins: fastestmirror
Setting up Update Process
.
.
.
```

3. Set up the SSH key from an anchor node to the replacement node. The anchor node can be a NorthStar application node or one of the analytics cluster nodes (other than the replacement node). Copy the public SSH key from the anchor node to the replacement node, from the replacement node to the other nodes (NorthStar application nodes and analytics cluster nodes), and from the other nodes (NorthStar application nodes and analytics cluster nodes) to the replacement node.

For example:

```
[root@NorthStarAnalytics1 network-scripts]# ssh-copy-id root@192.168.10.202
root@192.168.10.202's password:
```

Try logging into the machine using `ssh root@192.168.10.202` and check in with `.ssh/authorized_keys`.

4. Run net_setup.py on the node you selected. The Main Menu is displayed:

```
Main Menu:
    ..........................................
    A.) Host Setting
    ..........................................
    B.) JunosVM Setting
    ..........................................
    C.) Check Network Setting
    ..........................................
    D.) Maintenance & Troubleshooting
    ..........................................
    E.) HA Setting
    ..........................................
    F.) Collect Trace/Log
    ..........................................
    G.) Analytics Data Collector Setting
      (External standalone/cluster analytics server)
    ..........................................
    H.) Setup SSH Key for external JunosVM  setup
    ..........................................
    I.) Internal Analytics Setting (HA)
    ..........................................
    X.) Exit
    ..........................................
Please select a letter to execute.
```

5. Select **G** Data Collector Setting. The Data Collector Configuration Settings menu is displayed.

```
    Analytics Data Collector Configuration Settings:
         (External standalone/cluster analytics server)
    ********************************************************
    Note: This configuration only applicable for analytics
    data collector installation in separate server
```

```
*****************************************************
.....................................................
    NorthStar App #1
         Hostname                              : NorthStarAppServer1
         Interface
              Name                             : eth1
              IPv4                             : 192.168.10.100
.....................................................
    NorthStar App #2
         Hostname                              : NorthStarAppServer2
         Interface
              Name                             : eth1
              IPv4                             : 192.168.10.101
.....................................................
    NorthStar App #3
         Hostname                              : NorthStarAppServer3
         Interface
              Name                             : eth1
              IPv4                             : 192.168.10.102
.....................................................
    Analytics Collector #1
         Hostname                              : NorthStarAnalytics1
         Priority                              : 10
         Interface
              Name                             : eth1
              IPv4                             : 192.168.10.200
.....................................................
    Analytics Collector #2
         Hostname                              : NorthStarAnalytics2
         Priority                              : 20
         Interface
              Name                             : eth1
              IPv4                             : 192.168.10.201
.....................................................
    Analytics Collector #3
         Hostname                              : NorthStarAnalytics3
         Priority                              : 30
         Interface
              Name                             : eth1
              IPv4                             : 192.168.10.202

1. ) Add NorthStar App
2. ) Add analytics data collector
```

```
    3. ) Modify NorthStar App
    4. ) Modify analytics data collector
    5A.) Remove NorthStar App
    5B.) Delete NorthStar App data
    6A.) Remove analytics data collector
    6B.) Delete analytics data collector data
    ....................................................
    7A.) Virtual IP for Northstar App                 : 192.168.10.249
    7B.) Delete Virtual IP for Northstar App
    8A.) Virtual IP for Collector                     : 192.168.10.250
    8B.) Delete Virtual IP for Analytics Collector
    ....................................................
    9. ) Test Analytics Data Collector Connectivity
    A. ) Prepare and Deploy SINGLE Data Collector Setting
    B. ) Prepare and Deploy HA Analytics Data Collector Setting
    C. ) Prepare and Deploy GEO-HA Analytics Data Collector Setting
    D. ) Copy Collector setting to other nodes
    E. ) Add a new Collector node to existing cluster
    F. ) Sync Config with NorthStar App
    ....................................................
    Please select a number to modify.
    [<CR>=return to main menu]:
```

6. Select option **9** to test connectivity to all NorthStar application nodes and analytics cluster nodes.

```
Checking NorthStar App connectivity...
NorthStar App #1 interface name eth1 ip 192.168.10.100: OK
NorthStar App #2 interface name eth1 ip 192.168.10.101: OK
NorthStar App #3 interface name eth1 ip 192.168.10.102: OK

Checking collector connectivity...
Collector #1 interface name eth1 ip 192.168.10.200: OK
Collector #2 interface name eth1 ip 192.168.10.201: OK
Collector #3 interface name eth1 ip 192.168.10.202: OK
```

7. Select option **D** to copy the analytics settings to the other nodes.

```
Validate NorthStar App configuration interface
Validate Collector configuration interface

Verifying the NorthStar version on each NorthStar App node:
```

```
NorthStar App #1 NorthStarAppServer1 : NorthStar-Bundle-3.1.0-20170517_195239_70090_547.x86_64
NorthStar App #2 NorthStarAppServer2 : NorthStar-Bundle-3.1.0-20170517_195239_70090_547.x86_64
NorthStar App #3 NorthStarAppServer3 : NorthStar-Bundle-3.1.0-20170517_195239_70090_547.x86_64

Verifying the NorthStar version on each Collector node:
Collector #1 NorthStarAnalytics1 : NorthStar-Bundle-3.1.0-20170517_195239_70090_547.x86_64
Collector #2 NorthStarAnalytics2 : NorthStar-Bundle-3.1.0-20170517_195239_70090_547.x86_64
Collector #3 NorthStarAnalytics3 : NorthStar-Bundle-3.1.0-20170517_195239_70090_547.x86_64

Checking NorthStar App connectivity...
NorthStar App #1 interface name eth1 ip 192.168.10.100: OK
NorthStar App #2 interface name eth1 ip 192.168.10.101: OK
NorthStar App #3 interface name eth1 ip 192.168.10.102: OK

Checking collector connectivity...
Collector #1 interface name eth1 ip 192.168.10.200: OK
Collector #2 interface name eth1 ip 192.168.10.201: OK
Collector #3 interface name eth1 ip 192.168.10.202: OK

Sync configuration for NorthStar App #1: OK
Sync configuration for NorthStar App #2: OK
Sync configuration for NorthStar App #3: OK

Sync configuration for Collector #1: OK
Sync configuration for Collector #2: OK
Sync configuration for Collector #3: OK
```

8. Select option **E** to add the replacement node to the cluster. Specify the node ID of the replacement node.

9. On any analytics cluster node, use the following command to check Elasticsearch cluster status. Verify that the status is “green” and the number of nodes is correct.

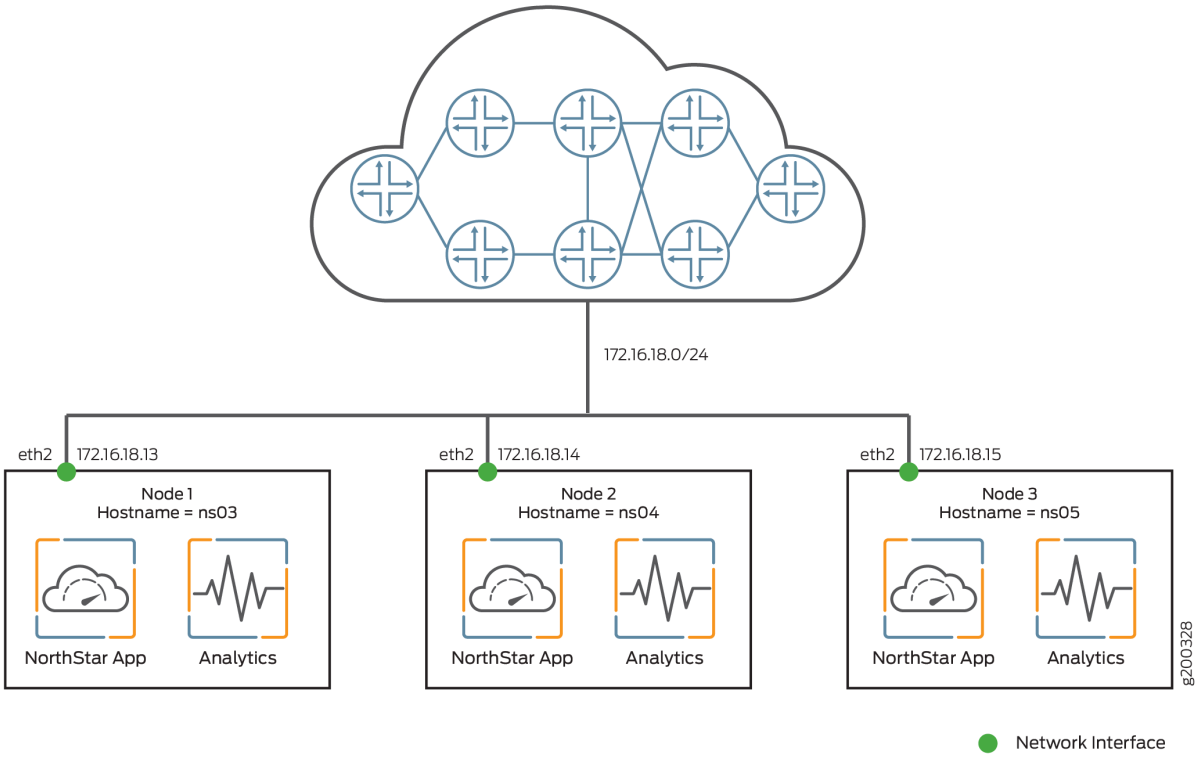
```
[root@NorthStarAnalytics1]# curl -XGET 'localhost:9200/_cluster/health?pretty'
{
"cluster_name" : "NorthStar",
"status" : "green",
"timed_out" : false,
"number_of_nodes" : 3,
"number_of_data_nodes" : 3,
"active_primary_shards" : 10,
"active_shards" : 10,
```

```
"relocating_shards" : 0,
"initializing_shards" : 0,
"unassigned_shards" : 0,
"delayed_unassigned_shards" : 0,
"number_of_pending_tasks" : 0,
"number_of_in_flight_fetch" : 0,
"task_max_waiting_in_queue_millis" : 0,
"active_shards_percent_as_number" : 100.0
}
```

Collectors Installed on the NorthStar HA Cluster Nodes

In a NorthStar HA environment, you can achieve failover protection simultaneously for the NorthStar application and for analytics by setting up each node in the NorthStar cluster to also serve as an analytics node. Because nothing is external to the NorthStar cluster, your total number of nodes is the number in the NorthStar cluster (minimum of three). Figure 12 on page 142 shows this installation scenario.

Figure 12: NorthStar HA Cluster Nodes with Analytics



To set up this scenario, you first install both the NorthStar application and analytics on each of the standalone nodes, configure the nodes to be an HA cluster, and finally, configure the nodes to be an analytics cluster. Follow these steps:

1. On each NorthStar application node, install the NorthStar Controller application, using the install.sh script. See the "Installing the NorthStar Controller" on page 47.

2. On each node, install northstar_bundle.rpm, and run the install-analytics.sh script. The script installs all required dependencies such as NorthStar-JDK, NorthStar-Python, and so on. For node ns03 in the example, it would look like this:

```
[root@ns03]# rpm -Uvh <rpm-filename>
[root@ns03]# cd /opt/northstar/northstar_bundle_x.x.x/
[root@ns03 northstar_bundle_x.x.x]# ./install-analytics.sh
groupadd: group 'pcs' already exists
package NorthStar-PCS is not installed
Loaded plugins: fastestmirror
Setting up Update Process
Loading mirror speeds from cached hostfile
northstar_bundle            | 2.9 kB     00:00 ...
```

```
No Packages marked for Update
Loaded plugins: fastestmirror
Setting up Update Process
Loading mirror speeds from cached hostfile
No Packages marked for Update
Loaded plugins: fastestmirror
Setting up Update Process
.
.
.
```

3. Use the following command on each node to ensure that the three analytics processes are installed and running:

```
[root@ns03 ~]# supervisorctl status | grep analytics:*
analytics:elasticsearch          RUNNING   pid 16238, uptime 20:58:37
analytics:esauthproxy            RUNNING   pid 16237, uptime 20:58:37
analytics:logstash               RUNNING   pid 3643, uptime 20:13:08
```

4. Follow the instructions in "Configuring a NorthStar Cluster for High Availability" on page 161 to configure the nodes for NorthStar HA. This involves running the net_setup.py utility, selecting **E** to access the HA Setup menu, and completing the HA setup steps using that menu.

5. From the HA Setup menu, press **Enter** to return to the main net_setup.py menu. The Main Menu is displayed:

```
Main Menu:
    ..........................................
    A.) Host Setting
    ..........................................
    B.) JunosVM Setting
    ..........................................
    C.) Check Network Setting
    ..........................................
    D.) Maintenance & Troubleshooting
    ..........................................
    E.) HA Setting
    ..........................................
    F.) Collect Trace/Log
    ..........................................
    G.) Data Collector Setting
```

```
      (External standalone/cluster analytics server)
    ........................................
    H.) Setup SSH Key for external JunosVM  setup
    ........................................
    I.) Internal Analytics Setting (HA)
    ........................................
    X.) Exit
    ........................................
Please select a letter to execute.
```

6. Select **I** to proceed. This menu option applies the settings you have already configured for your NorthStar HA cluster, so you do not need to make any changes.

```
    Internal Analytics Configuration HA Settings:
    ******************************************************
    Note: This configuration only applicable for analytics
    installation in the same server
    ******************************************************
   ......................................................
   Node #1
       Hostname                        : ns03
       Priority                        : 10
       Cluster Communication Interface : eth2
       Cluster Communication IP        : 172.16.18.13
          Interfaces
            Interface #1
               Name                    : eth2
               IPv4                    : 172.16.18.13
               Switchover              : yes
            Interface #2
               Name                    : mgmt0
               IPv4                    :
               Switchover              : yes
            Interface #3
            Interface #4
            Interface #5
   Node #2
       Hostname                        : ns04
       Priority                        : 20
       Cluster Communication Interface : eth2
       Cluster Communication IP        : 172.16.18.14
          Interfaces
```

```
            Interface #1
               Name                    : eth2
               IPv4                    : 172.16.18.14
               Switchover              : yes
            Interface #2
             Name                    : mgmt0
               IPv4                    :
               Switchover              : yes
            Interface #3
            Interface #4
            Interface #5
   Node #3
       Hostname                        : ns05
       Priority                        : 30
       Cluster Communication Interface : eth2
       Cluster Communication IP        : 172.16.18.15
          Interfaces
            Interface #1
               Name                    : eth2
               IPv4                    : 172.16.18.15
               Switchover              : yes
            Interface #2
               Name                    : mgmt0
               IPv4                    :
               Switchover              : yes
            Interface #3
            Interface #4
            Interface #5

   ..........................................................
   1.) Prepare and Deploy Internal Analytics HA configs
   ..........................................................

Please select a number to modify.
[<CR>=return to main menu]:
```

NOTE: Depending on the geographical location of the nodes, you might want to use analytics geo-HA instead of setting up internal analytics. In that case, instead of selecting **I**, you would select **G** to access the Analytics Data Collector Configuration Settings. After updating those settings, select **C** (Prepare and Deploy GEO-HA Analytics Data Collector Setting). Step 7 below would not apply.

7. Select **1** to set up the NorthStar HA cluster for analytics.

```
WARNING !
The selected menu will restart analytics processes in each cluster member
Type YES to continue...
YES

Checking connectivity of cluster_communication_interface...
Cluster communications status for node ns03 cluster interface eth2 ip 172.16.18.13: OK
Cluster communications status for node ns04 cluster interface eth2 ip 172.16.18.14: OK
Cluster communications status for node ns05 cluster interface eth2 ip 172.16.18.15: OK


Verifying the NorthStar version on each node:
ns03 : NorthStar-Bundle-18.1.0-20180412_071430_72952_187.x86_64
ns04 : NorthStar-Bundle-18.1.0-20180412_071430_72952_187.x86_64
ns05 : NorthStar-Bundle-18.1.0-20180412_071430_72952_187.x86_64

Checking analytics process in each node ...
Detected analytics in node #1 ns03: OK
Detected analytics in node #2 ns04: OK
Detected analytics in node #3 ns05: OK

Applying analytics config files
Deploying analytics configuration in node #1 ns03
Deploying analytics configuration in node #2 ns04
Deploying analytics configuration in node #3 ns05


Restart Analytics at node #1 ns03

Restart Analytics at node #2 ns04

Restart Analytics at node #3 ns05


Internal analytics configurations has been applied successfully

Press any key to return to menu
```

8. On any analytics node, use the following command to check elasticsearch cluster status. Verify that the status is “green” and the number of nodes is correct.

```
[root@ns03 ~]# curl -XGET 'localhost:9200/_cluster/health?pretty'
{
  "cluster_name" : "NorthStar",
  "status" : "green",
  "timed_out" : false,
  "number_of_nodes" : 3,
  "number_of_data_nodes" : 3,
  "active_primary_shards" : 10,
  "active_shards" : 10,
  "relocating_shards" : 0,
  "initializing_shards" : 0,
  "unassigned_shards" : 0,
  "delayed_unassigned_shards" : 0,
  "number_of_pending_tasks" : 0,
  "number_of_in_flight_fetch" : 0,
  "task_max_waiting_in_queue_millis" : 0,
  "active_shards_percent_as_number" : 100.0
}
```

Troubleshooting Logs

The following logs are available to help with troubleshooting:

- /opt/northstar/logs/elasticsearch.msg
- /opt/northstar/logs/logstash.msg
- /opt/northstar/logs/logstash.log

See Logs in the *NorthStar Controller User Guide* for more information.

RELATED DOCUMENTATION

Installing the NorthStar Controller | 47

Configuring Routers to Send JTI Telemetry Data and RPM Statistics to the Data Collectors | 148

Configuring a NorthStar Cluster for High Availability | 161

Configuring Routers to Send JTI Telemetry Data and RPM Statistics to the Data Collectors

Junos Telemetry Interface (JTI) sensors generate data from the PFE (LSP traffic data, logical and physical interface traffic data), and will only send probes through the data plane. So, in addition to connecting the routing engine to the management network, a data port must be connected to the collector on one of your devices. The rest of the devices in the network can use that interface to reach the collector.

NOTE: You must use Junos OS Release 15.1F6 or later for NorthStar analytics.

To configure the routers, use the following procedure:

1. Configure the devices (both devices running Junos OS and Junos OS Evolved) for sending telemetry data to the Controller. On each device, the following configuration is required. The device needs to be set to enhanced-ip mode, which might require a full reboot.

   NOTE:

   - You must configure Junos OS devices that have MPC10 or higher linecards to send telemetry data. Junos OS devices that have line cards older than MPC10 do not need the configuration for sending telemetry data.

     Devices running newer releases of EVO and Junos OS devices with MPC10+ line cards need not be configured to send telemetry data.

   - For NorthStar to correctly process the JTI telemetry packets from MPC10 line cards on routers running Junos OS, ensure that you:

     - Configure remote-port in `[services analytics streaming-server server]` as 3000

     - Add `set services analytics export-profile profile-name payload-size 1400`

- If you are configuring for NorthStar integration with Paragon Insights (formerly HealthBot), use default remote-port 4000 and set the remote-address to the Paragon Insights server IP address.

```
set chassis network-services enhanced-ip
set services analytics streaming-server ns remote-address 192.168.10.100
set services analytics streaming-server ns remote-port 3000
set services analytics export-profile ns local-address 10.10.0.10
set services analytics export-profile ns reporting-rate 2
set services analytics export-profile ns format gpb
set services analytics export-profile ns transport udp
set services analytics sensor ifd server-name ns
set services analytics sensor ifd export-name ns
set services analytics sensor ifd resource /junos/system/linecard/interface/
set services analytics sensor ifl server-name ns
set services analytics sensor ifl export-name ns
set services analytics sensor ifl resource /junos/system/linecard/interface/logical/usage/
set services analytics sensor lsp server-name ns
set services analytics sensor lsp export-name ns
set services analytics sensor lsp resource /junos/services/label-switched-path/usage/
set services analytics sensor sr-te-color server-name ns
set services analytics sensor sr-te-color export-name ns
set services analytics sensor sr-te-color resource /junos/services/segment-routing/traffic-
engineering/ingress/usage/
set services analytics sensor sid server-name ns
set services analytics sensor sid export-name ns
set services analytics sensor sid resource /junos/services/segment-routing/sid/usage/
set services analytics sensor sr-te-tunnels server-name ns
set services analytics sensor sr-te-tunnels export-name ns
set services analytics sensor sr-te-tunnels resource /junos/services/segment-routing/traffic-
engineering/tunnel/ingress/usage/
set protocols mpls sensor-based-stats
set protocols source-packet-routing telemetry statistics
```

In this configuration, the remote address is the IP address of the collector (reachable though a data port). The local address should be the loopback, or router-id, whichever is configured on the device profile to identify the device.

2. NorthStar supports bandwidth sizing and container LSPs for SR-TE LSPs. Junos OS release 19.2R1 or later is required for this functionality. There is additional configuration required on the router to enable collection of segment routing data. For example:

```
set groups jvision services analytics sensor sr-te-tunnels server-name ns
set groups jvision services analytics sensor sr-te-tunnels export-name ns
set groups jvision services analytics sensor sr-te-tunnels resource /junos/services/segment-
routing/traffic-engineering/tunnel/ingress/usage/
```

3. Real-time performance monitoring (RPM) enables you to monitor network performance in real time and to assess and analyze network efficiency. To achieve this, RPM exchanges a set of probes with other IP hosts in the network for monitoring and network tracking purposes.

   Configure RPM probes to measure the interface delays. The following examples show the configuration of probes out of interface ge-0/1/1.0 to the remote address 10.101.105.2 (on devices running Junos OS) and et-0/0/3:0.0 to the remote address 10.10.10.1 (on devices running Junos OS Evolved). This remote address should be the IP address of the node at the other end of the link.

   NOTE: The test name must match the interface being measured (test ge-0/1/1.0 and test et-0/0/3:0.0, in the following examples).

   - The following is a sample to configure RPM probes in a device running Junos OS.

```
set services rpm probe northstar-ifl test ge-0/1/1.0 target address 10.101.105.2
set services rpm probe northstar-ifl test ge-0/1/1.0 probe-count 11
set services rpm probe northstar-ifl test ge-0/1/1.0 probe-interval 5
set services rpm probe northstar-ifl test ge-0/1/1.0 test-interval 60
set services rpm probe northstar-ifl test ge-0/1/1.0 source-address 10.101.105.1
set services rpm probe northstar-ifl test ge-0/1/1.0 moving-average-size 12
set services rpm probe northstar-ifl test ge-0/1/1.0 traps test-completion
set services rpm probe northstar-ifl test ge-0/1/1.0 hardware-timestamp
```

   - The following is a sample to configure RPM probes in a device running Junos OS Evolved.

```
set services monitoring rpm owner northstar-ifl test et-0/0/3:0.0 target 10.10.10.1
set services monitoring rpm owner northstar-ifl test et-0/0/3:0.0 source-address 10.10.10.2
set services monitoring rpm owner northstar-ifl test et-0/0/3:0.0 probe-count 15
set services monitoring rpm owner northstar-ifl test et-0/0/3:0.0 probe-interval 1
set services monitoring rpm owner northstar-ifl test et-0/0/3:0.0 test-interval 20
```

```
set services monitoring rpm owner northstar-ifl test et-0/0/3:0.0 history-size 512
set services monitoring rpm owner northstar-ifl test et-0/0/3:0.0 moving-average-size 60
```

4. Configure the syslog host using the following commands for both devices running Junos OS and Junos OS Evolved. :

   NOTE: IMPORTANT: To prevent the population of duplicate delay data to the PCS, do not perform this step if you are configuring for integration with Paragon Insights.

```
set system syslog host 192.168.18.1 daemon info
set system syslog host 192.168.18.1 port 1514
set system syslog host 192.168.18.1 match-strings RPM_TEST_RESULTS
```

5. RPM probes do not yet generate telemetry data, but you can use the **rpm-log.slax** script (for devices running Junos OS) or **evo-rpm-log.slax** (for devices running Junos OS Evolved) to push the results. The script is located in **/opt/northstar/data/logstash/utils/junoscripts**. Install the script to **/var/db/scripts/event** on the router.

   Use a text editor such as vi to make one addition to the **rpm-log.slax** script as follows (the bundled script will be updated in a future release):

```
/* Add embedded event policy to trigger the script */
var $event-definition = {
    <event-options> {
      <policy> {
        <name> “rpm-log”;
        <events> “ping_test_completed”;
        <events> “ping_test_failed”;     <- Add this line
        <then> {
          <event-script> {
            <name> “rpm-log.slax”;
            <output-format> “xml”;
```

   Enable the script by adding it to the event/scripts configuration:

   NOTE: IMPORTANT: To prevent the population of duplicate delay data to the PCS, do not perform this step if you are configuring for integration with Paragon Insights.

- To enable the **rpm-log.slax** script:

```
router> start shell csh command "chmod 770 /var/db/scripts/event/rpm-log"

router# set event-options event-script file rpm-log.slax
```

- To enable the **evo-rpm-log.slax** script:

```
router> start shell csh command "chmod 770 /var/db/scripts/event/evo-rpm-log.slax"

router# set event-options event-script file evo-rpm-log.slax
```

  Add the following lines to the **evo-rpm-log.slax** script to trigger the script to ping the device every sixty seconds:

```
router> set event-options generate-event ns-evo-rpm-log time-interval 60
router> set event-options policy ns-evo-rpm-log-policy events ns-evo-rpm-log then event-
script evo-rpm-log.slax
```

The text of the **rpm-log.slax** script is as follows. Comments are enclosed in /* */.

```
version 1.2;
ns junos = "http://xml.juniper.net/junos/*/junos";
ns xnm = "http://xml.juniper.net/xnm/1.1/xnm";
ns jcs = "http://xml.juniper.net/junos/commit-scripts/1.0"; import "../import
/junos.xsl";
param $test-owner = event-script-input/trigger-event/attribute-list/attribute
[name=="test-owner"]/value;
param $test-name = event-script-input/trigger-event/attribute-list/attribute
[name=="test-name"]/value;
param $delay-value;
var $arguments = {
  <argument> {
      <name> “test-name”;
      <description> “Name of the RPM test”;
  }
  <argument> {
     <name> “test-owner”;
     <description> “ Name of the RPM probe owner”;
```

```
  }
  <argument> {
     <name> “delay-value”;
     <description> “Delay value to send out, used to generate fake
data”;
  }
}
/* Add embedded event policy to trigger the script */
var $event-definition = {
    <event-options> {
      <policy> {
        <name> “rpm-log”;
        <events> “ping_test_completed”;
        <then> {
          <event-script> {
            <name> “rpm-log.slax”;
            <output-format> “xml”;
          }
        }
      }
    }
}
match / {
  <op-script-results> {
        /* Load Probe results */
        var $get-probe-resultsrpc = <get-probe-results> { <owner> $test-
owner; <test> $test-name;}
        var $probe-results = jcs:invoke($get-probe-resultsrpc);
        /* Extract data of interest */
        var $target-address = $probe-results/probe-test-results/target-address;
        var $probe-type = $probe-results/probe-test-results/probe-type;
        var $loss-percentage = format-number(number($probe-results/probe-test-
results/probe-test-moving-results/probe-test-generic-results/loss-percentage), '#.##');
        var $jitter = format-number(number($probe-results/probe-test-results/probe-
test-moving-results/probe-test-generic-results/probe-test-rtt/probe-summary-results/
jitter-delay) div 1000, '#.###');
        var $avg-delay = {
          if ($delay-value) {
                number($delay-value);
          } else {
                expr format-number(number($probe-results/probe-test-results/probe-test-
moving-results/probe-test-generic-results/probe-test-egress/probe-summary-results/avg-
delay) div 1000, '#.##');
```

```
          }
        }
        var $min-delay = {
          if ($delay-value) {
                number($delay-value);
          } else {
                expr format-number(number($probe-results/probe-test-results/probe-test-
moving-results/probe-test-generic-results/probe-test-egress/probe-summary-results/min-
delay) div 1000, '#.##');
          }
        }
        var $max-delay = {
          if ($delay-value) {
                number($delay-value);
          } else {
                expr format-number(number($probe-results/probe-test-results/probe-test-
moving-results/probe-test-generic-results/probe-test-egress/probe-summary-results/max-
delay) div 1000, '#.##');
          }
        }

        expr jcs:syslog("daemon.info","RPM_TEST_RESULTS: ","test-owner=",$test-owner,"
test-name=",$test-name," loss=",$loss-percentage," min-rtt=",$min-delay," max-rtt=",
$max-delay," avgerage-rtt=",$avg-delay," jitter=",$jitter);
  }
}
```

The text of the **evo-rpm-log.slax** script is as follows. Comments are enclosed in /* */.

```
version 1.2;

ns junos = "http://xml.juniper.net/junos/*/junos";
ns xnm = "http://xml.juniper.net/xnm/1.1/xnm";
ns jcs = "http://xml.juniper.net/junos/commit-scripts/1.0"; import "../import/junos.xsl";

var $connection = jcs:open();

/* Create the RPC to get the probe results */
var $get-rpm-rpc = <rpc> {
    <get-rpm-probe-results> ;
}
/* Get the probe results */
```

```
var $probe-results = jcs:execute($connection, $get-rpm-rpc);

match / {

    /* Load Probe results */
    for-each($probe-results//probe-test-results) {
        /* Extract data of interest */
        var $target-address = target-address;
        var $test-type = test-type;
        var $test-owner = owner-name;
        var $test-name = test-name;
        var $generic-aggregate-results = generic-aggregate-results;
        var $one-way = 'false';

        mvar $min-delay = 0;
        mvar $max-delay = 0;
        mvar $avg-delay = 0;
        mvar $jitter = 0;
        mvar $loss-percentage = 0;
        for-each(generic-aggregate-results) {
            if (aggregate-type == 'moving average') {
                set $loss-percentage = format-number(number(loss-percentage), '#.##');
                set $min-delay = -1;
                set $max-delay = -1;
                set $avg-delay = -1;
                set $jitter = -1;
                if (generic-aggregate-measurement) {
                    for-each(generic-aggregate-measurement) {
                        var $rtt_measurement = measurement-type[starts-with(normalize-
space(), "Round trip time")];
                        var $jitter_measurement = measurement-type[starts-with(normalize-
space(), "Round trip jitter")];
                        if ($rtt_measurement) {
                            set $min-delay = format-number(number(measurement-min) div 1000,
'#.##');
                            set $max-delay = format-number(number(measurement-max) div 1000,
'#.##');
                            set $avg-delay = format-number(number(measurement-avg) div 1000,
'#.##');
                        } else if ($jitter_measurement){
                            set $jitter = format-number(number(measurement-avg) div 1000,
'#.##');
                        }
```

```
                    }
                }
                expr jcs:syslog("daemon.info","RPM_TEST_RESULTS: ","test-owner=",$test-
owner," test-name=",$test-name," loss=",$loss-percentage," min-rtt=",$min-delay," max-
rtt=",$max-delay," avgerage-rtt=",$avg-delay," jitter=",$jitter," one-way=",$one-way);
            }
        }
    }
}
```

RELATED DOCUMENTATION

Installing Data Collectors for Analytics | 113

Collector Worker Installation Customization

When you install the NorthStar application, a default number of collector workers are installed on the NorthStar server, depending on the number of cores in the CPU. This is regulated in order to optimize server resources, but you can change the number by using a provided script. Each installed worker starts a number of celery processes equal to the number of cores in the CPU plus one.

Table 14 on page 156 describes the default number of workers installed according to the number of cores in the CPU.

Table 14: Default Worker Groups and Processes by Number of CPU Cores

CPU Cores	Worker Groups Installed	Total Worker Processes	Minimum RAM Required
1-4	4	8-20 (CPUs +1) x 4 = 20	1 GB
5-8	2	12-18 (CPUs +1) x 2 = 18	1 GB

Table 14: Default Worker Groups and Processes by Number of CPU Cores *(Continued)*

CPU Cores	Worker Groups Installed	Total Worker Processes	Minimum RAM Required
16	1	17 (CPUs +1) x 1 = 17	1 GB
32	1	33 (CPUs +1) x 1 = 33	2 GB

Use the config_celery_workers.sh script to change the number of worker groups installed (post-initial installation). You might want to make a change if, for example:

- You upgrade your hardware with additional CPU cores and you want to increase the worker groups based on the new total number of cores.
- You want to manually determine the number of workers to be started rather than using the automatically-applied formula.

"NorthStar Controller System Requirements" on page 6 provides some guidance about memory requirements for various server uses and sizes.

NOTE: You can also use the config_celery_workers.sh script to change the number of slave workers installed on a slave collector server. See "Secondary Collector Installation for Distributed Data Collection" on page 158 for more information about distributed data collection.

To change the number of worker groups installed, launch the config_celery_workers.sh script:

```
/opt/northstar/snmp-collector/scripts/config_celery_workers.sh <option>
```

The available options are:

- -c

  This option automatically determines the number of cores and calculates the number of worker groups to add accordingly, per the formulas in Table 14 on page 156.

For example:

```
/opt/northstar/snmp-collector/scripts/config_celery_workers.sh -c
```

- -w *worker-groups*

  This option adds the specified number of worker groups. The following example starts six worker groups:

```
/opt/northstar/snmp-collector/scripts/config_celery_workers.sh -w 6
```

RELATED DOCUMENTATION

Secondary Collector Installation for Distributed Data Collection | 158

Secondary Collector Installation for Distributed Data Collection

When you install NorthStar Controller, a primary collector is installed, for use by Netconf and SNMP collection. You can improve performance of the collection tasks by also installing secondary collector workers to distribute the work. Each secondary collector worker starts a number of worker processes which is equal to the number of cores in the CPU plus one. You can create as many secondary collector servers as you wish to help with collection tasks. The primary collector manages all of the workers automatically.

Secondary collectors must be installed in a separate server from the NorthStar Controller. **You cannot install secondary collectors together with the NorthStar application in the same server.**

To install secondary collectors, follow this procedure:

1. On the secondary collector server, run the following:

```
rpm -Uvh rpm-filename
```

2. On the secondary collector server, run the collector.sh script:

```
[root@ns-sec-coll]# cd /opt/northstar/northstar_bundle_x.x.x/
[root@ns-sec-coll northstar]# ./collector.sh install
```

The script prompts you for the NorthStar application IP address, login, and password. If the NorthStar application is in HA mode, you need to provide the VIP address of the NorthStar application. The IP address is used by the secondary collectors to communicate with the primary collector:

```
Config file /opt/northstar/data/northstar.cfg does not exist copying it from Northstar APP
server, Please enter below info:
-------------------------------------------------------------------------------------------
----------------------------
Please enter application server IP address or host name: 10.49.166.211
Please enter Admin Web UI username: admin
Please enter Admin Web UI password: <not displayed>
retrieving config file from application server...

Saving to /opt/northstar/data/northstar.cfg
Secondary collector installed....
collector: added process group
collector:worker1: stopped
collector:worker3: stopped
collector:worker2: stopped
collector:worker4: stopped
collector:worker1: started
collector:worker3: started
collector:worker2: started
collector:worker4: started
```

3. Run the following command to confirm the secondary collector (worker) processes are running:

```
[root@ns-sec-coll]# supervisorctl status
collector:worker1                RUNNING   pid 15574, uptime 0:01:28
collector:worker2                RUNNING   pid 15576, uptime 0:01:28
collector:worker3                RUNNING   pid 15575, uptime 0:01:28
collector:worker4                RUNNING   pid 15577, uptime 0:01:28
```

4. Optionally, use the config_celery_workers.sh script to change the number of workers that are installed.

The collector.sh script installs a default number of workers, depending on the number of CPU cores on the server. After the initial installation, you can change the number of workers installed using the config_celery_workers.sh script. Table 15 on page 160 shows the default workers installed, the number of total celery processes started, and the amount of RAM required.

Table 15: Default Worker Groups and Processes by Number of CPU Cores

CPU Cores	Worker Groups Installed	Total Worker Processes	Minimum RAM Required
1-4	4	8-20 (CPUs +1) x 4 = 20	1 GB
5-8	2	12-18 (CPUs +1) x 2 = 18	1 GB
16	1	17 (CPUs +1) x 1 = 17	1 GB
32	1	33 (CPUs +1) x 1 = 33	2 GB

To change the number of workers, run the config_celery_workers.sh script:

```
[root@pcs02-q-pod08 ~]#/opt/northstar/snmp-collector/scripts/config_celery_workers.sh <option>
```

Use the **-w *worker-groups*** option to add a specified number of worker groups. Since this installation is on a server dedicated to providing distributed data collection, you can increase the number of workers installed up to the server storage capacity to improve performance. The following example starts six worker groups:

```
/opt/northstar/snmp-collector/scripts/config_celery_workers.sh -w 6
```

RELATED DOCUMENTATION

Collector Worker Installation Customization | 156

Configuring a NorthStar Cluster for High Availability

IN THIS SECTION

Before You Begin

Configuring a NorthStar application cluster for high availability (HA) is an optional process. This topic describes the steps for configuring, testing, deploying, and maintaining an HA cluster. If you are not planning to use the NorthStar application HA feature, you can skip this topic.

NOTE: See High Availability Overview in the *NorthStar Controller User Guide* for overview information about HA. For information about analytics HA, see "Installing Data Collectors for Analytics" on page 113.

NOTE: Throughout your use of NorthStar Controller HA, be aware that you must replicate any changes you make to northstar.cfg to all cluster nodes so the configuration is uniform across the cluster. NorthStar CLI configuration changes, on the other hand, are replicated across the cluster nodes automatically.

- Download the NorthStar Controller and install it on each server that will be part of the cluster. Each server must be completely enabled as a single node implementation before it can become part of a cluster.

This includes:

- Creating passwords
- License verification steps
- Connecting to the network for various protocol establishments such as PCEP or BGP-LS

> **NOTE**: All of the servers must be configured with the same database and RabbitMQ passwords.

- All server time must be synchronized by NTP using the following procedure:

  1. Install NTP.

     ```
     yum -y install ntp
     ```

  2. Specify the preferred NTP server in ntp.conf.
  3. Verify the configuration.

     ```
     ntpq -p
     ```

  > **NOTE**: All cluster nodes must have the same time zone and system time settings. This is important to prevent inconsistencies in the database storage of SNMP and LDP task collection delta values.

- Run the net_setup.py utility to complete the required elements of the host and JunosVM configurations. Keep that configuration information available.

  > **NOTE**: If you are using an OpenStack environment, you will have one JunosVM that corresponds to each NorthStar Controller VM.

- Know the virtual IPv4 address you want to use for Java Planner client and web UI access to NorthStar Controller (required). This VIP address is configured for the router-facing network for single interface configurations, and for the user-facing network for dual interface configurations. This address is always associated with the active node, even if failover causes the active node to change.

- A virtual IP (VIP) is required when setting up a NorthStar cluster. Ensure that all servers that will be in the cluster are part of the same subnet as the VIP.
- Decide on the priority that each node will have for active node candidacy upon failover. The default value for all nodes is 0, the highest priority. If you want all nodes to have equal priority for becoming the active node, you can just accept the default value for all nodes. If you want to rank the nodes in terms of their active node candidacy, you can change the priority values accordingly—the lower the number, the higher the priority.

Set Up SSH Keys

Set up SSH keys between the selected node and each of the other nodes in the cluster, and each JunosVM.

1. Obtain the public SSH key from one of the nodes. You will need the ssh-rsa string from the output:

```
[root@rw01-ns ~]# cat /root/.ssh/id_rsa.pub
```

2. Copy the public SSH key from each node to each of the other nodes, from each machine.

   From node 1:

```
[root@rw01-ns northstar_bundle_x.x.x]# ssh-copy-id root@node-2-ip

[root@rw01-ns northstar_bundle_x.x.x]# ssh-copy-id root@node-3-ip
```

   From node 2:

```
[root@rw02-ns northstar_bundle_x.x.x]# ssh-copy-id root@node-1-ip

[root@rw02-ns northstar_bundle_x.x.x]# ssh-copy-id root@node-3-ip
```

   From node 3:

```
[root@rw03-ns northstar_bundle_x.x.x]# ssh-copy-id root@node-1-ip

[root@rw03-ns northstar_bundle_x.x.x]# ssh-copy-id root@node-2-ip
```

3. Copy the public SSH key from the selected node to each remote JunosVM (JunosVM hosted on each other node). To do this, log in to each of the other nodes and connect to its JunosVM.

```
[root@rw02-ns ~]# ssh northstar@JunosVM-ip
[root@rw02-ns ~]# configure
[root@rw02-ns ~]# set system login user northstar authentication ssh-rsa replacement-string
[root@rw02-ns ~]# commit
```

```
[root@rw03-ns ~]# ssh northstar@JunosVM-ip
[root@rw03-ns ~]# configure
[root@rw03-ns ~]# set system login user northstar authentication ssh-rsa replacement-string
[root@rw03-ns ~]# commit
```

Access the HA Setup Main Menu

The /opt/northstar/utils/net_setup.py utility (the same utility you use to configure NorthStar Controller) includes an option for configuring high availability (HA) for a node cluster. Run the /opt/northstar/utils/net_setup.py utility on one of the servers in the cluster to set up the entire cluster.

1. Select one of the nodes in the cluster on which to run the setup utility to configure all the nodes in the cluster.
2. On the selected node, launch the NorthStar setup utility to display the NorthStar Controller Setup Main Menu.

```
[root@northstar]# /opt/northstar/utils/net_setup.py
Main Menu:
    ..........................................
    A.) Host Setting
    ..........................................
    B.) JunosVM Setting
    ..........................................
    C.) Check Network Setting
    ..........................................
    D.) Maintenance & Troubleshooting
    ..........................................
    E.) HA Setting
    ..........................................
```

```
    F.) Collect Trace/Log
    ........................................
    G.) Analytics Data Collector Setting
      (External standalone/cluster analytics server)
    ........................................
    H.) Setup SSH Key for external JunosVM  setup
    ........................................
    I.) Internal Analytics Setting (HA)
    ........................................
    X.) Exit
    ........................................
Please select a letter to execute.
```

3. Type **E** and press **Enter** to display the HA Setup main menu.

   Figure 13 on page 166 shows the top portion of the HA Setup main menu in which the current configuration is listed. It includes the five supported interfaces for each node, the VIP addresses, and the ping interval and timeout values. In this figure, only the first of the nodes is included, but you would see the corresponding information for all three of the nodes in the cluster configuration template. HA functionality requires an odd number of nodes in a cluster, and a minimum of three.

> **NOTE**: If you have a cRPD installation, the JunosVM information is not displayed as it is not applicable.

Figure 13: HA Setup Main Menu, Top Portion

```
HA Setup:
   ------------------------------------------------------------
   Node #1
      Hostname                          :
      Site Name                         : site1
      Priority                          : 0
      Cluster Communication Interface   : external0
      Cluster Communication IP          :
        Interfaces
          Interface #1
            Name                        : external0
            IPv4                        :
            Switchover                  : yes
          Interface #2
            Name                        : mgmt0
            IPv4                        :
            Switchover                  : yes
          Interface #3
            Name                        :
            IPv4                        :
            Switchover                  : yes
          Interface #4
            Name                        :
            IPv4                        :
            Switchover                  : yes

...
   ------------------------------------------------------------
   JunosVM #1
       Hostname                         :
       IPv4                             :
   JunosVM #2
       Hostname                         :
       IPv4                             :
   JunosVM #3
       Hostname                         :
       IPv4                             :
   ------------------------------------------------------------
   VIP Interfaces
       VIP Interface #1                 :
       VIP Interface #2                 :
       VIP Interface #3                 :
       VIP Interface #4                 :
       VIP Interface #5                 :

   Ping Interval(s)                     : 10
   Ping Timeout(s)                      : 30
```

NOTE: If you are configuring a cluster for the first time, the IP addresses are blank and other fields contain default values. If you are modifying an existing configuration, the current cluster configuration is displayed, and you have the opportunity to change the values.

NOTE: If the servers are located in geodiverse locations, you can use Site Name to indicate which servers are in the same or different geographical locations.

Figure 14 on page 167 shows the lower portion of the HA Setup main menu. To complete the configuration, you type the number or letter of an option and provide the requested information. After each option is complete, you are returned to the HA Setup main menu so you can select another option.

Figure 14: HA Setup Main Menu, Lower Portion

```
.....
 1.) Add node
 2.) Remove node
 3.) Add JunosVM
 4.) Remove JunosVM
 5.) Modify Node
 6.) Modify Node interface
 7.) Delete Node interface data
 8.) Modify JunosVM
 9.) Modify VIP interfaces
 A.) Delete VIP interface data
 B.) Modify ping interval
 C.) Modify ping timeout
 .....
 D.) Setup Mode (single/cluster)   : single
 E.) PCEP Session (physical_ip/vip): physical_ip
 .....
 F.) Test HA Connectivity for cluster communication interface only
 G.) Test HA Connectivity for all interfaces
 H.) Prepare and Deploy HA configs
 I.) Copy HA setting to other nodes
 J.) Add a new node to existing cluster
 K.) Check cluster status
 .....

Please select a number to modify.
[<CR>=return to main menu]:
```

NOTE: If you have a cRPD installation, options 3, 4, and 8 are not displayed as they are not applicable. The remaining options are not renumbered.

Configure the Three Default Nodes and Their Interfaces

The HA Setup main menu initially offers three nodes for configuration because a cluster must have a minimum of three nodes. You can add more nodes as needed.

For each node, the menu offers five interfaces. Configure as many of those as you need.

1. Type **5** and press **Enter** to modify the first node.
2. When prompted, enter the number of the node to be modified, the hostname, the site name, and the priority, pressing **Enter** between entries.

   NOTE: The NorthStar Controller uses **root** as a username to access other nodes.

   The default priority is **0**. You can just press **Enter** to accept the default or you can type a new value.

   For each interface, enter the interface name, IPv4 address, and switchover (yes/no), pressing **Enter** between entries.

   NOTE: For each node, interface #1 is reserved for the cluster communication interface which is used to facilitate communication between nodes. For this interface, it is required that switchover be set to Yes, and you cannot change that parameter.

   When finished, you are returned to the HA Setup main menu.

   The following example configures Node #1 and two of its available five interfaces.

```
Please select a number to modify.
[<CR>=return to main menu]
5
Node ID : 1

    HA Setup:
       .....................................................
       Node #1
       Hostname                        :
       Site Name                       : site1
       Priority                        : 0
       Cluster Communication Interface : external0
       Cluster Communication IP        :
          Interfaces
            Interface #1
```

```
               Name                    : external0
               IPv4                    :
               Switchover              : yes
            Interface #2
               Name                    : mgmt0
               IPv4                    :
               Switchover              : yes
            Interface #3
               Name                    :
               IPv4                    :
               Switchover              : yes
            Interface #4
               Name                    :
               IPv4                    :
               Switchover              : yes
            Interface #5
               Name                    :
               IPv4                    :
               Switchover              : yes

current node 1 Node hostname (without domain name) :
new node 1 Node hostname (without domain name) : node-1

current node 1 Site Name : site1
new node 1 Site Name : site1

current node 1 Node priority : 0
new node 1 Node priority : 10

current node 1 Node cluster communication interface : external0
new node 1 Node cluster communication interface : external0

current node 1 Node cluster communication IPv4 address :
new node 1 Node cluster communication IPv4 address : 10.25.153.6


current node 1 Node interface #2 name : mgmt0
new node 1 Node interface #2 name : external1

current node 1 Node interface #2 IPv4 address :
new node 1 Node interface #2 IPv4 address : 10.100.1.1

current node 1 Node interface #2 switchover (yes/no) : yes
```

```
new node 1 Node interface #2 switchover (yes/no) :

current node 1 Node interface #3 name :
new node 1 Node interface #3 name :

current node 1 Node interface #3 IPv4 address :
new node 1 Node interface #3 IPv4 address :

current node 1 Node interface #3 switchover (yes/no) : yes
new node 1 Node interface #3 switchover (yes/no) :

current node 1 Node interface #4 name :
new node 1 Node interface #4 name :

current node 1 Node interface #4 IPv4 address :
new node 1 Node interface #4 IPv4 address :

current node 1 Node interface #4 switchover (yes/no) : yes
new node 1 Node interface #4 switchover (yes/no) :

current node 1 Node interface #5 name :
new node 1 Node interface #5 name :

current node 1 Node interface #5 IPv4 address :
new node 1 Node interface #5 IPv4 address :

current node 1 Node interface #5 switchover (yes/no) : yes
new node 1 Node interface #5 switchover (yes/no) :
```

3. Type **5** and press **Enter** again to repeat the data entry for each of the other two nodes.

Configure the JunosVM for Each Node

To complete the node-specific setup, configure the JunosVM for each node in the cluster.

1. From the HA Setup main menu, type **8** and press **Enter** to modify the JunosVM for a node.
2. When prompted, enter the node number, the JunosVM hostname, and the JunosVM IPv4 address, pressing **Enter** between entries.

   Figure 15 on page 171 shows these JunosVM setup fields.

Figure 15: Node 1 JunosVM Setup Fields

```
Please select a number to modify.
[<CR>=return to main menu]:
8
Node ID : 1

current node 1 JunOSVM hostname :
new node 1 JunOSVM hostname : junosVM_node1

current node 1 JunosVM IPv4 address :
new node 1 JunosVM IPv4 address : 172.25.152.238
```

When finished, you are returned to the HA Setup main menu.

3. Type **8** and press **Enter** again to repeat the JunosVM data entry for each of the other two nodes.

(Optional) Add More Nodes to the Cluster

If you want to add additional nodes, type **1** and press **Enter**. Then configure the node and the node's JunosVM using the same procedures previously described. Repeat the procedures for each additional node.

NOTE: HA functionality requires an odd number of nodes and a minimum of three nodes per cluster.

The following example shows adding an additional node, node #4, with two interfaces.

```
Please select a number to modify.
[<CR>=return to main menu]:
1
New Node ID : 4

current node 4 Node hostname (without domain name) :
new node 4 Node hostname (without domain name) : node-4

current node 4 Site Name : site1
```

```
new node 4 Site Name : site1

current node 4 Node priority : 0
new node 4 Node priority : 40

current node 4 Node cluster communication interface : external0
new node 4 Node cluster communication interface : external0

current node 4 Node cluster communication IPv4 address :
new node 4 Node cluster communication IPv4 address : 10.25.153.12


current node 4 Node interface #2 name : mgmt0
new node 4 Node interface #2 name : external1

current node 4 Node interface #2 IPv4 address :
new node 4 Node interface #2 IPv4 address : 10.100.1.7

current node 4 Node interface #2 switchover (yes/no) : yes
new node 4 Node interface #2 switchover (yes/no) :

current node 4 Node interface #3 name :
new node 4 Node interface #3 name :

current node 4 Node interface #3 IPv4 address :
new node 4 Node interface #3 IPv4 address :

current node 4 Node interface #3 switchover (yes/no) : yes
new node 4 Node interface #3 switchover (yes/no) :

current node 4 Node interface #4 name :
new node 4 Node interface #4 name :

current node 4 Node interface #4 IPv4 address :
new node 4 Node interface #4 IPv4 address :

current node 4 Node interface #4 switchover (yes/no) : yes
new node 4 Node interface #4 switchover (yes/no) :

current node 4 Node interface #5 name :
new node 4 Node interface #5 name :

current node 4 Node interface #5 IPv4 address :
```

```
new node 4 Node interface #5 IPv4 address :

current node 4 Node interface #5 switchover (yes/no) : yes
new node 4 Node interface #5 switchover (yes/no) :
```

The following example shows configuring the JunosVM that corresponds to node #4.

```
Please select a number to modify.
[<CR>=return to main menu]
3
New JunosVM ID : 4
current junosvm 4 JunOSVM hostname :
new junosvm 4 JunOSVM hostname : junosvm-4

current junosvm 4 JunOSVM IPv4 address :
new junosvm 4 JunOSVM IPv4 address : 10.25.153.13
```

Configure Cluster Settings

The remaining settings apply to the cluster as a whole.

1. From the HA Setup main menu, type **9** and press **Enter** to configure the VIP address for the external (router-facing) network. This is the virtual IP address that is always associated with the active node, even if failover causes the active node to change. The VIP is required, even if you are configuring a separate user-facing network interface. If you have upgraded from an earlier NorthStar release in which you did not have VIP for external0, you must now configure it.

   NOTE: Make a note of this IP address. If failover occurs while you are working in the NorthStar Planner UI, the client is disconnected and you must re-launch it using this VIP address. For the NorthStar Controller web UI, you would be disconnected and would need to log back in.

   The following example shows configuring the VIP address for the external network.

   ```
   Please select a number to modify.
   [<CR>=return to main menu]
   9
   current VIP interface #1 IPv4 address :
   ```

```
new VIP interface #1 IPv4 address : 10.25.153.100

current VIP interface #2 IPv4 address :
new VIP interface #2 IPv4 address : 10.100.1.1

current VIP interface #3 IPv4 address :
new VIP interface #3 IPv4 address :

current VIP interface #4 IPv4 address :
new VIP interface #4 IPv4 address :

current VIP interface #5 IPv4 address :
new VIP interface #5 IPv4 address :
```

2. Type **9** and press **Enter** to configure the VIP address for the user-facing network for dual interface configurations. If you do not configure this IP address, the router-facing VIP address also functions as the user-facing VIP address.
3. Type **D** and press **Enter** to configure the setup mode as **cluster** (local cluster).
4. Type **E** and press **Enter** to configure the PCEP session. The default is **physical_ip**. If you are using the cluster VIP for your PCEP session, configure the PCEP session as **vip**.

> **NOTE**: All of your PCC sessions must use either physical IP or VIP (no mixing and matching), and that must also be reflected in the PCEP configuration on the router.

Test and Deploy the HA Configuration

You can test and deploy the HA configuration from within the HA Setup main menu.

1. Type **G** to test the HA connectivity for all the interfaces. You must verify that all interfaces are up before you deploy the HA cluster.
2. Type **H** and press **Enter** to launch a script that connects to and deploys all the servers and all the JunosVMs in the cluster. The process takes approximately 15 minutes, after which the display is returned to the HA Setup menu. You can view the log of the progress at /opt/northstar/logs/net_setup.log.

> **NOTE**: If the execution has not completed within 30 minutes, a process might be stuck. You can sometimes see this by examining the log at /opt/northstar/logs/net_setup.log. You can press **Ctrl-C** to cancel the script, and then restart it.

3. To check if the election process has completed, examine the processes running on each node by logging into the node and executing the **supervisorctl status** script.

```
[root@node-1]# supervisorctl status
```

For the active node, you should see all processes listed as RUNNING as shown here.

NOTE: The actual list of processes depends on the version of NorthStar and your deployment setup.

```
[root@node-1 ~]# supervisorctl status
bmp:bmpMonitor                   RUNNING   pid 2957, uptime 0:58:02
collector:worker1                RUNNING   pid 19921, uptime 0:01:42
collector:worker2                RUNNING   pid 19923, uptime 0:01:42
collector:worker3                RUNNING   pid 19922, uptime 0:01:42
collector:worker4                RUNNING   pid 19924, uptime 0:01:42
collector_main:beat_scheduler    RUNNING   pid 19770, uptime 0:01:53
collector_main:es_publisher      RUNNING   pid 19771, uptime 0:01:53
collector_main:task_scheduler    RUNNING   pid 19772, uptime 0:01:53
config:cmgd                      RUNNING   pid 22087, uptime 0:01:53
config:cmgd-rest                 RUNNING   pid 22088, uptime 0:01:53
docker:dockerd                   RUNNING   pid 4368, uptime 0:57:34
epe:epeplanner                   RUNNING   pid 9047, uptime 0:50:34
infra:cassandra                  RUNNING   pid 2971, uptime 0:58:02
infra:ha_agent                   RUNNING   pid 9009, uptime 0:50:45
infra:healthmonitor              RUNNING   pid 9172, uptime 0:49:40
infra:license_monitor            RUNNING   pid 2968, uptime 0:58:02
infra:prunedb                    RUNNING   pid 19770, uptime 0:01:53
infra:rabbitmq                   RUNNING   pid 7712, uptime 0:52:03
infra:redis_server               RUNNING   pid 2970, uptime 0:58:02
infra:zookeeper                  RUNNING   pid 2965, uptime 0:58:02
ipe:ipe_app                      RUNNING   pid 2956, uptime 0:58:02
listener1:listener1_00           RUNNING   pid 9212, uptime 0:49:29
netconf:netconfd_00              RUNNING   pid 19768, uptime 0:01:53
northstar:anycastGrouper         RUNNING   pid 19762, uptime 0:01:53
northstar:configServer           RUNNING   pid 19767, uptime 0:01:53
northstar:mladapter              RUNNING   pid 19765, uptime 0:01:53
northstar:npat                   RUNNING   pid 19766, uptime 0:01:53
northstar:pceserver              RUNNING   pid 19441, uptime 0:02:59
northstar:privatet1vproxy        RUNNING   pid 19432, uptime 0:02:59
```

```
northstar:prpdclient              RUNNING   pid 19763, uptime 0:01:53
northstar:scheduler               RUNNING   pid 19764, uptime 0:01:53
northstar:topologyfilter          RUNNING   pid 19760, uptime 0:01:53
northstar:toposerver              RUNNING   pid 19762, uptime 0:01:53
northstar_pcs:PCServer            RUNNING   pid 19487, uptime 0:02:49
northstar_pcs:PCViewer            RUNNING   pid 19486, uptime 0:02:49
web:app                           RUNNING   pid 19273, uptime 0:03:18
web:gui                           RUNNING   pid 19280, uptime 0:03:18
web:notification                  RUNNING   pid 19272, uptime 0:03:18
web:proxy                         RUNNING   pid 19275, uptime 0:03:18
web:restconf                      RUNNING   pid 19271, uptime 0:03:18
web:resthandler                   RUNNING   pid 19275, uptime 0:03:18
```

For a standby node, processes beginning with “northstar”and “northstar_pcs” should be listed as STOPPED. Also, if you have analytics installed, some of the processes beginning with “collector” are STOPPED. Other processes, including those needed to preserve connectivity, remain RUNNING. An example is shown here.

NOTE: This is just an example; the actual list of processes depends on the version of NorthStar, your deployment setup, and the optional features you have installed.

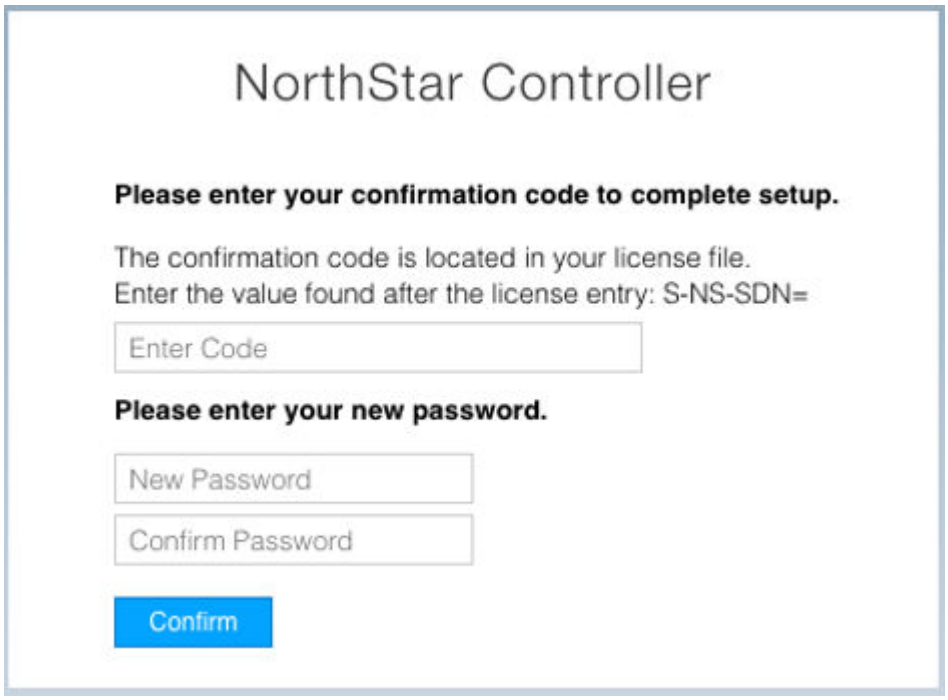
```
[root@node-1 ~]# supervisorctl status
bmp:bmpMonitor                RUNNING   pid 2957, uptime 0:58:02
collector:worker1                 RUNNING   pid 19921, uptime 0:01:42
collector:worker2                 RUNNING   pid 19923, uptime 0:01:42
collector:worker3                 RUNNING   pid 19922, uptime 0:01:42
collector:worker4                 RUNNING   pid 19924, uptime 0:01:42
collector:main:beat_scheduler     STOPPED   Dec 24, 05:12 AM
collector_main:es_publisher       STOPPED   Dec 24, 05:12 AM
collector_main:task_scheduler     STOPPED   Dec 24, 05:12 AM
config:cmgd                       STOPPED   Dec 24, 05:12 AM
config:cmgd-rest                  STOPPED   Dec 24, 05:12 AM
docker:dockerd                    RUNNING   pid 4368, uptime 0:57:34
epe:epeplanner                    RUNNING   pid 9047, uptime 0:50:34
infra:cassandra                   RUNNING   pid 2971, uptime 0:58:02
infra:ha_agent                    RUNNING   pid 9009, uptime 0:50:45
infra:healthmonitor               RUNNING   pid 9172, uptime 0:49:40
infra:license_monitor             RUNNING   pid 2968, uptime 0:58:02
infra:prunedb                     STOPPED   Dec 24, 05:12 AM
infra:rabbitmq                    RUNNING   pid 7712, uptime 0:52:03
infra:redis_server                RUNNING   pid 2970, uptime 0:58:02
```

```
infra:zookeeper                  RUNNING   pid 2965, uptime 0:58:02
ipe:ipe_app                      STOPPED   Dec 24, 05:12 AM
listener1:listener1_00           RUNNING   pid 9212, uptime 0:49:29
netconf:netconfd_00              RUNNING   pid 19768, uptime 0:01:53
northstar:anycastGrouper         STOPPED   Dec 24, 05:12 AM
northstar:configServer           STOPPED   Dec 24, 05:12 AM
northstar:mladapter              STOPPED   Dec 24, 05:12 AM
northstar:npat                   STOPPED   Dec 24, 05:12 AM
northstar:pceserver              STOPPED   Dec 24, 05:12 AM
northstar:privatet1vproxy        STOPPED   Dec 24, 05:12 AM
northstar:prpdclient             STOPPED   Dec 24, 05:12 AM
northstar:scheduler              STOPPED   Dec 24, 05:12 AM
northstar:topologyfilter         STOPPED   Dec 24, 05:12 AM
northstar:toposerver             STOPPED   Dec 24, 05:12 AM
northstar_pcs:PCServer           STOPPED   Dec 24, 05:12 AM
northstar_pcs:PCViewer           STOPPED   Dec 24, 05:12 AM
northstar_pcs:SRPCServer         STOPPED   Dec 24, 05:12 AM
web:app                          STOPPED   Dec 24, 05:12 AM
web:gui                          STOPPED   Dec 24, 05:12 AM
web:notification                 STOPPED   Dec 24, 05:12 AM
web:proxy                        STOPPED   Dec 24, 05:12 AM
web:restconf                     STOPPED   Dec 24, 05:12 AM
web:resthandler                  STOPPED   Dec 24, 05:12 AM
```

4. Set the web UI admin password using either the web UI or net_setup.

   - For the web UI method, use the external IP address that was provided to you when you installed the NorthStar application. Type that address into the address bar of your browser (for example, https://10.0.1.29:8443). A window is displayed requesting the confirmation code in your license file (the characters after S-NS-SDN=), and the password you wish to use. See Figure 16 on page 178.

Figure 16: Web UI Method for Setting the Web UI Password



- For the net_setup method, select **D** from the net_setup Main Menu (Maintenance & Troubleshooting), and then **3** from the Maintenance & Troubleshooting menu (Change UI Admin Password).

```
Main Menu:
    ..........................................
    A.) Host Setting
    ..........................................
    B.) JunosVM Setting
    ..........................................
    C.) Check Network Setting
    ..........................................
    D.) Maintenance & Troubleshooting
    ..........................................
    E.) HA Setting
    ..........................................
    F.) Collect Trace/Log
    ..........................................
    G.) Analytics Data Collector Setting
       (External standalone/cluster analytics server)
    ..........................................
    H.) Setup SSH Key for external JunosVM  setup
```

```
    ........................................
    I.) Internal Analytics Setting (HA)
    ........................................
    X.) Exit
    ........................................
Please select a letter to execute.
D


Maintenance & Troubleshooting:
   ..............................................
   1.) Backup JunosVM Configuration
   2.) Restore JunosVM Configuration
   3.) Change UI Admin Password
   4.) Change Database Password
   5.) Change MQ Password
   6.) Change Host Root Password
   7.) Change JunosVM root and northstar User Password
   8.) Initialize all credentials ( 3,4,5,6,7 included)
   ..............................................

Please select a number to modify.

[<CR>=return to main menu]:
3
```

Type Y to confirm you wish to change the UI Admin password, and enter the new password when prompted.

```
Change UI Admin Password
Are you sure you want to change the UI Admin password? (Y/N) y

Please enter new UI Admin password :
Please confirm new UI Admin password :
Changing UI Admin password ...
UI Admin password has been changed successfully
```

5. Once the web UI admin password has been set, return to the HA Setup menu (select **E** from the Main Menu). View cluster information and check the cluster status by typing **K**, and pressing **Enter**. In

addition to providing general cluster information, this option launches the ns_check_cluster.sh script. You can also run this script outside of the setup utility by executing the following commands:

```
[root@northstar]# cd /opt/northstar/utils/
[root@northstar utils]# ./ns_check_cluster.sh
```

Replace a Failed Node if Necessary

On the HA Setup menu, options I and J can be used when physically replacing a failed node. They allow you to replace a node without having to redeploy the entire cluster which would wipe out all the data in the database.

CAUTION: While a node is being replaced in a three-node cluster, HA is not guaranteed.

1. Replace the physical node in the network and install NorthStar Controller on the replacement node.
2. Run the NorthStar setup utility to configure the replaced node with the necessary IP addresses. Be sure you duplicate the previous node setup, including:
   - IP address and hostname
   - Initialization of credentials
   - Licensing
   - Network connectivity
3. Go to one of the existing cluster member nodes (preferably the same node that was used to configure the HA cluster initially). Going forward, we will refer to this node as the *anchor node*.
4. Set up the SSH key from the anchor node to the replacement node and JunosVM.

   Copy the public SSH key from the anchor node to the replacement node, from the replacement node to the other cluster nodes, and from the other cluster nodes to the replacement node.

NOTE: Remember that in your initial HA setup, you had to copy the public SSH key from each node to each of the other nodes, *from each machine*.

Copy the public SSH key from the anchor node to the replacement node's JunosVM (the JunosVM hosted on each of the other nodes). To do this, log in to each of the replacement nodes and connect to its JunosVM.

```
[root@node-1 ~]# ssh northstar@JunosVM-ip
[root@node-1 ~]# configure
[root@node-1 ~]# set system login user northstar authentication ssh-rsa replacement-string
[root@node-1 ~]# commit
```

5. From the anchor node, remove the failed node from the Cassandra database. Run the command `nodetool removenode host-id`. To check the status, run the command `nodetool status`.

   The following example shows removing the failed node with IP address 10.25.153.10.

```
[root@node-1 ~]# . /opt/northstar/northstar.env
[root@node-1 ~]# nodetool status
Datacenter: datacenter1
=======================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
--  Address        Load       Tokens       Owns    Host ID                               Rack
UN  10.25.153.6  5.06 MB    256          ?       507e572c-0320-4556-85ec-443eb160e9ba  rack1
UN  10.25.153.8  651.94 KB  256          ?       cd384965-cba3-438c-bf79-3eae86b96e62  rack1
DN  10.25.153.10  4.5 MB     256          ?       b985bc84-e55d-401f-83e8-5befde50fe96  rack1

 [root@node-1 ~]# nodetool removenode b985bc84-e55d-401f-83e8-5befde50fe96
[root@node-1 ~]# nodetool status
Datacenter: datacenter1
=======================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
--  Address        Load       Tokens       Owns    Host ID                               Rack
UN  10.25.153.6  5.06 MB    256          ?       507e572c-0320-4556-85ec-443eb160e9ba  rack1
UN  10.25.153.8  639.61 KB  256          ?       cd384965-cba3-438c-bf79-3eae86b96e62  rack1
```

6. From the HA Setup menu on the anchor node, select option I to copy the HA configuration to the replacement node.
7. From the HA Setup menu on the anchor node, select option J to deploy the HA configuration, only on the replacement node.

Configure Fast Failure Detection Between JunosVM and PCC

You can use Bidirectional Forward Detection (BFD) in deploying the NorthStar application to provide faster failure detection as compared to BGP or IGP keepalive and hold timers. The BFD feature is supported in PCC and JunosVM.

To utilize this feature, configure **bfd-liveness-detection minimum-interval *milliseconds*** on the PCC, and mirror this configuration on the JunosVM. We recommend a value of 1000 ms or higher for each cluster node. Ultimately, the appropriate BFD value depends on your requirements and environment.

Using a Remote Server for NorthStar Planner

IN THIS SECTION

You can install NorthStar Controller with a remote Planner server (a server separate from the NorthStar application server), to distribute the NorthStar Operator and NorthStar Planner server loads. This also helps ensure that the processes of each do not interfere with the processes of the other. The desktop Planner application then runs from the remote server. Be aware that to work in NorthStar Planner, you must still log in from the NorthStar Controller web UI login page.

NOTE: Using a remote server for NorthStar Planner does not make NorthStar Planner independent of NorthStar Controller. As of now, there is no standalone Planner.

We recommend using a remote Planner server if any of the following are true:

- Your network has more than 250 nodes
- You typically run multiple concurrent Planner users and/or multiple concurrent Planner sessions

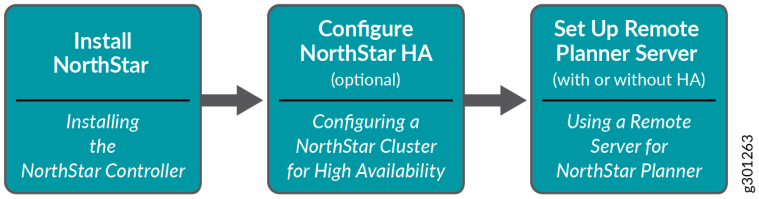
- You work extensively with Planner simulations

In NorthStar HA cluster setups, there is one remote Planner server for the whole cluster. It is configured to communicate with the virtual IP of the cluster.

Process Overview: Installing and Configuring Remote Planner Server

Figure 17 on page 183 shows the high level process you would follow to install NorthStar Controller with a remote Planner server. Installing and configuring NorthStar comes first. If you want a NorthStar HA cluster, you would set that up next. Finally, you would install and configure the remote Planner server. The text in italics indicates the topics in the *NorthStar Getting Started Guide* that cover the steps.

Figure 17: High Level Process Flow for Installing a Remote Planner Server



Setting up the remote Planner server is a two-step process that you perform on the remote server:

1. Run the remote Planner server installation script.

   This script prepares the remote server for the tasks that will be performed by the setup utility.

2. Run the remote Planner server setup utility.

   This utility copies your NorthStar license file from the NorthStar Controller to the remote Planner, and restarts the necessary NorthStar processes on the Controller. It disables Planner components on the NorthStar Controller application server so they can be run from the remote server instead. It also ensures that communication between the application server and the remote Planner server works. In an HA scenario, you run the setup utility once for each application server in the cluster, ending with the active node.

The setup utility requires you to provide the IP addresses of the NorthStar application server (physical IP or in the case of HA deployment, both the physical and virtual IP addresses) and the remote Planner server. Also, you must enter your server root password for access to the application server.

Download the Software to the Remote Planner Server

The NorthStar Controller software download page is available at https://www.juniper.net/support/downloads/?p=northstar#sw.

1. From the Version drop-down list, select the version number.
2. Click the NorthStar Application (which includes the RPM bundle) to download it. You do not need to download the Junos VM to the remote Planner server.

Install the Remote Planner Server

On the remote Planner server, execute the following commands:

```
[root@remote-server~]# yum install <rpm-filename>
[root@remote-server~]# cd /opt/northstar/northstar_bundle_x.x.x/
[root@remote-server~]# ./install-remote_planner.sh
```

Run the Remote Planner Server Setup Utility

This utility (setup_remote_planner.py) configures the NorthStar application server and the remote Planner server so the processes are distributed correctly between the two and so the servers can communicate with each other.

Three scenarios are described here. The effects on certain NorthStar components on the NorthStar Controller (application server) are as follows:

	Effect on Cassandra Database	Effect on Web Processes
Non-HA Setup Use this procedure if you do not have a NorthStar HA cluster configured.	modified and restarted	modified and restarted

(Continued)

	Effect on Cassandra Database	Effect on Web Processes
HA Active Node Setup Use this procedure to configure the active node in your NorthStar HA cluster.	not modified and not restarted	modified and restarted
HA Standby Node Setup Use this procedure to configure the standby nodes in your NorthStar HA cluster.	not modified and not restarted	modified, but not restarted

Non-HA Setup Procedure

The following procedure guides you through using the setup utility in interactive mode. As an alternative to interactive mode, you can launch the utility and enter the two required IP addresses directly in the CLI. The setup utility is located in **/opt/northstar/utils**/. To launch the utility:

```
[root@hostname~]# ./setup_remote_planner.py remote-planner-ip controller-ip
```

You will be required to enter your server root password to continue.

If you use interactive mode:

1. On the remote Planner server, launch the remote Planner server setup utility:

```
[root@hostname~]# ./setup_remote_planner.py
Remote Planner requires Controller to have port 6379 and 9042 opened.
```

2. Respond to the prompts. You'll need to provide the IP addresses for the NorthStar application server and the remote Planner server, and your server root password. For example:

```
Is Controller running in HA setup? (Y/N) N
Controller IP Address for Remote Planner to connect to: 10.53.65.245
Remote Planner IP Address for Controller to connect to: 10.53.65.244

Cassandra on Controller will be restarted. Are you sure you want to proceed? (Y/N) Y
```

```
Controller 10.53.65.245 root password:
Configuring Controller...
```

3. The setup utility proceeds to configure the NorthStar Controller application server, configure the remote Planner server, test the Cassandra database connection, and check the Redis connection. The following sample output shows the progress you would see and indicates which processes end up on the application server and which on the remote Planner server. All processes should be RUNNING when the setup is complete.

```
Configuring Controller...
bmp:bmpMonitor                   RUNNING   pid 20150, uptime 12 days, 20:30:10
collector:worker1                RUNNING   pid 21794, uptime 12 days, 20:28:25
collector:worker2                RUNNING   pid 21796, uptime 12 days, 20:28:25
collector:worker3                RUNNING   pid 21795, uptime 12 days, 20:28:25
collector:worker4                RUNNING   pid 21797, uptime 12 days, 20:28:25
collector_main:beat_scheduler    RUNNING   pid 22721, uptime 12 days, 20:26:39
collector_main:es_publisher      RUNNING   pid 22722, uptime 12 days, 20:26:39
collector_main:task_scheduler    RUNNING   pid 22723, uptime 12 days, 20:26:39
config:cmgd                      RUNNING   pid 22087, uptime 12 days, 20:28:08
config:cmgd-rest                 RUNNING   pid 22088, uptime 12 days, 20:28:08
docker:dockerd                   RUNNING   pid 20167, uptime 12 days, 20:30:10
epe:epeplanner                   RUNNING   pid 20151, uptime 12 days, 20:30:10
infra:cassandra                  RUNNING   pid 26573, uptime 0:00:25
infra:ha_agent                   RUNNING   pid 20160, uptime 12 days, 20:30:10
infra:healthmonitor              RUNNING   pid 21760, uptime 12 days, 20:28:40
infra:license_monitor            RUNNING   pid 20161, uptime 12 days, 20:30:10
infra:prunedb                    RUNNING   pid 20157, uptime 12 days, 20:30:10
infra:rabbitmq                   RUNNING   pid 20159, uptime 12 days, 20:30:10
infra:redis_server               RUNNING   pid 20163, uptime 12 days, 20:30:10
infra:zookeeper                  RUNNING   pid 20158, uptime 12 days, 20:30:10
ipe:ipe_app                      RUNNING   pid 22720, uptime 12 days, 20:26:39
listener1:listener1_00           RUNNING   pid 21788, uptime 12 days, 20:28:29
netconf:netconfd_00              RUNNING   pid 22719, uptime 12 days, 20:26:39
northstar:anycastGrouper         RUNNING   pid 22714, uptime 12 days, 20:26:40
northstar:configServer           RUNNING   pid 22718, uptime 12 days, 20:26:39
northstar:mladapter              RUNNING   pid 22716, uptime 12 days, 20:26:40
northstar:npat                   RUNNING   pid 22717, uptime 12 days, 20:26:39
northstar:pceserver              RUNNING   pid 22566, uptime 12 days, 20:27:56
northstar:privatetlvproxy        RUNNING   pid 22582, uptime 12 days, 20:27:46
northstar:prpdclient             RUNNING   pid 22713, uptime 12 days, 20:26:40
northstar:scheduler              RUNNING   pid 22715, uptime 12 days, 20:26:40
northstar:topologyfilter         RUNNING   pid 22711, uptime 12 days, 20:26:40
```

```
northstar:toposerver             RUNNING   pid 22712, uptime 12 days, 20:26:40
northstar_pcs:PCServer           RUNNING   pid 22602, uptime 12 days, 20:27:35
northstar_pcs:PCViewer           RUNNING   pid 22601, uptime 12 days, 20:27:35
northstar_pcs:SRPCServer         RUNNING   pid 22603, uptime 12 days, 20:27:35
web:app                          RUNNING   pid 26877, uptime 0:00:12
web:gui                          RUNNING   pid 26878, uptime 0:00:12
web:notification                 RUNNING   pid 26876, uptime 0:00:12
web:proxy                        RUNNING   pid 26879, uptime 0:00:12
web:restconf                     RUNNING   pid 19271, uptime 0:00:28
web:resthandler                  RUNNING   pid 19275, uptime 0:00:12

Controller configuration completed!

Configuring Remote Planner...
infra:rabbitmq                   RUNNING   pid 3587, uptime 3 days, 2:53:21
northstar:npat                   RUNNING   pid 3585, uptime 3 days, 2:53:21
web:planner                      RUNNING   pid 6557, uptime 0:00:11
Remote Planner configuration completed!

Testing Cassandra connection...
Connected to Test Cluster at 10.49.128.199:9042.
Cassandra connection OK.

Checking Redis connection...
Redis port is listening.
```

HA Standby Node Setup

As in the non-HA scenario, you don't have to use interactive mode for the setup script. You can launch the utility and enter the required IP addresses directly in the CLI as follows (three IP addresses in the case of cluster nodes):

```
[root@hostname~]# ./setup_remote_planner.py remote-planner-ip controller-physical-ip controller-vip
```

NOTE: This is the same for both active and standby nodes. The setup utility automatically distinguishes between the two.

You will be required to enter your server root password to continue.

Whether you use interactive or direct CLI mode, you run the utility once for each standby node in the HA cluster, using the appropriate IP addresses for each node.

If you use interactive mode:

1. On the remote Planner server, launch the remote Planner server setup utility:

```
[root@hostname~]# ./setup_remote_planner.py
Remote Planner requires Controller to have port 6379 and 9042 opened.
```

2. Respond to the prompts. You'll need to provide both the virtual IP and physical IP addresses for the standby application node, the IP address for the remote Planner server, and your server root password. For example:

```
Remote Planner requires Controller to have port 6379 and 9042 opened.

Is Controller running in HA setup? (Y/N) Y
Controller Cluster Virtual IP Address for Remote Planner to connect to: 10.51.132.196
Controller Cluster Physical IP Address for Remote Planner to connect to: 10.51.132.206
Remote Planner IP Address for Controller to connect to: 10.51.132.206

Controller 10.49.237.69 root password:
```

3. The setup utility proceeds to configure the NorthStar Controller application server, configure the remote Planner server, test the Cassandra database connection, and check the Redis connection. The following sample output shows the progress you would see and indicates which processes end up on the application server and which on the remote Planner server. Because this is a standby node in an HA cluster configuration, it is normal and expected for some processes to be STOPPED when the setup is complete.

```
Configuring Controller...
analytics:elasticsearch          RUNNING   pid 7942, uptime 2:27:50
analytics:esauthproxy            RUNNING   pid 7946, uptime 2:27:50
analytics:logstash               RUNNING   pid 7945, uptime 2:27:50
analytics:netflowd               RUNNING   pid 7944, uptime 2:27:50
analytics:pipeline               RUNNING   pid 7943, uptime 2:27:50
bmp:bmpMonitor                   RUNNING   pid 14342, uptime 3:48:40
collector:worker1                RUNNING   pid 7172, uptime 2:32:31
collector:worker2                RUNNING   pid 7174, uptime 2:32:31
collector:worker3                RUNNING   pid 7173, uptime 2:32:31
collector:worker4                RUNNING   pid 7175, uptime 2:32:31
```

```
collector_main:beat_scheduler    STOPPED   Sep 28 01:43 PM
collector_main:es_publisher      STOPPED   Sep 28 01:43 PM
collector_main:task_scheduler    STOPPED   Sep 28 01:42 PM
config:cmgd                      STOPPED   Sep 28 01:43 PM
config:cmgd-rest                 STOPPED   Sep 28 01:43 PM
config:ns_config_monitor         STOPPED   Sep 28 01:43 PM
docker:dockerd                   RUNNING   pid 15181, uptime 3:48:22
epe:epeplanner                   RUNNING   pid 18685, uptime 3:38:46
infra:cassandra                  RUNNING   pid 3919, uptime 2:55:54
infra:ha_agent                   RUNNING   pid 6972, uptime 2:33:50
infra:healthmonitor              RUNNING   pid 7131, uptime 2:32:45
infra:license_monitor            RUNNING   pid 3916, uptime 2:55:54
infra:prunedb                    STOPPED   Sep 28 02:20 PM
infra:rabbitmq                   RUNNING   pid 3914, uptime 2:55:54
infra:redis_server               RUNNING   pid 3918, uptime 2:55:54
infra:zookeeper                  RUNNING   pid 3907, uptime 2:55:54
ipe:ipe_app                      STOPPED   Sep 28 01:43 PM
listener1:listener1_00           RUNNING   pid 7166, uptime 2:32:35
netconf:netconfd_00              STOPPED   Sep 28 01:42 PM
northstar:anycastGrouper         STOPPED   Sep 28 01:43 PM
northstar:configServer           STOPPED   Sep 28 01:36 PM
northstar:mladapter              STOPPED   Sep 28 01:36 PM
northstar:npat                   STOPPED   Sep 28 01:36 PM
northstar:pceserver              STOPPED   Sep 28 01:42 PM
northstar:prpdclient             STOPPED   Sep 28 01:36 PM
northstar:scheduler              STOPPED   Sep 28 01:36 PM
northstar:toposerver             STOPPED   Sep 28 01:36 PM
northstar_pcs:PCServer           STOPPED   Sep 28 01:36 PM
northstar_pcs:PCViewer           STOPPED   Sep 28 01:36 PM
web:app                          STOPPED   Not started
web:gui                          STOPPED   Not started
web:notification                 STOPPED   Not started
web:proxy                        STOPPED   Not started
web:resthandler                  STOPPED   Not started
Controller configuration completed!

Configuring Remote Planner...
infra:rabbitmq                   RUNNING   pid 3587, uptime 3 days, 3:03:30
northstar:npat                   RUNNING   pid 3585, uptime 3 days, 3:03:30
web:planner                      RUNNING   pid 6650, uptime 0:00:10
Remote Planner configuration completed!

Testing Cassandra connection...
```

```
Connected to NorthStar Cluster at 10.49.237.68:9042.
Cassandra connection OK.

Checking Redis connection...
Redis port is listening.
```

HA Active Node Setup

After you have completed the remote server setup for all the standby nodes, you are ready to run setup_remote_planner.py for the active node in the cluster.

If you launch the utility and enter the three required IP addresses directly in the CLI, it looks like this:

```
[root@hostname~]# ./setup_remote_planner.py remote-planner-ip controller-physical-ip controller-vip
```

NOTE: This is the same for both active and standby nodes. The setup utility automatically distinguishes between the two.

You will be required to enter your server root password to continue.

If you use interactive mode:

1. On the remote Planner server, launch the remote Planner server setup utility:

```
[root@hostname~]# ./setup_remote_planner.py
Remote Planner requires Controller to have port 6379 and 9042 opened.
```

2. Respond to the prompts. You'll need to provide both the virtual IP and physical IP addresses for the active application node, the IP address for the remote Planner server, and your server root password. For example:

```
Controller Cluster Virtual IP Address for Remote Planner to connect to: 10.49.237.68
Controller Cluster Physical IP Address for Remote Planner to connect to: 10.49.237.70
Remote Planner IP Address for Controller to connect to: 10.49.128.198

Controller 10.49.237.70 root password:
```

3. The setup utility proceeds to configure the NorthStar Controller application server, configure the remote Planner server, test the Cassandra database connection, and check the Redis connection. The

following sample output shows the progress you would see and highlights which processes end up on the application server and which on the remote Planner server. Because this is the active node in an HA cluster configuration, all processes should be RUNNING when the setup is complete.

```
Configuring Controller...
analytics:elasticsearch          RUNNING   pid 28516, uptime 2:38:47
analytics:esauthproxy            RUNNING   pid 28520, uptime 2:38:47
analytics:logstash               RUNNING   pid 28519, uptime 2:38:47
analytics:netflowd               RUNNING   pid 28518, uptime 2:38:47
analytics:pipeline               RUNNING   pid 28517, uptime 2:38:47
bmp:bmpMonitor                   RUNNING   pid 14360, uptime 3:57:22
collector:worker1                RUNNING   pid 27521, uptime 2:40:34
collector:worker2                RUNNING   pid 27523, uptime 2:40:34
collector:worker3                RUNNING   pid 27522, uptime 2:40:34
collector:worker4                RUNNING   pid 27525, uptime 2:40:34
collector_main:es_publisher      RUNNING   pid 27388, uptime 2:40:48
collector_main:task_scheduler    RUNNING   pid 27389, uptime 2:40:48
config:cmgd                      RUNNING   pid 26496, uptime 2:42:18
config:cmgd-rest                 RUNNING   pid 26498, uptime 2:42:18
config:ns_config_monitor         RUNNING   pid 26497, uptime 2:42:18
docker:dockerd                   RUNNING   pid 15199, uptime 3:57:02
epe:epeplanner                   RUNNING   pid 6764, uptime 3:47:03
infra:cassandra                  RUNNING   pid 27016, uptime 3:10:43
infra:ha_agent                   RUNNING   pid 26101, uptime 2:43:27
infra:healthmonitor              RUNNING   pid 26074, uptime 2:43:38
infra:license_monitor            RUNNING   pid 27010, uptime 3:10:43
infra:prunedb                    RUNNING   pid 26059, uptime 2:43:49
infra:rabbitmq                   RUNNING   pid 27007, uptime 3:10:43
infra:redis_server               RUNNING   pid 27012, uptime 3:10:43
infra:zookeeper                  RUNNING   pid 27006, uptime 3:10:43
ipe:ipe_app                      RUNNING   pid 27387, uptime 2:40:48
listener1:listener1_00           RUNNING   pid 29161, uptime 3:01:56
netconf:netconfd_00              RUNNING   pid 27386, uptime 2:40:48
northstar:anycastGrouper         RUNNING   pid 19762, uptime 0:01:53
northstar:configServer           RUNNING   pid 27385, uptime 2:40:48
northstar:mladapter              RUNNING   pid 27383, uptime 2:40:48
northstar:npat                   RUNNING   pid 27384, uptime 2:40:48
northstar:pceserver              RUNNING   pid 27154, uptime 2:41:54
northstar:prpdclient             RUNNING   pid 27381, uptime 2:40:48
northstar:scheduler              RUNNING   pid 27382, uptime 2:40:48
northstar:toposerver             RUNNING   pid 27380, uptime 2:40:48
northstar_pcs:PCServer           RUNNING   pid 27192, uptime 2:41:44
```

```
northstar_pcs:PCViewer           RUNNING   pid 27191, uptime 2:41:44
web:app                          RUNNING   pid 16817, uptime 0:00:11
web:gui                          RUNNING   pid 16818, uptime 0:00:11
web:notification                 RUNNING   pid 16816, uptime 0:00:11
web:proxy                        RUNNING   pid 16819, uptime 0:00:11
web:resthandler                  RUNNING   pid 16820, uptime 0:00:11
Controller configuration completed!

Configuring Remote Planner...
infra:rabbitmq                   RUNNING   pid 3587, uptime 3 days, 3:12:12
northstar:npat                   RUNNING   pid 3585, uptime 3 days, 3:12:12
web:planner                      RUNNING   pid 6777, uptime 0:00:10
Remote Planner configuration completed!

Testing Cassandra connection...
Connected to NorthStar Cluster at 10.49.237.68:9042.
Cassandra connection OK.

Checking Redis connection...
Redis port is listening.
```

Installing Remote Planner Server at a Later Time

An alternative to installing and setting up the remote Planner server when you initially install or upgrade NorthStar, is to add the remote Planner server later. In this case, be aware that because running setup_remote_planner.py in a non-HA scenario includes a restart of the Cassandra database, all existing Planner sessions should be saved and closed before starting the process. We recommend performing the setup during a maintenance window. In the case of a NorthStar cluster, the Cassandra database is not restarted.

In a network where remote Planner server is already installed and functioning, you might need to switch from one NorthStar Controller application server to another, such as if the server should go down. It's also possible for the IP address of the application server to change, which would interrupt communication between the application server and the remote Planner server. In those situations, you need only run the setup_remote_planner.py utility again from the remote Planner server, entering the new IP address of the application server.

RELATED DOCUMENTATION

Installing the NorthStar Controller | 47

Configuring a NorthStar Cluster for High Availability | 161

5

CHAPTER

Configuring Topology Acquisition and Connectivity Between the NorthStar Controller and the Path Computation Clients

Understanding Network Topology Acquisition on the NorthStar Controller

After you use BGP-LS to establish BGP peering between the Junos VM and one or more routers in the backbone network, the NorthStar Controller acquires real-time topology changes, which are recorded in the traffic engineering database (TED). To compute optimal paths through the network, the NorthStar Controller requires a consolidated view of the network topology. This routing view of the network includes the nodes, links, and their attributes (metric, link utilization bandwidth, and so on) that comprise the network topology. Thus, any router CLI configuration changes to IGP metric, RSVP bandwidth, Priority/Hold values, and so on are instantly available from the NorthStar Controller UI topology view.

To provide a network view, the NorthStar Controller runs Junos OS in a virtual machine (JunosVM) that uses routing protocols to communicate with the network and dynamically learn the network topology. To provide real-time updates of the network topology, the JunosVM, which is based on a virtual Route Reflector (vRR), establishes a BGP-LS peering session with one or more routers from the existing MPLS TE backbone network. A router from the MPLS TE backbone advertises its traffic engineering database (TED) in BGP-LS. The JunosVM receives real-time BGP-LS updates and forwards this topology data into the Network Topology Abstractor Daemon (NTAD), which is a server daemon that runs in the JunosVM.

The NorthStar Controller stores network topology data in the following routing tables:

- lsdist.0—stores the network topology from TED
- lsdist.1—stores the network topology from IGP database

NTAD then forwards a copy of the updated topology information to the Path Computation Server (PCS), which displays the live topology update from the NorthStar Controller UI.

To provide a real-time topology update of the network, you can configure direct IS-IS or OSPF adjacency between the NorthStar Controller and an existing MPLS TE backbone router, but we recommend that you use BGP-LS rather than direct IGP adjacency or IGP adjacency over GRE.

NOTE: The current BGP-LS implementation only considers TED information, and some IGP-specific attributes might not be forwarded during topology acquisition. The following IGP attributes are not forwarded:

- Link net mask.
- IGP metric (TED provides TE metric only).

> In some cases, using IS-IS or OSPF adjacency instead of BGP-LS might produce stale data because IS-IS and OSPF have a database lifetime period that is not automatically cleared when the adjacency is down. In this case, NTAD will export all information in the OSPF or IS-IS database to the NorthStar Path Computation Server (PCS), so the NorthStar Controller might show incorrect topology.

Starting with NorthStar 4.3.0, BGP Monitoring Protocol (BMP) can be used as an alternative to NTAD. BMP runs automatically when you install NorthStar, but is not used unless you configure NorthStar and the JunosVM to make it the active topology acquisition method.

Unlike NTAD, BMP is a standard protocol which has the advantage of relieving the user of responsibility for version control to prevent mismatches. BMP also has the potential to be more compatible than NTAD with third-party routers. The third party router needs to support BGP-LS and BMP, and receive topology via BGP-LS. One disadvantage however, is that BMP only has access to the lsdist.0 routing table while NTAD accesses both lsdist.0 and lsdist.1.

With BMP, NorthStar can obtain the topology information from the BGP-LS data. When using BMP, only traffic engineering entries (from the TED) are available. NTAD also provides IGP entries if the router is peering with the IGP area. Topology data learned via IGP is not available through BMP.

See "Configuring Topology Acquisition" on page 196 for information about configuring both NTAD and BMP.

RELATED DOCUMENTATION

Configuring Topology Acquisition | 196

Configuring Topology Acquisition

IN THIS SECTION

Overview

After you have successfully established a connection between the NorthStar Controller and the network, you can configure topology acquisition using Border Gateway Protocol Link State (BGP-LS) or an IGP (OSPF or IS-IS). For BGP-LS topology acquisition, you must configure both the NorthStar Controller and the PCC routers.

We recommend that you use BGP-LS instead of IGP adjacency because:

- The OSPF and IS-IS databases have lifetime timers. If the OSPF or IS-IS neighbor goes down, the corresponding database is not immediately removed, making it impossible for the NorthStar Controller to determine whether the topology is valid.
- Using BGP-LS minimizes the risk of making the Junos VM a transit router between AS areas if the GRE metric is not properly configured.
- Typically, the NorthStar Controller is located in a network operations center (NOC) data center, multihops away from the backbone and MPLS TE routers. This is easily accommodated by BGP-LS, but more difficult for IGP protocols because they would have to employ a tunneling mechanism such as GRE to establish adjacency.

NOTE: If BGP-LS is used, the Junos VM is configured to automatically accept any I-BGP session. However, you must verify that the Junos VM is correctly configured and that it has IP reachability to the peering router.

Before You Begin

Before you begin, complete the following tasks:

- Verify IP connectivity between a switch (or router) and the x86 appliance on which the NorthStar Controller software is installed.
- Configure the Network Topology Acquisition Daemon (NTAD). The NTAD forwards topology information from the network to the NorthStar application, and it must be running on the Junos VM.

  Use the following command to enable the NTAD:

```
junosVM# set protocols topology-export
```

Use the following command to verify that the NTAD is running; if the topology-export statement is missing, the match produces no results:

```
junosVM> show system processes extensive | match ntad
 2462 root         1  96    0  6368K  1176K select   1:41  0.00% ntad
```

- Configure BGP Monitoring Protocol (BMP) if you have decided to use BMP as an alternative to NTAD. BMP must be enabled on both the NorthStar and Junos VM sides.

  1. Use a text editing tool such as vi to modify the /opt/northstar/data/northstar.cfg file, changing topology_src_protocol from 1 (which is NTAD) to 2 (which is BMP):

     ```
     vi /opt/northstar/data/northstar.cfg
     .
     .
     .
     topology_src_protocol=2
     ```

  2. Restart toposerver so the change takes effect:

     ```
     supervisorctl restart northstar:toposerver
     ```

  3. On the Junos VM, disable NTAD by deleting the **protocols topology-export** statement.

  4. On the Junos VM, under “firewall”, configure the firewall filter to permit BMP TCP segments from NorthStar toward the Junos VM.

     NOTE: Be sure to insert “term bmp” before “term default-discard”.

     ```
     filter protect-re {
       term mgmt-intf {
           from {
               interface-set mgmt-intf;
           }
           then accept;
       }
      .
     ```

```
    .
    .
     term bmp {
         from {
            protocol tcp;
            port 10001;
         }
         then accept;
     }
     term default-discard {
         then {
             syslog;
             discard;
         }
     }
   }
```

5. On the Junos VM, under “routing options”, enable BMP:

```
    bmp {
        connection-mode active;
        monitor enable;
        station northstar {
            station-address station-address;
            station-port 10001;
        }
    }
}
```

Where, *station-address* is any IP address of the controller that can be accessed externally. If the JunosVM and the controller are present on the same network segment, the *station-address* must be in the shared network segment.

Configuring Topology Acquisition Using BGP-LS

IN THIS SECTION

Configure BGP-LS Topology Acquisition on the NorthStar Controller

To configure BGP-LS topology acquisition on the NorthStar Controller, perform the following configuration steps from the NorthStar Junos VM:

1. Initiate an SSH or a telnet session to the Junos VM external IP or management IP address.
2. Specify the autonomous system (AS) number for the node (BGP peer).

```
[edit routing-options]
user@northstar_junosvm# set autonomous-system AS_number
```

3. Specify the BGP group name and type for the node.

```
[edit protocols bgp]
user@northstar_junosvm# set group group_1 type internal
```

4. Specify a description for the BGP group for the node.

```
[edit protocols bgp group group_1]
user@northstar_junosvm# set description "NorthStar BGP-TE Peering"
```

5. Specify the address of the local end of a BGP session.

   This is the IP address for the Junos VM external IP address that is used to accept incoming connections to the Junos VM peer and to establish connections to the remote peer.

```
[edit protocols bgp group group_1]
user@northstar_junosvm# set local-address <junosVM IP address>
```

6. Enable the traffic engineering features for the BGP routing protocol.

```
[edit protocols bgp group group_1]
user@northstar_junosvm# set family traffic-engineering unicast
```

7. Specify the IP address for the neighbor router that connects with the NorthStar Controller.

```
[edit protocols bgp group group_1]
user@northstar_junosvm# set neighbor <router loopback IP address>
```

NOTE: You can specify the router loopback address if it is reachable by the BGP peer on the other end. But for loopback to be reachable, usually some IGP has to be enabled between the NorthStar Junos VM and the peer on the other end.

Configure the Peering Router to Support Topology Acquisition

To enable the NorthStar Controller to discover the network, you must add the following configuration on each router that peers with the NorthStar Controller. The NorthStar Junos VM must peer with at least one router from each area (autonomous system).

To enable topology acquisition, initiate a telnet session to each PCC router and add the following configuration:

1. Configure a policy.

```
[edit policy-options]
user@PE1# set policy-statement TE term 1 from family traffic-engineering
user@PE1# set policy-statement TE term 1 then accept
```

NOTE: This configuration is appropriate for both OSPF and IS-IS.

2. Import the routes into the traffic-engineering database.

```
[edit protocols mpls traffic-engineering database]
user@PE1# set import policy TE
```

3. Configure a BGP group by specifying the IP address of the router that peers with the NorthStar Controller as the local address (typically the loopback address) and the Junos VM external IP address as the neighbor.

```
[edit routing-options]
user@PE1# set autonomous-system AS Number
[edit protocols bgp group northstar]
user@PE1# set type internal
user@PE1# set description "NorthStar BGP-TE Peering"
user@PE1# set local-address <router-IP-address>
user@PE1# set family traffic-engineering unicast
user@PE1# set export TE
user@PE1# set neighbor <JunosVM IP-address>
```

Configuring Topology Acquisition Using OSPF

IN THIS SECTION

Configure OSPF on the NorthStar Controller

To configure OSPF on the NorthStar Controller:

1. Configure the policy.

```
[edit policy-options]
user@northstar_junosvm# set policy-statement TE term 1 from family traffic-engineering
user@northstar_junosvm# set policy-statement TE term 1 then accept
```

2. Populate the traffic engineering database.

```
[edit]
user@northstar_junosvm# set protocols mpls traffic-engineering database import policy TE
```

3. Configure OSPF.

```
[edit]
user@northstar_junosvm# set protocols ospf area area interface interface interface-type p2p
```

Configure OSPF over GRE on the NorthStar Controller

Once you have configured OSPF on the NorthStar Controller, you can take the following additional steps to configure OSPF over GRE:

1. Initiate an SSH or telnet session using the NorthStar Junos VM external IP address.
2. Configure the tunnel.

```
[edit interfaces]
user@northstar_junosvm# set gre unit 0 tunnel source local-physical-ip
user@northstar_junosvm# set gre unit 0  tunnel destination destination-ip
user@northstar_junosvm# set gre unit 0  family inet address tunnel-ip-addr
user@northstar_junosvm# set gre unit 0  family iso
user@northstar_junosvm# set gre unit 0  family mpls
```

3. Enable OSPF traffic engineering on the Junos VM and add the GRE interface to the OSPF configuration.

```
[edit protocols ospf]
user@northstar_junosvm# set traffic-engineering
user@northstar_junosvm# set area area interface gre.0 interface-type p2p
user@northstar_junosvm# set area area interface gre.0 metric 65530
```

Configuring Topology Acquisition Using IS-IS

IN THIS SECTION

Configure IS-IS on the NorthStar Controller

To configure IS-IS topology acquisition and enable IS-IS routing, perform the following steps on the NorthStar Junos VM:

1. Configure interfaces for IS-IS routing. For example:

```
[edit]
user@northstar_junosvm# set interfaces em0 unit 0 family inet address 172.16.16.2/24
user@northstar_junosvm# set interfaces em1 unit 0 family inet address 192.168.179.117/25
user@northstar_junosvm# set interfaces em0 unit 0 family inet address 172.16.16.2/24
user@northstar_junosvm# set interfaces em2 unit 0 family mpls
user@northstar_junosvm# set interfaces lo0 unit 0 family inet address 88.88.88.88/32 primary
user@northstar_junosvm# set routing-options static route 0.0.0.0/0 next-hop 192.168.179.126
user@northstar_junosvm# set routing-options autonomous-system 1001
```

2. Configure the policy.

```
[edit policy-options]
user@northstar_junosvm# set policy-statement TE term 1 from family traffic-engineering
user@northstar_junosvm# set policy-statement TE term 1 then accept
```

3. Populate the traffic engineering database.

```
[edit protocols]
user@northstar_junosvm# set mpls traffic-engineering database import policy TE
```

4. Configure IS-IS.

```
[edit protocols]
user@northstar_junosvm# set isis interface interface level level metric metric
user@northstar_junosvm# set isis interface interface point-to-point
```

Configure IS-IS over GRE on the NorthStar Controller

Once you have configured IS-IS on the NorthStar Controller, you can take the following additional steps to configure IS-IS over GRE:

1. Initiate an SSH or telnet session using the IP address for the NorthStar Junos VM external IP address.
2. Configure the tunnel.

```
[edit interfaces]
user@northstar_junosvm# set gre unit 0 tunnel source local-physical-ip
user@northstar_junosvm# set gre unit 0  tunnel destination destination
user@northstar_junosvm# set gre unit 0  family inet addresstunnel-ip-addr
user@northstar_junosvm# set gre unit 0  family iso
user@northstar_junosvm# set gre unit 0  family mpls
```

3. Add the GRE interface to the IS-IS configuration.

```
[edit protocols isis]
user@northstar_junosvm# set interface gre.0 level level metric 65530
user@northstar_junosvm# set interface gre.0 point-to-point
```

RELATED DOCUMENTATION

Configuring PCEP on a PE Router (from the CLI) | 206

Configuring PCEP on a PE Router (from the CLI)

IN THIS SECTION

A Path Computation Client (PCC) supports the configurations related to the Path Computation Element (PCE) and communicates with the NorthStar Controller, which by default is configured to accept a Path Computation Element Protocol (PCEP) connection from any source address. However, you must configure PCEP on each PE router to configure the router as a PCC and establish a connection between the PCC and the NorthStar Controller. A PCC initiates path computation requests, which are then executed by the NorthStar Controller.

Configuring a PE Router as a PCC

Each PCC in the network that the NorthStar Controller can access must be running a Junos OS release that is officially supported by the NorthStar Controller as designated in the *NorthStar Controller Release Notes* (jinstall 32 bit).

NOTE: For a PCEP connection, the PCC can connect to the NorthStar Controller using an in-band or out-of-band management network, provided that IP connectivity is established between the Path Computation Server (PCS) and the specified PCEP local address. In some cases, an additional static route might be required from the NorthStar Controller to reach the PCC, if the IP address is unreachable from the NorthStar Controller default gateway.

To configure a PE router as a PCC:

1. Enable external control of LSPs from the PCC router to the NorthStar Controller.

```
[edit protocols]
user@PE1# set mpls lsp-external-controller pccd
```

2. Specify the loopback address of the PCC router as the local address, for example:

```
[edit protocols]
user@PE1# set pcep pce northstar1 local-address 10.0.0.101
```

> **NOTE**: As a best practice, the router ID is usually the loopback address, but it is not necessarily configured that way.

3. Specify the NorthStar Controller (**northstar1**) as the PCE that the PCC connects to, and specify the NorthStar Controller host external IP address as the destination address.

```
[edit protocols]
user@PE1# set pcep pce northstar1 destination-ipv4-address 10.99.99.1
```

4. Configure the destination port for the PCC router that connects to the NorthStar Controller (PCE server) using the TCP-based PCEP.

```
[edit protocols]
user@PE1# set pcep pce northstar1 destination-port 4189
```

5. Configure the PCE type.

```
[edit protocols]
user@PE1# set pcep pce northstar1 pce-type active
user@PE1# set pcep pce northstar1 pce-type stateful
```

6. Enable LSP provisioning.

```
[edit protocols]
user@PE1# set pcep pce northstar1 lsp-provisioning
```

7. To verify that PCEP has been configured on the router, open a telnet session to access the router, and run the following commands:

```
user@PE1> show configuration protocols mpls
```

Sample output:

```
lsp-external-controller pccd;
```

```
user@PE1> show configuration protocols pcep
```

Sample output:

```
pce northstar1 {
    local-address 10.0.0.101;
    destination-ipv4-address 10.99.99.1;
    destination-port 4189;
    pce-type active-stateful;
    lsp-provisioning;
}
```

Setting the PCC Version for Non-Juniper Devices

The PCEP protocol used by the Junos OS and NorthStar Controller supports *PCEP Extensions for establishing relationships between sets of LSPs* (draft-minei-pce-association-group-00) which defines the format and usage of AssociationObject, the optional object that makes association between LSP groups possible. There are later versions of this draft that might be supported by other equipment vendors, which introduces the possibility of mismatch between AssociationObject formats. Such a mismatch could cause non-Juniper PCCs to discard LSP provisioning requests from NorthStar. To prevent this, we recommend that you configure all non-Juniper PCCs to omit AssociationObject altogether.

NOTE: The result of omitting AssociationObject in non-Juniper PCC configuration is that NorthStar cannot associate groups of LSPs on those devices. For example, you would not be able to associate a primary LSP with secondary LSPs or a primary LSP with standby LSPs. This does not affect NorthStar's ability to create associations between LSP groups on Juniper PCCs.

Omitting AssociationObject on non-Juniper PCCs involves updating the **pcc_version.config** file on the NorthStar server and activating the update on the non-Juniper PCCs, using the following procedure:

1. Edit the **pcc_version.config** file on the NorthStar server to include the IP addresses of all non-Juniper PCCs. For each IP address, specify **3** as the PCC version. PCC version 3 omits AssociationObject.

   The **pcc_version.config** file is located in **/opt/pcs/db/config/**. The syntax of the configuration is **ver=***ip_address*:*pcc_version*.

   For example:

   ```
   [root@northstar]# cat /opt/pcs/db/config/pcc_version.config
   ver=192.168.2.100:3
   ver=192.168.2.200:3
   ver=192.168.2.215:3
   ```

2. At the PCEP CLI (**pcep_cli** command at the NorthStar Linux shell), execute the **set pcc-version** command to activate the change in PCC version.

   Executing this command restarts the PCEP sessions to the non-Juniper PCCs, applying the new PCC version 3. You can then provision LSPs from the NorthStar UI.

Mapping a Path Computation Client PCEP IP Address

A Path Computation Client (PCC) supports the configurations related to the Path Computation Element (PCE) and communicates with the NorthStar Controller, which by default is configured to accept a PCEP connection from any source address. Use the Device Profile window in the NorthStar Controller web UI to map a PCEP IP address for a PCC device.

A PCEP IP address (the local address of the PCC) is required when both of the following are true:

- PCEP is established through an IP address that is not supplied in the TED, such as an out-of-band IP address that uses an fxp0 management interface.
- There is no PCC-owned or PCC-delegated LSP configured on the router.

Before you begin, you must perform the configuration steps described in "Configuring PCEP on a PE Router (from the CLI)" on page 206 to configure the PE router as a PCC and establish a connection between the PCC and the NorthStar Controller.

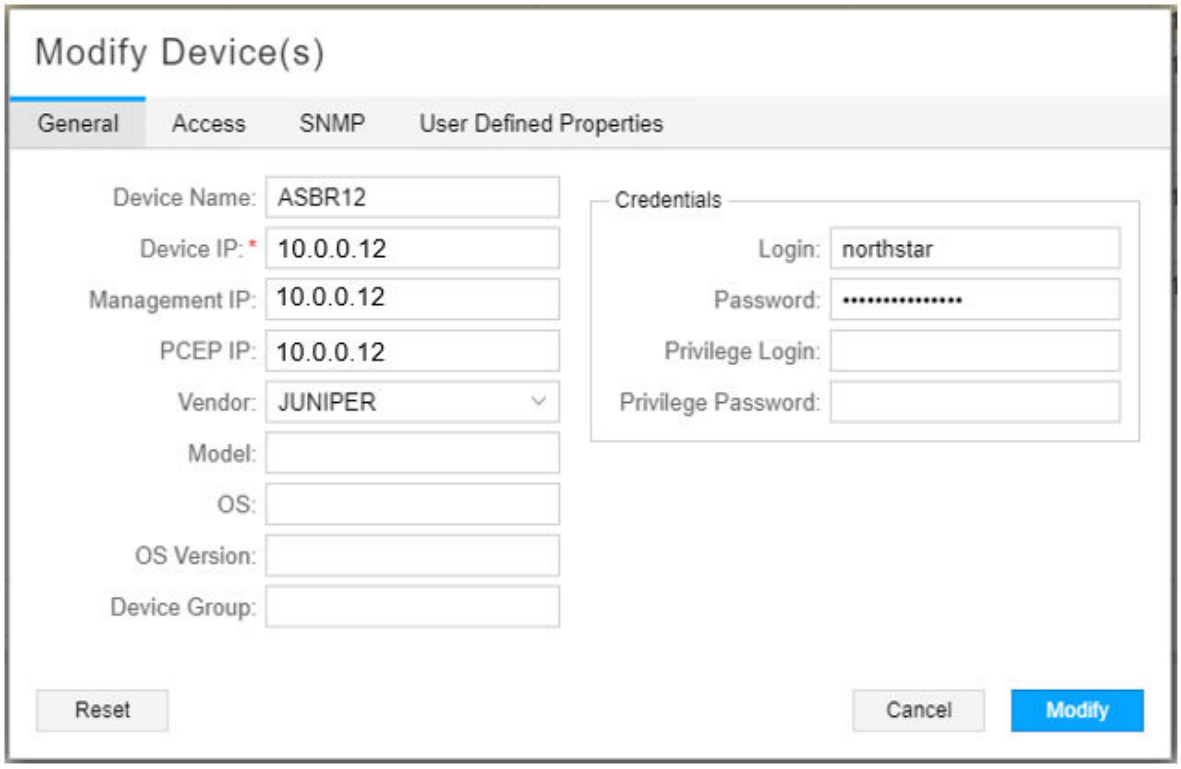
To map a PCEP IP address for a PCC to the NorthStar Controller:

1. Log in to the NorthStar Controller web UI.

2. Navigate to **More Options>Administration**.
3. From the Administration menu at the far left of the screen, select **Device Profile**.
4. The Device List pane shows all the devices in the selected profile along with many of their properties, including the PCEP IP address, if they are already known. If they are not already known, the fields are blank.

   To add or change a PCEP IP address, select the device row and click the Modify button. Figure 18 on page 210 shows the Modify Device window.

Figure 18: Modify Device Window



5. In the PCEP IP field, enter the PCEP IP address for the PCC.

   You can find the PCEP IP address in the PCE statement stanza block. Either of the following two CLI **show** commands can help you locate it:

```
northstar@vmx101> show path-computation-client statistics

PCE jnc
-----------------------------------------
General
    PCE IP address          : 172.25.152.134
```

```
    Local IP address         : 172.25.157.129
    Priority                 : 0
    PCE status               : PCE_STATE_UP
    Session type             : PCE_TYPE_STATEFULACTIVE
    LSP provisioning allowed : On
    PCE-mastership           : main

Counters
    PCReqs              Total: 0           last 5min: 0           last hour: 0
    PCReps              Total: 0           last 5min: 0           last hour: 0
    PCRpts              Total: 204         last 5min: 0           last hour: 0
    PCUpdates           Total: 9           last 5min: 0           last hour: 0
    PCCreates           Total: 21          last 5min: 0           last hour: 0

Timers
    Local  Keepalive timer:   30 [s]  Dead timer:  120 [s]  LSP cleanup timer:    0 [s]
    Remote Keepalive timer:   30 [s]  Dead timer:  120 [s]  LSP cleanup timer:    0 [s]

Errors
    PCErr-recv
    PCErr-sent
    PCE-PCC-NTFS
    PCC-PCE-NTFS
```

```
northstar@vmx101> show configuration protocols pcep
pce jnc {
    local-address 172.25.157.129;
    destination-ipv4-address 172.25.152.134;
    destination-port 4189;
    pce-type active stateful;
    lsp-provisioning;
}
```

6. Click **Submit**.
7. Repeat this process for each PCC device for which you want to map a PCEP IP address.

RELATED DOCUMENTATION

Configuring PCEP on a PE Router (from the CLI) | **206**

CHAPTER 6

Accessing the User Interface

NorthStar Application UI Overview

IN THIS SECTION

NorthStar has two user interfaces (UIs):

- NorthStar Controller—web UI for working with a live network.
- NorthStar Planner—Simulates the effect of various scenarios on the network, without affecting the live network. The Planner is currently in transition from a desktop application to a web UI. Until the transition is complete, both the full-featured desktop application and the in-development web UI are available and documented separately.

Comparison Between NorthStar Controller and NorthStar Planner

Table 16 on page 214 summarizes the major use cases for the NorthStar Controller and NorthStar Planner.

NOTE: All user administration (adding, modifying, and deleting users) must be done from the NorthStar Controller web UI.

NOTE: A subset of the Planner functionality shown here is currently available in the Planner web UI.

Table 16: Controller Versus Planner Comparison

NorthStar Controller (web client)	NorthStar Planner (Java client)
Manage, monitor, and provision a live network in real-time.	Design, simulate, and analyze a network offline.
Live network topology map shows node status, link utilization, and LSP paths.	Network topology map shows simulated or imported data for nodes, links, and LSP paths.
Network information table shows live status of nodes, links, and LSPs.	Network information table shows simulated or imported data for nodes, links, and LSPs.
Discover nodes, links, and LSPs from the live network using PCEP or NETCONF.	Import or add nodes, links, and LSPs for network modeling.
Provision LSPs directly to the network.	Add and stage LSPs for provisioning to the network.
Create or schedule maintenance events to re-route LSPs around the impacted nodes and links.	Create or schedule simulation events to analyze the network model from failure scenarios.
Dashboard reports shows current status and KPIs of the live network.	Report manager provides extensive reports for simulation and planning.
Analytics collects real-time interface traffic or delay statistics and stores the data for querying and chart displays.	Import interface data or aggregate archived data to generate historical statistics for querying and chart displays.

Browser Compatibility

For accessing the NorthStar Controller web UI, we recommend using Google Chrome and Mozilla Firefox browsers for Windows and Mac OS. We also recommend that you keep your browser updated to a recent version.

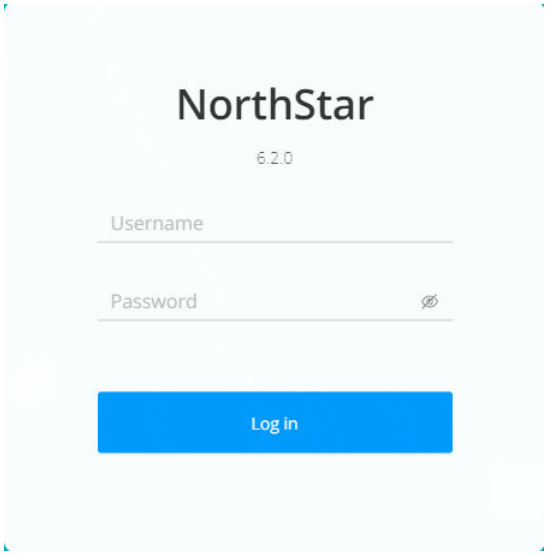
Logging in to NorthStar

Use this procedure to log in to the NorthStar controller. You can launch the NorthStar Planner (both web UI and desktop) from within the NorthStar Controller.

1. Enter your external IP address and port number that was provided to you when you installed NorthStar (for example, `https://10.0.1.29:8443`).

   The NorthStar login window is displayed, as shown in Figure 19 on page 215.

   Figure 19: NorthStar Login Window

2. Select **Operator**.

3. Enter your username and password, and click **SIGN IN**.

   You have now logged in to the NorthStar Controller.

4. If you want to log in to the NorthStar Planner, click the four-square icon in the top right corner and select **Planner** for the web UI or **Planner Desktop** for the desktop application. If you choose the web UI, a new tab opens in your browser. If you choose the desktop application, you will be prompted to download and run the desktop .jnlp executable.

NOTE: If you attempt to reach the login window, but instead, are routed to a message window that says, “Please enter your confirmation code to complete setup,” you must go to your license file and obtain the confirmation code as directed. Enter the confirmation code

along with your administrator password to be routed to the web UI login window. The requirement to enter the confirmation code only occurs if the installation process was not completed correctly and the NorthStar application needs to confirm that you have the authorization to continue.

WARNING: To avoid a Browser Exploit Against SSL/TLS (BEAST) attack, whenever you log in to NorthStar through a browser tab or window, make sure that the tab or window was not previously used to surf a non-HTTPS website. A best practice is to close your browser and relaunch it before logging in to NorthStar.

User Inactivity Timer

(System Administrator only) You can configure an inactivity timer and apply it to any user who is idle (and has not performed any actions (keystrokes or mouse clicks), so they are automatically logged out of NorthStar after a specified number of minutes. By default, the timer is disabled. To set the timer, select **System Settings** from the **Administration** menu.

NorthStar Controller Web UI Overview

IN THIS SECTION

You have two options for the NorthStar Controller Web UI: Classic, New.

The Classic option is also known as the Ext UI. *Ext* refers to the Extended Javascript (ExtJS) framework upon which this UI is based.

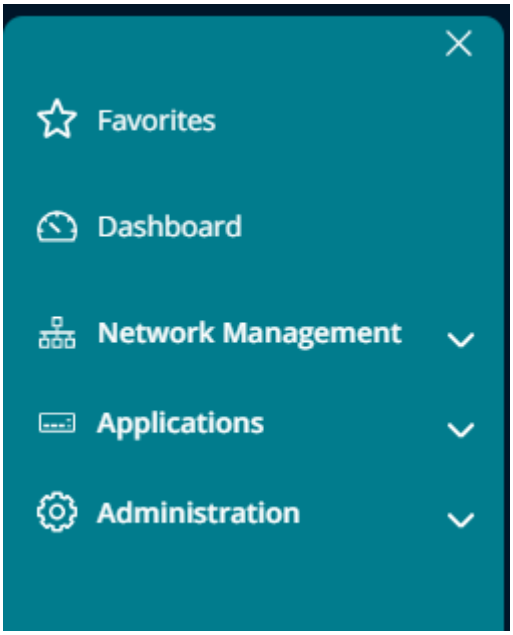
To switch between the two UI options, see "Switching Between the Classic and New UI" on page 224. This document shows screenshots from a mix of the two UIs.

The NorthStar Controller web UI has the following menus and sub-menus:

- Favorites
- Dashboard
- Network Management: Includes sub-menus for Topology, Nodes, Analytics, and Provisioning.
- Applications: Includes sub-menus for Bandwidth Calendar. EPE Planner, Network, Events, Path Optimization, Reports, and Maintenance.
- Administration: Includes sub-menus for Analytics, Authentication, Device Profile, License, Subscribers, System Health, System Settings, Task Scheduler, Transport Controller, Topology Filter, and Users.

Figure 20 on page 217 shows the navigation bar (or left-nav bar) that appears on the left side of each page. From this bar, you can select the different menu items and sub-menus. You can access the sub-menus by clicking the down arrows.

Figure 20: Web UI Main Menus

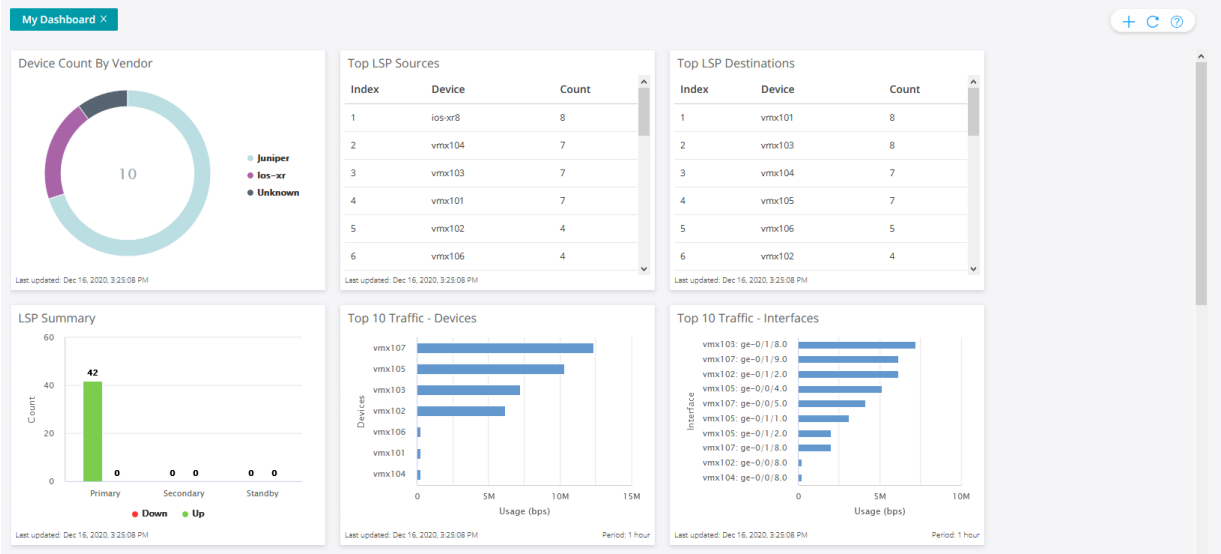


NOTE: The availability of some functions and features is dependent on user group permissions.

If you have certain pages that you use frequently, you can save them to Favorites in the left-nav bar for quick access.

The Dashboard (landing page) shows the network status and statistics information in the form of widgets. This page is displayed when you first log in to the web UI. Figure 21 on page 218 shows a sample of the available widgets.

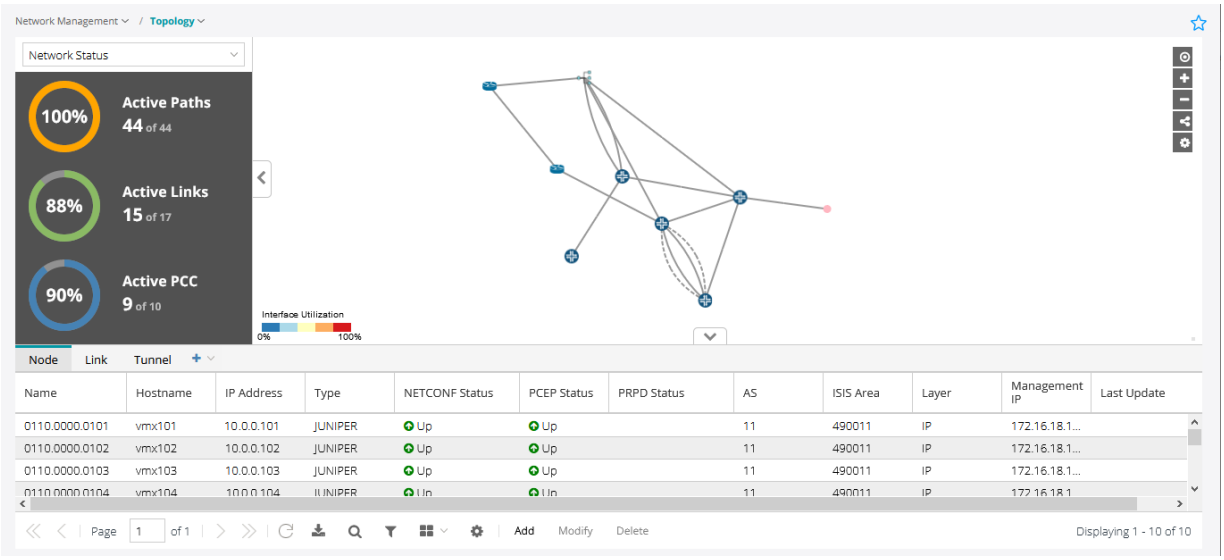
Figure 21: Dashboard



The **Network Management > Topology** page is the main work area for the live network you load into the system.

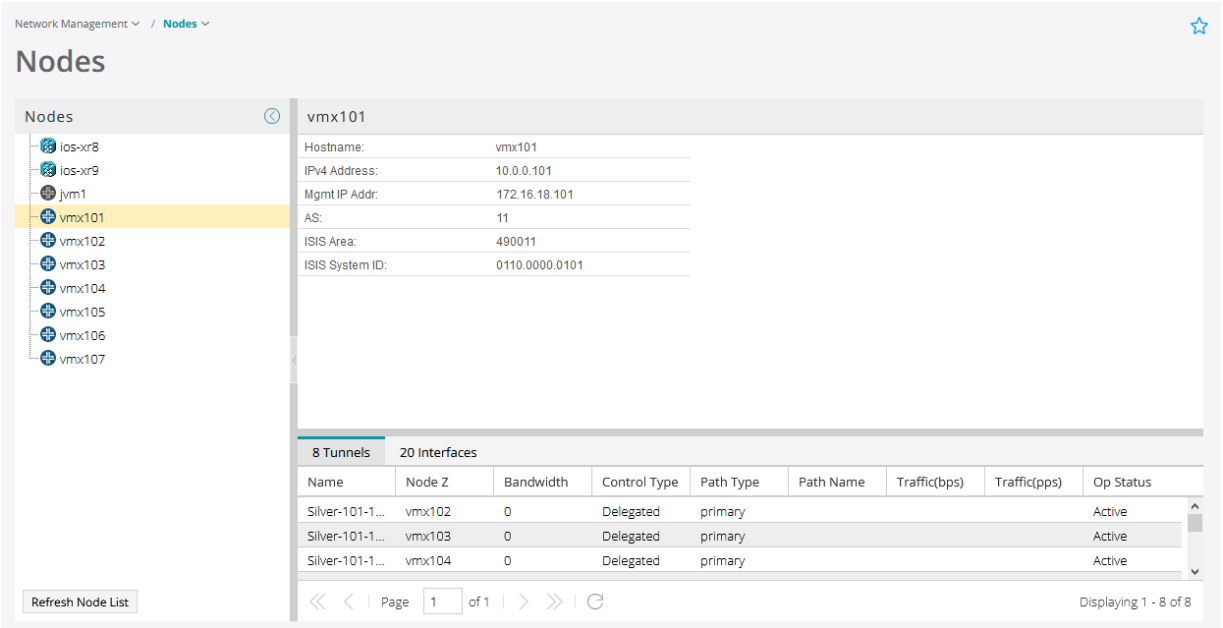
Figure 22 on page 219 shows an example of the Topology page.

Figure 22: Topology Page



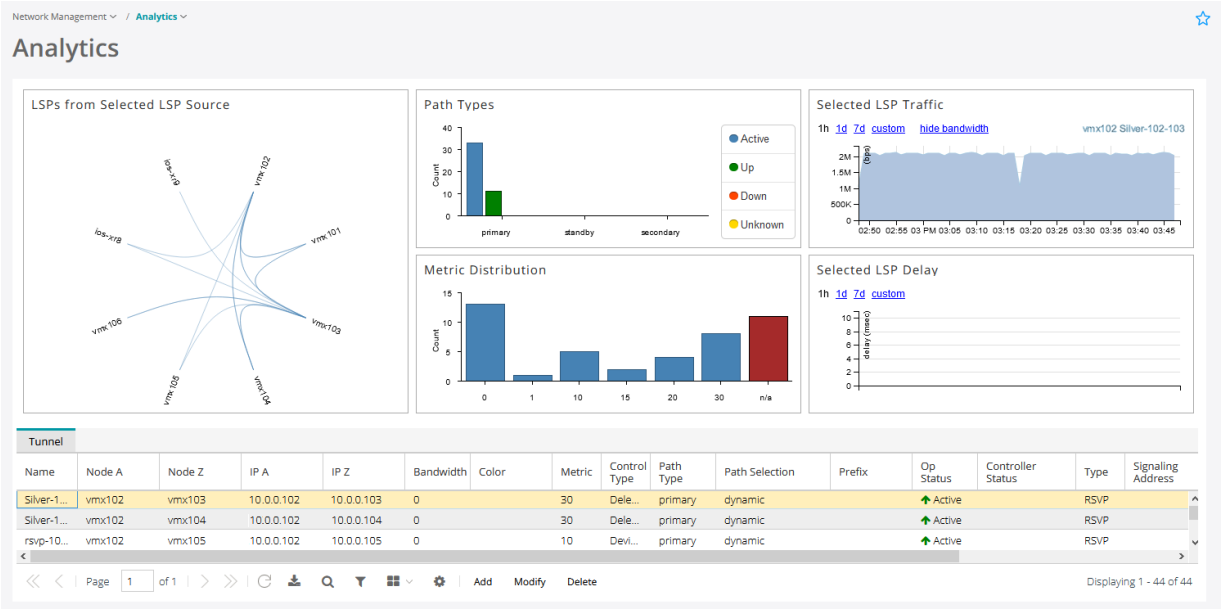
The **Network Management > Nodes** page, shown in Figure 23 on page 219, displays detailed information about the nodes in the network. With this view, you can see node details, tunnel and interface summaries, groupings, and geographic placement (if enabled), all in one place.

Figure 23: Nodes Page



The **Network Management > Analytics** page , shown in Figure 24 on page 220, provides a collection of quick-reference widgets related to analytics.

Figure 24: Analytics Page



The **Network Management > Provisioning** sub-menu has options for provisioning LSPs, configuring devices, and managing work orders (Figure 25 on page 220 through Figure 30 on page 223).

Figure 25: Provision LSP

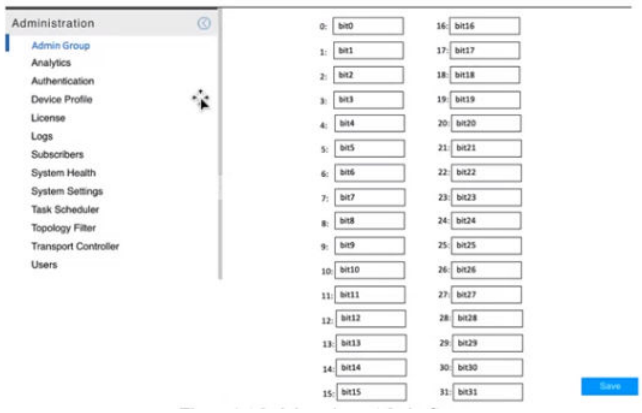


Figure 26: Provision Diverse LSP

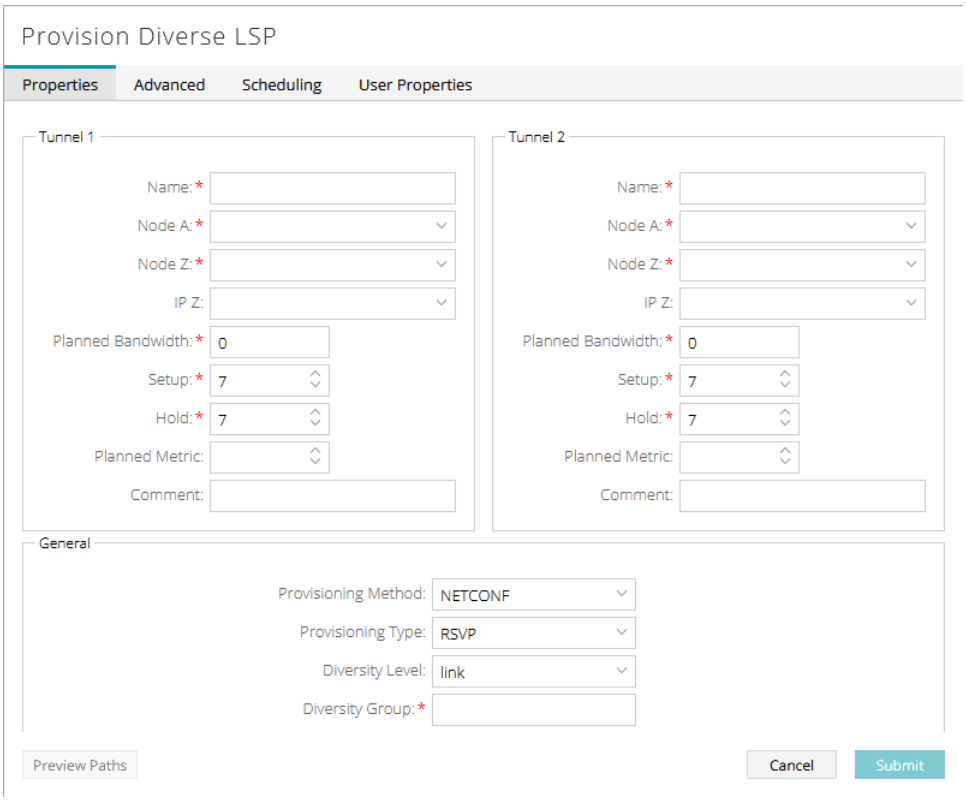


Figure 27: Provision Multiple LSPs

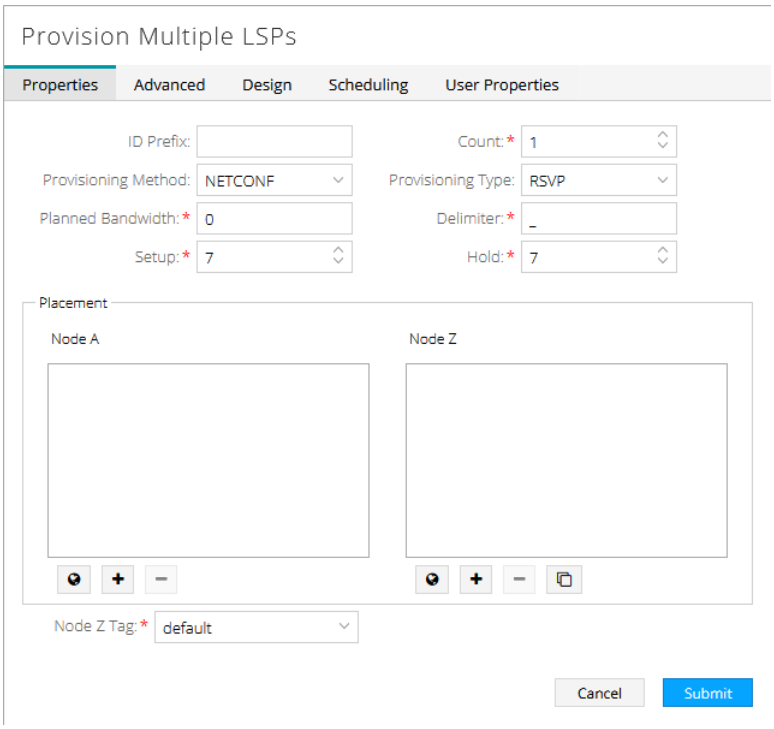


Figure 28: Configure LSP Delegation

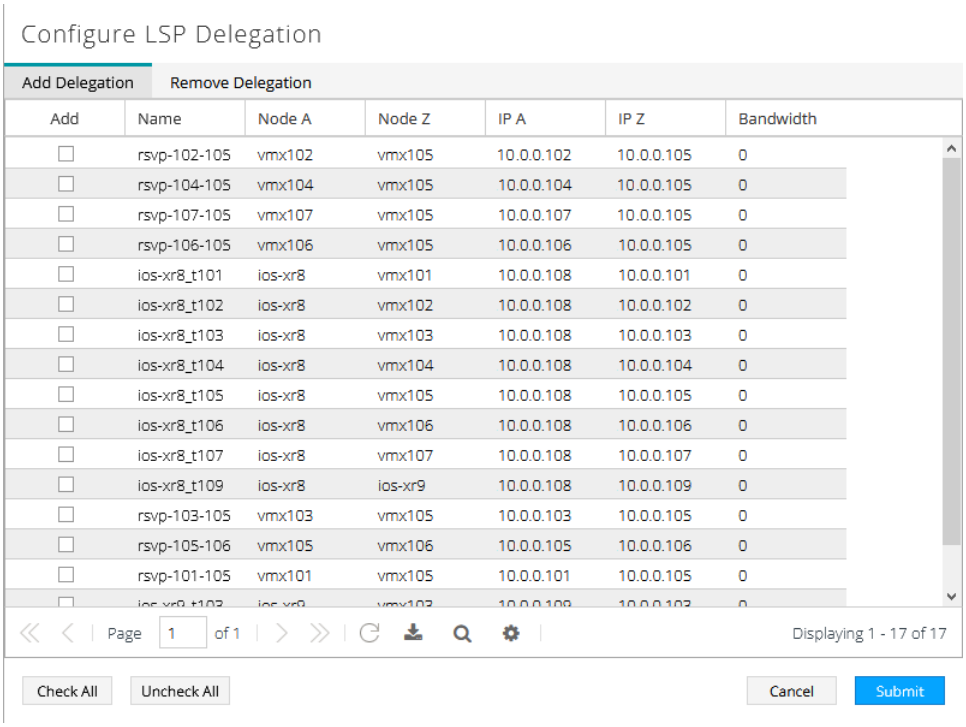


Figure 29: Device Configuration

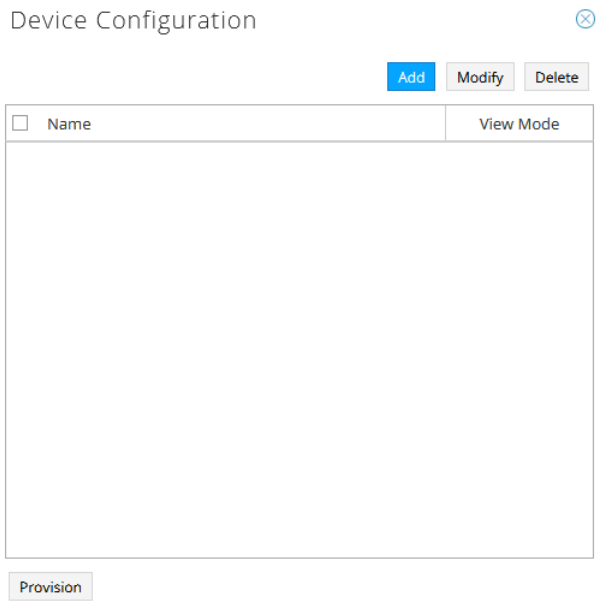
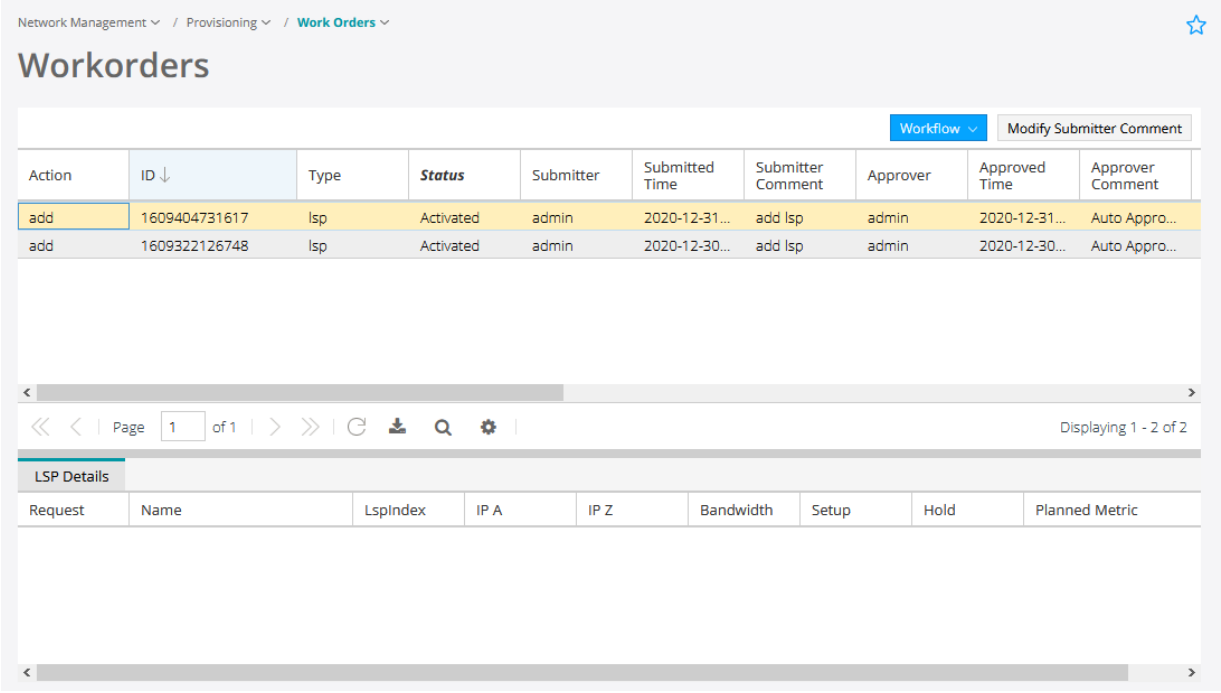
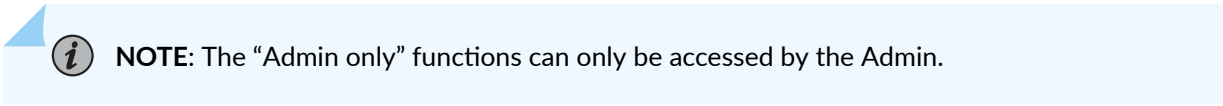


Figure 30: Work Orders



The **Administration** menu contains the following items:



- Admin Group
- Authentication (Admin only)
- Device Profile
- License (Admin only)
- Logs
- Subscribers (Admin only)
- System Health
- System Settings (Admin only)
- Task Scheduler
- Transport Controller
- Topology Filter
- Users (Admin only)

Additionally, you can access user and help functions in the top right corner of the window.

- User Options (user icon)
  - Account Settings
  - Log Out
- Help Options (help icon)
  - Documentation (link to NorthStar customer documentation)
  - About (version and license information)

Switching Between the Classic and New UI

When you log in to the Northstar Controller for the first time after installing or upgrading, you are placed in the Classic UI.

When you subsequently log in to the Northstar Controller, you are placed in the UI that you last used. Your last selection is saved per user per browser.

To switch from the Classic UI to the New UI, click the hamburger icon in the top right corner and select **New UI**.

To switch from the New UI to the Classic UI, click the four-square icon in the top right corner and select **Ext UI**.

CHAPTER 7

Appendix

Upgrading from Pre-4.3 NorthStar with Analytics

IN THIS SECTION

The special upgrade procedure described in this topic is only necessary if you use NorthStar Analytics **and** you are upgrading NorthStar from a version of NorthStar earlier than Release 4.3. In all other cases, you can install NorthStar using the procedure described in "Installing the NorthStar Controller" on page 47.

This procedure involves uninstalling and reinstalling NorthStar analytics in addition to upgrading the NorthStar application, and is a little different depending on whether you have analytics installed on the same server as the NorthStar application or not.

Export Existing Data from the NorthStar Application Server (Recommended)

This procedure involves running a utility called es_export_import_util.py on the NorthStar application server to export and save your existing data prior to upgrade.

The utility exports data to a file called exportdata.tar.Z in the **/opt/northstar/northstar_bundle_x.x.x/db_migration** directory. Ensure that this directory has available space equivalent to at least 10% the size of your Elasticsearch database. For example, a 10GB Elasticsearch database would require 1GB in the **/opt/northstar/northstar_bundle_x.x.x/db_migration** directory. When you begin this procedure, the script will tell you how much memory is required and give you the option to stop the procedure if you do not have enough space to continue.

The amount of time required for the export utility to complete depends on the number of export days and Elasticsearch cache memory/CPU cores you have. You can use the utility's -l option to reduce the number of export days.

You can use the following command to see the utility's full set of supported export options:

```
es_export_import_util.py --help
```

To export and save your existing data, use the following procedure:

1. Log in to the NorthStar application server.
2. Navigate to the db_migration directory:

```
[root db_migration]# cd /opt/northstar/northstar_bundle_x.x.x/db_migration
```

3. Launch the export utility and type **yes** to continue if you agree with the setup information provided. An example follows:

```
[root db_migration]# ./es_export_import_util.py -exp

Warning 1: Time taken for export util depends on number of export days and ES cache memory/
CPU cores. Reduce the considered number of days using -l option.
Warning 2: Total indexes=6 and total DB indexes size=599MB. Average DB size of an index=99MB
and max export DB limit=50000MB.
Considering export data for last 6 days, requires disk space of 59.4MB under current dir

Please enter 'yes' or 'no' to proceed:yes
Starting ES data export
Logs stored at /opt/northstar/logs/es_export_import.log
Export required for datatype=jvision-lsp for the last 6 days
Export required for datatype=jvision-ifd for the last 6 days
Export required for datatype=jvision-ifl for the last 6 days
Export required for datatype=demands for the last 6 days
Export required for datatype=as_demands for the last 6 days
Export required for datatype=rpm-ifl for the last 6 days
Export required for datatype=jnx-cos for the last 6 days
Starting 2 EsExportWorker process with total ES query count in queue=144
Starting process EsExportWorker-0
Starting process EsExportWorker-1
Total rollups data added=13187( jvision-lsp=743 jvision-ifd=8136 jvision-ifl=3230 demands=0
as_demands=0 rpm-ifl=1079 jnx-cos=0) EsExportWorkers-1 completed in minutes=1
Total rollups data added=13378( jvision-lsp=637 jvision-ifd=8424 jvision-ifl=3325 demands=0
as_demands=0 rpm-ifl=993 jnx-cos=0) EsExportWorkers-0 completed in minutes=1
```

```
All EsExportWorker processes completed.
Rollup data exported to ./exportdata.tar.gz . Total time taken in minutes=1
```

Upgrade Procedure with NorthStar Application and NorthStar Analytics on the Same Server

Use the following procedure:

1. Download the NorthStar software and untar the NorthStar_Bundle. See "Installing the NorthStar Controller" on page 47.
2. Navigate to the NorthStar directory:

```
[root]# cd /opt/northstar/northstar_bundle_x.x.x
```

3. Uninstall analytics:

```
[root]# ./uninstall-analytics.sh
```

4. Upgrade the NorthStar application:

```
[root]# ./install.sh
```

5. Reinstall NorthStar analytics:

```
[root]# ./install-analytics.sh
```

Upgrade Procedure with NorthStar Application and NorthStar Analytics on Separate Servers

Use the following procedure:

1. Download the NorthStar software and untar the NorthStar_Bundle. See "Installing the NorthStar Controller" on page 47.

2. On the NorthStar application server, navigate to the NorthStar directory and upgrade the NorthStar application:

```
[root]# cd /opt/northstar/northstar_bundle_x.x.x
[root]# ./install.sh
```

3. On each analytics server, uninstall and reinstall NorthStar analytics:

```
[root]# ./uninstall-analytics.sh
(wait for completion)
[root]# ./install-analytics.sh
```

4. On the NorthStar application server, prepare and redeploy HA analytics data collector settings:

```
[root]# /opt/northstar/utils/net_setup.py
```

   Select **G** (Analytics Data Collector Setting) from the main menu, and then **B** (Prepare and Deploy HA Analytics Data Collector Setting) from the Analytics Data Collector Configuration Settings menu. See "Installing Data Collectors for Analytics" on page 113 for more information.

5. On the NorthStar application server, ensure that analytics data collector connectivity is UP. From the net_setup.py utility main menu, select **G** (Analytics Data Collector Setting), and then select **9** (Test Analytics Data Collector Connectivity) from the Analytics Data Collector Configuration Settings menu.

Update the Netflow Aggregation Setting

The netflowd process analyzes traffic from the router and displays it in the Demands tab in the network information table. By default, Netflow aggregates traffic by PE, but for some applications (such as EPE), you would want the traffic aggregated by prefix. The Netflow aggregate by prefix setting controls how traffic is aggregated. If the parameter is enabled, Netflow aggregates all traffic from a specific ingress PE router to a specific destination (prefix) within the defined period of time.

Between NorthStar Controller Releases 4.2.0 and 4.3.0, the possible values for the Netflow aggregate_by_prefix parameter changed. If you are upgrading from Release 4.2.0 or earlier, you must modify the setting to reflect a valid value for your post-4.2.0 release. See Table 17 on page 231

Table 17: Netflow Aggregate by Prefix Values by NorthStar Release

	NorthStar Releases Earlier than 4.2.0	NorthStar Releases 4.3.0 and Later
Netflow aggregate by prefix possible values	• **0** = aggregation by prefix is disabled (this is the default) • **1** = aggregation by prefix is enabled **NOTE**: NorthStar Releases 4.2.0 and earlier: netflow_aggregate_by_prefix parameter maintained in the northstar.cfg file.	• **always** = aggregation by prefix is enabled • **disable** = aggregation by prefix is disabled • **unknown-destination** = aggregation by prefix is enabled even though the flow is missing a BGP next hop (BGP_NH) or has a BGP_NH of 0.0.0.0 **NOTE**: NorthStar Releases 4.3.0 through 6.0.0: modify the netflow_aggregate_by_prefix parameter in the **/opt/northstar/data/northstar.cfg** file. NorthStar Releases 6.1.0 and later: modify the aggregate-by-prefix setting using the NorthStar CLI `set northstar analytics netflowd aggregate-by-prefix` command.

Import Existing Data (Recommended)

In this procedure, you run the es_export_import_util.py utility again on the NorthStar application server to import the data you previously exported and saved.

1. Log in to the NorthStar application server.
2. Navigate to the db_migration directory:

```
[root db_migration]# cd /opt/northstar/northstar_bundle_x.x.x/db_migration
```

3. Launch the import utility:

```
[root db_migration]# ./es_export_import_util.py -imp
```

RELATED DOCUMENTATION

Installing the NorthStar Controller | 47