

# Cumulus NetQ 2.1 Deployment Guide



## **Table of Contents**

Deployment Preface	6
Contents	
What's New in Cumulus NetQ 2.1	7
Available Documentation	7
Document Formatting	8
Typographical Conventions	8
Note Conventions	8
Cumulus NetQ Primer	9
Contents	10
Cumulus NetQ Operational Advantages	7
Demystify Overlay Networks	1
Protect Network Integrity with NetQ Validation	1
Active Network-wide Troubleshooting	12
Track Connectivity with NetQ Trace	12
Cumulus NetQ Components	] =
NetQ Agents	13
Switch Agents	73
Host Agents	74
NetQ Platform	74
Data Aggregation Layer	74
Data Stores	74
Real-time Streaming Layer	74
Network Service Layer	]5
User Interfaces	15
Data Center Network Deployments	15
Out-of-Band Management Deployment	15
In-band Management Deployment	17
NetQ Operation	17
The NetQ Agent	17
The NetQ Platform	18
Timestamps in NetQ	23
Exporting NetQ Data	27
Key File Locations	29
	20
Install NetQ	
Contents	
Prerequisites	
Hardware Requirements	
NetQ Platform HyperVisor Requirements  Operating System Requirements	5
CORIGINIO SYSTEM RECOMEMIS	7/





NetQ Application Support	32
Install Workflow	. 32
Install the NetQ Platform	. 34
Install Using VM	34
Verify the Installation	42
Install the NetQ Agent	. 45
Install NetQ Agent on a Cumulus Linux Switch	45
Install NetQ Agent on an Ubuntu Server (Optional)	. 46
Install NetQ Agent on a Red Hat or CentOS Server (Optional)	. 49
Configure Optional NetQ Agent Settings	51
Configure the Agent to Use a VRF	51
Configure the Agent to Communicate over a Specific Port	
Integrate with Event Notification Tools	. 52
Set Up Security	. 52
Upgrade from NetQ 1.x to NetQ 2.1.x	52
Contents	
Prerequisites	. 53
Hardware Requirements	53
NetQ Platform HyperVisor Requirements	54
Operating System Requirements	54
NetQ Application Support	
Upgrade Workflow	. 55
Upgrade the NetQ Platform	. 57
Install Using VM	57
Verify the Installation	
Upgrade the NetQ Agent	
Upgrade NetQ Agent on a Cumulus Linux Switch	
Upgrade NetQ Agent on an Ubuntu Server (Optional)	. 70
Upgrade NetQ Agent on a Red Hat or CentOS Server (Optional)	
Configure Optional NetQ Agent Settings	
Configure the Agent to Use a VRF	
Configure the Agent to Communicate over a Specific Port	
Integrate with Event Notification Tools	
Set Up Security	. 75
Upgrade from NetQ 2.0.x to NetQ 2.1.0	75
Contents	
Prerequisites	
Upgrade the NetQ Platform	
Upgrade the NetQ Agents	
Upgrade NetQ Agent on a Cumulus Linux Switch	
Upgrade NetQ Agent on an Ubuntu Server (Optional)	
Upgrade NetQ Agent on a Red Hat or CentOS Server (Optional)	
Set Up the NetQ Agents	. 86



4



Basic Configuration  Configure the Agent to Use a VRF (Optional)	
Configure the Agent to Communicate over a Specific Port (Optional)	87
Upgrade Tips	87
Scenario 1: NetQ Appliance IP Address Was Changed	88
Scenario 2: No IP Address Assigned to the NetQ Appliance on Boot	88
Upgrade from NetQ 2.1.0 to NetQ 2.1.x	88
Contents	
Prerequisites	
Upgrade the NetQ Platform or NetQ Appliance	90
Upgrade the NetQ Agents	94
Upgrade NetQ Agent on a Cumulus Linux Switch	94
Upgrade NetQ Agent on an Ubuntu Server (Optional)	95
Upgrade NetQ Agent on a Red Hat or CentOS Server (Optional)	
Configure Optional NetQ Agent Settings	98
Configure the Agent to Use a VRF Interface	
Configure the Agent to Communicate over a Specific Port	98
Upgrade Tips	98
Scenario 1: NetQ Appliance IP Address Was Changed	99
Scenario 2: No IP Address Assigned to the NetQ Appliance on Boot	99
Integrate with Third-party Software and Hardware	100
Integrate with Third-party Software and Hardware	
Integrate NetQ with an Event Notification Application	
Event Message Format	
Notification Commands Overview	
Configure a Proxy Server	
Create Channels	
Create Rules	
Create Filters	
Example Notification Configurations	
Create a Notification for BGP Events from a Selected Switch	
Create a Notification for Warnings on a Given EVPN VNI	
Create a Notification for Configuration File Changes	
Create a Notification for When a Service Goes Down	
Create a Filter to Drop Notifications from a Given Interface	
Create a Notification for a Given Device that has a Tendency to Overheat (using multiple)	
124	uics)
View Notification Configurations in JSON Format	125
Manage Event Notification Integrations	125
Remove an Event Notification Channel	
Delete an Event Notification Rule	12
Delete an Event Notification Filter	
Integrate with a Hardware Chassis	128





#### ©2019 Cumulus Networks. All rights reserved

CUMULUS, the Cumulus Logo, CUMULUS NETWORKS, and the Rocket Turtle Logo (the "Marks") are trademarks and service marks of Cumulus Networks, Inc. in the U.S. and other countries. You are not permitted to use the Marks without the prior written consent of Cumulus Networks. The registered trademark Linux® is used pursuant to a sublicense from LMI, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis. All other marks are used under fair use or license from their respective owners.





This guide is intended for network administrators who are responsible for installation, setup, and maintenance of Cumulus NetQ in their data center environment. NetQ offers the ability to monitor and manage your data center network infrastructure and operational health with simple tools based on open source Linux. This guide provides instructions and information about installing NetQ core capabilities, configuring optional capabilities, and upgrading an existing NetQ installation. This guide assumes you have already installed Cumulus Linux on your network switches and you are ready to add these NetQ capabilities.

For information about monitoring and troubleshooting your network, refer to the Cumulus NetQ CLI User Guide or the Cumulus NetQ UI User Guide.



Before you get started, you should review the release notes for this version.



## **Deployment Preface**

A variety of resources are available for you to become familiar with Cumulus NetQ and aid in its deployment. These are identified here along with information about how the content is presented.

#### **Contents**

This topic describes...

- What's New in Cumulus NetQ 2.1 (see page 7)
- Available Documentation (see page 7)
- Document Formatting (see page 8)
  - Typographical Conventions (see page 8)
  - Note Conventions (see page 8)

## What's New in Cumulus NetQ 2.1

Cumulus NetQ has been reinvented to scale with the fast adoption rate of open networks and rapid network growth. In addition to the real-time data collection and fabric-wide performance analysis that are already supported in Cumulus NetQ 1.4.1, Cumulus NetQ now offers a graphical user interface for:

- Data visualizations of the overlay and underlay networks,
- Simplified network troubleshooting with network-wide roll-ups of health and alarm status in a single card, and
- Proactive validation of the network status and configuration to regularly detect network issues.

For further information regarding new features, improvements, bug fixes, and known issues present in this release, refer to the release notes.

## **Available Documentation**

The NetQ documentation set has been reorganized and updated from prior releases. They still provide the information you need to proactively monitor your Linux-based network fabric using Cumulus NetQ. They assume that you have already installed Cumulus Linux and NetQ.

You may start anywhere in the documentation or read it from start to finish depending on your role and familiarity with the NetQ software and Linux networking. If you are new to NetQ, you may want to read the Cumulus NetQ Primer before reading the other available documents.

The following NetQ documents are available:

- Cumulus NetQ Deployment Guide (this guide)
- Cumulus NetQ CLI User Guide
- Cumulus NetO UI User Guide
- Cumulus NetQ Release Notes
- What the NetQ Validation System Checks
- Cumulus NetQ Release Versioning and Support Policy



## **Document Formatting**

The Cumulus NetQ Deployment Guide uses the following typographical and note conventions.

#### **Typographical Conventions**

Throughout the guide, text formatting is used to convey contextual information about the content.

Text Format	Meaning
Green text	Link to additional content within the topic or to another topic
Text in Monospace font	Filename, directory and path names, and command usage
[Text within square brackets]	Optional command parameters; may be presented in mixed case or all caps text
<text angle="" brackets="" within=""></text>	Required command parameter values-variables that are to be replaced with a relevant value; may be presented in mixed case or all caps text

#### **Note Conventions**

Several note types are used throughout the document. The formatting of the note indicates its intent and urgency.

## **⊘** Tip or Best Practice

Offers information to improve your experience with the tool, such as time-saving or shortcut options, or i ndicates the common or recommended method for performing a particular task or process

## ① Information

Provides additional information or a reminder about a task or process that may impact your next step or selection

#### Caution

Advises that failure to take or avoid specific action can result in possible data loss





## ① Warning

Advises that failure to take or avoid specific action can result in possible physical harm to yourself, hardware equipment, or facility



## **Cumulus NetQ Primer**

Cumulus® NetQ is a highly-scalable, modern network operations tool set that provides visibility and troubleshooting of your overlay and underlay networks in real-time. NetQ delivers actionable insights and operational intelligence about the health of your data center — from the container, virtual machine, or host, all the way to the switch and port. NetQ correlates configuration and operational status, and instantly identifies and tracks state changes while simplifying management for the entire Linux-based data center. With NetQ, network operations change from a manual, reactive, box-by-box approach to an automated, informed and agile one.

Cumulus NetQ performs three primary functions:

- **Data collection**: real-time and historical telemetry and network state information
- Data analytics: deep processing of the data
- **Data visualization**: rich graphical user interface (GUI) for actionable insight

This documentation is current as of April 23, 2019 for version 2.1.0. Please visit the Cumulus Networks Web site for the most up to date documentation.

#### **Contents**

This topic describes...

- Cumulus NetQ Operational Advantages (see page 11)
  - Demystify Overlay Networks (see page 11)
  - Protect Network Integrity with NetQ Validation (see page 11)
  - Active Network-wide Troubleshooting (see page 12)
  - Track Connectivity with NetQ Trace (see page 12)
- Cumulus NetQ Components (see page 13)
- NetQ Agents (see page 13)
  - Switch Agents (see page 13)
  - Host Agents (see page 14)
- NetQ Platform (see page 14)
  - Data Aggregation Layer (see page 14)
  - Data Stores (see page 14)
  - Real-time Streaming Layer (see page 14)
  - Network Service Layer (see page 15)
  - User Interfaces (see page 15)
- Data Center Network Deployments (see page 15)
  - Out-of-Band Management Deployment (see page 15)
  - In-band Management Deployment (see page 17)



- NetQ Operation (see page 17)
  - The NetQ Agent (see page 17)
  - The NetQ Platform (see page 18)
  - Timestamps in NetQ (see page 23)
  - Exporting NetQ Data (see page 27)
  - Key File Locations (see page 29)

## **Cumulus NetQ Operational Advantages**

Unlike other network operations tools, NetQ delivers significant operational improvements to your network management and maintenance processes. It simplifies the data center network by reducing the complexity through real-time visibility into hardware and software status and eliminating the guesswork associated with investigating issues through the analysis and presentation of detailed, focused data.

#### **Demystify Overlay Networks**

While overlay networks provide significant advantages in network management, it can be difficult to troubleshoot issues that occur in the overlay one box at a time. You are unable to correlate what events (configuration changes, power outages, etc.) may have caused problems in the network and when they occurred. Only a sampling of data is available to use for your analysis. By contrast, with Cumulus NetQ deployed, you have a network-wide view of the overlay network, can correlate events with what is happening now or in the past, and have real-time data to fill out the complete picture of your network health and operation.

#### In summary:

Without NetQ	With NetQ
Difficult to debug overlay network	View network-wide status of overlay network
Hard to find out what happened in the past	View historical activity with time-machine view
Periodically sampled data	Real-time collection of telemetry data for a more complete data set

## **Protect Network Integrity with NetQ Validation**

Network configuration changes can cause numerous trouble tickets because you are not able to test a new configuration before deploying it. When the tickets start pouring in, you are stuck with a large amount of data that is collected and stored in multiple tools making correlation of the events to the resolution required difficult at best. Isolating faults in the past is challenging. By contract, with Cumulus NetQ deployed, you can proactively verify a configuration change as inconsistencies and misconfigurations can be caught prior to deployment. And historical data is readily available to correlate past events with current issues.

In summary:



Without NetQ	With NetQ
Reactive to trouble tickets	Catch inconsistencies and misconfigurations prior to deployment with integrity checks/validation
Large amount of data and multiple tools to correlate the logs/events with the issues	Correlate network status, all in one place
Periodically sampled data	Readily available historical data for viewing and correlating changes in the past with current issues

#### **Active Network-wide Troubleshooting**

Troubleshooting networks is challenging in the best of times, but trying to do so manually, one box at a time, and digging through a series of long and ugly logs make the job harder than it needs to be. Cumulus NetQ provides rolled up and correlated network status on a regular basis, enabling you to get down to the root of the problem quickly, whether it occurred recently or over a week ago. The graphical user interface make this possible visually to speed the analysis.

In summary:

Without NetQ	With NetQ
Large amount of data and multiple tools to correlate the logs/events with the issues	Rolled up and correlated network status, view events and status together
Past events are lost	Historical data gathered and stored for comparison with current network state
Manual, box-by-box troubleshooting	View issues on all devices all at once, pointing to the source of the problem

#### **Track Connectivity with NetQ Trace**

Conventional trace only traverses the data path looking for problems, and does so on a node to node basis. For paths with a small number of hops that might be fine, but in larger networks, it can become extremely time consuming. With Cumulus NetQ both the data and control paths are verified providing additional information. It discovers misconfigurations along all of the hops in one go, speeding the time to resolution.

In summary:

Without NetQ	With NetQ
Trace covers only data path; hard to check control path	Both data and control paths are verified

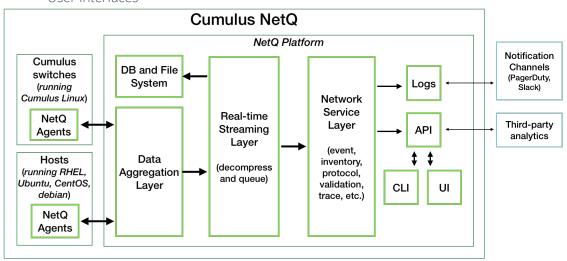


Without NetQ	With NetQ
View portion of entire path	View all paths between devices all at once to find problem paths
Node-to-node check on misconfigurations	View any misconfigurations along all hops from source to destination

## **Cumulus NetQ Components**

NetQ contains the following applications and key components:

- Telemetry data collection and aggregation
  - NetQ Switch Agents
  - NetQ Host Agents
  - Telemetry Aggregator
  - Database
- Data streaming layer
- Network service layer
- User Interfaces



NetQ interfaces with event notification applications, third-party analytics tools.

Each of the NetQ components used to gather, store and process data about the network state are described here.

## **NetQ Agents**

NetQ Agents are software installed and running on every monitored *node* in the network — including Cumulus® Linux® switches, Linux bare-metal hosts, and virtual machines. The NetQ Agents push network data regularly and event information immediately to the NetQ Platform.



#### **Switch Agents**

The NetQ Agents running on Cumulus Linux switches gather the following network data via Netlink:

- Interfaces
- IP addresses (v4 and v6)
- IP routes (v4 and v6)
- Links
- Bridge FDB (MAC Address table)
- ARP Entries/Neighbors (IPv4 and IPv6)

for the following protocols:

- Bridging protocols: LLDP, STP, MLAG
- Routing protocols: BGP, OSPF
- Network virtualization: EVPN, LNV, VXLAN

The NetQ Agent is supported on Cumulus Linux 3.7.0 and later.

#### **Host Agents**

The NetQ Agents running on hosts gather the same information as that for switches, plus the following network data:

- Network IP and MAC addresses
- Container IP and MAC addresses

The NetQ Agent obtains container information by listening to the Kubernetes orchestration tool.

The NetQ Agent is supported on hosts running Ubuntu 16.04, Red Hat® Enterprise Linux 7, and CentOS 7 Operating Systems.

#### **NetQ Platform**

The NetQ Platform performs the data collection, storage, and processing for delivery to various user interfaces. It is comprised of a collection of scalable components running entirely within a single server. The NetQ software queries this server, rather than individual devices enabling greater scalability of the system. Each of these components is described briefly here.

#### **Data Aggregation Layer**

The data aggregation layer collects data coming from all of the NetQ Agents. It then filters, compresses, and forwards the data to the streaming layer. The server monitors for missing messages and also monitors the NetQ Agents themselves, providing alarms when appropriate. In addition to the telemetry data collected from the NetQ Agents, the aggregation server collects information from the switches and hosts, such as vendor, model, version, and basic operational state.

#### **Data Stores**

Two types of data stores are used in the NetQ product. The first stores the raw data, data aggregations, and discrete events needed for quick response to data requests. The second stores data based on correlations, transformations and processing of the raw data.



#### **Real-time Streaming Layer**

The streaming layer processes the incoming raw data from the aggregation server in real time. It reads the metrics and stores them as a time series, and triggers alarms based on anomaly detection, thresholds, and events.

#### **Network Service Layer**

The network service layer monitors services network-wide and stores status details.

#### **User Interfaces**

NetQ data is available through several user interfaces:

- NetQ CLI (command line interface)
- NetQ UI (graphical user interface )
- NetQ RESTful API (representational state transfer application programming interface )
- Logs

The CLI and UI query the RESTful API for the data to present. Standard integrations can be configured to integrate with third-party notification tools.

## **Data Center Network Deployments**

There are two deployment types that are commonly deployed for network management in the data center:

- Out-of-Band Management (recommended)
- In-band Management

A summary of each type is provided here.



NetQ operates over layer 3, and can be used in both layer 2 bridged and layer 3 routed environments. Cumulus Networks always recommends layer 3 routed environments whenever possible.

#### **Out-of-Band Management Deployment**

Cumulus Networks recommends deploying NetQ on an out-of-band (OOB) management network to separate network management traffic from standard network data traffic, but it is not required. This figure shows a sample CLOS-based network fabric design for a data center using an OOB management network overlaid on top, where NetQ is deployed.

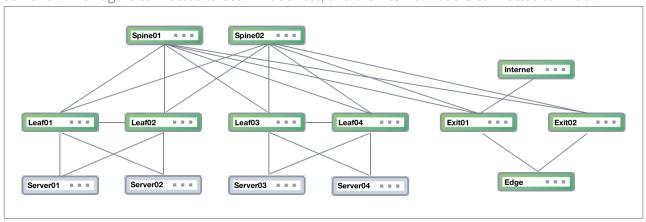
The physical *network* hardware includes:

- **Spine** switches: where data is aggregated and distributed; also known as an aggregation switch, end-of-row (EOR) switch or distribution switch
- **Leaf** switches: where servers connect to the network; also known as a Top of Rack (TOR) or access switch
- **Server** hosts: where applications are hosted and data served to the user through the network
- **Exit** switch: where connections to outside the data center occur; also known as Border Leaf or Service Leaf



- **Edge** server (optional): where the firewall is the demarcation point, peering may occur through the exit switch layer to Internet (PE) devices
- **Internet** device (PE): where provider edge (PE) equipment communicates at layer 3 with the network fabric

The diagram shows physical connections (in the form of grey lines) between Spine 01 and four Leaf devices and two Exit devices, and Spine 02 and the same four Leaf devices and two Exit devices. Leaf 01 and Leaf 02 are connected to each other over a peerlink and act as an MLAG pair for Server 01 and Server 02. Leaf 03 and Leaf 04 are connected to each other over a peerlink and act as an MLAG pair for Server 03 and Server 04. The Edge is connected to both Exit devices, and the Internet node is connected to Exit 01.

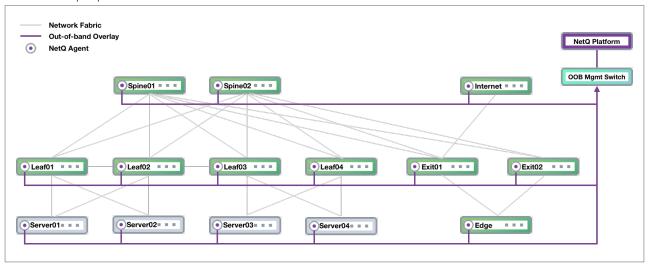


Data Center Network Example

The physical management hardware includes:

- OOB Mgmt Switch: aggregation switch that connects to all of the network devices through communications with the NetQ Agent on each node
- NetQ Platform: hosts the telemetry software, database and user interfaces (refer to description above).

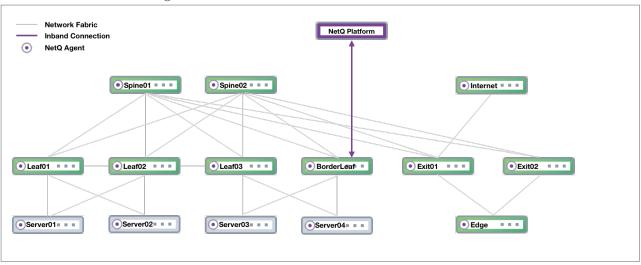
These switches are connected to each of the physical network devices through a virtual network overlay, shown with purple lines.





#### **In-band Management Deployment**

While not the preferred deployment method, you might choose to implement NetQ within your data network. In this scenario, there is no overlay and all traffic to and from the NetQ Agents and the NetQ Platform traverses the data paths along with your regular network traffic. The roles of the switches in the CLOS network are the same, except that the NetQ Platform performs the aggregation function that the OOB management switch performed. If your network goes down, you might not have access to the NetQ Platform for troubleshooting.



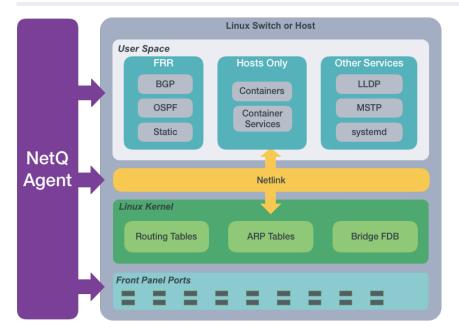
## **NetQ Operation**

In any of the above deployments, NetQ offers network-wide configuration and device management, proactive monitoring capabilities, and performance diagnostics for complete management of your network. Each component of the solution provides a critical element to make this possible.

## **The NetQ Agent**

From a software perspective, a network switch has software associated with the hardware platform, the operating system, and communications. For data centers, the software on a Cumulus Linux network switch would be similar to the diagram shown here.





The NetQ Agent interacts with the various components and software on switches and hosts and provides the gathered information to the NetQ Platform. You can view the data using the NetQ CLI or UI.

The NetQ Agent p olls the user space applications for information about the performance of the various routing protocols and services that are running on the switch. Cumulus Networks supports BGP and OSPF Free Range Routing (FRR) protocols as well as static addressing. Cumulus Linux also supports LLDP and MSTP among other protocols, and a variety of services such as systemd and sensors. For hosts, the NetQ Agent also polls for performance of containers managed with Kubernetes. All of this information is used to provide the current health of the network and verify it is configured and operating correctly.

For example, if the NetQ Agent learns that an interface has gone down, a new BGP neighbor has been configured, or a container has moved, it provides that information to the NetQ Platform . That information can then be used to notify users of the operational state change through various channels. By default, data is logged in the database, but you can use the CLI (netq show events) or configure the Event Service in NetQ to send the information to a third-party notification application as well. NetQ supports PagerDuty and Slack integrations.

The NetQ Agent interacts with the Netlink communications between the Linux kernel and the user space, listening for changes to the network state, configurations, routes and MAC addresses. NetQ uses this information to enable notifications about these changes so that network operators and administrators can respond quickly when changes are not expected or favorable.

For example, if a new route is added or a MAC address removed, NetQ Agent records these changes and sends that information to the NetQ Platform . Based on the configuration of the Event Service, these changes can be sent to a variety of locations for end user response.

The NetQ Agent also interacts with the hardware platform to obtain performance information about various physical components, such as fans and power supplies, on the switch. Operational states and temperatures are measured and reported, along with cabling information to enable management of the hardware and cabling, and proactive maintenance.

For example, as thermal sensors in the switch indicate that it is becoming very warm, various levels of alarms are generated. These are then communicated through notifications according to the Event Service configuration.

## The NetQ Platform

Once the collected data is sent to and stored in the NetQ database, you can:



- Validate configurations, identifying misconfigurations in your current network, in the past, or prior to deployment,
- Monitor communication paths throughout the network,
- Notify users of issues and management information,
- Anticipate impact of connectivity changes,
- and so forth.

#### **Validate Configurations**

The NetQ CLI enables validation of your network health through two sets of commands: netq check and netq show. They extract the information from the Network Service Layer and Event service. The Network Service Layer is continually validating the connectivity and configuration of the devices and protocols running on the network. Using the netq check and netq show commands displays the status of the various components and services on a network-wide and complete software stack basis. For example, you can perform a network-wide check on all sessions of BGP with a single netq check bgp command. The command lists any devices that have misconfigurations or other operational errors in seconds. When errors or misconfigurations are present, using the netq show bgp command displays the BGP configuration on each device so that you can compare and contrast each device, looking for potential causes. netq check and netq show commands are available for numerous components and services as shown in the following table.

Component or Service	Check	Show	Component or Service	Check	Show
Agents	*	*	LLDP		*
BGP	*	*	LNV	*	*
CLAG (MLAG)	*	*	MACs		*
Events		*	MTU	*	
EVPN	*	*	NTP	*	*
Interfaces	*	*	OSPF	*	*
Inventory		*	Sensors	*	*
IPv4/v6		*	Services		*
Kubernetes		*	VLAN	*	*
License	*		VXLAN	*	*

#### **Monitor Communication Paths**



The trace engine is used to validate the available communication paths between two network devices. The corresponding netq trace command enables you to view all of the paths between the two devices and if there are any breaks in the paths. This example shows two successful paths between server12 and leaf11, all with an MTU of 9152. The first command shows the output in path by path tabular mode. The second command show the same output as a tree.

```
cumulus@switch:~$ netq trace 10.0.0.13 from 10.0.0.21
Number of Paths: 2
Number of Paths with Errors: 0
Number of Paths with Warnings: 0
Path MTU: 9152
Id Hop Hostname InPort
                              InTun, RtrIf OutRtrIf, Tun
OutPort
___ ___ ____
1 1 server12
bond1.1002
   2 leaf12 swp8
                                            vlan1002
peerlink-1
               swpб
                       vlan1002
  3 leaf11
vlan1002
2 1 server12
bond1.1002
  2 leaf11 swp8
vlan1002
cumulus@switch:~$ netq trace 10.0.0.13 from 10.0.0.21 pretty
Number of Paths: 2
Number of Paths with Errors: 0
Number of Paths with Warnings: 0
Path MTU: 9152
hostd-12 bond1.1002 -- swp8 leaf12 <vlan1002> peerlink-1 -- swp6
<vlan1002> leaf11 vlan1002
        bond1.1002 -- swp8 leaf11 vlan1002
```

#### This output is read as:

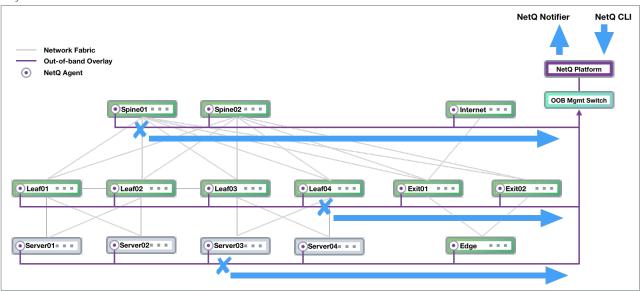
- Path 1 traverses the network from server12 out bond1.1002 into leaf12 interface swp8 out VLAN1002 peerlink-1 into VLAN1002 interface swp6 on leaf11
- Path 2 traverses the network from server12 out bond1.1002 into VLAN1002 interface swp8 on leaf11.

If the MTU does not match across the network, or any of the paths or parts of the paths have issues, that data is called out in the summary at the top of the output and shown in red along the paths, giving you a starting point for troubleshooting.

#### **View Historical State and Configuration**



All of the check, show and trace commands can be run for the current status and for a prior point in time. For example, this is useful when you receive messages from the night before, but are not seeing any problems now. You can use the netq check command to look for configuration or operational issues around the time that the messages are timestamped. Then use the netq show commands to see information about how the devices in question were configured at that time or if there were any changes in a given timeframe. Optionally, you can use the netq trace command to see what the connectivity looked like between any problematic nodes at that time. This example shows problems occurred on spine01, leaf04, and server03 last night. The network administrator received notifications and wants to investigate. The diagram is followed by the commands to run to determine the cause of a BGP error on spine01. Note that the commands use the around option to see the results for last night and that they can be run from any switch in the network.



```
cumulus@switch:~$ netq check bgp around 30m
Total Nodes: 25, Failed Nodes: 3, Total Sessions: 220, Failed
Sessions: 24,
                  VRF
Hostname
                                  Peer Name
                                                    Peer Hostname
Reason
                                              Last Changed
exit-1
                  DataVrf1080
                                                    firewall-1
                                  swp6.2
BGP session with peer firewall-1 swp6.2: AFI/ 1d:2h:6m:21s
SAFI evpn not activated on peer
exit-1
                  DataVrf1080
                                  swp7.2
                                                    firewall-2
BGP session with peer firewall-2 (swp7.2 vrf 1d:1h:59m:43s
DataVrf1080) failed,
reason: Peer not configured
exit-1
                  DataVrf1081
                                  swp6.3
                                                    firewall-1
BGP session with peer firewall-1 swp6.3: AFI/ 1d:2h:6m:21s
SAFI evpn not activated on peer
```



```
DataVrf1081 swp7.3 firewall-2
exit-1
BGP session with peer firewall-2 (swp7.3 vrf 1d:1h:59m:43s
DataVrf1081) failed,
reason: Peer not configured
exit-1 DataVrf1082 swp6.4 firewall-1
BGP session with peer firewall-1 swp6.4: AFI/ 1d:2h:6m:21s
SAFI evpn not activated on peer
               DataVrf1082 swp7.4
                                              firewall-2
BGP session with peer firewall-2 (swp7.4 vrf 1d:1h:59m:43s
DataVrf1082) failed,
reason: Peer not configured
                       swp6 firewall-1
              default
exit-1
BGP session with peer firewall-1 swp6: AFI/SA 1d:2h:6m:21s
FI evpn not activated on peer
              default swp7
                                              firewall-2
BGP session with peer firewall-2 (swp7 vrf de 1d:1h:59m:43s
cumulus@switch:~$ netq exit-1 show bgp
Matching bgp records:
Hostname
               Neighbor
                                          VRF
ASN Peer ASN PfxRx Last Changed
exit-1 swp3(spine-1) default 655537 655435 27/24/412 Fri Feb 15 17:20:00 2019
        swp3.2(spine-1) DataVrf1080 655435 14/12/0 Fri Feb 15 17:20:00 2019
exit-1
655537
         swp3.3(spine-1)
exit-1
                                         DataVrf1081
655537 655435 14/12/0 Fri Feb 15 17:20:00 2019
         swp3.4(spine-1)
exit-1
                                         DataVrf1082
        655435 14/12/0 Fri Feb 15 17:20:00 2019
655537
exit-1
              swp4(spine-2)
                                         default
         655435 27/24/412 Fri Feb 15 17:20:00 2019
655537
               swp4.2(spine-2)
exit-1
                                         DataVrf1080
         655435 14/12/0 Fri Feb 15 17:20:00 2019
655537
exit-1
              swp4.3(spine-2)
                                         DataVrf1081
655537
          655435 14/12/0 Fri Feb 15 17:20:00 2019
              swp4.4(spine-2)
exit-1
                                          DataVrf1082
655537
         655435 13/12/0 Fri Feb 15 17:20:00 2019
         swp5(spine-3) default
655435 28/24/412 Fri Feb 15 17:20:00 2019
swp5.2(spine-3) DataVrf1080
exit-1
655537
exit-1
         655435 14/12/0 Fri Feb 15 17:20:00 2019
655537
             swp5.3(spine-3)
exit-1
                                         DataVrf1081
         655435 14/12/0 Fri Feb 15 17:20:00 2019
655537
```

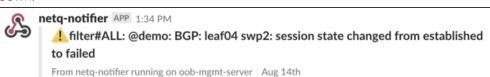


exit-1		swp5.4(spine-3) DataVrf1082
		14/12/0 Fri Feb 15 17:20:00 2019
exit-1		swp6(firewall-1) default
655537	655539	73/69/- Fri Feb 15 17:22:10 2019
exit-1		swp6.2(firewall-1) DataVrf1080
655537	655539	73/69/- Fri Feb 15 17:22:10 2019
exit-1		swp6.3(firewall-1) DataVrf1081
655537	655539	73/69/- Fri Feb 15 17:22:10 2019
exit-1		swp6.4(firewall-1) DataVrf1082
655537	655539	73/69/- Fri Feb 15 17:22:10 2019
exit-1		swp7 default
655537	-	NotEstd Fri Feb 15 17:28:48 2019
exit-1		swp7.2 DataVrf1080
655537	-	NotEstd Fri Feb 15 17:28:48 2019
exit-1		swp7.3 DataVrf1081
655537	_	NotEstd Fri Feb 15 17:28:48 2019
exit-1		swp7.4 DataVrf1082
655537	-	NotEstd Fri Feb 15 17:28:48 2019

#### **Manage Network Events**

The NetQ notifier manages the events that occur for the devices and components, protocols and services that it receives from the NetQ Agents. The notifier enables you to capture and filter events that occur to manage the behavior of your network. This is especially useful when an interface or routing protocol goes down and you want to get them back up and running as quickly as possible, preferably before anyone notices or complains. You can improve resolution time significantly by creating filters that focus on topics appropriate for a particular group of users. You can easily create filters around events related to BGP, LNV, and MLAG session states, interfaces, links, NTP and other services, fans, power supplies, and physical sensor measurements.

For example, for operators responsible for routing, you can create an integration with a notification application that notifies them of routing issues as they occur. This is an example of a Slack message received on a *netq-notifier* channel indicating that the BGP session on switch *leaf04* interface *swp2* has gone down.



## **Timestamps in NetQ**

Every event or entry in the NetQ database is stored with a timestamp of when the event was captured by the NetQ Agent on the switch or server. This timestamp is based on the switch or server time where the NetQ Agent is running, and is pushed in UTC format. It is important to ensure that all devices are NTP synchronized to prevent events from being displayed out of order or not displayed at all when looking for events that occurred at a particular time or within a time window.

Interface state, IP addresses, routes, ARP/ND table (IP neighbor) entries and MAC table entries carry a timestamp that represents the time the event happened (such as when a route is deleted or an interface comes up) — *except* the first time the NetQ agent is run. If the network has been running and stable when a NetQ agent is brought up for the first time, then this time reflects when the agent was started. Subsequent changes to these objects are captured with an accurate time of when the event happened.



Data that is captured and saved based on polling, and just about all other data in the NetQ database, including control plane state (such as BGP or MLAG), has a timestamp of when the information was *captured* rather than when the event *actually happened*, though NetQ compensates for this if the data extracted provides additional information to compute a more precise time of the event. For example, BGP uptime can be used to determine when the event actually happened in conjunction with the timestamp.

When retrieving the timestamp, command outputs display the time in three ways:

- For non-JSON output when the timestamp represents the Last Changed time, time is displayed in actual date and time when the time change occurred
- For non-JSON output when the timestamp represents an Uptime, time is displayed as days, hours, minutes, and seconds from the current time.
- For JSON output, time is displayed in microseconds that have passed since the Epoch time ( January 1, 1970 at 00:00:00 GMT) .

This example shows the difference between the timestamp displays.

Hostname		Neighbor	VRF	
		SN PfxRx	Last Changed	
 exit-1			default	
655537	655435	27/24/412	default Fri Feb 15 17:20:00 2019	
exit-1		swp3.2(spine-1)	DataVrf1080	
655537	655435		Fri Feb 15 17:20:00 2019	
exit-1	033133	swp3.3(spine-1)		
655537			Fri Feb 15 17:20:00 2019	
exit-1		swp3.4(spine-1)		
655537			Fri Feb 15 17:20:00 2019	
exit-1		swp4(spine-2)	default	
655537			Fri Feb 15 17:20:00 2019	
exit-1		swp4.2(spine-2)		
655537	655435		Fri Feb 15 17:20:00 2019	
exit-1		swp4.3(spine-2)	DataVrf1081	
655537	655435	14/12/0	Fri Feb 15 17:20:00 2019	
exit-1		swp4.4(spine-2)	DataVrf1082	
655537	655435	13/12/0	Fri Feb 15 17:20:00 2019	
• • •				
		netq show agents		
Matching a	gents re	ecords:		
Hostname		Status	NTP Sync	
Version			Sys Uptime	Agent
Uptime			ime Last Changed	



```
leaf01
                                  yes 2.0.0-cl3u11~1549993210.
                 Fresh
e902a94
            2h:32m:33s
                                      2h:26m:19s
26m:19s
                       Tue Feb 12 18:13:28 2019
leaf02
                                  yes 2.0.0-cl3u11~1549993210.
                 Fresh
                                     2h:26m:14s
e902a94
            2h:32m:33s
                                                                2h:
26m:14s
                       Tue Feb 12 18:13:33 2019
                                 yes 2.0.0-ub16.
leaf11
                 Fresh
04u11~1549993314.e902a94 2h:32m:28s
                                                   2h:25m:
                 2h:25m:49s
                                             Tue Feb 12 18:17:32 2019
                                      2.0.0-rh7u11~1549992132.
leaf12
                 Fresh
                                  yes
c42c08f
            2h:32m:0s
                                      2h:25m:44s
                                                                2h:
                    Tue Feb 12 18:17:36 2019
25m:44s
                                 yes 2.0.0-ub16.
leaf21
                 Fresh
04u11~1549993314.e902a94 2h:32m:28s
                                                   2h:25m:
                  2h:25m:39s
                                             Tue Feb 12 18:17:42 2019
39s
leaf22
                                         2.0.0-rh7u11~1549992132.
                Fresh
                                  yes
c42c08f
            2h:32m:0s
                                      2h:25m:35s
25m:35s
                      Tue Feb 12 18:17:46 2019
spine01
                 Fresh
                                  yes
                                           2.0.0-cl3u11~1549993210.
e902a94
            2h:32m:33s
                                      2h:27m:11s
                       Tue Feb 12 18:13:06 2019
27m:11s
                                         2.0.0-cl3u11~1549993210.
spine02
                 Fresh
                                 yes
e902a94
            2h:32m:33s
                                      2h:27m:6s
                                                                2h:
                       Tue Feb 12 18:13:11 2019
27m:6s
cumulus@switch:~$ netq show agents json
    "agents":[
       {
            "status": "Fresh",
           "lastChanged":1549995208.3039999008,
           "reinitializeTime":1549995146.0,
           "hostname": "leaf01",
           "version": "2.0.0-cl3u11~1549993210.e902a94",
            "sysUptime":1549994772.0,
           "ntpSync": "yes",
           "agentUptime":1549995146.0
       },
           "status": "Fresh",
           "lastChanged":1549995213.3399999142,
           "reinitializeTime":1549995151.0,
           "hostname": "leaf02",
            "version": "2.0.0-cl3u11~1549993210.e902a94",
           "sysUptime":1549994772.0,
           "ntpSync": "yes",
           "agentUptime":1549995151.0
            "status": "Fresh",
           "lastChanged":1549995434.3559999466,
```



```
"reinitializeTime":1549995157.0,
    "hostname": "leaf11",
    "version": "2.0.0-ub16.04u11~1549993314.e902a94",
    "sysUptime":1549994772.0,
    "ntpSync": "yes",
    "agentUptime":1549995157.0
    "status": "Fresh",
    "lastChanged":1549995439.3770000935,
    "reinitializeTime":1549995164.0,
    "hostname": "leaf12",
    "version": "2.0.0-rh7u11~1549992132.c42c08f",
    "sysUptime":1549994809.0,
    "ntpSync": "yes",
    "agentUptime":1549995164.0
    "status": "Fresh",
    "lastChanged":1549995452.6830000877,
    "reinitializeTime":1549995176.0,
    "hostname": "leaf21",
    "version": "2.0.0-ub16.04u11~1549993314.e902a94",
    "sysUptime":1549994777.0,
    "ntpSync": "yes",
    "agentUptime":1549995176.0
},
    "status": "Fresh",
    "lastChanged":1549995456.4500000477,
    "reinitializeTime":1549995181.0,
    "hostname": "leaf22",
    "version": "2.0.0-rh7u11~1549992132.c42c08f",
    "sysUptime":1549994805.0,
    "ntpSync": "yes",
    "agentUptime":1549995181.0
    "status": "Fresh",
    "lastChanged":1549995186.3090000153,
    "reinitializeTime":1549995094.0,
    "hostname": "spine01",
    "version": "2.0.0-cl3u11~1549993210.e902a94",
    "sysUptime":1549994772.0,
    "ntpSync": "yes",
    "agentUptime":1549995094.0
    "status": "Fresh",
    "lastChanged":1549995191.4530000687,
    "reinitializeTime":1549995099.0,
    "hostname": "spine02",
```



```
"version": "2.0.0-cl3u11~1549993210.e902a94",
            "sysUptime":1549994772.0,
            "ntpSync": "yes",
            "agentUptime":1549995099.0
        },
. . .
```



If a NetQ Agent is restarted on a device, the timestamps for existing objects are not updated to reflect this new restart time. Their timestamps are preserved relative to the original start time of the Agent. A rare exception is if the device is rebooted between the time it takes the Agent being stopped and restarted; in this case, the time is once again relative to the start time of the Agent.

#### **Exporting NetQ Data**

Data from the NetQ Platform can be exported in a couple of ways:

- use the json option to output command results to ISON format for parsing in other applications
- use the UI to export data from the full screen cards

#### **Example Using the CLI**

You can check the state of BGP on your network with netq check bgp:

```
cumulus@leaf01:~$ netq check bgp
Total Nodes: 25, Failed Nodes: 3, Total Sessions: 220, Failed
Sessions: 24,
              VRF Peer Name
                                    Peer Hostname
Hostname
Reason
                                   Last Changed
_____
             DataVrf1080 swp6.2 firewall01
exit01
BGP session with peer firewall01 swp6.2: AFI/ Tue Feb 12 18:11:16 2019
SAFI evpn not activated on peer
             DataVrf1080 swp7.2 firewall02
BGP session with peer firewall02 (swp7.2 vrf Tue Feb 12 18:11:27 2019
DataVrf1080) failed,
reason: Peer not configured
             DataVrf1081 swp6.3
                                        firewall01
BGP session with peer firewall01 swp6.3: AFI/ Tue Feb 12 18:11:16 2019
SAFI evpn not activated on peer
exit01 DataVrf1081 swp7.3 firewall02
BGP session with peer firewall02 (swp7.3 vrf Tue Feb 12 18:11:27 2019
DataVrf1081) failed,
```



```
reason: Peer not configured ...
```

When you show the output in JSON format, this same command looks like this:

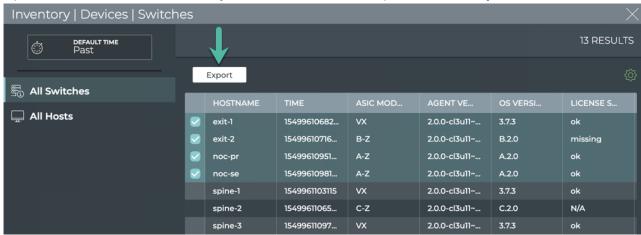
```
cumulus@leaf01:~$ netq check bgp json
    "failedNodes":[
            "peerHostname": "firewall01",
            "lastChanged":1549995080.0,
            "hostname": "exit01",
            "peerName": "swp6.2",
            "reason": "BGP session with peer firewall01 swp6.2: AFI
/SAFI evpn not activated on peer",
            "vrf": "DataVrf1080"
            "peerHostname": "firewall02",
            "lastChanged":1549995449.7279999256,
            "hostname": "exit01",
            "peerName": "swp7.2",
            "reason": "BGP session with peer firewall02 (swp7.2 vrf
DataVrf1080) failed, reason: Peer not configured",
            "vrf": "DataVrf1080"
            "peerHostname": "firewall01",
            "lastChanged":1549995080.0,
            "hostname": "exit01",
            "peerName": "swp6.3",
            "reason": "BGP session with peer firewall01 swp6.3: AFI
/SAFI evpn not activated on peer",
            "vrf": "DataVrf1081"
            "peerHostname": "firewall02",
            "lastChanged":1549995449.7349998951,
            "hostname": "exit01",
            "peerName": "swp7.3",
            "reason": "BGP session with peer firewall02 (swp7.3 vrf
DataVrf1081) failed, reason: Peer not configured",
            "vrf": "DataVrf1081"
        },
. . .
    "summary": {
        "checkedNodeCount": 25,
```



```
"failedSessionCount": 24,
    "failedNodeCount": 3,
    "totalSessionCount": 220
}
```

#### **Example Using the UI**

Open the full screen Switch Inventory card, select the data to export, and click **Export**.



#### **Key File Locations**

The primary configuration file for all Cumulus NetQ tools, netq.yml, resides in /etc/netq by default. Log files are stored in /var/logs/ by default.

Refer to Investigate NetQ Issues for a complete listing of configuration files and logs for use in issue resolution.



## Install NetQ

Installing NetQ can be accomplished in one of three ways:

- If you have purchased a NetQ Appliance, the only installation required is to load the NetQ Agent on any switches and hosts you want to monitor.
- If you already have a switch (running Cumulus Linux version 3.3.0 or later) and you want to add NetQ functionality to it, installation involves three steps:
  - Verify your server meets the hardware and software requirements.
  - Load the software onto the switch.
  - Load the NetQ Agent onto the switches and hosts you want to monitor.
- If you upgrading from a prior version of NetQ, please follow the instructions in the relevant upgrade topic, rather than the instructions here.

#### **Contents**

This topic describes...

- Prerequisites (see page 31)
  - Hardware Requirements (see page 31)
  - NetQ Platform HyperVisor Requirements (see page 31)
  - Operating System Requirements (see page 32)
  - NetQ Application Support (see page 32)
- Install Workflow (see page 32)
- Install the NetQ Platform (see page 34)
  - Install Using VM (see page 34)
  - Verify the Installation (see page 42)
- Install the NetQ Agent (see page 45)
  - Install NetQ Agent on a Cumulus Linux Switch (see page 45)
  - Install NetQ Agent on an Ubuntu Server (Optional) (see page 46)
  - Install NetQ Agent on a Red Hat or CentOS Server (Optional) (see page 49)
- Configure Optional NetQ Agent Settings (see page 51)
  - Configure the Agent to Use a VRF (see page 51)
  - Configure the Agent to Communicate over a Specific Port (see page 52)
- Integrate with Event Notification Tools (see page 52)
- Set Up Security (see page 52)



## **Prerequisites**

#### **Hardware Requirements**

NetQ is supported on the NetQ Appliance and a variety of other hardware.



#### **M** IMPORTANT

You must meet these hardware requirements to install the VM and have it run properly.

In all cases, the NetQ software requires a server with the following:

Requirement	Description
Processing power	Eight (8) virtual CPUs
Memory	64 GB RAM
Local disk storage	256 GB SSD ( <b>Note</b> : This must be an SSD; use of other storage options can lead to system instability and are not supported.)
Network interface speed	1 G NIC or higher

If you are not using the NetQ Appliance, you must open the following ports on the NetQ server to use the NetQ software:

Port	Component Access
31980	NetQ Platform
32708	API Gateway
32666	Web-based User Interface



These ports have changed from NetQ 1.4 and earlier.

## **NetQ Platform HyperVisor Requirements**

The NetQ Platform can be installed as a Virtual Machine (VM) using one of the following hypervisors:

• VMware ESXi™ 6.5 for servers running Cumulus Linux, CentOS, Ubuntu and RedHat operating systems.



• KVM/QCOW (QEMU Copy on Write) image for servers running CentOS, Ubuntu and RedHat operating systems.

#### **Operating System Requirements**

NetQ 2.1 Agents are supported on the following switch or host operating systems:

- Cumulus Linux 3.3.0 and later
- Ubuntu 16.04
- Red Hat<sup>®</sup> Enterprise Linux (RHEL) 7.1
- CentOS 7

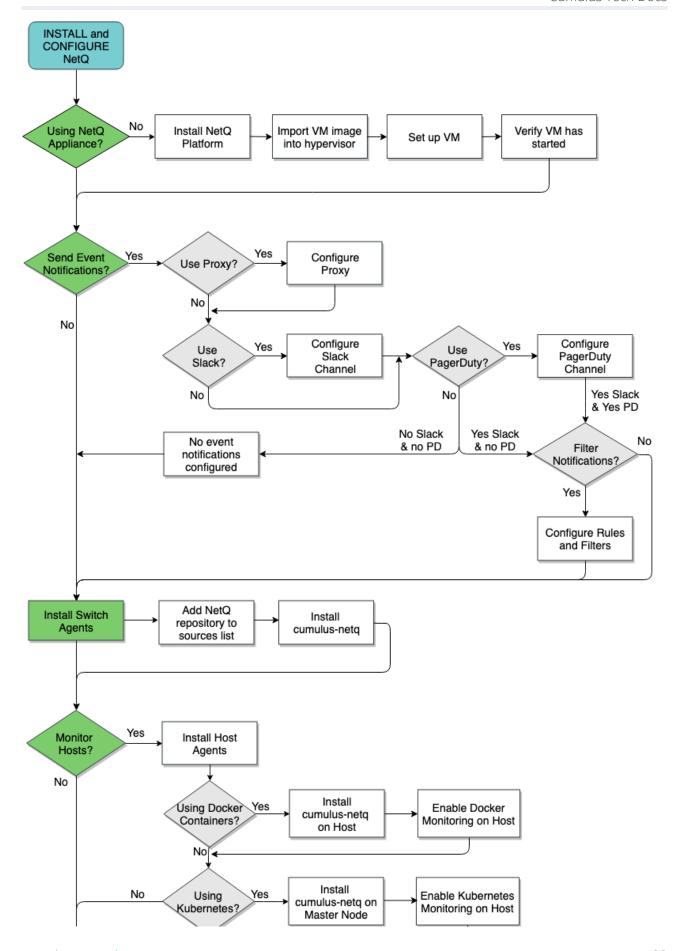
#### **NetQ Application Support**

The NetQ CLI, UI, and RESTful API are supported on NetQ 2.1.0 and later. NetQ 1.4, and earlier, applications are not supported in NetQ 2.x.

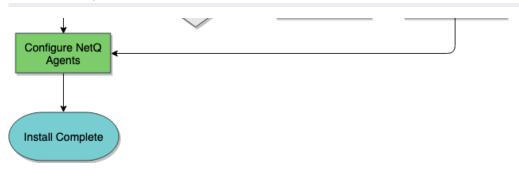
#### **Install Workflow**

Installation of NetQ involves installing the NetQ Platform, and installing and configuring the NetQ Agents. Additional steps are needed to Integrate NetQ with Event Notification Applications . This flow chart shows the required steps to install and setup NetQ to start validating your network and the optional steps of integrating with event notification applications and monitoring hosts.









## Install the NetQ Platform

If you *are not* using the NetQ Appliance, you must install the NetQ Platform on your selected hardware. If you have deployed the NetQ Appliance, you can skip to Install NetQ Agent (see page 45).

The NetQ Platform is comprised of the following components:

- NetQ applications: network monitoring and analytics functionality
- NetQ CLI: command line user interface for monitoring network and administering NetQ through a terminal session
- **NetQ UI**: graphical interface for monitoring network and administering NetQ
- NetQ API: Restful application programming interface for accessing NetQ data and integrating with third-party tools
- **NetQ notifier**: application used to send event notifications to third-party notification tools

#### **(i)** Best Practice

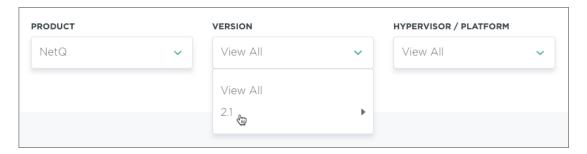
Cumulus Networks recommends you install the NetQ software on a server that is part of an outof-band management network to ensure it can monitor in-band network issues without being affected itself. You should run the software on a separate, powerful server to ensure proper operation and for maximum usability and performance. Refer to Hardware Requirements (see page 31) for specifics.

#### **Install Using VM**

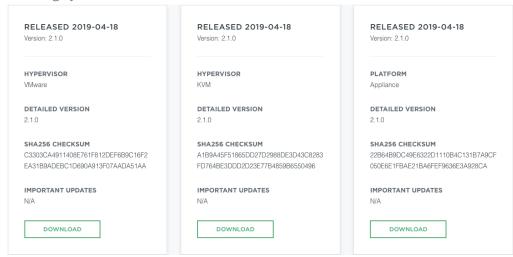
To install the NetQ Platform using your own hardware using a VM image:

- 1. **IMPORTANT**: Confirm that your server hardware meets the requirements set out here (see page 31).
- 2. Download the NetQ Platform image.
  - a. On the Cumulus Downloads page, select *NetQ* from the **Product** list box.
  - b. Click 2.1 from the **Version** list box, and then select 2.1.x from the submenu.
  - c. Optionally, select the hypervisor you wish to use (*VMware* or *KVM*) from the **Hypervisor** /**Platform** list box.





d. Scroll down to review the images that match your selection criteria, and click **Download** for the image you want.



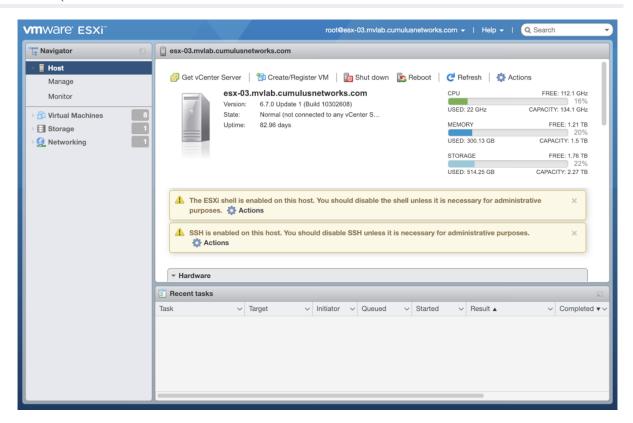
3. Open your hypervisor and set up your VM. You can use these examples for reference or use your own hypervisor instructions.

#### VMware example

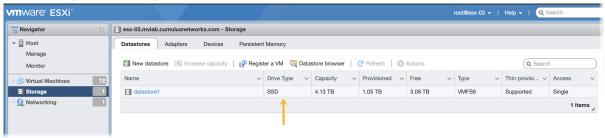
This example shows the VM setup process using an OVA file with VMware ESXi.

- 1. Enter the address of the hardware in your browser.
- 2. Log in to VMware using credentials with root access.





3. Click Storage in the Navigator to verify you have an SSD installed.

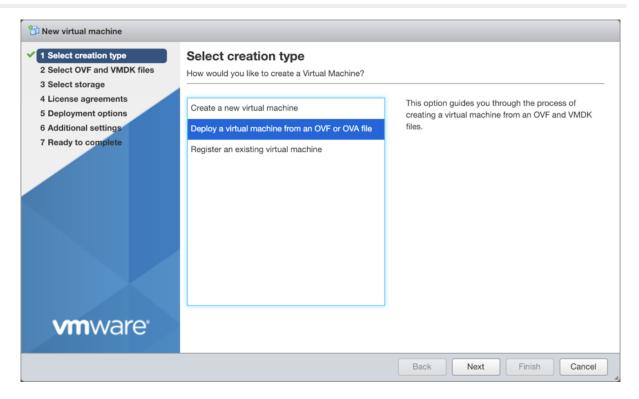


4. Click **Create/Register VM** at the top of the right pane.

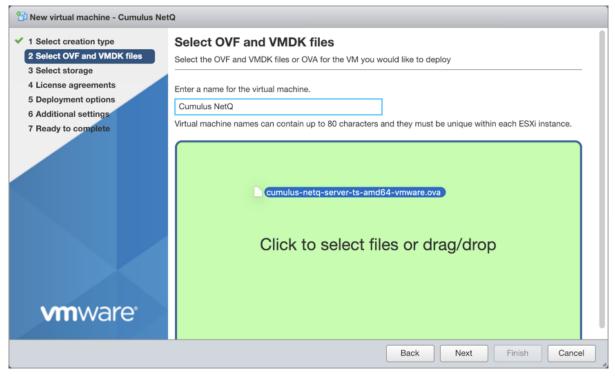


5. Select Deploy a virtual machine from and OVF or OVA file, and click Next.



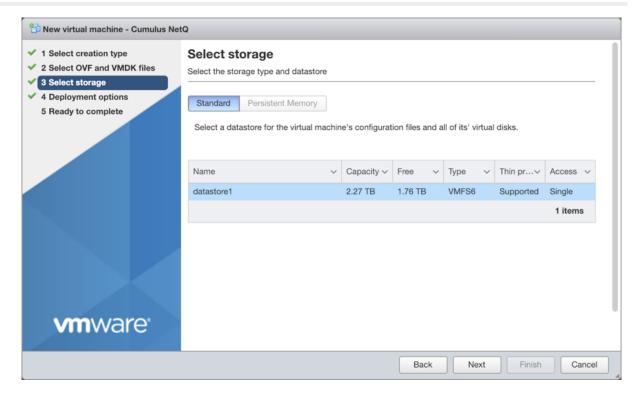


- 6. Provide a name for the VM, for example Cumulus NetQ.
- 7. Drag and drop the NetQ Platform image file you downloaded in Step 1 above.
- 8. Click Next.

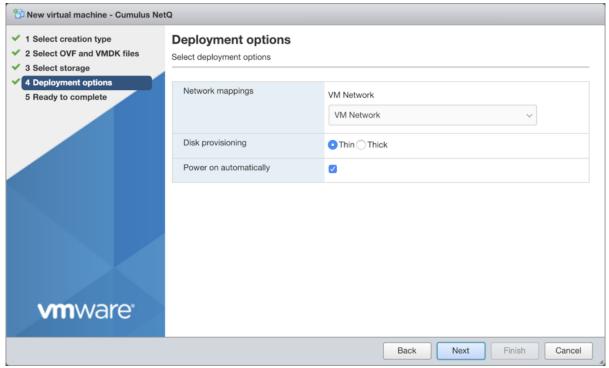


9. Select the storage type and data store for the image to use, then click **Next**. In this example, only one is available.



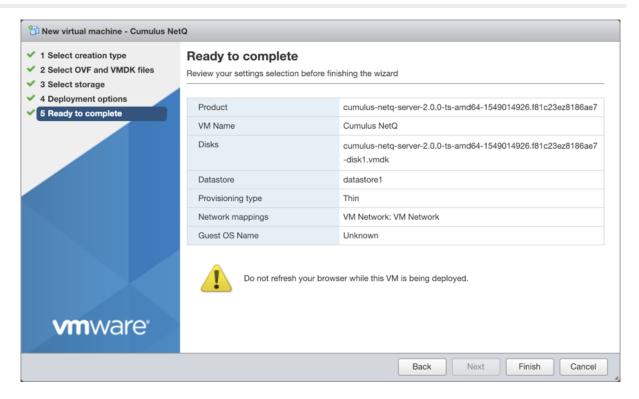


10. Accept the default deployment options or modify them according to your network needs. Click **Next** when you are finished.



11. Review the configuration summary. Click **Back** to change any of the settings, or click **Finish** to continue with the creation of the VM.

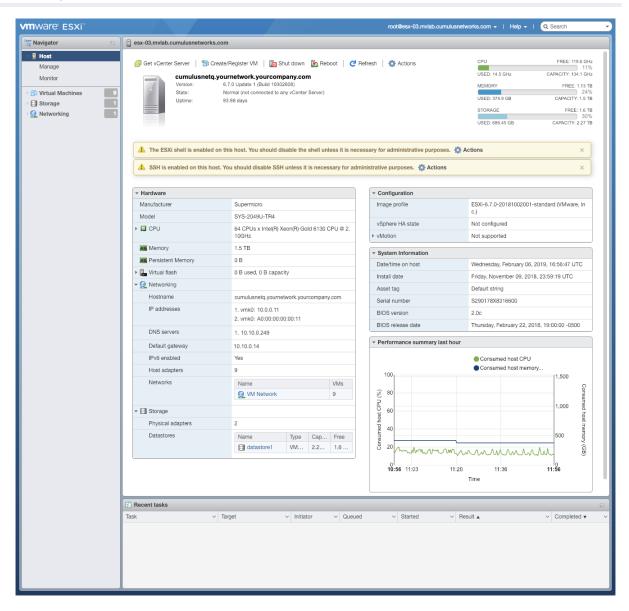




The progress of the request is shown in the Recent Tasks window at the bottom of the application. This may take some time, so continue with your other work until the upload finishes.

12. Once completed, view the full details of the VM and hardware.





#### KVM example

This example shows the VM setup process for a system with Libvirt and KVM/QEMU installed.

1. Confirm that the SHA256 checksum matches the one posted on the Cumulus Downloads website to ensure the image download has not been corrupted.

```
$ sha256sum ./Downloads/cumulus-netq-server-2.1.1-ts-amd64-qemu.
qcow2
$
6fff5f2ac62930799b4e8cc7811abb6840b247e2c9e76ea9ccba03f991f42424
./Downloads/cumulus-netq-server-2.1.1-ts-amd64-qemu.qcow2
```

2. Copy the Qcow2 image to a directory where you want to run it.





Copy, instead of moving, the original Qcow2 image that was downloaded to avoid redownloading it again later should you need to perform this process again.

```
$ sudo mkdir /vms
$ sudo cp ./Downloads/cumulus-netq-server-2.1.1-ts-amd64-qemu.
qcow2 /vms/ts.qcow2
```

3. Create the VM.

For a Direct VM, where the VM uses a MACVLAN interface to sit on the host interface for its connectivity:

```
$ virt-install --name=netq_ts --vcpus=8 --memory=65536 --os-
type=linux --os-variant=debian7 \
   --disk path=/vms/ts.qcow2,format=qcow2,bus=virtio,cache=none \
   --network=type=direct,source=eth0,model=virtio --import --
noautoconsole
```

Replace the disk path value with the location where the Qcow2 image is to reside. Replace network model value (eth0 in the above example) with the name of the interface where the VM is connected to the external network.

Or, for a Bridged VM, where the VM attaches to a bridge which has already been setup to allow for external access:

```
$ virt-install --name=netq_ts --vcpus=8 --memory=65536 --os-
type=linux --os-variant=debian7 \
   --disk path=/vms/ts.qcow2,format=qcow2,bus=virtio,cache=none \
   --network=bridge=br0,model=virtio --import --noautoconsole
```

- Replace network bridge value (br0 in the above example) with the name of the (pre-existing) bridge interface where the VM is connected to the external network.
- 4. Watch the boot process in another terminal window.

```
$ virsh console netq_ts
```

5. From the Console of the VM, check to see which IP address Eth0 has obtained via DHCP, or alternatively set a static IP address with NCLU on the NetQ Appliance or Platform VM.

```
$ ip addr show eth0
```



```
$ net add interface eth0 ip address 10.0.0.1
$ net commit
```



If you need to change the IP address, reboot the VM in order to let the new IP Address be registered properly with the containers running on the NetQ Appliance or Platform.

Please allow 5-10 minutes for the VM to boot.

#### Verify the Installation

- 1. Verify you can access the NetQ CLI.
  - a. From a terminal window, log in to the NetQ Platform using the default credentials (*cumulus /CumulusLinux!*).

```
<computer>:~<username>$ ssh cumulus@<netq-platform-ipaddress>
Warning: Permanently added '<netq-platform-hostname>,
192.168.1.254' (ECDSA) to the list of known hosts.
cumulus@<netq-platform-hostname>'s password: <enter
CumulusLinux! here>

Welcome to Cumulus (R) Linux (R)

For support and online technical documentation, visit
http://www.cumulusnetworks.com/support

The registered trademark Linux (R) is used pursuant to a
sublicense from LMI,
the exclusive licensee of Linus Torvalds, owner of the mark
on a world-wide
basis.

cumulus@<netq-platform-hostname>:~$
```

b. Run the following command to verify all applications are operating properly. **Note**: Please allow 10-15 minutes for all applications to come up and report their status.



Decided   Deci				
true	netg-app-vlan		UP	
2019 netq-app-sensors up f37272c Mon Jun 3 20:20:34 2019 netq-app-topology true 3f4a887 Mon Jun 3 20:20:34 2019 kafka-broker UP x0120:35 2019 netq-app-mstpinfo up ef5565d Mon Jun 3 20:20:35 2019 netq-app-address up 7e0d03d Mon Jun 3 20:20:35 2019 netq-app-kube UP up fbcaa9d Mon Jun 3 20:20:35 2019 netq-app-link UP up fc2771 Mon Jun 3 20:20:34 2019 netq-app-ptm UP true up 7162771 Mon Jun 3 20:20:34 2019 netq-app-clagsession UP true 2019 netq-app-clagsession UP true 2019 netq-app-ospf UP up go582de Mon Jun 3 20:20:34 2019 netq-app-lindp UP true 295e9ed Mon Jun 3 20:20:34 2019 netq-app-lindp UP true 295e9ed Mon Jun 3 20:20:34 2019 netq-app-lindp UP true 295e9ed Mon Jun 3 20:20:34 2019 netq-app-lindp UP true 295e9ed Mon Jun 3 20:20:34 2019 netq-app-lindp UP true 295e9ed Mon Jun 3 20:20:34 2019 netq-app-lindp UP true 295e9ed Mon Jun 3 20:20:34 2019 netq-app-lindp UP true 295e9ed Mon Jun 3 20:20:34 2019 netq-app-lindp UP true 295e9ed Mon Jun 3 20:20:34 2019 netq-app-lindp UP true 295e9ed Mon Jun 3 20:20:34 2019 netq-app-inventory UP true 295e9ed Mon Jun 3 20:20:34 2019 netq-app-infra UP true 2019 netq-app-infra UP true 5484c68 Mon Jun 3 20:20:34 2019 netq-app-infra UP true 5484c68 Mon Jun 3 20:20:34 2019 netq-app-pinfra UP true 5484c68 Mon Jun 3 20:20:34 2019 netq-app-pinfra UP true 5484c68 Mon Jun 3 20:20:34 2019 netq-app-prodevstats UP netq-app-prodevstats UP netq-app-prodevstats UP up b8e280e Mon Jun 3 20:20:34 2019 netq-app-prodevstats UP netq-app-vxlan UP true up b8e280e Mon Jun 3 20:20:34 2019 netq-app-prodevstats UP netq-app-vxlan UP true up b8e280e Mon Jun 3 20:20:34 2019 netq-app-vxlan UP true up b8e280e Mon Jun 3 20:20:34 2019 true			4daed85	Mon Jun 3 20:20:35
netq-app-sensors         UP         true           up         f37272c         Mon Jun         3 20:20:34 2019           netq-app-topology         UP         UP           true         3f4a887         Mon Jun         3 20:20:34 2019           kafka-broker         UP         Mon Jun         3 20:20:34 2019           up         ef5565d         Mon Jun         3 20:20:35 2019         metq-app-address           up         7e0d03d         Mon Jun         3 20:20:35 2019         metq-app-address           up         7e0d03d         Mon Jun         3 20:20:35 2019         metq-app-address         Mon Jun         3 20:20:35 2019           netq-app-duju         DP         True         Mon Jun         3 20:20:35 2019         metq-app-link         UP         true           up         fbcaa9d         Mon Jun         3 20:20:35 2019         metq-app-link         UP         true           up         6c2b2la         Mon Jun         3 20:20:35 2019         metq-app-link         UP         true           up         7162771         Mon Jun         3 20:20:34 2019         metq-app-link         UP         true           up         356dda9         Mon Jun         3 20:20:34 2019         metq-app-link	2019			
up         f37272c         Mon Jun         3 20:20:34 2019           netq-app-topology         UP           true         3f4a887         Mon Jun         3 20:20:34           2019         kafka-broker         Won         Jun         3 20:20:34           2019         kafka-broker         Won         Jun         3 20:20:35         2019           netq-app-mstpinfo         UP         true         UP         true         UP         true         UP         true         UP         U		S	ΙΙΡ	true
Netq-app-topology				
true 3f4a887 Mon Jun 3 20:20:34 2019	_			3 20 20 31 2019
2019 kafka-broker UP		3 Y		Mon Tun 3 20:20:24
kafka-broker         UP       Mon Jun 3         20:20:35 2019       netq-app-mstpinfo       UP       true         up       ef5565d       Mon Jun 3       20:20:35 2019       netq-app-address         up       7e0d03d       Mon Jun 3       20:20:35 2019       netq-gui         UP       Mon Jun 3       20:20:35 2019       Mon Jun 3       20:20:34 2019         netq-app-kube       UP       true       true         up       fbcaa9d       Mon Jun 3       20:20:34 2019       true         up       fbcaa9d       Mon Jun 3       20:20:35 2019       true         up       fbcaa9d       Mon Jun 3       20:20:35 2019       true         up       fbcaa9d       Mon Jun 3       20:20:35 2019       true         up       7162771       Mon Jun 3       20:20:35 2019       true         2019       true       WP       true       WD       true         2019       metq-app-clagsession       UP       WD       true       WD       T			JI 4aoo7	Mon dan 3 20.20.34
UP				
20:20:35 2019         netq-app-mstpinfo         UP         true           up         ef5565d         Mon Jun         3 20:20:35 2019         true           up         7e0d03d         Mon Jun         3 20:20:35 2019         netq-app-address           up         7e0d03d         Mon Jun         3 20:20:35 2019         netq-app-link           up         fbcaa9d         Mon Jun         3 20:20:34 2019         netq-app-link           up         fbcaa9d         Mon Jun         3 20:20:35 2019         true           up         fbcaa9d         Mon Jun         3 20:20:34 2019         true           up         fbcaa9d         Mon Jun         3 20:20:35 2019         true           up         fbcaa9d         Mon Jun         3 20:20:35 2019         true           up         fbcaa9d         Mon Jun         3 20:20:35 2019         true           up         7162771         Mon Jun         3 20:20:34 2019         true           up         356dda9         Mon Jun         3 20:20:34 2019         true           up         356dda9         Mon Jun         3 20:20:34 2019         true           up         e0e2ab0         Mon Jun         3 20:20:34 2019         true				Mon Tun 2
netq-app-mstpinfo         UP         true           up         ef5565d         Mon Jun         3 20:20:35 2019           netq-app-address         UP         true           up         7e0d03d         Mon Jun         3 20:20:35 2019           netq-gui         Mon Jun         3 20:20:35 2019         Mon Jun         3 20:20:34 2019           netq-app-kube         UP         true         up         true           up         fbcaa9d         Mon Jun         3 20:20:34 2019         true           up         fbcaa9d         Mon Jun         3 20:20:35 2019         true           up         fbcaa9d         Mon Jun         3 20:20:34 2019         true           up         fbcaa9d         Mon Jun         3 20:20:35 2019         true           up         fbcaa9d         Mon Jun         3 20:20:34 2019         true           up         fbcaa9d         Mon Jun         3 20:20:34 2019         true           up         356da9         Mon Jun         3 20:20:34 2019         true           up         e0e2ab0         Mon Jun         3 20:20:34 2019         true           up         e0e2ab0         Mon Jun         3 20:20:35 2019         true <t< td=""><td></td><td></td><td></td><td>MOII Juli 3</td></t<>				MOII Juli 3
up         ef5565d         Mon Jun         3         20:20:35         2019           netq-app-address         UP         true           up         7e0d03d         Mon Jun         3         20:20:35         2019           netq-gui         UP         Mon Jun         3         20:20:35         2019           netq-app-kube         UP         true         up         6c2b21a         Mon Jun         3         20:20:34         2019           netq-app-link         UP         true         up         true         up         true           up         6c2b21a         Mon Jun         3         20:20:35         2019         true           up         6c2b21a         Mon Jun         3         20:20:35         2019         true           up         7162771         Mon Jun         3         20:20:34         2019         true           2019         109		c _	IID	£
netq-app-address         UP         true           up         7e0d03d         Mon Jun         3 20:20:35 2019           netq-gui         Mon Jun         3 20:20:35 2019           netq-app-kube         UP         true           up         fbcaa9d         Mon Jun         3 20:20:34 2019           netq-app-link         UP         true           up         6c2b2la         Mon Jun         3 20:20:35 2019           netq-app-ptm         UP         true           up         7162771         Mon Jun         3 20:20:34 2019           netq-opta         UP         true           2019         Won Jun         3 20:20:34 2019           netq-opta         UP         true           2019         true         Mon Jun         3 20:20:34 2019           netq-app-clagsession         UP         true           up         356dda9         Mon Jun         3 20:20:34 2019           netq-endpoint-gateway         UP         true           up         e0e2ab0         Mon Jun         3 20:20:34 2019           netq-app-lidp         UP         true           up         90582de         Mon Jun         3 20:20:35 2019           netq-app-				
up         7e0d03d         Mon Jun         3 20:20:35 2019           netq-gui         Mon Jun         3 20:20:35 2019           netq-app-kube         UP         true           up         fbcaa9d         Mon Jun         3 20:20:34 2019           netq-app-link         UP         true           up         6c2b21a         Mon Jun         3 20:20:35 2019           netq-app-ptm         UP         true           up         7162771         Mon Jun         3 20:20:34 2019           netq-opta         UP         true           up         7162771         Mon Jun         3 20:20:34 2019           netq-opta         UP         true           up         356dda9         Mon Jun         3 20:20:34 2019           netq-app-clagsession         UP         true           up         402         Mon Jun         3 20:20:34 2019           netq-endpoint-gateway         UP         true           up         e0e2ab0         Mon Jun         3 20:20:34 2019           netq-app-ospf         UP         true           up         90582de         Mon Jun         3 20:20:35 2019           netq-app-infra         UP         true	_			
netq-gui UP 20:20:35 2019 netq-app-kube UP perpose because the content of the con			~ -	
UP true  up fbcaa9d Mon Jun 3 20:20:34 2019  netq-app-kube up fbcaa9d Mon Jun 3 20:20:35 2019  netq-app-link up 6c2b2la Mon Jun 3 20:20:35 2019  netq-app-ptm up 7162771 Mon Jun 3 20:20:34 2019  netq-opta UP  true up 356dda9 Mon Jun 3 20:20:34 2019  netq-app-clagsession UP true 295e9ed Mon Jun 3 20:20:34  2019  netq-app-ospf UP true up e0e2ab0 Mon Jun 3 20:20:34 2019  netq-app-lldp UP true up 90582de Mon Jun 3 20:20:35 2019  netq-app-inventory UP true up bbf9938 Mon Jun 3 20:20:35 2019  netq-app-tracecheck-scheduler UP true 5484c68 Mon Jun 3 20:20:34  2019  netq-app-infra UP true up 13f9e7c Mon Jun 3 20:20:34 2019  kafka-connect UP true 5484c68 Mon Jun 3 20:20:34  20:20:35 2019  netq-app-search UP true up b8e280e Mon Jun 3 20:20:34 2019  netq-app-procdevstats UP true up b8e280e Mon Jun 3 20:20:34 2019  netq-app-vxlan UP true true up b8e280e Mon Jun 3 20:20:34 2019	_	7e0d03d	Mon Jun	3 20:20:35 2019
20:20:35 2019       netq-app-kube       UP       true         up       fbcaa9d       Mon Jun       3 20:20:34 2019         netq-app-link       UP       true         up       6c2b21a       Mon Jun       3 20:20:35 2019         netq-app-ptm       UP       true         up       7162771       Mon Jun       3 20:20:34 2019         netq-opta       UP       true         2019       Mon Jun       3 20:20:34 2019         netq-app-clagsession       UP       true         up       356dda9       Mon Jun       3 20:20:34 2019         netq-endpoint-gateway       UP       true         2019       100       true       100         netq-endpoint-gateway       UP       true       100         up       e0e2ab0       Mon Jun       3 20:20:34 2019       100         netq-app-ospf       UP       true       100 <td></td> <td></td> <td></td> <td></td>				
netq-app-kube         UP         true           up         fbcaa9d         Mon Jun         3 20:20:34 2019           netq-app-link         UP         true           up         6c2b21a         Mon Jun         3 20:20:35 2019           netq-app-ptm         UP         true           up         7162771         Mon Jun         3 20:20:34 2019           netq-opta         UP         true           2019         Mon Jun         3 20:20:34 2019           netq-app-clagsession         UP         true           up         356dda9         Mon Jun         3 20:20:34 2019           netq-endpoint-gateway         UP         true           up         295e9ed         Mon Jun         3 20:20:34 2019           netq-endpoint-gateway         UP         true           up         e0e2ab0         Mon Jun         3 20:20:34 2019           netq-endpoint-gateway         UP         true           up         e0e2ab0         Mon Jun         3 20:20:34 2019           netq-app-lospf         UP         true           up         bf9938         Mon Jun         3 20:20:35 2019           netq-app-infra         UP         true           u				Mon Jun 3
up         fbcaa9d         Mon Jun         3 20:20:34 2019           netq-app-link         UP         true           up         6c2b2la         Mon Jun         3 20:20:35 2019           netq-app-ptm         UP         true           up         7162771         Mon Jun         3 20:20:34 2019           netq-opta         UP         true           2019         Mon Jun         3 20:20:34 2019           netq-app-clagsession         UP         true           up         356dda9         Mon Jun         3 20:20:34 2019           netq-endpoint-gateway         UP         true           2019         UP         true           up         e0e2ab0         Mon Jun         3 20:20:34 2019           netq-app-ospf         UP         true           up         90582de         Mon Jun         3 20:20:34 2019           netq-app-inventory         UP         true           up         bbf9938         Mon Jun         3 20:20:34 2019           netq-app-infra         UP         true           up         13f9e7c         Mon Jun         3 20:20:34 2019           kafka-connect         UP         true           up         4				
netq-app-link         UP         true           up         6c2b21a         Mon Jun         3 20:20:35 2019           netq-app-ptm         UP         true           up         7162771         Mon Jun         3 20:20:34 2019           netq-opta         UP         mon Jun         3 20:20:34 2019           true         Mon Jun         3 20:20:34 2019           netq-app-clagsession         UP         true           up         356dda9         Mon Jun         3 20:20:34 2019           netq-endpoint-gateway         UP         true           2019         UP         true           up         e0e2ab0         Mon Jun         3 20:20:34 2019           netq-app-ospf         UP         true           up         90582de         Mon Jun         3 20:20:34 2019           netq-app-inventory         UP         true           up         bbf9938         Mon Jun         3 20:20:34 2019           netq-app-tracecheck-scheduler         UP         true           up         13f9e7c         Mon Jun         3 20:20:34 2019           kafka-connect         UP         Mon Jun         3 20:20:34 2019           netq-app-search         UP         tru	netq-app-kube			
up         6c2b21a         Mon Jun         3 20:20:35 2019           netq-app-ptm         UP         true           up         7162771         Mon Jun         3 20:20:34 2019           netq-opta         UP         true           true         Mon Jun         3 20:20:34 2019           netq-app-clagsession         UP         true           up         356dda9         Mon Jun         3 20:20:34 2019           netq-endpoint-gateway         UP         true           2019         Mon Jun         3 20:20:34 2019           netq-endpoint-gateway         UP         true           2019         true         295e9ed         Mon Jun         3 20:20:34 2019           netq-endpoint-gateway         UP         true         401 <t< td=""><td>up</td><td>fbcaa9d</td><td>Mon Jun</td><td>3 20:20:34 2019</td></t<>	up	fbcaa9d	Mon Jun	3 20:20:34 2019
netq-app-ptm         UP         true           up         7162771         Mon Jun         3 20:20:34 2019           netq-opta         UP         true           true         Mon Jun         3 20:20:34           2019         netq-app-clagsession         UP         true           up         356dda9         Mon Jun         3 20:20:34 2019           netq-endpoint-gateway         UP         true           2019         true         295e9ed         Mon Jun         3 20:20:34 2019           netq-app-ospf         UP         true           up         e0e2ab0         Mon Jun         3 20:20:34 2019           netq-app-lidp         UP         true           up         90582de         Mon Jun         3 20:20:35 2019           netq-app-inventory         UP         true           up         bbf9938         Mon Jun         3 20:20:34 2019           netq-app-tracecheck-scheduler         UP         true           up         13f9e7c         Mon Jun         3 20:20:34 2019           kafka-connect         UP         true           UP         true           up         e47aaba         Mon Jun         3 20:20:34 2019	netq-app-link		UP	true
up       7162771       Mon Jun       3 20:20:34 2019         netq-opta       UP         true       Mon Jun       3 20:20:34         2019       true       Description         netq-app-clagsession       UP       true         up       356dda9       Mon Jun       3 20:20:34 2019         netq-endpoint-gateway       UP       true         2019       true       295e9ed       Mon Jun       3 20:20:34 2019         netq-app-ospf       UP       true         up       e0e2ab0       Mon Jun       3 20:20:34 2019         netq-app-lidp       UP       true         up       bbf9938       Mon Jun       3 20:20:35 2019         netq-app-tracecheck-scheduler       UP       true         2019       true       5484c68       Mon Jun       3 20:20:34 2019         hetq-app-infra       UP       true         up       13f9e7c       Mon Jun       3 20:20:34 2019         kafka-connect       UP       true         up       e47aaba       Mon Jun       3 20:20:34 2019         netq-app-procdevstats       UP       true         up       b8e280e       Mon Jun       3 20:20:34 2019	up	6c2b21a	Mon Jun	3 20:20:35 2019
netq-opta         UP           true         Mon Jun 3 20:20:34           2019         netq-app-clagsession         UP         true           up         356dda9         Mon Jun 3 20:20:34 2019         netq-endpoint-gateway         UP           true         295e9ed         Mon Jun 3 20:20:34         20:20:34           2019         Detrue         UP         true           up         e0e2ab0         Mon Jun 3 20:20:34 2019         Detrue           up         po582de         Mon Jun 3 20:20:35 2019         Detrue           up         bbf9938         Mon Jun 3 20:20:34 2019         Detrue           up         bbf9938         Mon Jun 3 20:20:34 2019         Detrue           true         5484c68         Mon Jun 3 20:20:34 2019         Mon Jun 3 20:20:34 2019           netq-app-infra         UP         true         Detrue         Mon Jun 3 20:20:34 2019           kafka-connect         UP         true         Mon Jun 3 20:20:34 2019         Detrue           up         e47aaba         Mon Jun 3 20:20:34 2019         Detrue         Detrue         Detrue           up         b8e280e         Mon Jun 3 20:20:34 2019         Detrue         Detrue         Detrue         Detrue         Detrue <t< td=""><td>netq-app-ptm</td><td></td><td>UP</td><td>true</td></t<>	netq-app-ptm		UP	true
true	up	7162771	Mon Jun	3 20:20:34 2019
2019       netq-app-clagsession       UP       true         up       356dda9       Mon Jun       3 20:20:34 2019         netq-endpoint-gateway       UP       true       295e9ed       Mon Jun       3 20:20:34 2019         true       295e9ed       Mon Jun       3 20:20:34 2019       10 10 10 10 10 10 10 10 10 10 10 10 10 1	netq-opta		UP	
netq-app-clagsession         UP         true           up         356dda9         Mon Jun         3 20:20:34 2019           netq-endpoint-gateway         UP           true         295e9ed         Mon Jun         3 20:20:34           2019         true         UP         true           up         e0e2ab0         Mon Jun         3 20:20:34 2019           netq-app-lidp         UP         true           up         90582de         Mon Jun         3 20:20:35 2019           netq-app-inventory         UP         true           up         bbf9938         Mon Jun         3 20:20:34 2019           netq-app-tracecheck-scheduler         UP           true         5484c68         Mon Jun         3 20:20:34           2019         true         up         13f9e7c         Mon Jun         3 20:20:34 2019           kafka-connect         UP         Mon Jun         3 20:20:34 2019           netq-app-search         UP         true           up         e47aaba         Mon Jun         3 20:20:34 2019           netq-app-procdevstats         UP         true           up         b8e280e         Mon Jun         3 20:20:34 2019           netq-app	true			Mon Jun 3 20:20:34
up       356dda9       Mon Jun       3 20:20:34 2019         netq-endpoint-gateway       UP         true       295e9ed       Mon Jun       3 20:20:34         2019       netq-app-ospf       UP       true         up       e0e2ab0       Mon Jun       3 20:20:34 2019         netq-app-lldp       UP       true         up       90582de       Mon Jun       3 20:20:35 2019         netq-app-inventory       UP       true         up       bbf9938       Mon Jun       3 20:20:34 2019         netq-app-tracecheck-scheduler       UP         true       5484c68       Mon Jun       3 20:20:34 2019         netq-app-infra       UP       true         up       13f9e7c       Mon Jun       3 20:20:34 2019         kafka-connect       UP       Mon Jun       3 20:20:34 2019         netq-app-search       UP       true         up       e47aaba       Mon Jun       3 20:20:34 2019         netq-app-procdevstats       UP       true         up       b8e280e       Mon Jun       3 20:20:34 2019         netq-app-vxlan       UP       true	2019			
netq-endpoint-gateway  true  295e9ed  Mon Jun 3 20:20:34  2019  netq-app-ospf  UP  true  up  e0e2ab0  Mon Jun 3 20:20:34 2019  netq-app-lldp  up  p0582de  Mon Jun 3 20:20:35 2019  netq-app-inventory  UP  true  up  bbf9938  Mon Jun 3 20:20:35 2019  netq-app-tracecheck-scheduler  UP  true  5484c68  Mon Jun 3 20:20:34  2019  netq-app-infra  UP  true  up  13f9e7c  Mon Jun 3 20:20:34  2019  kafka-connect  UP  kafka-connect  UP  Mon Jun 3 20:20:34  2019  netq-app-search  UP  true  up  e47aaba  Mon Jun 3 20:20:34  2019  netq-app-procdevstats  UP  true  up  b8e280e  Mon Jun 3 20:20:34  2019  netq-app-vxlan  UP  true  up  b8e280e  Mon Jun 3 20:20:34  2019  true  up  b8e280e  Mon Jun 3 20:20:34  2019  true  up  b8e280e  Mon Jun 3 20:20:34  2019  true	netq-app-clagses	ssion	UP	true
netq-endpoint-gateway  true  295e9ed  Mon Jun 3 20:20:34  2019  netq-app-ospf  UP  true  up  e0e2ab0  Mon Jun 3 20:20:34 2019  netq-app-lldp  up  p0582de  Mon Jun 3 20:20:35 2019  netq-app-inventory  UP  true  up  bbf9938  Mon Jun 3 20:20:35 2019  netq-app-tracecheck-scheduler  UP  true  5484c68  Mon Jun 3 20:20:34  2019  netq-app-infra  UP  true  up  13f9e7c  Mon Jun 3 20:20:34  2019  kafka-connect  UP  kafka-connect  UP  Mon Jun 3 20:20:34  2019  netq-app-search  UP  true  up  e47aaba  Mon Jun 3 20:20:34  2019  netq-app-procdevstats  UP  true  up  b8e280e  Mon Jun 3 20:20:34  2019  netq-app-vxlan  UP  true  up  b8e280e  Mon Jun 3 20:20:34  2019  true  up  b8e280e  Mon Jun 3 20:20:34  2019  true  up  b8e280e  Mon Jun 3 20:20:34  2019  true			Mon Jun	3 20:20:34 2019
true 295e9ed Mon Jun 3 20:20:34 2019  netq-app-ospf UP true  up e0e2ab0 Mon Jun 3 20:20:34 2019  netq-app-lldp UP true  up 90582de Mon Jun 3 20:20:35 2019  netq-app-inventory UP true  up bbf9938 Mon Jun 3 20:20:34 2019  netq-app-tracecheck-scheduler UP  true 5484c68 Mon Jun 3 20:20:34 2019  netq-app-infra UP true  up 13f9e7c Mon Jun 3 20:20:34 2019  kafka-connect  UP Mon Jun 3 20:20:34 2019  netq-app-search UP true  up e47aaba Mon Jun 3 20:20:34 2019  netq-app-procdevstats UP true  up b8e280e Mon Jun 3 20:20:34 2019  netq-app-vxlan UP true  up true  up b8e280e Mon Jun 3 20:20:34 2019  netq-app-vxlan UP true	netg-endpoint-ga	ateway	UP	
2019       netq-app-ospf       UP       true         up       e0e2ab0       Mon Jun       3 20:20:34 2019         netq-app-lldp       UP       true         up       90582de       Mon Jun       3 20:20:35 2019         netq-app-inventory       UP       true         up       bbf9938       Mon Jun       3 20:20:34 2019         netq-app-tracecheck-scheduler       UP         true       5484c68       Mon Jun       3 20:20:34         2019       netq-app-infra       UP       true         up       13f9e7c       Mon Jun       3 20:20:34 2019         kafka-connect       UP       Mon Jun       3 20:20:34 2019         netq-app-search       UP       true         up       e47aaba       Mon Jun       3 20:20:34 2019         netq-app-procdevstats       UP       true         up       b8e280e       Mon Jun       3 20:20:34 2019         netq-app-vxlan       UP       true		-	295e9ed	Mon Jun 3 20:20:34
netq-app-ospf         UP         true           up         e0e2ab0         Mon Jun         3 20:20:34 2019           netq-app-lldp         UP         true           up         90582de         Mon Jun         3 20:20:35 2019           netq-app-inventory         UP         true           up         bbf9938         Mon Jun         3 20:20:34 2019           netq-app-tracecheck-scheduler         UP           true         5484c68         Mon Jun         3 20:20:34           2019         netq-app-infra         UP         true           up         13f9e7c         Mon Jun         3 20:20:34 2019           kafka-connect         UP         Mon Jun         3 20:20:34 2019           netq-app-search         UP         true           up         e47aaba         Mon Jun         3 20:20:34 2019           netq-app-procdevstats         UP         true           up         b8e280e         Mon Jun         3 20:20:34 2019           netq-app-vxlan         UP         true				
up       e0e2ab0       Mon Jun       3 20:20:34 2019         netq-app-lldp       UP       true         up       90582de       Mon Jun       3 20:20:35 2019         netq-app-inventory       UP       true         up       bbf9938       Mon Jun       3 20:20:34 2019         netq-app-tracecheck-scheduler       UP         true       5484c68       Mon Jun       3 20:20:34         2019       netq-app-infra       UP       true         up       13f9e7c       Mon Jun       3 20:20:34 2019         kafka-connect       UP       Mon Jun       3 20:20:34 2019         netq-app-search       UP       true         up       e47aaba       Mon Jun       3 20:20:34 2019         netq-app-procdevstats       UP       true         up       b8e280e       Mon Jun       3 20:20:34 2019         netq-app-vxlan       UP       true			ΙΙΡ	true
netq-app-lldp         UP         true           up         90582de         Mon Jun         3 20:20:35 2019           netq-app-inventory         UP         true           up         bbf9938         Mon Jun         3 20:20:34 2019           netq-app-tracecheck-scheduler         UP         true           true         5484c68         Mon Jun         3 20:20:34           2019         netq-app-infra         UP         true           up         13f9e7c         Mon Jun         3 20:20:34 2019           kafka-connect         UP         Mon Jun         3 20:20:34 2019           netq-app-search         UP         true           up         e47aaba         Mon Jun         3 20:20:34 2019           netq-app-procdevstats         UP         true           up         b8e280e         Mon Jun         3 20:20:34 2019           netq-app-vxlan         UP         true		e0e2ab0		
up       90582de       Mon Jun       3 20:20:35 2019         netq-app-inventory       UP       true         up       bbf9938       Mon Jun       3 20:20:34 2019         netq-app-tracecheck-scheduler       UP         true       5484c68       Mon Jun       3 20:20:34         2019       netq-app-infra       UP       true         up       13f9e7c       Mon Jun       3 20:20:34 2019         kafka-connect       UP       Mon Jun       3 20:20:34 2019         netq-app-search       UP       true         up       e47aaba       Mon Jun       3 20:20:34 2019         netq-app-procdevstats       UP       true         up       b8e280e       Mon Jun       3 20:20:34 2019         netq-app-vxlan       UP       true	_	coczazo		
netq-app-inventory UP true  up bbf9938 Mon Jun 3 20:20:34 2019  netq-app-tracecheck-scheduler UP  true 5484c68 Mon Jun 3 20:20:34  2019  netq-app-infra UP true  up 13f9e7c Mon Jun 3 20:20:34 2019  kafka-connect  UP Mon Jun 3  20:20:35 2019  netq-app-search UP true  up e47aaba Mon Jun 3 20:20:34 2019  netq-app-procdevstats UP true  up b8e280e Mon Jun 3 20:20:34 2019  netq-app-vxlan UP true		9058246		
up       bbf9938       Mon Jun       3 20:20:34 2019         netq-app-tracecheck-scheduler       UP         true       5484c68       Mon Jun       3 20:20:34         2019       netq-app-infra       UP       true         up       13f9e7c       Mon Jun       3 20:20:34 2019         kafka-connect       UP       Mon Jun       3 20:20:34 2019         netq-app-search       UP       true         up       e47aaba       Mon Jun       3 20:20:34 2019         netq-app-procdevstats       UP       true         up       b8e280e       Mon Jun       3 20:20:34 2019         netq-app-vxlan       UP       true	_			
netq-app-tracecheck-scheduler UP true 5484c68 Mon Jun 3 20:20:34 2019 netq-app-infra UP true up 13f9e7c Mon Jun 3 20:20:34 2019 kafka-connect UP Mon Jun 3 20:20:35 2019 netq-app-search UP true up e47aaba Mon Jun 3 20:20:34 2019 netq-app-procdevstats UP true up b8e280e Mon Jun 3 20:20:34 2019 netq-app-vxlan UP true		_		
true 5484c68 Mon Jun 3 20:20:34 2019 netq-app-infra UP true up 13f9e7c Mon Jun 3 20:20:34 2019 kafka-connect UP Mon Jun 3 20:20:35 2019 netq-app-search UP true up e47aaba Mon Jun 3 20:20:34 2019 netq-app-procdevstats UP true up b8e280e Mon Jun 3 20:20:34 2019 netq-app-vxlan UP true	_			3 20.20.34 2019
2019 netq-app-infra		ileck-sciled		Mon Tun 2 20:20:24
netq-app-infra       UP       true         up       13f9e7c       Mon Jun 3 20:20:34 2019         kafka-connect       Won Jun 3         UP       Mon Jun 3         20:20:35 2019       UP       true         up       e47aaba       Mon Jun 3 20:20:34 2019         netq-app-procdevstats       UP       true         up       b8e280e       Mon Jun 3 20:20:34 2019         netq-app-vxlan       UP       true			5464000	Mon Jun 3 20.20.34
up       13f9e7c       Mon Jun       3 20:20:34 2019         kafka-connect       Won Jun       3 20:20:34 2019         up       Mon Jun       3 20:20:34 2019         netq-app-procdevstats       UP       true         up       b8e280e       Mon Jun       3 20:20:34 2019         netq-app-vxlan       UP       true			IID	+ m. o
kafka-connect UP		1250-7-		
UP	-	1319e/c	Mon Jun	3 20:20:34 2019
20:20:35 2019  netq-app-search  up  e47aaba  Mon Jun  3 20:20:34 2019  netq-app-procdevstats  UP  true  up  b8e280e  Mon Jun  3 20:20:34 2019  netq-app-vxlan  UP  true				
netq-app-search UP true up e47aaba Mon Jun 3 20:20:34 2019 netq-app-procdevstats UP true up b8e280e Mon Jun 3 20:20:34 2019 netq-app-vxlan UP true				Mon Jun 3
upe47aabaMon Jun 3 20:20:34 2019netq-app-procdevstatsUPtrueupb8e280eMon Jun 3 20:20:34 2019netq-app-vxlanUPtrue				
netq-app-procdevstats UP true up b8e280e Mon Jun 3 20:20:34 2019 netq-app-vxlan UP true		45		
up b8e280e Mon Jun 3 20:20:34 2019 netq-app-vxlan UP true				
netq-app-vxlan UP true	netq-app-procder			
	_	b8e280e		3 20:20:34 2019
up 123c577 Mon Jun 3 20:20:34 2019	netq-app-vxlan			
	up	123c577	Mon Jun	3 20:20:34 2019



zookeeper UP			Mon Jun 3
20:20:35 2019			MOII UUII 3
netq-app-resour	ce-util	UP	true
up	41dfb07		3 20:20:34 2019
netq-app-evpn	TIGIDO /	UP	true
up	05a4003	_	3 20:20:34 2019
netq-api-gatewa		UP	3 20.20.34 2019
true	I	_	Mon Jun 3 20:20:34
2019		C102514	11011 0 011 3 20 20 31
netq-app-port		UP	true
up	4592b70		3 20:20:35 2019
netq-app-macs	1332270	UP	3 20 20 33 2019
true		dd6cd96	Mon Jun 3 20:20:35
2019		aavearo	
netq-app-notifi	er	UP	true
up	da57b69		3 20:20:35 2019
netq-app-events		UP	true
up	8f7b4d9	Mon Jun	3 20:20:34 2019
netq-app-servic		UP	true
up	5094f4a	Mon Jun	3 20:20:34 2019
cassandra			
UP			Mon Jun 3
20:20:35 2019			
netq-app-config	diff	UP	true
up	3be2ef1	Mon Jun	3 20:20:34 2019
netq-app-neighb	or	UP	true
up	9ebe479	Mon Jun	3 20:20:35 2019
netq-app-bgp		UP	true
up	e68f7a8	Mon Jun	3 20:20:35 2019
schema-registry			
UP			Mon Jun 3
20:20:35 2019			
netq-app-lnv		UP	true
up	a9ca80a	Mon Jun	3 20:20:34 2019
netq-app-health	dashboard	UP	
true		eea044c	Mon Jun 3 20:20:34
2019			
netq-app-ntp		UP	true
up	651c86f	Mon Jun	3 20:20:35 2019
netq-app-custom	ermgmt	UP	
true		7250354	Mon Jun 3 20:20:34
2019			
netq-app-node		UP	true
up	f676c9a	Mon Jun	3 20:20:34 2019
netq-app-route		UP	true
up	6e31f98	Mon Jun	3 20:20:35 2019
cumulus@ <netq-p< td=""><td>latform-ho</td><td>stname&gt;:~\$</td><td></td></netq-p<>	latform-ho	stname>:~\$	







If any of the applications or services display Status as DOWN after 30 minutes, open a support ticket and attach the output of the opta-support command.

- 2. Verify that NTP is configured and running. NTP operation is critical to proper operation of NetQ. Refer to Setting Date and Time in the *Cumulus Linux User Guide* for details and instructions.
- 3. Continue the NetQ installation by loading the NetQ Agent on each switch or host you want to monitor. Refer to the next section for instructions.

## **Install the NetQ Agent**

Whether using the NetQ Appliance or your own hardware, the NetQ Agent must be installed on each node you want to monitor. The node can be a:

- Switch running Cumulus Linux version 3.7.0 or later
- Server running Red Hat RHEL 7.1, Ubuntu 16.04 or CentOS 7
- Linux virtual machine running any of the above Linux operating systems

To install the NetQ Agent you need to install the OS-specific meta package, <code>cumulus-netq</code>, on each switch. Optionally, you can install it on hosts. The meta package contains the NetQ Agent, the NetQ command line interface (CLI), and the NetQ library. The library contains modules used by both the NetQ Agent and the CLI.

Instructions for installing the meta package on each node type are included here:

- Install NetQ Agent on a Cumulus Linux Switch (see page 45)
- Install NetQ Agent on an Ubuntu Server (see page 46)
- Install NetQ Agent on a Red Hat or CentOS Server (see page 49)



If your network uses a proxy server for external connections, you should first configure a global proxy so apt-get can access the meta package on the Cumulus Networks repository.

### **Install NetQ Agent on a Cumulus Linux Switch**

A simple process installs the NetQ Agent on a Cumulus switch.

1. Edit the /etc/apt/sources.list file to add the repository for Cumulus NetQ. **Note** that NetQ has a separate repository from Cumulus Linux.

```
cumulus@switch:~$ sudo nano /etc/apt/sources.list
...
deb http://apps3.cumulusnetworks.com/repos/deb CumulusLinux-3
netq-2.1
...
```

0

The repository deb http://apps3.cumulusnetworks.com/repos/deb CumulusLinux-3 netq-latest can be used if you want to always retrieve the latest posted version of NetQ.



2. Update the local apt repository, then install the NetQ meta package on the switch.

```
cumulus@switch:~$ sudo apt-get update
cumulus@switch:~$ sudo apt-get install cumulus-netq
```

3. Verify that NTP is running on the host node. Nodes must be in time synchronization with the NetQ Platform to enable useful statistical analysis.

4. Restart rsyslog so log files are sent to the correct destination.

```
cumulus@switch:~$ sudo systemctl restart rsyslog.service
```

5. Configure the NetQ Agent to send telemetry data to the NetQ Platform and, optionally, configure the switch or host to run the NetQ CLI. In this example, the IP address for the agent and cli servers is 192 168 1 254

**Note:** If you intend to use VRF, skip to Configure the Agent to Use VRF (see page 51). If you intend to specify a port for communication, skip to Configure the Agent to Communicate over a Specific Port (see page 52).

```
cumulus@switch:~$ netq config add agent server 192.168.1.254 cumulus@switch:~$ netq config add cli server 192.168.1.254
```

This command updates the configuration in the /etc/netq/netq.yml file and enables the NetQ CLI.

6. Restart NetQ Agent and CLI.

```
cumulus@switch:~$ netq config restart agent cumulus@switch:~$ netq config restart cli
```

Repeat these steps for each Cumulus switch, or use an automation tool to install NetQ Agent on multiple Cumulus Linux switches.



#### Install NetQ Agent on an Ubuntu Server (Optional)

Before you install the NetQ Agent on an Ubuntu server, make sure the following packages are installed and running these minimum versions:

- iproute 1:4.3.0-1ubuntu3.16.04.1 all
- iproute2 4.3.0-1ubuntu3 amd64
- Ildpd 0.7.19-1 amd64
- ntp 1:4.2.8p4+dfsg-3ubuntu5.6 amd64



Make sure you are running lldpd, not lldpad. Ubuntu does not include 11dpd by default, which is required for the installation. To install this package, run the following commands:

```
root@ubuntu:~# apt-get update
root@ubuntu:~# apt-get install lldpd
root@ubuntu:~# systemctl enable lldpd.service
root@ubuntu:~# systemctl start lldpd.service
```

To install the NetQ Agent on an Ubuntu server:

1. Reference and update the local apt repository.

```
root@ubuntu:~# wget -O- https://apps3.cumulusnetworks.com/setup
/cumulus-apps-deb.pubkey | apt-key add -
```

2. Create the file /etc/apt/sources.list.d/cumulus-host-ubuntu-xenial.list and add the following lines:

```
root@ubuntu:~# vi /etc/apt/sources.list.d/cumulus-apps-deb-
xenial.list
deb [arch=amd64] https://apps3.cumulusnetworks.com/repos/deb
xenial netg-latest
```



The use of netg-latest in this example means that a get to the repository always. retrieves the latest version of NetQ, even in the case where a major version update has been made. If you want to keep the repository on a specific version — such as netg-2.1 — use that instead.

3. Install NTP on the server, if not already installed.



#### root@ubuntu:~# sudo apt-get install ntp

- 4. Configure the NTP server.
  - a. Open the /etc/ntp.conf file in your text editor of choice.
  - b. Under the Server section, specify the NTP server IP address or hostname.
- 5. Enable and start the NTP service.

```
root@ubuntu:~# sudo systemctl enable ntp.service root@ubuntu:~# sudo systemctl start ntp.service
```

6. Verify NTP is operating correctly. Look for an asterisk (\*) or a plus sign (+) that indicates the clock is synchronized.

7. Install the meta package on the server.

```
root@ubuntu:~# apt-get update
root@ubuntu:~# apt-get install cumulus-netq
```

8. Configure the NetQ Agent to send telemetry data to the NetQ Platform.

```
user@ubuntu:~# netq config add agent server <netq-platform-ip-address>
Updated agent server 192.168.1.254 vrf default. Please restart netq-agent (netq config restart agent).
```

9. Restart the NetQ Agent

```
user@ubuntu:~# netq config restart agent
```

10. Optionally, configure the Ubuntu server to run the NetQ CLI.



```
user@ubuntu:~# netq config add cli server <netq-platform-ip-address>
Updated cli server 192.168.1.254 vrf default. Please restart netqd (netq config restart cli).
```

11. Restart the CLL

```
user@ubuntu:~# netq config restart cli
```

#### Install NetQ Agent on a Red Hat or CentOS Server (Optional)

Before you install the NetQ Agent on a Red Hat or CentOS server, make sure the following packages are installed and running these minimum versions:

- iproute-3.10.0-54.el7\_2.1.x86\_64
- Ildpd-0.9.7-5.el7.x86 64
  - ①

Make sure you are running lldpd, not lldpad.

CentOS does not include 11dpd by default, nor does it include wget, which is required for the installation. To install this package, run the following commands:

```
root@rhel7:~# yum -y install epel-release
root@rhel7:~# yum -y install lldpd
root@rhel7:~# systemctl enable lldpd.service
root@rhel7:~# systemctl start lldpd.service
root@rhel7:~# yum install wget
```

- ntp-4.2.6p5-25.el7.centos.2.x86\_64
- ntpdate-4.2.6p5-25.el7.centos.2.x86 64

To install the NetQ Agent on a Red Hat or CentOS server:

1. Reference and update the local yum repository.

```
root@rhel7:~# rpm --import https://apps3.cumulusnetworks.com
/setup/cumulus-apps-rpm.pubkey
root@rhel7:~# wget -O- https://apps3.cumulusnetworks.com/setup
/cumulus-apps-rpm-el7.repo > /etc/yum.repos.d/cumulus-host-el.
repo
```

2. Edit /etc/yum.repos.d/cumulus-host-el.repo to set the enabled=1 flag for the two NetQ repositories.



```
root@rhel7:~# vi /etc/yum.repos.d/cumulus-host-el.repo
...
[cumulus-arch-netq-2.1]
name=Cumulus netq packages
baseurl=https://apps3.cumulusnetworks.com/repos/rpm/el/7/netq-2.1
/$basearch
gpgcheck=1
enabled=1
[cumulus-noarch-netq-2.1]
name=Cumulus netq architecture-independent packages
baseurl=https://apps3.cumulusnetworks.com/repos/rpm/el/7/netq-2.1
/noarch
gpgcheck=1
enabled=1
...
```

3. Install NTP on the server.

```
root@rhel7:~# yum install ntp
```

- 4. Configure the NTP server.
  - a. Open the /etc/ntp.conf file in your text editor of choice.
  - b. Under the Server section, specify the NTP server IP address or hostname.
- 5. Enable and start the NTP service.

```
root@rhel7:~# sudo systemctl enable ntpd.service
root@rhel7:~# sudo systemctl start ntpd.service
```

6. Verify NTP is operating correctly. Look for an asterisk (\*) or a plus sign (+) that indicates the clock is synchronized.

```
root@rhel7:~# ntpq -pn
              refid st t when poll reach delay
   remote
offset jitter
______
=========
+173.255.206.154 132.163.96.3 2 u 86 128 377 41.354
2.834 0.602
+12.167.151.2
           198.148.79.209 3 u 103 128 377 13.395
-4.025 0.198
2a00:7600::41 .STEP. 16 u - 1024 0
                                        0.000
0.000 0.000
*129.250.35.250 249.224.99.213 2 u 101 128 377 14.588
-0.299 0.243
```



7. Install the Bash completion and NetQ meta packages on the server.

```
root@rhel7:~# yum -y install bash-completion root@rhel7:~# yum install cumulus-netq
```

8. Configure the NetQ Agent to send telemetry data to the NetQ Platform.

```
root@rhel7:~# netq config add agent server <netq-platform-ip-
address>
Updated agent server 192.168.1.254 vrf default. Please restart
netq-agent (netq config restart agent).
```

9. Restart the NetQ Agent

```
root@rhel7:~# netq config restart agent
```

10. Optionally, configure the RHEL/CentOS server to run the NetQ CLI.

```
root@rhel7:~# netq config add cli server <netq-platform-ip-address>
Updated cli server 192.168.1.254 vrf default. Please restart netqd (netq config restart cli).
```

11. Restart the CLI.

```
root@rhel7:~# netq config restart cli
```

## **Configure Optional NetQ Agent Settings**

Once the NetQ Agents have been installed on the network nodes you want to monitor, the NetQ Agents must be configured to obtain useful and relevant data. The code examples shown in this section illustrate how to configure the NetQ Agent on a Cumulus switch, but it is exactly the same for the other type of nodes. Depending on your deployment, follow the relevant additional instructions after the basic configuration steps:

- Configuring the Agent to Use a VRF (see page 51)
- Configuring the Agent to Communicate over a Specific Port (see page 52)

## Configure the Agent to Use a VRF

While optional, Cumulus strongly recommends that you configure NetQ Agents to communicate with the NetQ Platform only via a VRF, including a management VRF. To do so, you need to specify the VRF name when configuring the NetQ Agent. For example, if the management VRF is configured and you want the agent to communicate with the NetQ Platform over it, configure the agent like this:



```
cumulus@leaf01:~$ netq config add agent server 192.168.1.254 vrf mgmt cumulus@leaf01:~$ netq config add cli server 192.168.254 vrf mgmt
```

You then restart the agent:

```
cumulus@leaf01:~$ netq config restart agent cumulus@leaf01:~$ netq config restart cli
```

#### Configure the Agent to Communicate over a Specific Port

By default, NetQ uses port 31980 for communication between the NetQ Platform and NetQ Agents. If you want the NetQ Agent to communicate with the NetQ Platform via a different port, you need to specify the port number when configuring the NetQ Agent like this:

```
cumulus@leaf01:~$ netq config add agent server 192.168.1.254 port 7379
```

You then restart the agent:

```
cumulus@leaf01:~$ netq config restart agent
```

## **Integrate with Event Notification Tools**

If you want to proactively monitor events in your network, you can integrate NetQ with the PagerDuty or Slack notification tools. To do so you need to configure both the notification application itself to receive the messages, and NetQ with what messages to send and where to send them. Refer to Integrate NetQ with Event Notification Applications to use the CLI for configuration.

## **Set Up Security**

When you set up and configured your Cumulus Linux switches, you likely configured a number of the security features available. Cumulus recommends the same security measures be followed for the NetQ Platform in the out-of-band-network. Refer to the Securing Cumulus Linux white paper for details.

Your Cumulus Linux switches have a number of ports open by default. A few additional ports must be opened to run the NetQ software (refer to Default Open Ports in Cumulus Linux and NetQ article).



# Upgrade from NetQ 1.x to NetQ 2.1.x

This document describes the steps required to upgrade from NetQ 1.x to NetQ 2.1.x.



Data collected in the NetQ 1.x environment cannot be migrated to the NetQ 2.1 environment at this time. Event notification configurations must also be reconfigured as the CLI commands have changed. Upgrading from NetQ 1.x version requires a fresh install as described here.

#### **Contents**

This topic describes how to...

- Prerequisites (see page 53)
  - Hardware Requirements (see page 53)
  - NetQ Platform HyperVisor Requirements (see page 54)
  - Operating System Requirements (see page 54)
  - NetQ Application Support (see page 55)
- Upgrade Workflow (see page 55)
- Upgrade the NetQ Platform (see page 57)
  - Install Using VM (see page 57)
  - Verify the Installation (see page 65)
- Upgrade the NetQ Agent (see page 68)
  - Upgrade NetQ Agent on a Cumulus Linux Switch (see page 68)
  - Upgrade NetQ Agent on an Ubuntu Server (Optional) (see page 70)
  - Upgrade NetQ Agent on a Red Hat or CentOS Server (Optional) (see page 72)
- Configure Optional NetQ Agent Settings (see page 74)
  - Configure the Agent to Use a VRF (see page 74)
  - Configure the Agent to Communicate over a Specific Port (see page 74)
- Integrate with Event Notification Tools (see page 75)
- Set Up Security (see page 75)

# **Prerequisites**

### **Hardware Requirements**

NetQ is supported on the NetQ Appliance and a variety of other hardware.





You must meet these hardware requirements to install the VM and have it run properly.

In all cases, the NetQ software requires a server with the following:

Requirement	Description
Processing power	Eight (8) virtual CPUs
Memory	64 GB RAM
Local disk storage	256 GB SSD ( <b>Note</b> : This must be an SSD; use of other storage options can lead to system instability and are not supported.)
Network interface speed	1 G NIC or higher

If you are *not* using the NetQ Appliance, you must open the following ports on the NetQ server to use the NetQ software:

Port	Component Access
31980	NetQ Platform
32708	API Gateway
32666	Web-based User Interface



These ports have changed from NetQ 1.4 and earlier.

## **NetQ Platform HyperVisor Requirements**

The NetQ Platform can be installed as a Virtual Machine (VM) using one of the following hypervisors:

- VMware ESXi™ 6.5 for servers running Cumulus Linux, CentOS, Ubuntu and RedHat operating systems.
- KVM/QCOW (QEMU Copy on Write) image for servers running CentOS, Ubuntu and RedHat operating systems.

# **Operating System Requirements**

NetQ 2.1 Agents are supported on the following switch or host operating systems:

- Cumulus Linux 3.3.0 and later
- Ubuntu 16.04
- Red Hat<sup>®</sup> Enterprise Linux (RHEL) 7.1



CentOS 7

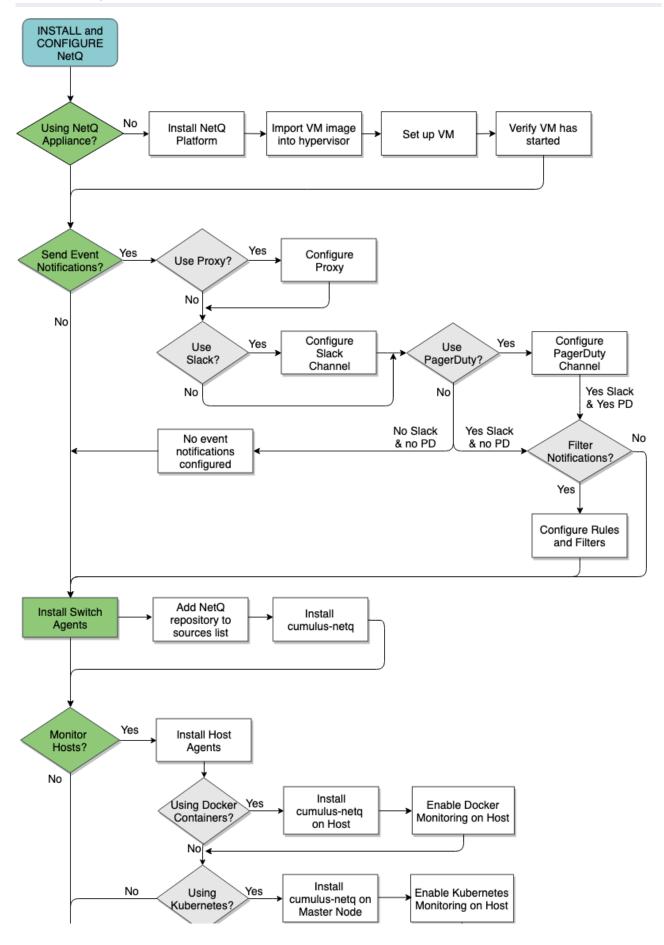
#### **NetQ Application Support**

The NetQ CLI, UI, and RESTful API are supported on NetQ 2.1.0 and later. NetQ 1.4, and earlier, applications are not supported in NetQ 2.x.

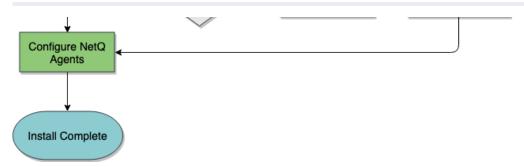
# **Upgrade Workflow**

Installation of NetQ involves installing the NetQ Platform, and installing and configuring the NetQ Agents. Additional steps are needed to Integrate NetQ with Event Notification Applications . This flow chart shows the required steps to install and setup NetQ to start validating your network and the optional steps of integrating with event notification applications and monitoring hosts.









## **Upgrade the NetQ Platform**

If you *are not* using the NetQ Appliance, you must install the NetQ Platform on your selected hardware. If you have deployed the NetQ Appliance, you can skip to Install NetQ Agent (see page 68).

The NetQ Platform is comprised of the following components:

- NetQ applications: network monitoring and analytics functionality
- NetQ CLI: command line user interface for monitoring network and administering NetQ through a terminal session
- NetQ UI: graphical interface for monitoring network and administering NetQ
- NetQ API: Restful application programming interface for accessing NetQ data and integrating with third-party tools
- **NetQ notifier**: application used to send event notifications to third-party notification tools

### **(i)** Best Practice

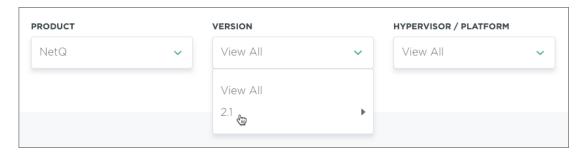
Cumulus Networks recommends you install the NetQ software on a server that is part of an outof-band management network to ensure it can monitor in-band network issues without being affected itself. You should run the software on a separate, powerful server to ensure proper operation and for maximum usability and performance. Refer to Hardware Requirements (see page 53) for specifics.

### **Install Using VM**

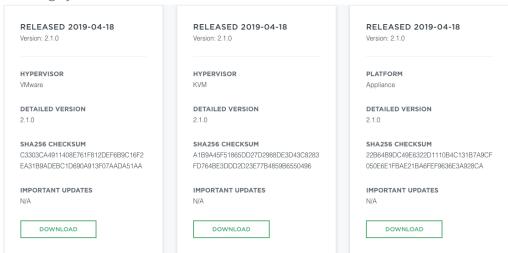
To install the NetQ Platform using a VM image:

- 1. **IMPORTANT**: Confirm that your server hardware meets the requirements set out here (see page 53).
- 2. Download the NetQ Platform image.
  - a. On the Cumulus Downloads page, select NetQ from the **Product** list box.
  - b. Click 2.1 from the **Version** list box, and then select 2.1.x from the submenu.
  - c. Optionally, select the hypervisor you wish to use (*VMware* or *KVM*) from the **Hypervisor** /**Platform** list box.





d. Scroll down to review the images that match your selection criteria, and click **Download** for the image you want.



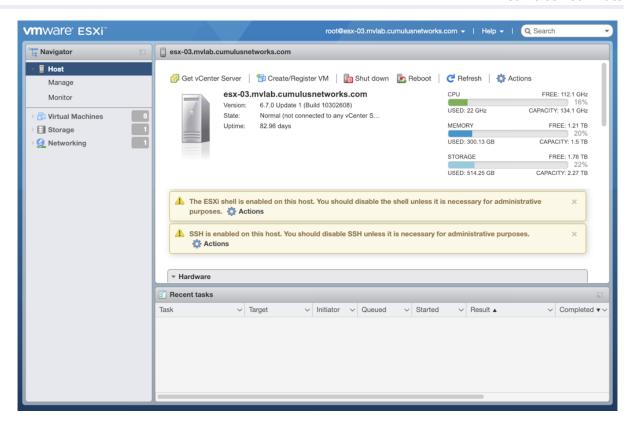
3. Open your hypervisor and set up your VM. You can use these examples for reference or use your own hypervisor instructions.

#### VMware example

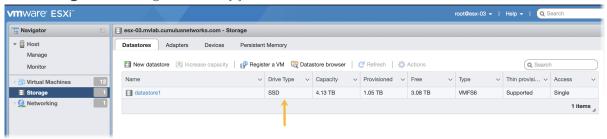
This example shows the VM setup process using an OVA file with VMware ESXi.

- 1. Enter the address of the hardware in your browser.
- 2. Log in to VMware using credentials with root access.





3. Click **Storage** in the Navigator to verify you have an SSD installed.

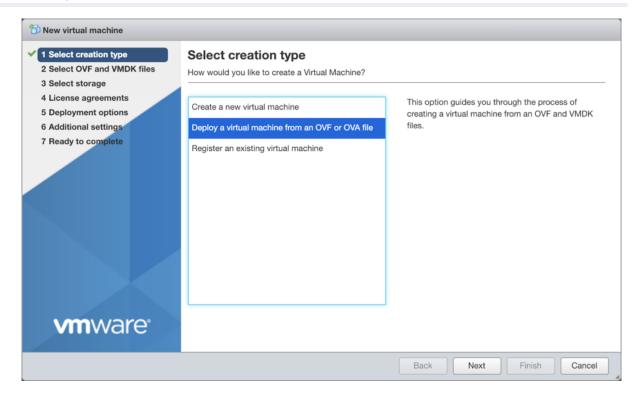


4. Click **Create/Register VM** at the top of the right pane.

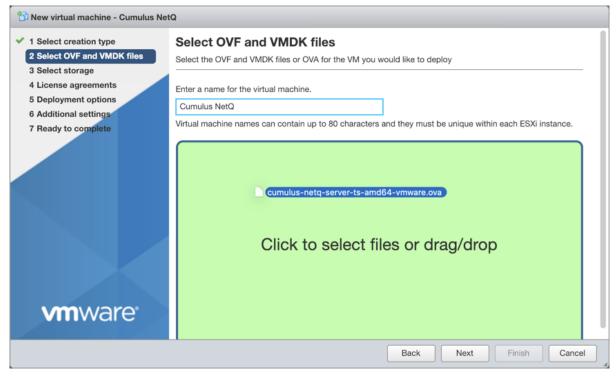


5. Select **Deploy a virtual machine from and OVF or OVA file**, and click **Next**.



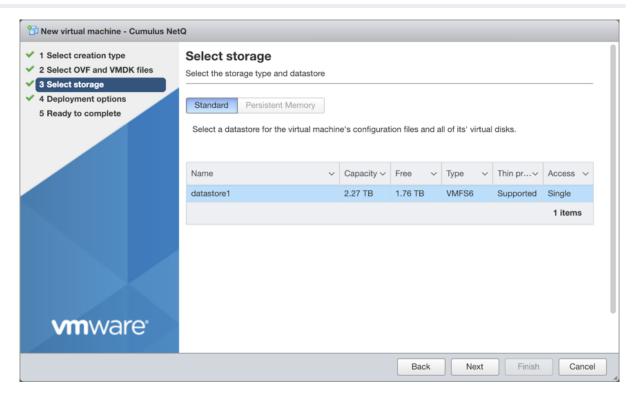


- 6. Provide a name for the VM, for example Cumulus NetQ.
- 7. Drag and drop the NetQ Platform image file you downloaded in Step 1 above.
- 8. Click Next.

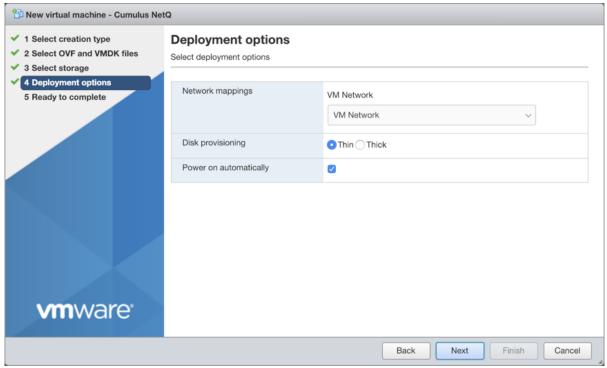


9. Select the storage type and data store for the image to use, then click **Next**. In this example, only one is available.



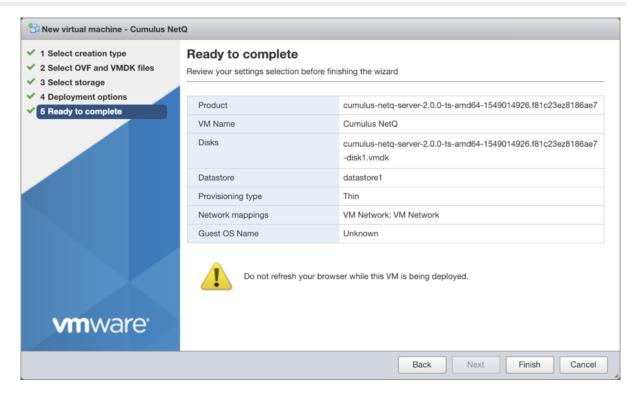


10. Accept the default deployment options or modify them according to your network needs. Click **Next** when you are finished.



11. Review the configuration summary. Click **Back** to change any of the settings, or click **Finish** to continue with the creation of the VM.

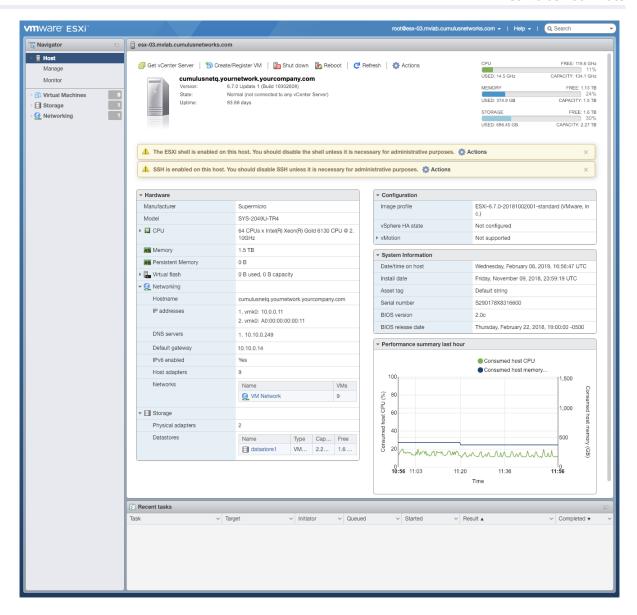




The progress of the request is shown in the Recent Tasks window at the bottom of the application. This may take some time, so continue with your other work until the upload finishes.

12. Once completed, view the full details of the VM and hardware.





#### KVM example

This example shows the VM setup process for a system with Libvirt and KVM/QEMU installed.

1. Confirm that the SHA256 checksum matches the one posted on the Cumulus Downloads website to ensure the image download has not been corrupted.

```
$ sha256sum ./Downloads/cumulus-netq-server-2.1.1-ts-amd64-qemu.
qcow2
$
6fff5f2ac62930799b4e8cc7811abb6840b247e2c9e76ea9ccba03f991f42424
./Downloads/cumulus-netq-server-2.1.1-ts-amd64-qemu.qcow2
```

2. Copy the Qcow2 image to a directory where you want to run it.

⊘



Copy, instead of moving, the original Qcow2 image that was downloaded to avoid redownloading it again later should you need to perform this process again.

```
$ sudo mkdir /vms
$ sudo cp ./Downloads/cumulus-netq-server-2.1.1-ts-amd64-qemu.
qcow2 /vms/ts.qcow2
```

3. Create the VM.

For a Direct VM, where the VM uses a MACVLAN interface to sit on the host interface for its connectivity:

```
$ virt-install --name=netq_ts --vcpus=8 --memory=65536 --os-
type=linux --os-variant=debian7 \
   --disk path=/vms/ts.qcow2,format=qcow2,bus=virtio,cache=none \
   --network=type=direct,source=eth0,model=virtio --import --
noautoconsole
```

Replace the disk path value with the location where the Qcow2 image is to reside. Replace network model value (eth0 in the above example) with the name of the interface where the VM is connected to the external network.

Or, for a Bridged VM, where the VM attaches to a bridge which has already been setup to allow for external access:

```
$ virt-install --name=netq_ts --vcpus=8 --memory=65536 --os-
type=linux --os-variant=debian7 \
   --disk path=/vms/ts.qcow2,format=qcow2,bus=virtio,cache=none \
   --network=bridge=br0,model=virtio --import --noautoconsole
```

- Replace network bridge value (br0 in the above example) with the name of the (pre-existing) bridge interface where the VM is connected to the external network.
- 4. Watch the boot process in another terminal window.

```
$ virsh console netq_ts
```

5. From the Console of the VM, check to see which IP address Eth0 has obtained via DHCP, or alternatively set a static IP address with NCLU on the NetQ Appliance or Platform VM.

```
$ ip addr show eth0
```



```
$ net add interface eth0 ip address 10.0.0.1
$ net commit
```



If you need to change the IP address, reboot the VM in order to let the new IP Address be registered properly with the containers running on the NetQ Appliance or Platform.

Please allow 5-10 minutes for the VM to boot.

#### **Verify the Installation**

- 1. Verify you can access the NetQ CLI.
  - a. From a terminal window, log in to the NetQ Platform using the default credentials (*cumulus /CumulusLinux!*).

```
<computer>:~<username>$ ssh cumulus@<netq-platform-ipaddress>
Warning: Permanently added '<netq-platform-hostname>,
192.168.1.254' (ECDSA) to the list of known hosts.
cumulus@<netq-platform-hostname>'s password: <enter
CumulusLinux! here>

Welcome to Cumulus (R) Linux (R)

For support and online technical documentation, visit
http://www.cumulusnetworks.com/support

The registered trademark Linux (R) is used pursuant to a
sublicense from LMI,
the exclusive licensee of Linus Torvalds, owner of the mark
on a world-wide
basis.

cumulus@<netq-platform-hostname>:~$
```

b. Run the following command to verify all applications are operating properly. **Note**: Please allow 10-15 minutes for all applications to come up and report their status.



netq-app-vlan	UP	
true	4daed85	Mon Jun 3 20:20:35
2019		
netq-app-sensors	UP	true
up f37272c		3 20:20:34 2019
netq-app-topology	UP	
true	3f4a887	Mon Jun 3 20:20:34
2019		
kafka-broker		
UP		Mon Jun 3
20:20:35 2019		
netq-app-mstpinfo	UP	true
up ef5565d	Mon Jun	3 20:20:35 2019
netq-app-address	UP	true
up 7e0d03d	Mon Jun	3 20:20:35 2019
netq-gui		
UP		Mon Jun 3
20:20:35 2019		
netq-app-kube	UP	true
up fbcaa9d	Mon Jun	3 20:20:34 2019
netq-app-link	UP	true
up 6c2b21a	Mon Jun	3 20:20:35 2019
netq-app-ptm	UP	true
up 7162771	Mon Jun	3 20:20:34 2019
netq-opta	UP	
true		Mon Jun 3 20:20:34
2019		
netq-app-clagsession	UP	true
up 356dda9	Mon Jun	3 20:20:34 2019
netq-endpoint-gateway	UP	
true	295e9ed	Mon Jun 3 20:20:34
2019		
netq-app-ospf	UP	true
up e0e2ab0	Mon Jun	3 20:20:34 2019
netq-app-lldp	UP	true
up 90582de		3 20:20:35 2019
netq-app-inventory	UP	true
_		3 20:20:34 2019
netq-app-tracecheck-sch		
true	5484c68	Mon Jun 3 20:20:34
2019		
netq-app-infra	UP	true
_	Mon Jun	3 20:20:34 2019
kafka-connect		
UP		Mon Jun 3
20:20:35 2019		h
netq-app-search	UP Mara Tura	true
up e47aaba		3 20:20:34 2019
netq-app-procdevstats	UP Man Tun	true
up b8e280e		3 20:20:34 2019
netq-app-vxlan	UP Mon Tun	true
up 123c577	Mon Jun	3 20:20:34 2019



zookeeper			
zookeeper UP			Mon Jun 3
20:20:35 2019			MOII daii 3
netq-app-resour	ae-util	UP	true
			3 20:20:34 2019
up	4101007	MOII UUII	
netq-app-evpn	05-4002	~ -	true
up 	05a4003		3 20:20:34 2019
netq-api-gatewa	Y	UP	Mars Tare 2 20.20.24
true		C40231a	Mon Jun 3 20:20:34
2019			
netq-app-port		UP	true
up	4592b70		3 20:20:35 2019
netq-app-macs		UP	
true		dd6cd96	Mon Jun 3 20:20:35
2019			
netq-app-notifi	er	UP	true
up	da57b69	Mon Jun	3 20:20:35 2019
netq-app-events		UP	true
up	8f7b4d9	Mon Jun	3 20:20:34 2019
netq-app-servic	es	UP	true
up	5094f4a	Mon Jun	3 20:20:34 2019
cassandra			
UP			Mon Jun 3
20:20:35 2019			
netq-app-config	diff	UP	true
up	3be2ef1	Mon Jun	3 20:20:34 2019
netq-app-neighb	or	UP	true
up	9ebe479	Mon Jun	3 20:20:35 2019
netq-app-bgp		UP	true
up	e68f7a8	Mon Jun	3 20:20:35 2019
schema-registry			
UP			Mon Jun 3
20:20:35 2019			non can
netq-app-lnv		UP	true
up	a9ca80a		3 20:20:34 2019
up netq-app-health			3 20.20.34 2019
necq-app-nearch true	aasiibuatu	eea044c	Mon Jun 3 20:20:34
2019		eea0440	11011 UUII 3 ZU•ZU•34
		IID	truo
netq-app-ntp	651 0065	UP Mon Tun	true
up	651c86f		3 20:20:35 2019
netq-app-custom	ermgmt	UP	Mars Trees 2 00:00:24
true		7250354	Mon Jun 3 20:20:34
2019			
netq-app-node	5686	UP	true
up	f676c9a		3 20:20:34 2019
netq-app-route		UP	true
	6e31f98	Mon Jun	3 20:20:35 2019
up cumulus@ <netq-p< td=""><td></td><td></td><td>3 20:20:35 2019</td></netq-p<>			3 20:20:35 2019





If any of the applications or services display Status as DOWN after 30 minutes, open a support ticket and attach the output of the opta-support command.

- 2. Verify that NTP is configured and running. NTP operation is critical to proper operation of NetQ. Refer to Setting Date and Time in the *Cumulus Linux User Guide* for details and instructions.
- 3. Continue the NetQ installation by loading the NetQ Agent on each switch or host you want to monitor. Refer to the next section for instructions.

## **Upgrade the NetQ Agent**

Whether using the NetQ Appliance or your own hardware, the NetQ Agent must be updated on each node you want to monitor. The node can be a:

- Switch running Cumulus Linux version 3.7.0 or later
- Server running Red Hat RHEL 7.1, Ubuntu 16.04 or CentOS 7
- Linux virtual machine running any of the above Linux operating systems

To upgrade the NetQ Agent you need to install the OS-specific meta package, cumulus-netq, on each switch. Optionally, you can install it on hosts. The meta package contains the NetQ Agent, the NetQ command line interface (CLI), and the NetQ library. The library contains modules used by both the NetQ Agent and the CLI.

Instructions for installing the meta package on each node type are included here:

- Upgrade NetQ Agent on a Cumulus Linux Switch (see page 68)
- Upgrade NetQ Agent on an Ubuntu Server (see page 70)
- Upgrade NetQ Agent on a Red Hat or CentOS Server (see page 72)



If your network uses a proxy server for external connections, you should first configure a global proxy so apt-get can access the meta package on the Cumulus Networks repository.

## **Upgrade NetQ Agent on a Cumulus Linux Switch**

A simple process installs the NetQ Agent on a Cumulus switch.

1. Stop the netg-agent service and netgd daemon running on your switch.

```
cumulus@switch:~$ sudo systemctl stop netq-agent
cumulus@switch:~$ sudo systemctl stop netqd
```



If you are running VRF, run these additional commands:

```
cumulus@switch:~$ sudo systemctl stop netq-agent@mgmt
cumulus@switch:~$ sudo systemctl stop netqd@mgmt
```

2. Remove the older NetQ packages.



```
cumulus@switch:~$ sudo apt -y purge cumulus-netq netq-agent netq-
apps python-netq-lib
```

3. Verify you have removed all older NetQ packages. You should not see any older version files after running the dpkg command here.

```
cumulus@switch:~# dpkg -1 | grep netq
```

4. Edit the /etc/apt/sources.list file to add the repository for Cumulus NetQ. **Note** that NetQ has a separate repository from Cumulus Linux.

```
cumulus@switch:~$ sudo nano /etc/apt/sources.list
...
deb http://apps3.cumulusnetworks.com/repos/deb CumulusLinux-3
netq-2.1
...
```



The repository deb http://apps3.cumulusnetworks.com/repos/deb CumulusLinux-3 netq-latest can be used if you want to always retrieve the latest posted version of NetQ.

5. Update the local apt repository, then install the NetQ meta package on the switch.

```
cumulus@switch:~$ sudo apt-get update
cumulus@switch:~$ sudo apt-get install cumulus-netq
```

6. Restart rsyslog so log files are sent to the correct destination.

```
cumulus@switch:~$ sudo systemctl restart rsyslog.service
```

7. Configure the NetQ Agent to send telemetry data to the NetQ Platform and, optionally, configure the switch or host to run the NetQ CLI. In this example, the IP address for the agent and cli servers is 192.168.1.254.

**Note:** If you intend to use VRF, skip to Configure the Agent to Use VRF (see page 74). If you intend to specify a port for communication, skip to Configure the Agent to Communicate over a Specific Port (see page 74).

```
cumulus@switch:~$ netq config add agent server 192.168.1.254
cumulus@switch:~$ netq config add cli server 192.168.1.254
```

This command updates the configuration in the /etc/netq/netq.yml file and enables the NetQ CLI.



8. Restart NetQ Agent and CLI.

```
cumulus@switch:~$ netq config restart agent
cumulus@switch:~$ netg config restart cli
```

Repeat these steps for each Cumulus switch, or use an automation tool to install NetQ Agent on multiple Cumulus Linux switches.

#### **Upgrade NetQ Agent on an Ubuntu Server (Optional)**

To install the NetQ Agent on an Ubuntu server:

1. Remove the current NetQ Agent and application software from your switch or host.

```
root@ubuntu:~# sudo systemctl stop netq-agent
root@ubuntu:~# sudo systemctl stop netqd
root@ubuntu:~# sudo apt-get purge --auto-remove cumulus-netq
netq-agent netq-apps python-netq-lib
```

2. Verify you have removed all older NetQ packages. You should not see any older version files.

```
root@ubuntu:~# dpkg -l | grep netg
```

3. Reference and update the local apt repository.

```
root@ubuntu:~# wget -O- https://apps3.cumulusnetworks.com/setup
/cumulus-apps-deb.pubkey | apt-key add -
```

4. In /etc/apt/sources.list.d/cumulus-host-ubuntu-xenial.list, verify the following repository is included:

```
root@ubuntu:~# vi /etc/apt/sources.list.d/cumulus-apps-deb-
xenial.list
deb [arch=amd64] https://apps3.cumulusnetworks.com/repos/deb
xenial netq-latest
```



riangle The use of  $\mathtt{netq-latest}$  in this example means that a  $\mathtt{get}$  to the repository always. retrieves the latest version of NetQ, even in the case where a major version update has been made. If you want to keep the repository on a specific version — such as netq-2.1 — use that instead.

70 06 June 2019



5. Verify NTP is operating correctly. Look for an asterisk (\*) or a plus sign (+) that indicates the clock is synchronized.

```
root@ubuntu:~# ntpg -pn
   remote
               refid
                        st t when poll reach
offset jitter
______
==========
+173.255.206.154 132.163.96.3 2 u 86 128 377
                                         41.354
2.834 0.602
+12.167.151.2 198.148.79.209 3 u 103 128 377 13.395
-4.025 0.198
2a00:7600::41 .STEP.
                        16 u - 1024
                                         0.000
                                     0
0.000 0.000
*129.250.35.250 249.224.99.213 2 u 101 128 377
                                         14.588
-0.299 0.243
```

6. Install the meta package on the server.

```
root@ubuntu:~# apt-get update
root@ubuntu:~# apt-get install cumulus-netq
```

7. Restart the NetQ daemon.

```
root@ubuntu:~# systemctl enable netqd
root@ubuntu:~# systemctl restart netqd
```

8. Configure the NetQ Agent to send telemetry data to the NetQ Platform.

```
user@ubuntu:~# netq config add agent server <netq-platform-ip-address>
Updated agent server 192.168.1.254 vrf default. Please restart netq-agent (netq config restart agent).
```

9. Restart the NetQ Agent

```
user@ubuntu:~# netq config restart agent
```

10. Optionally, configure the Ubuntu server to run the NetQ CLI.

```
user@ubuntu:~# netq config add cli server <netq-platform-ip-address>
Updated cli server 192.168.1.254 vrf default. Please restart netqd (netq config restart cli).
```



11. Restart the CLI.

```
user@ubuntu:~# netq config restart cli
```

#### **Upgrade NetQ Agent on a Red Hat or CentOS Server (Optional)**

To install the NetQ Agent on a Red Hat or CentOS server:

1. Remove the current NetQ Agent and application software from your switch or host.

```
root@rhel7:~# sudo systemctl stop netq-agent
root@rhel7:~# sudo systemctl stop netqd
root@rhel7:~# yum remove netq-apps netq-agent cumulus-netq
```

2. Verify you have removed all older NetQ packages. You should not see any older version files.

```
root@rhel7:~# yum list | grep netq
```

3. Reference and update the local yum repository.

```
root@rhel7:~# rpm --import https://apps3.cumulusnetworks.com
/setup/cumulus-apps-rpm.pubkey
root@rhel7:~# wget -O- https://apps3.cumulusnetworks.com/setup
/cumulus-apps-rpm-el7.repo > /etc/yum.repos.d/cumulus-host-el.
repo
```

4. Edit /etc/yum.repos.d/cumulus-host-el.repo to set the enabled=1 flag for the two NetQ repositories.

```
root@rhel7:~# vi /etc/yum.repos.d/cumulus-host-el.repo
...
[cumulus-arch-netq-2.1]
name=Cumulus netq packages
baseurl=https://apps3.cumulusnetworks.com/repos/rpm/el/7/netq-2.1
/$basearch
gpgcheck=1
enabled=1
[cumulus-noarch-netq-2.1]
name=Cumulus netq architecture-independent packages
baseurl=https://apps3.cumulusnetworks.com/repos/rpm/el/7/netq-2.1
/noarch
gpgcheck=1
enabled=1
```



. . .

5. Verify NTP is operating correctly. Look for an asterisk (\*) or a plus sign (+) that indicates the clock is synchronized.

```
root@rhel7:~# ntpq -pn
                                          delay
   remote
               refid st t when poll reach
offset jitter
______
+173.255.206.154 132.163.96.3 2 u 86 128 377
                                         41.354
2.834 0.602
+12.167.151.2 198.148.79.209 3 u 103 128 377
                                         13.395
-4.025 0.198
2a00:7600::41 .STEP.
                        16 u - 1024 0
                                         0.000
0.000 0.000
*129.250.35.250 249.224.99.213 2 u 101 128 377
                                         14.588
-0.299 0.243
```

6. Update the NetQ meta packages on the server.

```
root@rhel7:~# yum update cumulus-netq.x86_64
```

7. Configure the NetQ Agent to send telemetry data to the NetQ Platform.

```
root@rhel7:~# netq config add agent server <netq-platform-ip-
address>
Updated agent server 192.168.1.254 vrf default. Please restart
netq-agent (netq config restart agent).
```

8. Restart the NetQ Agent

```
root@rhel7:~# netq config restart agent
```

9. Optionally, configure the RHEL/CentOS server to run the NetQ CLI.

```
root@rhel7:~# netq config add cli server <netq-platform-ip-address>
Updated cli server 192.168.1.254 vrf default. Please restart netqd (netq config restart cli).
```

10. Restart the CLI.

```
root@rhel7:~# netq config restart cli
```



## **Configure Optional NetQ Agent Settings**

Once the NetQ Agents have been installed on the network nodes you want to monitor, the NetQ Agents must be configured to obtain useful and relevant data. The code examples shown in this section illustrate how to configure the NetQ Agent on a Cumulus switch, but it is exactly the same for the other type of nodes. Depending on your deployment, follow the relevant additional instructions after the basic configuration steps:

- Configuring the Agent to Use a VRF (see page 74)
- Configuring the Agent to Communicate over a Specific Port (see page 74)

#### Configure the Agent to Use a VRF

While optional, Cumulus strongly recommends that you configure NetQ Agents to communicate with the NetQ Platform only via a VRF, including a management VRF. To do so, you need to specify the VRF name when configuring the NetQ Agent. For example, if the management VRF is configured and you want the agent to communicate with the NetQ Platform over it, configure the agent like this:

```
cumulus@leaf01:~$ netq config add agent server 192.168.1.254 vrf mgmt cumulus@leaf01:~$ netq config add cli server 192.168.254 vrf mgmt
```

You then restart the agent:

```
cumulus@leaf01:~$ netq config restart agent cumulus@leaf01:~$ netq config restart cli
```

## Configure the Agent to Communicate over a Specific Port

By default, NetQ uses port 8981 for communication between the NetQ Platform and NetQ Agents. If you want the NetQ Agent to communicate with the NetQ Platform via a different port, you need to specify the port number when configuring the NetQ Agent like this:

```
cumulus@switch:~$ netq config add agent server 192.168.1.254 port 7379
```

You then restart the agent:

```
cumulus@leaf01:~$ netq config restart agent
```



## **Integrate with Event Notification Tools**

If you want to proactively monitor events in your network, you can integrate NetQ with the PagerDuty or Slack notification tools. To do so you need to configure both the notification application itself to receive the messages, and NetQ with what messages to send and where to send them. Refer to Integrate NetQ with Event Notification Applications to use the CLI for configuration.

## **Set Up Security**

When you set up and configured your Cumulus Linux switches, you likely configured a number of the security features available. Cumulus recommends the same security measures be followed for the NetQ Platform in the out-of-band-network. Refer to the Securing Cumulus Linux white paper for details.

Your Cumulus Linux switches have a number of ports open by default. A few additional ports must be opened to run the NetQ software (refer to Default Open Ports in Cumulus Linux and NetQ article).



# Upgrade from NetQ 2.0.x to NetQ 2.1.0

This document describes the steps required to upgrade from NetQ 2.0.x to NetQ 2.1.0.



Cumulus Networks recommends only upgrading NetQ during a network maintenance window. Any data you have collected while using NetQ 2.0.x is maintained during this upgrade process.



Events generated during the upgrade process will not be available in the database. Once the upgrade process is complete, the agents re-sync with the current state of the Host or Cumulus Linux switch with the NetQ Platform.

To upgrade from NetQ 1.x to NetQ 2.1.0, please follow the instructions in here. Instructions for installing NetQ 2.1.0 for the first time can be found in here.

#### **Contents**

This topic describes how to...

- Prerequisites (see page 76)
- Upgrade the NetQ Platform (see page 77)
- Upgrade the NetQ Agents (see page 80)
  - Upgrade NetQ Agent on a Cumulus Linux Switch (see page 81)
  - Upgrade NetQ Agent on an Ubuntu Server (Optional) (see page 81)
  - Upgrade NetQ Agent on a Red Hat or CentOS Server (Optional) (see page 83)
- Set Up the NetQ Agents (see page 86)
  - Basic Configuration (see page 86)
  - Configure the Agent to Use a VRF (Optional) (see page 87)
  - Configure the Agent to Communicate over a Specific Port (Optional) (see page 87)
- Upgrade Tips (see page 87)
  - Scenario 1: NetQ Appliance IP Address Was Changed (see page 88)
  - Scenario 2: No IP Address Assigned to the NetQ Appliance on Boot (see page 88).

## **Prerequisites**

Before you begin the upgrade process, please note the following:

- The minimum supported Cumulus Linux version for NetQ 2.0.1 is 3.3.0.
- You must upgrade your NetQ Agents as well as the NetQ Platform.
- You can upgrade to NetQ 2.1.0 without upgrading Cumulus Linux.

76 06 June 2019



• The NetQ installer pod netq-installer should be up in either the *Containercreating* or *Running* state. The netq-installer pod state could also be *ContainerCreating*, in which case the host's kubelet is looking to initialize SSH keys.

## **Upgrade the NetQ Platform**

To upgrade the NetQ Platform:

- 1. Obtain the NetQ-2.1.0.tgz and NetQ-2.1.0-installer.tgz files from your sales engineer.
- 2. From a terminal window, log in to the NetQ Platform (*netq-appliance*) using your login credentials. This example uses the default *cumulus/CumulusLinux!* credentials.

```
<computer>:~<username>$ ssh cumulus@netq-appliance
cumulus@netq-appliance's password:
cumulus@netq-appliance:~$
```

3. Change to the root user.

```
cumulus@netq-appliance:~$ sudo -i
[sudo] password for cumulus:
root@netq-appliance:~#
```

4. Create an *installables* subdirectory in the mount directory.

```
root@netq-appliance:~# mkdir -p /mnt/installables/
root@netq-appliance:~#
```

5. Copy the NetQ-2.1.0.tgz and NetQ-2.1.0-installer.tgz files into your new directory.

```
root@netq-appliance:~# cd /mnt/installables/
root@netq-appliance:/mnt/installables# cp /home/usr/dir/{NetQ-
2.1.0.tgz,NetQ-2.1.0-installer.tgz} ./
```

6. Export the installer script.

```
\label{lem:notenergy} root@netq-appliance:/mnt/installables\#tar -xvf \ NetQ-2.1.0.tgz \ . \\ /netq-install.sh
```

7. Verify the contents of the directory. You should have the two TGZ files and the netq-install.sh script.

```
root@netq-appliance:/mnt/installables# ls -l total 9607744
```



```
-rw-r--r-- 1 cumulus cumulus 253283595 Apr 23 20:36 NetQ-2.1.0-installer.tgz
-rw-r--r-- 1 cumulus cumulus 5911383922 Apr 23 11:13 NetQ-2.1.0.
tgz
-rwxr-xr-x 1 _lldpd _lldpd 4309 Apr 23 10:34 netq-install.sh
root@netq-appliance:/mnt/installables#
```

- 8. Configure SSH access.
  - (i) If you perform the upgrade more than once, skip to step 8c.
    - a. Generate the SSH key to enable you to run the script.
      - ① Leave the passphrase blank to simplify running the script.

```
root@netq-appliance:/mnt/installables# ssh-keygen -t rsa -b 4096

Generating public/private rsa key pair.

Enter file in which to save the key (/root/.ssh/id_rsa):

Created directory '/root/.ssh'.

Enter passphrase (empty for no passphrase):

Enter same passphrase again:

Your identification has been saved in /root/.ssh/id_rsa.

Your public key has been saved in /root/.ssh/id_rsa.pub.
```

b. Copy the key to the authorized\_keys directory.

```
root@netq-appliance:/mnt/installables# cat ~/.ssh/id_rsa.pub
>> ~/.ssh/authorized_keys
root@netq-appliance:/mnt/installables# chmod 0600 ~/.ssh
/authorized_keys
root@netq-appliance:/mnt/installables#
```

c. Associate the key with the installer.

```
root@netq-appliance:~#/mnt/installables/./netq-install.sh --
usekey ~/.ssh/id_rsa
[Fri 22 Mar 2019 06:34:47 AM UTC] - This Script can only be
invoked by user: root
[Fri 22 Mar 2019 06:34:47 AM UTC] - The logged in user is
root
[Fri 22 Mar 2019 06:34:47 AM UTC] - Install directory /mnt
/installables exists on system.
```



```
[Fri 22 Mar 2019 06:34:47 AM UTC] - File /root/.ssh/id_rsa exists on system...

[Fri 22 Mar 2019 06:34:47 AM UTC] - checking the presence of existing instaler-ssh-keys secret/instaler-ssh-keys created

[Fri 22 Mar 2019 06:34:48 AM UTC] - Unable to find netq-installer up and running. Sleeping for 10 seconds ...

[Fri 22 Mar 2019 06:34:58 AM UTC] - Unable to find netq-installer up and running. Sleeping for 10 seconds ...

[Fri 22 Mar 2019 06:35:08 AM UTC] - Able to find the netq-installer up and running...
```

9. Upgrade the netg-installer pod to the 2.1.0 version.

```
root@netq-appliance:/mnt/installables# ./netq-install.sh --
installbundle /mnt/installables/NetQ-2.1.0-installer.tgz
[Tue Apr 23 02:55:52 2019] - File /mnt/installables/NetQ-2.1.0-
installer.tgz exists on system for updating netq installer ...
[Tue Apr 23 02:55:52 2019] - Updating the netq installer ...
[Tue Apr 23 02:55:52 2019] - Able to execute the command for
updating netq installer ...
[Tue Apr 23 02:55:52 2019] - Checking initialization of
installer update ...
[Tue Apr 23 02:55:57 2019] - Update of netq installer is in
progress ...
******[Tue Apr 23 02:58:57 2019] - Finished updating the netq-
installer
[Tue Apr 23 02:58:57 2019] - Finished updating netq installer
!
0,/mnt/installables/NetQ-2.1.0-installer.tgz
cumulus@sm-telem-02:/mnt/installables$
```

- Please allow 3-4 minutes for the installer upgrade to complete.
- 10. Confirm the netq-installer has been upgraded to the 2.1.0 version.

```
root@netq-appliance:~# kubectl get pod -l app=netq-installer -
o=jsonpath={.items[0].status.containerStatuses[0].image}
netq-installer:2.1.0
```

11. Upgrade the NetQ software.

```
root@netq-appliance:/mnt/installables/# ./netq-install.sh --
updateapps /mnt/installables/NetQ-2.1.0.tgz
[Tue Apr 23 02:55:52 2019] - File /mnt/installables/NetQ-2.1.0-
installer.tgz exists on system for updating netq installer ...
```



```
[Tue Apr 23 02:55:52 2019] - Updating the netq installer ...
[Tue Apr 23 02:55:52 2019] - Able to execute the command for
updating netq installer ...
[Tue Apr 23 02:55:52 2019] - Checking initialization of
installer update ...
[Tue Apr 23 02:55:57 2019] - Update of netq installer is in
******[Tue Apr 23 02:58:57 2019] - Finished updating the netq-
installer
[Tue Apr 23 02:58:57 2019] - Finishshed updating netg installer
0,/mnt/installables/NetQ-2.1.0-installer.tgz
cumulus@sm-telem-02:/mnt/installables$ sudo /mnt/installables/.
/netq-install.sh --updateapps /mnt/installables/NetQ-2.1.0.tgz
[Tue Apr 23 03:01:13 2019] - File /mnt/installables/NetQ-2.1.0.
tgz exists on system for updating netq apps...
[Tue Apr 23 03:01:13 2019] - User selected to update netq-apps
. . .
[Tue Apr 23 03:04:13 2019] - Able to execute the command for
netq apps updates ...
[Tue Apr 23 03:04:13 2019] - Checking initialization of apps
[Tue Apr 23 03:04:18 2019] - netq apps update is in progress ...
*******************
*[Tue Apr 23 03:37:18 2019] - Finished updating the netq apps
[Tue Apr 23 03:37:18 2019] - Finished updating apps ....
```

①

Please allow about an hour for the upgrade to complete.

You are now running NetQ 2.1.0.

## **Upgrade the NetQ Agents**

Whether using the NetQ Appliance or your own hardware, the NetQ Agent must be upgraded on each node you want to monitor. The node can be a:

- Switch running Cumulus Linux version 3.7.0 or later
- Server running Red Hat RHEL 7.1, Ubuntu 16.04 or CentOS 7
- Linux virtual machine running any of the above Linux operating systems

To upgrade the NetQ Agent you need to install the OS-specific meta package, cumulus-netq, on each switch. Optionally, you can install it on hosts. The meta package contains the NetQ Agent, the NetQ command line interface (CLI), and the NetQ library. The library contains modules used by both the NetQ Agent and the CLI.

Instructions for installing the meta package on each node type are included here:

- Upgrade NetQ Agent on a Cumulus Linux Switch (see page 81)
- Upgrade NetQ Agent on an Ubuntu Server (see page 81)



Upgrade NetQ Agent on a Red Hat or CentOS Server (see page 83)



If your network uses a proxy server for external connections, you should first configure a global proxy so apt-get can access the meta package on the Cumulus Networks repository .

#### **Upgrade NetQ Agent on a Cumulus Linux Switch**

A simple process upgrades the NetQ Agent on a Cumulus switch.

1. Edit the /etc/apt/sources.list file to add the repository for Cumulus NetQ. **Note** that NetQ has a separate repository from Cumulus Linux.

```
cumulus@switch:~$ sudo nano /etc/apt/sources.list
...
deb http://apps3.cumulusnetworks.com/repos/deb CumulusLinux-3
netq-2.1
...
```



The repository deb http://apps3.cumulusnetworks.com/repos/deb CumulusLinux-3 netq-latest can be used if you want to always retrieve the latest posted version of NetQ.

2. Update the local apt repository, then install the NetQ meta package on the switch.

```
cumulus@switch:~$ sudo apt-get update
cumulus@switch:~$ sudo apt-get install cumulus-netq
```

Repeat these steps for each Cumulus switch, or use an automation tool to install NetQ Agent on multiple Cumulus Linux switches.

## Upgrade NetQ Agent on an Ubuntu Server (Optional)

Before you upgrade the NetQ Agent on an Ubuntu server, make sure the following packages are installed and running these minimum versions:

- iproute 1:4.3.0-1ubuntu3.16.04.1 all
- iproute2 4.3.0-1ubuntu3 amd64
- Ildpd 0.7.19-1 amd64
- ntp 1:4.2.8p4+dfsg-3ubuntu5.6 amd64



Make sure you are running lldpd, not lldpad. Ubuntu does not include 11dpd by default, which is required for the installation. To install this package, run the following commands:



```
root@ubuntu:~# apt-get update
root@ubuntu:~# apt-get install lldpd
root@ubuntu:~# systemctl enable lldpd.service
root@ubuntu:~# systemctl start lldpd.service
```

To upgrade the NetQ Agent on an Ubuntu server:

1. Remove the current NetQ Agent and application software from your switch or host.

```
root@ubuntu:~# sudo systemctl stop netq-agent
root@ubuntu:~# sudo systemctl stop netqd
root@ubuntu:~# yum remove netq-agent netq-apps
```

2. Verify you have removed all older NetQ packages. You should not see any older version files.

```
root@ubuntu:~# dpkg -l | grep netq
```

3. Reference and update the local apt repository.

```
root@ubuntu:~# wget -O- https://apps3.cumulusnetworks.com/setup
/cumulus-apps-deb.pubkey | apt-key add -
```

4. Create the file /etc/apt/sources.list.d/cumulus-host-ubuntu-xenial.list and add the following lines:

```
root@ubuntu:~# vi /etc/apt/sources.list.d/cumulus-apps-deb-
xenial.list
deb [arch=amd64] https://apps3.cumulusnetworks.com/repos/deb
xenial netg-latest
```



riangle The use of netq-latest in this example means that a get to the repository always retrieves the lastest version of NetQ, even in the case where a major version update has been made. If you want to keep the repository on a specific version — such as netq-2.1 — use that instead.

5. Install NTP on the server.

```
root@ubuntu:~# sudo apt-get install ntp
```

82 06 June 2019



- 6. Configure the NTP server.
  - a. Open the /etc/ntp.conf file in your text editor of choice.
  - b. Under the Server section, specify the NTP server IP address or hostname.
- 7. Enable and start the NTP service.

```
root@ubuntu:~# sudo systemctl enable ntp.service root@ubuntu:~# sudo systemctl start ntp.service
```

8. Verify NTP is operating correctly. Look for an asterisk (\*) or a plus sign (+) that indicates the clock is synchronized.

```
root@ubuntu:~# ntpq -pn
   remote
                refid
                         st t when poll reach
                                            delay
offset jitter
______
=========
+173.255.206.154 132.163.96.3 2 u 86 128
                                      377
                                           41.354
2.834 0.602
+12.167.151.2
             198.148.79.209 3 u 103 128
                                      377
                                           13.395
-4.025 0.198
2a00:7600::41
             .STEP.
                         16 u
                                - 1024
                                            0.000
0.000 0.000
*129.250.35.250 249.224.99.213 2 u 101 128 377
                                           14.588
-0.299 0.243
```

9. Install the meta package on the server.

```
root@ubuntu:~# apt-get update
root@ubuntu:~# apt-get install cumulus-netq
```

10. Restart the NetQ daemon.

```
root@ubuntu:~# systemctl enable netqd
root@ubuntu:~# systemctl restart netqd
```

## Upgrade NetQ Agent on a Red Hat or CentOS Server (Optional)

Before you upgrade the NetQ Agent on a Red Hat or CentOS server, make sure the following packages are installed and running these minimum versions:

- iproute-3.10.0-54.el7\_2.1.x86\_64
- Ildpd-0.9.7-5.el7.x86 64

Make sure you are running lldp**d**, not lldp**ad**.



CentOS does not include 11dpd by default, nor does it include wget, which is required for the installation. To install this package, run the following commands:

```
root@centos:~# yum -y install epel-release
root@centos:~# yum -y install lldpd
root@centos:~# systemctl enable lldpd.service
root@centos:~# systemctl start lldpd.service
root@centos:~# yum install wget
```

- ntp-4.2.6p5-25.el7.centos.2.x86\_64
- ntpdate-4.2.6p5-25.el7.centos.2.x86\_64

To upgrade the NetQ Agent on a Red Hat or CentOS server:

1. Remove the current NetQ Agent and application software from your switch or host.

```
root@rhel7:~# sudo systemctl stop netq-agent
root@rhel7:~# sudo systemctl stop netqd
root@rhel7:~# apt-get purge --auto-remove cumulus-netq netq-
agent netq-apps
```

2. Verify you have removed all older NetQ packages. You should not see any older version files.

```
root@rhel7:~# dpkg -l | grep netq
```

3. Reference and update the local yum repository.

```
root@rhel7:~# rpm --import https://apps3.cumulusnetworks.com
/setup/cumulus-apps-rpm.pubkey
root@rhel7:~# wget -O- https://apps3.cumulusnetworks.com/setup
/cumulus-apps-rpm-el7.repo > /etc/yum.repos.d/cumulus-host-el.
repo
```

4. Edit /etc/yum.repos.d/cumulus-host-el.repo to set the enabled=1 flag for the two NetQ repositories.

```
root@rhel7:~# vi /etc/yum.repos.d/cumulus-host-el.repo
...
[cumulus-arch-netq-2.1]
name=Cumulus netq packages
baseurl=https://apps3.cumulusnetworks.com/repos/rpm/el/7/netq-2.1
/$basearch
gpgcheck=1
enabled=1
```



```
[cumulus-noarch-netq-2.1]
name=Cumulus netq architecture-independent packages
baseurl=https://apps3.cumulusnetworks.com/repos/rpm/el/7/netq-2.1
/noarch
gpgcheck=1
enabled=1
...
```

5. Install NTP on the server.

```
root@rhel7:~# yum install ntp
```

- 6. Configure the NTP server.
  - a. Open the /etc/ntp.conf file in your text editor of choice.
  - b. Under the Server section, specify the NTP server IP address or hostname.
- 7. Enable and start the NTP service.

```
root@rhel7:~# sudo systemctl enable ntpd.service root@rhel7:~# sudo systemctl start ntpd.service
```

8. Verify NTP is operating correctly. Look for an asterisk (\*) or a plus sign (+) that indicates the clock is synchronized.

```
root@rhel7:~# ntpq -pn
   remote
              refid
                       st t when poll reach
                                       delay
offset jitter
______
=========
+173.255.206.154 132.163.96.3 2 u 86 128 377
                                       41.354
2.834 0.602
+12.167.151.2 198.148.79.209 3 u 103 128 377
                                       13.395
-4.025 0.198
2a00:7600::41 .STEP. 16 u - 1024 0
                                       0.000
0.000 0.000
*129.250.35.250 249.224.99.213 2 u 101 128 377
                                       14.588
-0.299 0.243
```

9. Install the Bash completion and NetQ meta packages on the server.

```
root@rhel7:~# yum -y install bash-completion root@rhel7:~# yum install cumulus-netq
```

10. Restart the NetQ daemon.



root@rhel7:~# systemctl enable netqd ; systemctl restart netqd

## **Set Up the NetQ Agents**

Once the NetQ Agents have been installed on the network nodes you want to monitor, the NetQ Agents must be configured to obtain useful and relevant data. The code examples shown in this section illustrate how to configure the NetQ Agent on a Cumulus switch, but it is exactly the same for the other type of nodes. Depending on your deployment, follow the relevant additional instructions after the basic configuration steps:

- Basic Configuration
- Configuring the Agent to Use a VRF (see page 87)
- Configuring the Agent to Communicate over a Specific Port (see page 87)

#### **Basic Configuration**

This is the minimum configuration required to properly monitor your nodes.

1. Verify that NTP is running on the host node. Nodes must be in time synchronization with the NetQ Platform to enable useful statistical analysis.

2. Restart rsyslog so log files are sent to the correct destination.

```
cumulus@switch:~$ sudo systemctl restart rsyslog.service
```

3. Link the node to the NetQ Platform you configured above. You must configure both the agent server and the cli server to link to the NetQ Platform. **Note:** If you intend to use VRF, skip to Configure the Agent to Use VRF (see page 87). If you intend to specify a port for communication, skip to Configure the Agent to Communicate over a Specific Port (see page 87). In this example, the IP address for the agent and cli servers is 192.168.1.254.

```
cumulus@switch:~$ netq config add agent server 192.168.1.254 cumulus@switch:~$ netq config add cli server 192.168.1.254
```



This command updates the configuration in the /etc/netq/netq.yml file and enables the NetQ CLI.

4. Restart NetQ Agent and CLI.

```
cumulus@switch:~$ netq config restart agent cumulus@switch:~$ netq config restart cli
```

#### Configure the Agent to Use a VRF (Optional)

While optional, Cumulus strongly recommends that you configure NetQ Agents to communicate with the NetQ Platform only via a VRF, including a management VRF. To do so, you need to specify the VRF name when configuring the NetQ Agent. For example, if the management VRF is configured and you want the agent to communicate with the NetQ Platform over it, configure the agent like this:

```
cumulus@leaf01:~$ netq config add agent server 192.168.1.254 vrf mgmt cumulus@leaf01:~$ netq config add cli server 192.168.254 vrf mgmt
```

You then restart the agent:

```
cumulus@leaf01:~$ netq config restart agent
cumulus@leaf01:~$ netq config restart cli
```

## Configure the Agent to Communicate over a Specific Port (Optional)

By default, NetQ uses port 8981 for communication between the NetQ Platform and NetQ Agents. If you want the NetQ Agent to communicate with the NetQ Platform via a different port, you need to specify the port number when configuring the NetQ Agent like this:

```
cumulus@switch:~$ netq config add agent server 192.168.1.254 port 7379
```

You then restart the agent:

```
cumulus@leaf01:~$ netq config restart agent
```

## **Upgrade Tips**

After your installation is complete, the following scenarios may be encountered.



#### Scenario 1: NetQ Appliance IP Address Was Changed

The NetQ Appliance came with a pre-configured IP address, which has since been changed. How do I recover the system?

You must remove and reconfigure the Kubernetes configuration. Follow these steps:

- Locate the Kubernetes pods.
   cumulus@switch:~\$ kubectl get pods
- 2. Reset all Kubernetes administrative settings. Run the command twice to make sure all directories and files have been reset.

```
cumulus@switch:~$ sudo kudeadm reset -f cumulus@switch:~$ sudo kudeadm
reset -f
```

- 3. Remove the Kubernetes configuration. cumulus@switch:~\$ sudo rm /home/cumulus/.kube/config
- 4. Reset the NetQ Appliance install daemon. cumulus@switch:~\$ sudo systemctl reset-failed
- 5. Reset the Kubernetes service.

  cumulus@switch:~\$ sudo systemctl restart cts-kubectl-config

  Note: Allow 15 minutes for the prompt to return.

#### Scenario 2: No IP Address Assigned to the NetQ Appliance on Boot

A user did not configure an IP address when the system was first booted. Later the user assigned an IP address to eth0, but the NetQ appliance does not appear to be functioning.

You must reset the install daemon and restart the Kubernetes service. Follow these steps:

- 1. Reset the NetQ Appliance install daemon. cumulus@switch:~\$ sudo systemctl reset-failed
- 2. Restart the Kubernetes service. cumulus@switch:~\$ sudo systemctl restart cts-kubectl-config



## Upgrade from NetQ 2.1.0 to NetQ 2.1.x

This document describes the steps required to upgrade from NetQ 2.1.0 to NetQ 2.1.1 or later.



Cumulus Networks recommends only upgrading NetQ during a network maintenance window. Any data you have collected while using NetQ 2.1.0 is maintained during this upgrade process.



Events generated during the upgrade process will not be available in the database. Once the upgrade process is complete, the agents re-sync with the current state of the Host or Cumulus Linux switch with the NetO Platform.

To upgrade from NetQ 1.x to NetQ 2.1.x, please follow the instructions here. Instructions for installing NetQ 2.1.x for the first time can be found here.

#### **Contents**

This topic describes how to...

- Prerequisites (see page 89)
- Upgrade the NetQ Platform or NetQ Appliance (see page 90)
- Upgrade the NetQ Agents (see page 94)
  - Upgrade NetQ Agent on a Cumulus Linux Switch (see page 94)
  - Upgrade NetQ Agent on an Ubuntu Server (Optional) (see page 95)
  - Upgrade NetQ Agent on a Red Hat or CentOS Server (Optional) (see page 96)
- Configure Optional NetQ Agent Settings (see page 98)
  - Configure the Agent to Use a VRF Interface (see page 98)
  - Configure the Agent to Communicate over a Specific Port (see page 98)
- Upgrade Tips (see page 98)
  - Scenario 1: NetQ Appliance IP Address Was Changed (see page 99)
  - Scenario 2: No IP Address Assigned to the NetQ Appliance on Boot (see page 99).

## **Prerequisites**

Before you begin the upgrade process, please note the following:

- The minimum supported Cumulus Linux version for NetQ 2.1.x is 3.3.0.
- You must upgrade your NetQ Agents as well as the NetQ Platform.
- You can upgrade to NetQ 2.1.x without upgrading Cumulus Linux.

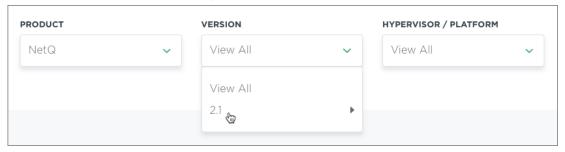


• The NetQ installer pod netq-installer should be up in either the *Containercreating* or *Running* state. The netq-installer pod state could also be *ContainerCreating*, in which case the host is initializing with the SSH keys.

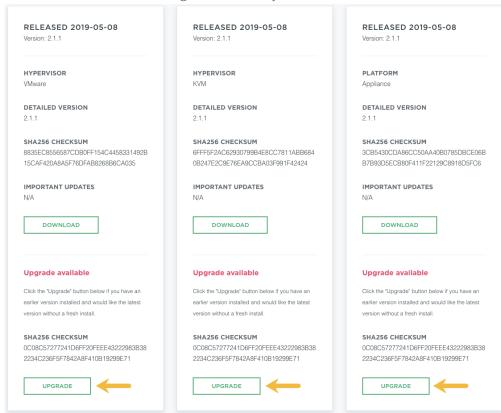
## **Upgrade the NetQ Platform or NetQ Appliance**

To upgrade the NetQ Platform:

- 1. Download the NetQ Platform or Appliance upgrade image (NetQ-2.1.x.tgz).
  - a. On the Cumulus Downloads page, select NetQ from the Product list box.
  - b. Click 2.1 from the **Version** list box, and then select 2.1.x from the submenu.



c. Scroll down to review the images that match your selection criteria.



d. Click **Upgrade** for the relevant version (VMware or KVM for NetQ Platform, Appliance for NetQ Appliance).



2. From a terminal window, log in to the NetQ Platform or Appliance using your login credentials. This example uses the default *cumulus/CumulusLinux!* credentials and is run on the NetQ Appliance.

```
<computer>:~<username>$ ssh cumulus@netq-appliance
cumulus@netq-appliance's password:
cumulus@netq-appliance:~$
```

3. Change to the root user.

```
cumulus@netq-appliance:~$ sudo -i
[sudo] password for cumulus:
root@netq-appliance:~#
```

4. Create an *installables* subdirectory in the mount directory.

```
root@netq-appliance:~# mkdir -p /mnt/installables/
root@netq-appliance:~#
```

5. Copy the NetQ-2.1.x.tgz file into your new directory.

```
root@netq-appliance:~# cd /mnt/installables/
root@netq-appliance:/mnt/installables# cp /home/usr/dir/NetQ-2.1.
x.tgz ./
```

6. Export the installer script.

```
root@netq-appliance:/mnt/installables# tar -xvf NetQ-2.1.x.tgz .
/netq-install.sh
```

7. Verify the contents of the directory. You should have the NetQ-2.1.x.tgz file and the netq-install. sh script.

```
root@netq-appliance:/mnt/installables# ls -l
total 9607744
-rw-r--r- 1 cumulus cumulus 5911383922 Apr 23 11:13 NetQ-2.1.x.
tgz
-rwxr-xr-x 1 _lldpd _lldpd 4309 Apr 23 10:34 netq-install.sh
root@netq-appliance:/mnt/installables#
```

8. Configure SSH access.





If you perform the upgrade more than once, you can skip this step after performing it once.

If you have an existing SSH key, skip to step 8c.

a. Generate the SSH key to enable you to run the script.



Leave the passphrase blank to simplify running the script.

```
root@netq-appliance:/mnt/installables# ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
```

b. Copy the key to the authorized\_keys directory.

```
root@netq-appliance:/mnt/installables# cat ~/.ssh/id_rsa.pub
>> ~/.ssh/authorized_keys
root@netq-appliance:/mnt/installables# chmod 0600 ~/.ssh
/authorized_keys
root@netq-appliance:/mnt/installables#
```

c. Associate the key with the installer.

```
root@netq-appliance:/mnt/installables/# ./netq-install.sh --
usekey ~/.ssh/id_rsa
[Fri 22 Mar 2019 06:34:47 AM UTC] - This Script can only be
invoked by user: root
[Fri 22 Mar 2019 06:34:47 AM UTC] - The logged in user is
root
[Fri 22 Mar 2019 06:34:47 AM UTC] - Install directory /mnt
/installables exists on system.
[Fri 22 Mar 2019 06:34:47 AM UTC] - File /root/.ssh/id_rsa
exists on system...
[Fri 22 Mar 2019 06:34:47 AM UTC] - checking the presence
of existing instaler-ssh-keys secret/instaler-ssh-keys
created
[Fri 22 Mar 2019 06:34:48 AM UTC] - Unable to find netq-
installer up and running. Sleeping for 10 seconds ...
```



[Fri 22 Mar 2019 06:34:58 AM UTC] - Unable to find netq-installer up and running. Sleeping for 10 seconds ... [Fri 22 Mar 2019 06:35:08 AM UTC] - Able to find the netq-installer up and running...

9. Upgrade the netq-installer pod to the 2.1.x version.

```
root@netq-appliance:/mnt/installables# ./netq-install.sh --
installbundle /mnt/installables/NetQ-2.1.x.tgz
[Tue Apr 23 02:55:52 2019] - File /mnt/installables/NetQ-2.1.x.
tgz exists on system for updating netq installer ...
[Tue Apr 23 02:55:52 2019] - Updating the netg installer ...
[Tue Apr 23 02:55:52 2019] - Able to execute the command for
updating netg installer ...
[Tue Apr 23 02:55:52 2019] - Checking initialization of
installer update ...
[Tue Apr 23 02:55:57 2019] - Update of netq installer is in
progress ...
******[Tue Apr 23 02:58:57 2019] - Finished updating the netq-
installer
[Tue Apr 23 02:58:57 2019] - Finished updating netq installer!
0,/mnt/installables/NetQ-2.1.x.tgz
root@netg-appliance:/mnt/installables#
```

- Please allow 3-4 minutes for the installer upgrade to complete.
- 10. Confirm the netq-installer has been upgraded to the 2.1.x version.

```
root@netq-appliance:~# kubectl get pod -l app=netq-installer -
o=jsonpath={.items[0].status.containerStatuses[0].image}
netq-installer:2.1.x
```

11. Upgrade the NetQ software.

```
root@netq-appliance:/mnt/installables# ./netq-install.sh --
updateapps /mnt/installables/NetQ-2.1.x.tgz
[Tue Apr 23 02:55:52 2019] - File /mnt/installables/NetQ-2.1.x.
tgz exists on system for updating netq installer ...
[Tue Apr 23 02:55:52 2019] - Updating the netq installer ...
[Tue Apr 23 02:55:52 2019] - Able to execute the command for
updating netq installer ...
[Tue Apr 23 02:55:52 2019] - Checking initialization of
installer update ...
[Tue Apr 23 02:55:57 2019] - Update of netq installer is in
progress ...
```



```
******[Tue Apr 23 02:58:57 2019] - Finished updating the netq-
installer
[Tue Apr 23 02:58:57 2019] - Finishshed updating netq installer
0,/mnt/installables/NetQ-2.1.x.tgz
cumulus@sm-telem-02:/mnt/installables$ sudo /mnt/installables/.
/netg-install.sh --updateapps /mnt/installables/NetO-2.1.x.tgz
[Tue Apr 23 03:01:13 2019] - File /mnt/installables/NetQ-2.1.x.
tgz exists on system for updating netq apps...
[Tue Apr 23 03:01:13 2019] - User selected to update netq-apps
[Tue Apr 23 03:04:13 2019] - Able to execute the command for
netq apps updates ...
[Tue Apr 23 03:04:13 2019] - Checking initialization of apps
update ...
[Tue Apr 23 03:04:18 2019] - netq apps update is in progress ...
***************
*[Tue Apr 23 03:37:18 2019] - Finished updating the netq apps
[Tue Apr 23 03:37:18 2019] - Finished updating apps ....
```

ധ

Please allow about an hour for the upgrade to complete.

You are now running NetQ 2.1.x.

## Upgrade the NetQ Agents

Whether using the NetQ Appliance or your own hardware, the NetQ Agent should be upgraded on each of the existing nodes you want to monitor. The node can be a:

- Switch running Cumulus Linux version 3.7.0 or later
- Server running Red Hat RHEL 7.1, Ubuntu 16.04 or CentOS 7
- Linux virtual machine running any of the above Linux operating systems

To upgrade the NetQ Agent you need to install the OS-specific meta package, cumulus-netq, on each switch. Optionally, you can install it on hosts. The meta package contains the NetQ Agent, the NetQ command line interface (CLI), and the NetQ library. The library contains modules used by both the NetQ Agent and the CLI.

- Upgrade NetQ Agent on a Cumulus Linux Switch (see page 94)
- Upgrade NetQ Agent on an Ubuntu Server (see page 95)
- Upgrade NetQ Agent on a Red Hat or CentOS Server (see page 96)

If your network uses a proxy server for external connections, you should first configure a global proxy, so apt-get can access the meta package on the Cumulus Networks repository.

94 06 June 2019



#### **Upgrade NetQ Agent on a Cumulus Linux Switch**

A simple process installs the NetQ Agent on a Cumulus switch.

1. Edit the /etc/apt/sources.list file to add the repository for Cumulus NetQ. **Note** that NetQ has a separate repository from Cumulus Linux.

```
cumulus@switch:~$ sudo nano /etc/apt/sources.list
deb http://apps3.cumulusnetworks.com/repos/deb CumulusLinux-3
netq-2.1
. . .
```



The repository deb http://apps3.cumulusnetworks.com/repos/deb CumulusLinux-3 netq-latest can be used if you want to always retrieve the latest posted version of NetQ.

2. Update the local apt repository, then install the NetQ meta package on the switch.

```
cumulus@switch:~$ sudo apt-get update
cumulus@switch:~$ sudo apt-get install cumulus-netq
```

3. Verify the upgrade.

```
cumulus@switch:~$ dpkg -l | grep netq
ii cumulus-netq
                                    2.1.2-cl3u17~1557345432.
a60ec9a all
                     This meta-package provides installation
of Cumulus NetQ packages.
                                    2.1.2-cl3u17~1559681411.
ii netq-agent
2bba220 amd64
                  Cumulus NetQ Telemetry Agent for Cumulus
Linux
ii netq-apps
                                    2.1.2-cl3u17~1559681411.
2bba220 amd64 Cumulus NetQ Fabric Validation
Application for Cumulus Linux
```

4. Repeat these steps for each Cumulus switch, or use an automation tool to install NetQ Agent on multiple Cumulus Linux switches.

## **Upgrade NetQ Agent on an Ubuntu Server (Optional)**

To install the NetQ Agent on an Ubuntu server:

1. Reference and update the local apt repository.



```
root@ubuntu:~# wget -O- https://apps3.cumulusnetworks.com/setup
/cumulus-apps-deb.pubkey | apt-key add -
```

2. In /etc/apt/sources.list.d/cumulus-host-ubuntu-xenial.list, verify the following repository is included:

```
root@ubuntu:~# vi /etc/apt/sources.list.d/cumulus-apps-deb-
xenial.list
deb [arch=amd64] https://apps3.cumulusnetworks.com/repos/deb
xenial netq-latest
. . .
```



 $ilde{\mathbb{L}}$  The use of  $\mathtt{netq-latest}$  in this example means that a  $\mathtt{get}$  to the repository always. retrieves the latest version of NetO, even in the case where a major version update has been made. If you want to keep the repository on a specific version — such as netq-2.1 — use that instead.

3. Install the meta package on the server.

```
root@ubuntu:~# apt-get update
root@ubuntu:~# apt-get install cumulus-netq
```

4. Verify the upgrade.

```
root@ubuntu:~$ dpkg -l | grep netq
                                     2.1.2-cl3u17~1557345432.
ii cumulus-netq
a60ec9a
          all
                       This meta-package provides installation
of Cumulus NetQ packages.
ii netq-agent
                                     2.1.2-cl3u17~1559681411.
2bba220
         amd64
                       Cumulus NetQ Telemetry Agent for Cumulus
Linux
ii netq-apps
                                     2.1.2-cl3u17~1559681411.
         amd64
                       Cumulus NetO Fabric Validation
2bba220
Application for Cumulus Linux
```

5. Repeat these steps for each switch/host running Ubuntu, or use an automation tool to install NetQ Agent on multiple switches/hosts.

## Upgrade NetQ Agent on a Red Hat or CentOS Server (Optional)

To install the NetQ Agent on a Red Hat or CentOS server:

1. Reference and update the local yum repository.

96 06 June 2019



```
root@rhel7:~# rpm --import https://apps3.cumulusnetworks.com
/setup/cumulus-apps-rpm.pubkey
root@rhel7:~# wget -O- https://apps3.cumulusnetworks.com/setup
/cumulus-apps-rpm-el7.repo > /etc/yum.repos.d/cumulus-host-el.
repo
```

2. Edit /etc/yum.repos.d/cumulus-host-el.repo to set the enabled=1 flag for the two NetQ repositories.

```
[cumulus@firewall-2 ~]$ cat /etc/yum.repos.d/cumulus-host-el.
[cumulus-arch-netq-2.0]
name=Cumulus netq packages
baseurl=http://rohbuild03.mvlab.cumulusnetworks.com/dev/rpm/el/7
/netq-latest/$basearch
gpgcheck=1
enabled=1
[cumulus-noarch-netq-2.0]
name=Cumulus netq architecture-independent packages
baseurl=http://rohbuild03.mvlab.cumulusnetworks.com/dev/rpm/el/7
/netq-latest/noarch
gpgcheck=1
enabled=1
[cumulus-src-netq-2.0]
name=Cumulus netq source packages
baseurl=http://rohbuild03.mvlab.cumulusnetworks.com/dev/rpm/el/7
/netq-latest/src
apacheck=1
enabled=1
```

3. Update the NetQ meta packages on the server.

```
root@rhel7:~# yum update cumulus-netq.x86_64
```

4. Verify the upgrade.

```
root@ubuntu:~$ yum list installed | grep netq
ii cumulus-netq 2.1.2-cl3u17~1557345432.
a60ec9a all This meta-package provides installation
of Cumulus NetQ packages.
ii netq-agent 2.1.2-cl3u17~1559681411.
2bba220 amd64 Cumulus NetQ Telemetry Agent for Cumulus
Linux
```



```
ii netq-apps 2.1.2-cl3u17~1559681411.
2bba220 amd64 Cumulus NetQ Fabric Validation
Application for Cumulus Linux
```

5. Repeat these steps for each switch/host running Ubuntu, or use an automation tool to install NetQ Agent on multiple switches/hosts.

## **Configure Optional NetQ Agent Settings**

Once the NetQ Agents have been installed on the network nodes you want to monitor, the NetQ Agents must be configured to obtain useful and relevant data. The code examples shown in this section illustrate how to configure the NetQ Agent on a Cumulus switch, but it is exactly the same for the other type of nodes. If you have already configured these settings, you do not need to do so again.

- Configuring the Agent to Use a VRF
- Configuring the Agent to Communicate over a Specific Port

#### Configure the Agent to Use a VRF Interface

While optional, Cumulus strongly recommends that you configure NetQ Agents to communicate with the NetQ Platform only via a VRF, including a management VRF. To do so, you need to specify the VRF name when configuring the NetQ Agent. For example, if the management VRF is configured and you want the agent to communicate with the NetQ Platform over it, configure the agent like this:

```
cumulus@leaf01:~$ netq config add agent server 192.168.1.254 vrf mgmt cumulus@leaf01:~$ netq config add cli server 192.168.254 vrf mgmt
```

You then restart the agent:

```
cumulus@leaf01:~$ netq config restart agent cumulus@leaf01:~$ netq config restart cli
```

## Configure the Agent to Communicate over a Specific Port

By default, NetQ uses port 8981 for communication between the NetQ Platform and NetQ Agents. If you want the NetQ Agent to communicate with the NetQ Platform via a different port, you need to specify the port number when configuring the NetQ Agent like this:

```
cumulus@switch:~$ netq config add agent server 192.168.1.254 port 7379
```

You then restart the agent:

```
cumulus@leaf01:~$ netq config restart agent
```



## **Upgrade Tips**

After you have upgraded NetQ, if you find that some issues remain, review these commonly encountered scenarios. If NetQ is still not operating as expected, please open a support ticket with a detailed description of your issues.

#### Scenario 1: NetQ Appliance IP Address Was Changed

The NetQ Appliance came with a pre-configured IP address, which has since been changed. How do I recover the system?

You must remove and reconfigure the Kubernetes configuration. Follow these steps:

1. Locate the Kubernetes pods.

```
cumulus@switch:~$ kubectl get pods
```

2. Reset all Kubernetes administrative settings. Run the command twice to make sure all directories and files have been reset.

```
cumulus@switch:~$ sudo kudeadm reset -f
cumulus@switch:~$ sudo kudeadm reset -f
```

3. Remove the Kubernetes configuration.

```
cumulus@switch:~$ sudo rm /home/cumulus/.kube/config
```

4. Reset the NetQ Appliance install daemon.

```
cumulus@switch:~$ sudo systemctl reset-failed
```

5. Reset the Kubernetes service.

```
cumulus@switch:~$ sudo systemctl restart cts-kubectl-config
```



Allow 15 minutes for the prompt to return.

## Scenario 2: No IP Address Assigned to the NetQ Appliance on Boot

A user did not configure an IP address when the system was first booted. Later the user assigned an IP address to eth0, but the NetQ appliance does not appear to be functioning.

You must reset the install daemon and restart the Kubernetes service. Follow these steps:



1. Reset the NetQ Appliance install daemon.

cumulus@switch:~\$ sudo systemctl reset-failed

2. Restart the Kubernetes service.

cumulus@switch:~\$ sudo systemctl restart cts-kubectl-config



# Integrate with Third-party Software and Hardware

After you have installed NetQ applications package and the NetQ Agents, you may want to configure some of the additional capabilities that NetQ offers. This topic describes how to install, setup, and configure these capabilities.

#### **Contents**

This topic describes how to...

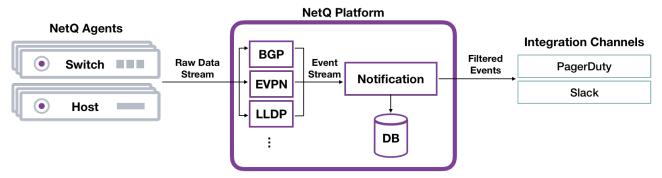
- Integrate NetQ with an Event Notification Application (see page 102)
  - Event Message Format (see page 103)
  - Notification Commands Overview (see page 104)
  - Configure a Proxy Server (see page 104)
  - Create Channels (see page 105)
    - Configure a PagerDuty Channel (see page 105)
    - Configure a Slack Channel (see page 106)
  - Create Rules (see page 108)
    - Example Rules (see page 113)
    - View the Rule Configurations (see page 114)
  - Create Filters (see page 115)
    - Example Filters (see page 116)
    - View the Filter Configurations (see page 117)
    - Reorder Filters (see page 118)
- Example Notification Configurations (see page 119)
  - Create a Notification for BGP Events from a Selected Switch (see page 119)
  - Create a Notification for Warnings on a Given EVPN VNI (see page 120)
  - Create a Notification for Configuration File Changes (see page 121)
  - Create a Notification for When a Service Goes Down (see page 122)
  - Create a Filter to Drop Notifications from a Given Interface (see page 123)
  - Create a Notification for a Given Device that has a Tendency to Overheat (using multiple rules) (see page 124)
  - View Notification Configurations in JSON Format (see page 125)
- Manage Event Notification Integrations (see page 127)
  - Remove an Event Notification Channel (see page 127)
  - Delete an Event Notification Rule (see page 127)
  - Delete an Event Notification Filter (see page 128)
- Integrate with a Hardware Chassis (see page 128)



## Integrate NetQ with an Event Notification Application

To take advantage of the numerous event messages generated and processed by NetQ, you must integrate with third-party event notification applications. You can integrate NetQ with the PagerDuty and Slack tools. You may integrate with one or both of these applications.

Each network protocol and service in the NetQ Platform receives the raw data stream from the NetQ Agents, processes the data and delivers events to the Notification function. Notification then stores, filters and sends messages to any configured notification applications . Filters are based on rules you create. You must have at least one rule per filter.



You may choose to implement a proxy server (that sits between the NetQ Platform and the integration channels) that receives, processes and distributes the notifications rather than having them sent directly to the integration channel. If you use such a proxy, you must configure NetQ with the proxy information.

In either case, notifications are generated for the following types of events:

- Network Protocols
  - BGP status and session state
  - CLAG (MLAG) status and session state
  - EVPN status and session state
  - LLDP status
  - LNV status and session state \*
  - OSFP status and session state
  - VLAN status and session state \*
  - VXLAN status and session state \*
- Interfaces
  - Link status
  - Ports and cables status
- Services status
  - NetQ Agent status
  - SSH \*
  - NTP status \*



- Trace status
- Sensors
  - Fan status
  - PSU (power supply unit) status
  - Temperature status
- System
  - Configuration File changes
  - Cumulus Linux License status \*
  - Cumulus Linux Support status

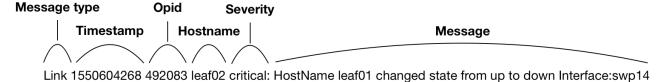
#### **Event Message Format**

Messages have the following structure: <message-

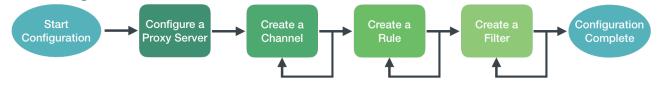
type><timestamp><opid><hostname><severity><message>

Element	Description	
message type	Category of event; bgp, clag, configdiff, evpn, link, lldp, lnv, node, ntp, ospf, port, sensor, services, trace, vlan or vxlan	
timestamp	Date and time event occurred	
opid	Identifier of the service or process that generated the event	
hostname	Hostname of network device where event occurred	
severity	Severity Severity level in which the given event is classified; <i>debug</i> , <i>error</i> , <i>info</i> , <i>warning</i> , or <i>critical</i>	
message Text description of event		

#### For example:



To set up the integrations, you must configure NetQ with at least one channel. Optionally, you can define rules and filters to refine what messages you want to view and where to send them. You can also configure a proxy server to receive, process, and forward the messages. This is accomplished using the NetQ CLI in the following order:



<sup>\*</sup> This type of event can only be viewed in the CLI with this release.



#### **Notification Commands Overview**

The NetQ Command Line Interface (CLI) is used to filter and send notifications to third-party tools based on severity, service, event-type, and device. You can use TAB completion or the help keyword to assist when needed. The command syntax is:

```
##Proxy
netq add notification proxy <text-proxy-hostname> [port <text-proxy-
netg show notification proxy
netq del notification proxy
##Channels
netq add notification channel slack <text-channel-name> webhook <text-
webhook-url> [severity info|severity warning|severity error|severity
debug] [tag <text-slack-tag>]
netq add notification channel pagerduty <text-channel-name>
integration-key <text-integration-key> [severity info|severity
warning|severity error|severity debug]
##Rules and Filters
netq add notification rule <text-rule-name> key <text-rule-key> value
<text-rule-value>
netq add notification filter <text-filter-name> [severity
info|severity warning|severity error|severity debug] [rule <text-rule-
name-anchor>] [channel <text-channel-name-anchor>] [before <text-
filter-name-anchor>|after <text-filter-name-anchor>]
##Management
netq del notification channel <text-channel-name-anchor>
netg del notification filter <text-filter-name-anchor>
netq del notification rule <text-rule-name-anchor>
netq show notification [channel|filter|rule] [json]
```

The options are described in the following sections where they are used.

## **Configure a Proxy Server**

To send notification messages through a proxy server instead of directly to a notification channel, you configure NetQ with the hostname and optionally a port of a proxy server. If no port is specified, NetQ defaults to port 80. Only one proxy server is currently supported. To simplify deployment, configure your proxy server before configuring channels, rules, or filters. To configure the proxy server:

```
cumulus@switch:~$ netq add notification proxy <text-proxy-hostname> [port <text-proxy-port] cumulus@switch:~$ netq add notification proxy proxy4 Successfully configured notifier proxy proxy4:80
```



You can view the proxy server settings by running the netq show notification proxy command.

You can remove the proxy server by running the netq del notification proxy command. This changes the NetQ behavior to send events directly to the notification channels.

cumulus@switch:~\$ netq del notification proxy
Successfully overwrote notifier proxy to null

#### **Create Channels**

Create one or more PagerDuty and Slack channels to present the notifications.

#### Configure a PagerDuty Channel

NetQ sends notifications to PagerDuty as PagerDuty events.

#### For example:

Status	Urgency T	Title	Created	Service	Assigned To
Resolved	Low	filter#default: NetQ Agent: spine-1: state change from fresh to rotten  B SHOW DETAILS (1 resolved alert) #106	3:08 PM	Puneet - Netq Notifier integration	-
Resolved	Low	filter#default: Service: noc-se clagd (vrf default) state changed from ok to warning  B SHOW DETAILS (1 resolved alert) #106	3:08 PM	Puneet - Netq Notifier integration	
Resolved	Low	filter#default: BGP: tor-2 uplink-1: session state changed from established to failed  B SHOW DETAILS (1resolved alert) #106	on Aug 31, 2017 at 3:08 PM	Puneet - Netq Notifier integration	
Resolved	Low	filter#default: BGP: torc-12 uplink-1: session state changed from established to failed  B SHOW DETAILS (1 resolved alert) #106:	3:08 PM	Puneet - Netq Notifier integration	

To configure the NetQ notifier to send notifications to PagerDuty:

1. Configure the following options using the netq add notification channel command:

Option	Description
CHANNEL_TYPE <text-channel-name></text-channel-name>	The third-party notification channel and name; use <i>pagerduty</i> in this case.



Option	Description
integration-key <text- integration-key&gt;</text- 	The integration key is also called the service_key or routing_key. The default is an empty string ("").
severity	(Optional) The log level to set, which can be one of <i>info</i> , warning, error, critical or debug. The severity defaults to <i>info</i> .

cumulus@switch:~\$ netq add notification channel pagerduty pd-netq-events integration-key c6d666e210a8425298ef7abde0d1998 Successfully added/updated channel pd-netq-events

2. Verify that the channel is configured properly.

#### Configure a Slack Channel

NetQ Notifier sends notifications to Slack as incoming webhooks for a Slack channel you configure. For example:

@NoName link event occurred at Mon, 25 Mar 2019 18:08:14

link: HostName noc-se changed state from up to down Interface:peerlink-1

From NetQ

@NoName link event occurred at Mon, 25 Mar 2019 18:08:24

link : HostName noc-se changed state from down to up Interface:swp1

From NetQ

@NoName link event occurred at Mon, 25 Mar 2019 18:08:24

link : HostName noc-se changed state from down to up Interface:swp10

From NetQ

To configure NetQ to send notifications to Slack:

1. If needed, create one or more Slack channels on which to receive the notifications.

106 June 2019



- a. Click + next to Channels.
- b. Enter a name for the channel, and click **Create Channel**.
- c. Navigate to the new channel.
- d. Click + Add an app link below the channel name to open the application directory.
- e. In the search box, start typing *incoming* and select **Incoming WebHooks** when it appears.
- f. Click **Add Configuration** and enter the name of the channel you created (where you want to post notifications).
- g. Click Add Incoming WebHooks integration.
- h. Save WebHook URL in a text file for use in next step.
- 2. Configure the following options in the netq config add notification channel command:

Option	Description
CHANNEL_TYPE <text-channel- name&gt;</text-channel- 	The third-party notification channel name; use <i>slack</i> in this case.
WEBHOOK	Copy the WebHook URL from the text file OR in the desired channel, locate the initial message indicating the addition of the webhook, click <b>incomingwebhook</b> link, click <b>Settings</b> .  Example URL: https://hooks.slack.com/services/text/moretext/evenmoretext
severity	The log level to set, which can be one of <i>error</i> , <i>warning</i> , <i>info</i> , or <i>debug</i> . The severity defaults to <i>info</i> .
tag	Optional tag appended to the Slack notification to highlight particular channels or people. The tag value must be preceded by the @ sign. For example, @netq-info.

cumulus@switch:~\$ netq add notification channel slack slk-netqevents webhook https://hooks.slack.com/services/text/moretext
/evenmoretext
Successfully added/updated channel netq-events

3. Verify the channel is configured correctly. From the CLI:



moretext

/evenmoretext

From the Slack Channel:

Administrator 2:12 PM

added an integration to this channel: incoming-webhook

#### **Create Rules**

Each rule is comprised of a single key-value pair. The key-value pair indicates what messages to include or drop from event information sent to a notification channel. You can create more than one rule for a single filter. Creating multiple rules for a given filter can provide a very defined filter. For example, you can specify rules around hostnames or interface names, enabling you to filter messages specific to those hosts or interfaces. You should have already defined the PagerDuty or Slack channels (as described earlier).

There is a fixed set of valid rule keys. Values are entered as regular expressions and vary according to your deployment.

Service	Rule Key	Description	Example Rule Values
BGP	message_type	Network protocol or service identifier	bgp
	hostname	User-defined, text-based name for a switch or host	server02, leaf11, exit01, spine-4
	peer	User-defined, text-based name for a peer switch or host	server4, leaf-3, exit02, spine06
	desc	Text description	
	vrf	Name of VRF interface	mgmt, default
	old_state	Previous state of the BGP service	Established, Failed
	new_state	Current state of the BGP service	Established, Failed
	old_last_reset_time	Previous time that BGP service was reset	Apr3, 2019, 4:17 pm
	new_last_reset_time	Most recent time that BGP service was reset	Apr8, 2019, 11:38 am
MLAG (CLAG)	message_type	Network protocol or service identifier	clag
	hostname		



Service	Rule Key	Description	Example Rule Values
		User-defined, text-based name for a switch or host	server02, leaf-9, exit01, spine04
	old_conflicted_bonds	Previous pair of interfaces in a conflicted bond	swp7 swp8, swp3 swp4
	new_conflicted_bonds	Current pair of interfaces in a conflicted bond	swp11 swp12, swp23 swp24
	old_state_protodownbond	Previous state of the bond	protodown, up
	new_state_protodownbond	Current state of the bond	protodown, up
ConfigDiff	message_type	Network protocol or service identifier	configdiff
	hostname User-defined, text-based name a switch or host		server02, leaf11, exit01, spine-4
	vni	Virtual Network Instance identifier	12, 23
	old_state	Previous state of the configuration file	created, modified
	new_state	Current state of the configuration file	created, modified
EVPN	message_type	Network protocol or service identifier	evpn
	hostname	User-defined, text-based name for a switch or host	server02, leaf-9, exit01, spine04
	vni	Virtual Network Instance identifier	12, 23
	old_in_kernel_state	Previous VNI state, in kernel or not	true, false
	new_in_kernel_state	Current VNI state, in kernel or not	true, false
	old_adv_all_vni_state	Previous VNI advertising state, advertising all or not	true, false
	new_adv_all_vni_state	Current VNI advertising state, advertising all or not	true, false



Service	Rule Key	Description	Example Rule Values
Link	message_type	Network protocol or service identifier	link
	hostname	User-defined, text-based name for a switch or host	server02, leaf-6, exit01, spine7
	ifname	Software interface name	eth0, swp53
LLDP	message_type	Network protocol or service identifier	lldp
	hostname	User-defined, text-based name for a switch or host	server02, leaf41, exit01, spine-5, tor-36
	ifname	Software interface name	eth1, swp12
	old_peer_ifname	Previous software interface name	eth1, swp12, swp27
	new_peer_ifname	Curent software interface name	eth1, swp12, swp27
	old_peer_hostname	Previous user-defined, text-based name for a peer switch or host	server02, leaf41, exit01, spine-5, tor-36
	new_peer_hostname	Current user-defined, text-based name for a peer switch or host	server02, leaf41, exit01, spine-5, tor-36
Node	message_type	Network protocol or service identifier	node
	hostname	User-defined, text-based name for a switch or host	server02, leaf41, exit01, spine-5, tor-36
	ntp_state	Current state of NTP service	in sync, not sync
	db_state	Current state of DB	Add, Update, Del, Dead
NTP	message_type	Network protocol or service identifier	ntp
	hostname	User-defined, text-based name for a switch or host	server02, leaf-9, exit01, spine04
	old_state	Previous state of service	in sync, not sync



Service	Rule Key	Description	Example Rule Values
	new_state	Current state of service	in sync, not sync
Port	message_type	Network protocol or service identifier	port
	hostname	User-defined, text-based name for a switch or host	server02, leaf13, exit01, spine-8, tor-36
	ifname	Interface name	eth0, swp14
	old_speed	Previous speed rating of port	10 G, 25 G, 40 G, unknown
	old_transreceiver	Previous transceiver	40G Base-CR4, 25G Base-CR
	old_vendor_name	Previous vendor name of installed port module	Amphenol, OEM, Mellanox, Fiberstore, Finisar
	old_serial_number	Previous serial number of installed port module	MT1507VS05177, AVE1823402U, PTN1VH2
	old_supported_fec	Previous forward error correction (FEC) support status	none, Base R, RS
	old_advertised_fec	Previous FEC advertising state	true, false, not reported
	old_fec	Previous FEC capability	none
	old_autoneg	Previous activation state of autonegotiation	on, off
	new_speed	Current speed rating of port	10 G, 25 G, 40 G
	new_transreceiver	Current transceiver	40G Base-CR4, 25G Base-CR
	new_vendor_name	Current vendor name of installed port module	Amphenol, OEM, Mellanox, Fiberstore, Finisar
	new_part_number	Current part number of installed port module	SFP-H10GB-CU1M, MC3309130-001, 603020003



Service	Rule Key	Description	Example Rule Values
	new_serial_number	Current serial number of installed port module	MT1507VS05177, AVE1823402U, PTN1VH2
	new_supported_fec	Current FEC support status	none, Base R, RS
	new_advertised_fec	Current FEC advertising state	true, false
	new_fec	Current FEC capability	none
	new_autoneg	Current activation state of autonegotiation	on, off
Sensors	sensor	Network protocol or service identifier	Fan: fan1, fan-2 Power Supply Unit: psu1 psu2 Temperature: psu1temp1, temp2
	hostname	User-defined, text-based name for a switch or host	server02, leaf-26, exit01, spine2-4
	old_state	Previous state of a fan, power supply unit, or thermal sensor	Fan: ok, absent, bad PSU: ok, absent, bad Temp: ok, busted, bad, critical
	new_state	Current state of a fan, power supply unit, or thermal sensor	Fan: ok, absent, bad PSU: ok, absent, bad Temp: ok, busted, bad, critical
	old_s_state	Previous state of a fan or power supply unit.	Fan: up, down PSU: up, down
	new_s_state	Current state of a fan or power supply unit.	Fan: up, down PSU: up, down
	new_s_max	Current maximum temperature threshold value	Temp: 110
	new_s_crit	Current critical high temperature threshold value	Temp: 85
	new_s_lcrit	Current critical low temperature threshold value	Temp: -25



Service	Rule Key	Description	Example Rule Values
	new_s_min	Current minimum temperature threshold value	Temp: -50
Services	message_type	Network protocol or service identifier	services
	hostname	User-defined, text-based name for a switch or host	server02, leaf03, exit01, spine-8
	name	Name of service	clagd, lldpd, ssh, ntp, netqd, net-agent
	old_pid	Previous process or service identifier	12323, 52941
	new_pid	Current process or service identifier	12323, 52941
	old_status	Previous status of service	up, down
	new_status	Current status of service	up, down

 Rule names are case sensitive, and no wildcards are permitted. Rule names may contain spaces, but must be enclosed with single quotes in commands. It is easier to use dashes in place of spaces or mixed case for better readability. For example, use bgpSessionChanges or BGP-sessionchanges or BGPsessions, instead of 'BGP Session Changes'.

Use Tab completion to view the command options syntax.

## Example Rules

Create a BGP Rule Based on Hostname:

cumulus@switch:~\$ netq add notification rule bgpHostname key hostname value spine-01 Successfully added/updated rule bgpHostname

Create a Rule Based on a Configuration File State Change:

cumulus@switch:~\$ netq add notification rule sysconf key configdiff value updated Successfully added/updated rule sysconf



Create an EVPN Rule Based on a VNI:

cumulus@switch:~\$ netq add notification rule evpnVni key vni value 42 Successfully added/updated rule evpnVni

Create an Interface Rule Based on FEC Support:

cumulus@switch:~\$ netq add notification rule fecSupport key
new\_supported\_fec value supported
Successfully added/updated rule fecSupport

Create a Service Rule Based on a Status Change:

cumulus@switch:~\$ netq add notification rule svcStatus key new\_status
value down
Successfully added/updated rule svcStatus

Create a Sensor Rule Based on a Threshold:

cumulus@switch:~\$ netq add notification rule overTemp key new\_s\_crit value 24 Successfully added/updated rule overTemp

Create an Interface Rule Based on Port:

cumulus@switch:~\$ netq add notification rule swp52 key port value swp52 Successfully added/updated rule swp52

#### View the Rule Configurations

Use the netq show notification command to view the rules on your platform.



#### **Create Filters**

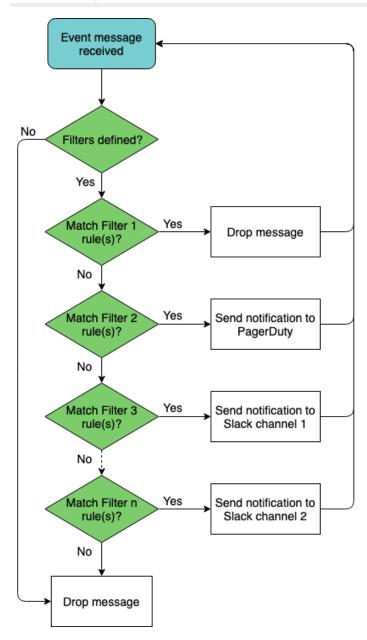
You can limit or direct event messages using filters. Filters are created based on rules you define; like those in the previous section. Each filter contains one or more rules. When a message matches the rule, it is sent to the indicated destination. Before you can create filters, you need to have already defined the rules and configured PagerDuty and/or Slack channels (as described earlier).

As filters are created, they are added to the bottom of a filter list. By default, filters are processed in the order they appear in this list (from top to bottom) until a match is found. This means that each event message is first evaluated by the first filter listed, and if it matches then it is processed, ignoring all other filters, and the system moves on to the next event message received. If the event does not match the first filter, it is tested against the second filter, and if it matches then it is processed and the system moves on to the next event received. And so forth. Events that do not match any filter are ignored.

You may need to change the order of filters in the list to ensure you capture the events you want and drop the events you do not want. This is possible using the *before* or *after* keywords to ensure one rule is processed before or after another.

This diagram shows an example with four defined filters with sample output results.





①

Filter names may contain spaces, but *must* be enclosed with single quotes in commands. It is easier to use dashes in place of spaces or mixed case for better readability. For example, use bgpSessionChanges or BGP-session-changes or BGPsessions, instead of 'BGP Session Changes'. Filter names are also case sensitive.

## Example Filters

Create a filter for BGP Events on a Particular Device:

cumulus@switch:~\$ netq add notification filter bgpSpine rule bgpHostname channel pd-netq-events Successfully added/updated filter bgpSpine



Create a Filter for a Given VNI in Your EVPN Overlay:

cumulus@switch:~\$ netq add notification filter vni42 severity warning
rule evpnVni channel pd-netq-events
Successfully added/updated filter vni42

Create a Filter for when a Configuration File has been Updated:

cumulus@switch:~\$ netq add notification filter configChange severity info rule sysconf channel slk-netq-events
Successfully added/updated filter configChange

Create a Filter to Monitor Ports with FEC Support:

cumulus@switch:~\$ netq add notification filter newFEC rule fecSupport channel slk-netq-events
Successfully added/updated filter newFEC

Create a Filter to Monitor for Services that Change to a Down State:

cumulus@switch:~\$ netq add notification filter svcDown severity error
rule svcStatus channel slk-netq-events
Successfully added/updated filter svcDown

Create a Filter to Monitor Overheating Platforms:

cumulus@switch:~\$ netq add notification filter critTemp severity
error rule overTemp channel pd-netq-events
Successfully added/updated filter critTemp

Create a Filter to Drop Messages from a Given Interface, and match against this filter before any other filters. To create a drop style filter, do not specify a channel. To put the filter first, use the *before* option.

cumulus@switch:~\$ netq add notification filter swp52Drop severity
error rule swp52 before bgpSpine
Successfully added/updated filter swp52Drop

#### View the Filter Configurations

Use the netg show notification command to view the filters on your platform.



cumulus@switch Matching confi	· -	now notification	filter	
Name	0rder 	Severity	Channels	Rules
swp52Drop	1	error	NetqDefaultCha el	nn swp52
bgpSpine bgpHostnam	2	info	pd-netq-events	
				е
vni42 evpnVni	3	warning	pd-netq-even	ts
configChange	4	info	slk-netq-events	sysconf
newFEC fecSupport	5	info	slk-netq-events	
svcDown svcStatus	6	critical	slk-netq-events	
critTemp	7	critical	pd-netq-events	overTemp

#### Reorder Filters

When you look at the results of the netg show notification filter command above, you might notice that although you have the drop-based filter first (no point in looking at something you are going to drop anyway, so that is good), but the critical severity events are processed last, per the current definitions. If you wanted to process those before lesser severity events, you can reorder the list using the before and after options.

For example, to put the two critical severity event filters just below the drop filter:

cumulus@switch:~\$ netq add notification filter critTemp after swp52Drop Successfully added/updated filter critTemp cumulus@switch:~\$ netq add notification filter svcDown before bgpSpine Successfully added/updated filter svcDown



✓ You do not need to reenter all the severity, channel, and rule information for existing rules if you only want to change their processing order.

Run the netg show notification command again to verify the changes:

cumulus@switc Matching conf	-	how notification	n filter	
Name	Order	Severity	Channels	Rules
swp52Drop	1	error	NetqDefault	tChann swp52
			el	

118 06 June 2019



critTemp	2	critical	pd-netq-events overTemp
svcDown	3	critical	slk-netq-events
svcStatus			
bgpSpine	4	info	pd-netq-events
bgpHostnam			
			е
vni42	5	warning	pd-netq-events
evpnVni			
configChange	6	info	slk-netq-events sysconf
newFEC	7	info	slk-netq-events
fecSupport			

# **Example Notification Configurations**

Putting all of these channel, rule, and filter definitions together you create a complete notification configuration. The following are example notification configurations are created using the three-step process outlined above. Refer to Integrate NetQ with an Event Notification Application (see page 102) for details and instructions for creating channels, rules, and filters.

#### Create a Notification for BGP Events from a Selected Switch

In this example, we created a notification integration with a PagerDuty channel called *pd-netq-events*. We then created a rule *bgpHostname* and a filter called *4bgpSpine* for any notifications from *spine-01*. The result is that any info severity event messages from Spine-01 are filtered to the *pd-netq-events* channel.

```
cumulus@switch:~$ netq add notification channel pagerduty pd-netq-
events integration-key 1234567890
Successfully added/updated channel pd-netq-events
cumulus@switch:~$ netq add notification rule bgpHostname key node
value spine-01
Successfully added/updated rule bgpHostname
cumulus@switch:~$ netq add notification filter bgpSpine rule
bgpHostname channel pd-netg-events
Successfully added/updated filter bgpSpine
cumulus@switch:~$ netq show notification channel
Matching config_notify records:
             Type Severity Channel Info
Name
pd-netq-events pagerduty info
                                             integration-key:
1234567
                                              890
cumulus@switch:~$ netg show notification rule
Matching config_notify records:
       Rule Key Rule Value
bgpHostname hostname spine-01
```



cumulus@switc Matching conf	· -	how notificatio	n filter	
Name	Order	Severity	Channels	Rules
				_
bgpSpine	1	info	pd-netq-events	
bgpHostnam				
				е

## Create a Notification for Warnings on a Given EVPN VNI

In this example, we created a notification integration with a PagerDuty channel called *pd-netq-events*. We then created a rule *evpnVni* and a filter called *3vni42* for any warnings messages from VNI 42 on the EVPN overlay network. The result is that any warning severity event messages from VNI 42 are filtered to the *pd-netg-events* channel.

```
cumulus@switch:~$ netq add notification channel pagerduty pd-netq-
events integration-key 1234567890
Successfully added/updated channel pd-netq-events
cumulus@switch:~$ netq add notification rule evpnVni key vni value 42
Successfully added/updated rule evpnVni
cumulus@switch:~$ netq add notification filter vni42 rule evpnVni
channel pd-netg-events
Successfully added/updated filter vni42
cumulus@switch:~$ netq show notification channel
Matching config_notify records:
      Type Severity Channel Info
Name
pd-netq-events pagerduty info integration-key:
1234567
                                            890
cumulus@switch:~$ netq show notification rule
Matching config_notify records:
Name Rule Key Rule Value
bgpHostname hostname spine-01
evpnVni vni 42
cumulus@switch:~$ netq show notification filter
Matching config_notify records:
Name Order Severity Channels Rules
```



bgpSpine bgpHostnam	1	info	pd-netq-events	
vni42	2	warning	pd-netq-events	e evpnVni

## **Create a Notification for Configuration File Changes**

In this example, we created a notification integration with a Slack channel called *slk-netq-events*. We then created a rule *sysconf* and a filter called *configChange* for any configuration file update messages. The result is that any configuration update messages are filtered to the *slk-netq-events* channel.

```
cumulus@switch:~$ netq add notification channel slack slk-netq-events
webhook https://hooks.slack.com/services/text/moretext/evenmoretext
Successfully added/updated channel slk-netq-events
cumulus@switch:~$ netg add notification rule sysconf key configdiff
value updated
Successfully added/updated rule sysconf
cumulus@switch:~$ netq add notification filter configChange severity
info rule sysconf channel slk-netq-events
Successfully added/updated filter configChange
cumulus@switch:~$ netq show notification channel
Matching config_notify records:
Name Type Severity Channel Info
slk-netq-events slack
                           info webhook:https://hooks.s
                                   lack.com/services/text/
                                    moretext
/evenmoretext
cumulus@switch:~$ netg show notification rule
Matching config_notify records:
            Rule Key Rule Value
______
bgpHostname hostname spine-01
evpnVni
            vni
                           42
          configdiff updated
sysconf
cumulus@switch:~$ netq show notification filter
Matching config_notify records:
       Order Severity Channels Rules
_____
            1 info
bgpSpine
                                      pd-netq-events
bgpHostnam
vni42
                        warning
                                      pd-netq-events evpnVni
configChange 3
                       info
                                      slk-netq-events sysconf
```



#### Create a Notification for When a Service Goes Down

In this example, we created a notification integration with a Slack channel called *slk-netg-events*. We then created a rule svcStatus and a filter called svcDown for any services state messages indicating a service is no longer operational. The result is that any service down messages are filtered to the slk-netg-events channel.

cumulus@switch:~\$ netq add notification channel slack slk-netq-events webhook https://hooks.slack.com/services/text/moretext/evenmoretext Successfully added/updated channel slk-netq-events

cumulus@switch:~\$ netq add notification rule svcStatus key new\_status value down

Successfully added/updated rule svcStatus

cumulus@switch:~\$ netq add notification filter svcDown severity error rule svcStatus channel slk-netq-events Successfully added/updated filter svcDown

cumulus@switch:~\$ netq show notification channel

Matching config\_notify records:

Severity Channel Info Type \_\_\_\_\_\_\_ info webhook:https://hooks.s slk-netq-events slack lack.com/services/text/ moretext

/evenmoretext

cumulus@switch:~\$ netq show notification rule

Matching config\_notify records:

Name Rule Key Rule Value \_\_\_\_\_\_ bgpHostname hostname spine-01 evpnVni vni 42 svcStatus new\_status down sysconf configdiff updated

cumulus@switch:~\$ netq show notification filter

Matching config\_notify records:

Order Severity Channels Rules bgpSpine 1 info pd-netq-events bgpHostnam vni42 2 configChange 3 svcDown 4 warning pd-netq-events evpnVni info slk-netq-events sysconf critical slk-netg-events svcStatus

122 06 June 2019



### Create a Filter to Drop Notifications from a Given Interface

In this example, we created a notification integration with a Slack channel called *slk-netq-events*. We then created a rule *swp52* and a filter called *swp52Drop* that drops all notifications for events from interface *swp52*.

cumulus@switch:~\$ netq add notification channel slack slk-netq-events webhook https://hooks.slack.com/services/text/moretext/evenmoretext Successfully added/updated channel slk-netg-events cumulus@switch:~\$ netq add notification rule swp52 key port value swp52 Successfully added/updated rule swp52 cumulus@switch:~\$ netq add notification filter swp52Drop severity error rule swp52 before bgpSpine Successfully added/updated filter swp52Drop cumulus@switch:~\$ netq show notification channel Matching config\_notify records: Severity Channel Info Name Type info webhook:https://hooks.s slk-netq-events slack lack.com/services/text/ moretext /evenmoretext cumulus@switch:~\$ netq show notification rule Matching config\_notify records: Name Rule Key Rule Value bgpHostname hostname spine-01 vni evpnVni svcStatus new\_status down port swp52 swp52 sysconf configdiff updated cumulus@switch:~\$ netq show notification filter Matching config\_notify records: Order Severity Channels Rules swp52Drop 1 error NetqDefaultChann swp52 el bgpSpine 2 info pd-netq-events bgpHostnam vni42 warning pd-netq-events evpnVni configChange 4 info slk-netq-events sysconf



	_			
svcDown	5	critical	slk-netq-events	
svcStatus				

# Create a Notification for a Given Device that has a Tendency to Overheat (using multiple rules)

In this example, we created a notification when switch *leaf04* has passed over the high temperature threshold. Two rules were needed to create this notification, one to identify the specific device and one to identify the temperature trigger. We sent the message to the *pd-netg-events* channel.

```
cumulus@switch:~$ netq add notification channel pagerduty pd-netq-
events integration-key 1234567890
Successfully added/updated channel pd-netq-events
cumulus@switch:~$ netg add notification rule switchLeaf04 key
hostname value leaf04
Successfully added/updated rule switchLeaf04
cumulus@switch:~$ netq add notification rule overTemp key new_s_crit
value 24
Successfully added/updated rule overTemp
cumulus@switch:~$ netq add notification filter critTemp rule
switchLeaf04 channel pd-netq-events
Successfully added/updated filter critTemp
cumulus@switch:~$ netq add notification filter critTemp severity
critical rule overTemp channel pd-netq-events
Successfully added/updated filter critTemp
cumulus@switch:~$ netq show notification channel
Matching config_notify records:
                               Severity Channel Info
               Type
pd-netg-events pagerduty info
                                                integration-key:
1234567
                                                 890
cumulus@switch:~$ netq show notification rule
Matching config_notify records:
Name Rule Key Rule Value
bgpHostname hostname spine-01
evpnVni vni 42
overTemp new_s_crit 24
svcStatus new_status down
switchLeaf04 hostname leaf04
swp52
               port
                                swp52
sysconf configdiff updated
cumulus@switch:~$ netq show notification filter
```



Matching confi Name	g_notify r Order	ecords: Severity	Channels	Rules
swp52Drop	1	error	NetqDefaultCha el	nn swp52
bgpSpine bgpHostnam	2	info	pd-netq-events	
51				е
vni42	3	warning	pd-netq-events	evpnVni
configChange	4	info	slk-netq-events	sysconf
svcDown svcStatus	5	critical	slk-netq-events	
critTemp switchLeaf	6	critical	pd-netq-events	
				04
overTemp				

## **View Notification Configurations in JSON Format**

You can view configured integrations using the netq show notification commands. To view the channels, filters, and rules, run the three flavors of the command. Include the json option to display JSON-formatted output.

For example:

```
cumulus@switch:~$ netq show notification channel json
    "config_notify":[
            "type": "slack",
            "name": "slk-netq-events",
            "channelInfo": "webhook: https://hooks.slack.com/services
/text/moretext/evenmoretext",
            "severity": "info"
            "type": "pagerduty",
            "name": "pd-netq-events",
            "channelInfo": "integration-key: 1234567890",
            "severity": "info"
    }
    "truncatedResult":false
}
cumulus@switch:~$ netq show notification rule json
    "config_notify":[
```



```
"ruleKey": "hostname",
             "ruleValue": "spine-01",
             "name": "bgpHostname"
        },
             "ruleKey": "vni",
             "ruleValue":42,
             "name": "evpnVni"
        },
             "ruleKey": "new_supported_fec",
             "ruleValue": "supported",
             "name": "fecSupport"
        },
             "ruleKey": "new_s_crit",
             "ruleValue":24,
             "name": "overTemp"
        },
             "ruleKey": "new_status",
             "ruleValue": "down",
             "name": "svcStatus"
             "ruleKey": "configdiff",
             "ruleValue": "updated",
             "name": "sysconf"
    "truncatedResult":false
}
cumulus@switch:~$ netq show notification filter json
    "config notify":[
             "channels": "pd-netq-events",
             "rules": "overTemp",
             "name":"1critTemp",
             "severity": "critical"
             "channels": "pd-netq-events",
             "rules": "evpnVni",
             "name": "3vni42",
             "severity": "warning"
        },
             "channels": "pd-netq-events",
             "rules": "bgpHostname",
```



```
"name": "4bgpSpine",
             "severity": "info"
             "channels": "slk-netq-events",
             "rules": "sysconf",
             "name": "configChange",
             "severity": "info"
             "channels": "slk-netq-events",
             "rules": "fecSupport",
             "name": "newFEC",
             "severity": "info"
             "channels": "slk-netq-events",
             "rules": "svcStatus",
             "name": "svcDown",
             "severity": "critical"
    ],
    "truncatedResult":false
}
```

## **Manage Event Notification Integrations**

You might need to modify event notification configurations at some point in the lifecycle of your deployment. Optionally, you might want to configure a proxy.

#### **Remove an Event Notification Channel**

You can delete an event notification integration using the netq config del notification command. You can verify it has been removed using the related show command.

For example, to remove a Slack integration and verify it is no longer in the configuration:



#### **Delete an Event Notification Rule**

To delete a rule, use the following command, then verify it has been removed:

```
cumulus@switch:~$ netq del notification rule swp52
cumulus@switch:~$ netq show notification rule
Matching config_notify records:
                          Rule Value
Name
              Rule Key
bgpHostname hostname
evpnVni vni
overTemp new_s_crit
                             spine-01
                               42
                             24
svcStatus
             new_status
                               down
switchLeaf04 hostname
                              leaf04
           configdiff updated
sysconf
```

#### **Delete an Event Notification Filter**

To delete a filter, use the following command, then verify it has been removed:

cumulus@switch:~\$ netq del notification filter bgpSpine cumulus@switch:~\$ netq show notification filter Matching config_notify records:						
Name	Order		Channels	Rules		
swp52Drop	1	error	NetqDefaultChan el	NetqDefaultChann swp52 el		
vni42	2	warning	pd-netq-events	evpnVni		
configChange	3	info	slk-netq-events	sysconf		
svcDown svcStatus	4	critical	slk-netq-events			
critTemp	5	critical	pd-netq-events			
switchLeaf	J	CIICICAI	pa-necq-evenes			
SWICCILLEAL				04		
overTemp						

# Integrate with a Hardware Chassis

NetQ can run within a Facebook Backpack chassis, Cumulus Express CX-10256-S chassis or Edgecore OMP-800 chassis.

Keep the following issues in mind if you intend to use NetQ with a chassis:

- You must assign a unique hostname to every node that runs the NetQ Agent. By default, all the fabric cards in the chassis have the same hostname.
- The NetQ Agent must be installed on every line card.



- No information is returned about the ASIC when you run netq show inventory asic. This is a known issue.
- Since the chassis sensor information is shared, every line card and fabric card can report the same sensor data. By default, sensor data is disabled on a chassis to avoid this duplication. To enable sensor data on a line card, edit /etc/netq/netq.yml or /etc/netq/config.d/user.yml and set the send\_chassis\_sensor\_data keyword to true, then restart the NetQ Agent with netq config agent restart. Configuring NetQ in this way prevents any duplication of data in the NetQ database.

```
cumulus@chassis:~$ sudo nano /etc/netq/netq.yml
...
netq-agent:
   send_chassis_sensor_data: true
...
```