



Cisco Network Insights for Resources Application for Cisco DCNM User Guide, Release 2.0.x

First Published: 2019-06-07

Last Modified: 2020-11-17

Americas Headquarters

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
<http://www.cisco.com>
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883



CONTENTS

CHAPTER 1	New and Changed Information	1
	New and Changed Information	1

CHAPTER 2	Cisco Network Insights for Resources Overview	3
	About Cisco Network Insights for Resources	3
	Hardware Requirements	3
	Downloading Cisco Network Insights for Resources from the Cisco App Center	4
	Installing Cisco NIR Application in Cisco DCNM	5

CHAPTER 3	Using Cisco Network Insights for Resources	7
	Cisco Network Insights for Resources Components in Cisco DCNM	7
	Guidelines and Limitations	8
	Cisco NIR App Initial Setup	8
	Cisco NIR App Settings	11
	Navigating Cisco NIR	13
	Using the Cisco Network Insights for Resources App	16

CHAPTER 4	Cisco NIR DCNM REST API Examples	27
	all_resources()	27
	anomalies_details()	28
	anomalies_summary()	29
	get_fabrics_anomaly_summary()	29
	get_fabrics_list()	30
	get_nodes_list()	30
	get_protocols_details()	31
	get_protocols_resources()	32

[get_protocols_topentities\(\)](#) 33

[get_protocols_topnodes\(\)](#) 34

[health_diagnostics\(\)](#) 35

[service_health\(\)](#) 35

[utilization_node_details\(\)](#) 36

[utilization_top_nodes\(\)](#) 37



CHAPTER 1

New and Changed Information

This chapter contains the following sections:

- [New and Changed Information, on page 1](#)

New and Changed Information

The following table provides an overview of the significant changes up to the current release. The table does not provide an exhaustive list of all changes or of the new features up to this release.

Table 1: New Features and Changed Behavior in the Cisco Network Insights for Resources application for Release 2.0.x

Feature	Description	Release
Cisco Network Insights for Resources Application	This guide was released to provide a description of Cisco Network Insights for Resources application.	2.0.2



CHAPTER 2

Cisco Network Insights for Resources Overview

This chapter contains the following sections:

- [About Cisco Network Insights for Resources, on page 3](#)
- [Downloading Cisco Network Insights for Resources from the Cisco App Center, on page 4](#)
- [Installing Cisco NIR Application in Cisco DCNM, on page 5](#)

About Cisco Network Insights for Resources

Cisco Network Insights for Resources (Cisco NIR) applications consist of monitoring utilities that can be added to the Cisco Data Center Network Manager (Cisco DCNM).

Hardware Requirements

This section describes the Cisco DCNM 11.2 LAN deployment requirements for Cisco NIA and Cisco NIR software telemetry. A Cisco DCNM-native HA deployment is recommended.

Table 2: Hardware Recommendations for Deployments up to 80 Switches

Node	Deployment Mode	CPU	Memory	Storage	Network
Cisco DCNM	OVA/ISO	16 vCPUs	32G	500G HDD	3x NIC
Computes (x3)	OVA/ISO	16 vCPUs	64G	500G HDD	3x NIC

Table 3: Hardware Recommendations for Deployments from 81 to 250 Switches

Node	Deployment Mode	CPU	Memory	Storage	Network
Cisco DCNM	OVA/ISO	16 vCPUs	32G	500G HDD	3x NIC
Computes (x3)	ISO	32 vCPUs	256G	2.4TB HDD	3x NIC*

* Network card: Quad-port 10/25G

Downloading Cisco Network Insights for Resources from the Cisco App Center

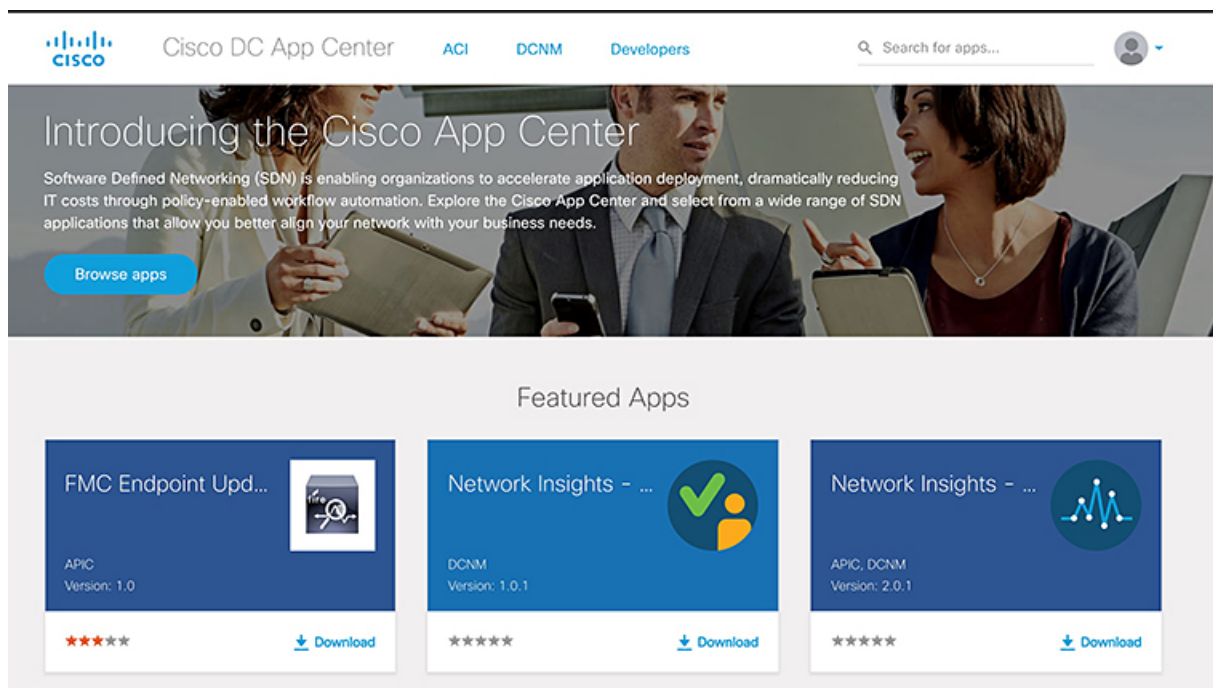
This section contains the steps required to download Cisco NIR app in the Cisco DCNM in preparation for installation.

Before you begin

You must have administrative credentials to download applications in Cisco DCNM.

-
- Step 1** Log in to the Cisco DCNM GUI with admin privileges.
- If you don't have admin privileges, you can log in to the [Cisco App Center](#) to download the application.
- Step 2** Choose **Applications**.
- Step 3** Click **Browse App Center** on the far-right side of the screen.
- Step 4** Search for Cisco Network Insights for Resources application on the search bar.
- Step 5** Select the Cisco Network Insights for Resources application you want to download and click **Download** for that app to begin the process of downloading the app to your local machine.

Figure 1: Cisco NIR in the DCNM-App Center Landing Page



- Step 6** Review the license agreement and, if OK, click **Agree and download**.
The Cisco NIR app is downloaded to your local machine.
-

What to do next

Make sure the following requirements are met:

- Note the download location of the Cisco NIR app file on your local machine.
- Make sure the downloaded file can be accessed by the Cisco DCNM. If it cannot, move the file to a device and/or location where it can be installed on the Cisco DCNM.

Installing Cisco NIR Application in Cisco DCNM

This section contains the steps required to install Cisco NIR app in the Cisco DCNM.

Before you begin

Before you begin installing a Cisco NIR app, make sure the following requirements are met:

Step 1

You must have administrator credentials to install Cisco NIR application.

Step 2

You must have three compute servers installed and in the “Joined” state. For more information regarding the installation, discovery, and addition of compute servers, refer to the following sections:

- **Compute Installation:** For details on compute installation, refer to the [Installing a DCNM Compute](#) section.
- **DVS Security Settings:** For details on DVS security settings, refer to the [Networking Policies for OVA Installation](#) section.
- **Subnet Requirements for OOB and IB pool:** For details on subnet requirements for OOB and IB pool, refer to the [Subnet Requirements](#) section.
- **Creating a Compute Cluster:** For details on creating a compute cluster, refer to the [Enabling the Compute Cluster](#) section.
- **Adding Computers in Web UI:** For details on adding computers in web UI, refer to the [Adding Computers into the Cluster Mode](#) section.

What to do next

When the installation is complete, the application opens to a Welcome dialog where initial setup is performed. Continue with the setup of the Cisco NIR app located in the Initial Setup section of the next chapter.



CHAPTER 3

Using Cisco Network Insights for Resources

This chapter contains the following sections:

- [Cisco Network Insights for Resources Components in Cisco DCNM, on page 7](#)
- [Guidelines and Limitations, on page 8](#)
- [Cisco NIR App Initial Setup, on page 8](#)
- [Cisco NIR App Settings, on page 11](#)
- [Navigating Cisco NIR, on page 13](#)
- [Using the Cisco Network Insights for Resources App, on page 16](#)

Cisco Network Insights for Resources Components in Cisco DCNM



The Cisco Network Insights for Resources (Cisco NIR) is a real-time monitoring and analytics application.

The Cisco NIR app consists of the following components:

- **Data Collection**—The streaming of telemetry data is done by the Operating System on the fabric nodes. As each data source is different and the format in which data is streamed is different, there are corresponding collectors running analytics that translate the telemetry events from the nodes into data records to be stored in the data lake. The data stored in the data lake is in a format that the analytics pipeline can understand and work upon.

The following telemetry information is collected from various nodes in the fabric:

- **Resources Analytics**—This includes monitoring software and hardware resources of fabric nodes on Cisco DCNM.
- **Environmental**—This includes monitoring environmental statistics such as fan, CPU, memory, and power of the fabric nodes.
- **Statistics Analytics**—This includes monitoring of nodes, interfaces, and protocol statistics on Cisco DCNM and fabric nodes.

- **Flow Analytics**—This includes detecting anomalies in the flow such as average latency, packet drop indication, and flow move indication across the fabric.
- **Resource Utilization and Environmental Statistics**—Resource analytics supports configuration, operational and hardware resources. Environmental covers CPU, memory, temperature, and fan speed. System analytics also covers, Anomalies, and trending information of each resource and graphing of parameters which help Network operators to debug over period of time.
- **Predictive Analytics and Correlation**—The value-add of this platform is predicting failures in the fabric and correlating internal fabric failures to the user-visible/interested failures.
- **Anomaly Detection**—Involves understanding the behavior of each component while using different machine learning algorithms and raising anomalies when the resource behavior deviates from the expected pattern. Anomaly detector applications use different supervised and unsupervised learning algorithms to detect the anomalies in the resources and they log the anomalies in an anomaly database.

Guidelines and Limitations

The following are the guidelines and limitations for the Cisco Network Insights for Resources (Cisco NIR) application in the Cisco Data Center Network Manager (Cisco DCNM):

- To enable telemetry on monitored fabric through Cisco NIR app, you must first delete all existing telemetry configurations on all the devices in the monitored fabric before you enable this fabric from Cisco NIR app. The telemetry then assigns the receiver IPs to these switches, which the Health page displays. The telemetry configuration will not push any telemetry configurations to the switches because they are monitored. Therefore, the user has to check the receiver IPs from the Health page and should configure the switches manually.
- IPv6 is not supported for Cisco NIR app.
- The Cisco NIR application requires that physical servers hosting Cisco DCNM computes as VMs are at least Cisco C220-M4 category. It is also required that a compute be hosted on a data store with a dedicated hard disk of at least 500GB.
- For instances where one or more fabrics do not recover from disabling state, you must stop and restart the Cisco NIR application in the Cisco DCNM. This will recover the failed disable state.

Cisco NIR App Initial Setup

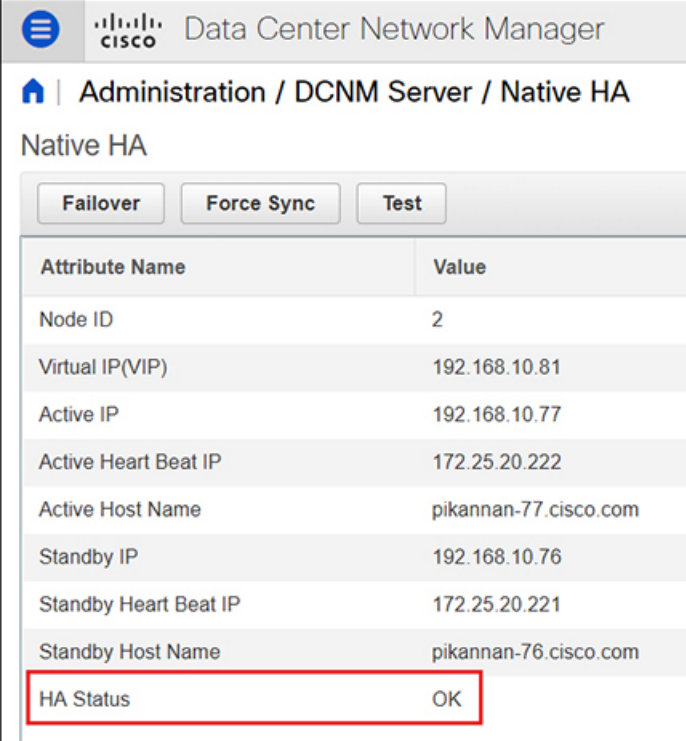
The first time you launch the Cisco NIR app, you are greeted with a **Welcome to Network Insights** dialog. Follow these steps to complete the initial setup of Cisco NIR app:

Before you begin

Before you begin the initial set up of the Cisco NIR application in the Cisco DCNM, make sure the following prerequisites are met:

- The primary and standby hosts (HA) return a status of OK:
 1. In Cisco DCNM, click **Administration**.

2. Under Cisco DCNM Server, click **Native HA**.
3. Check the **HA Status** attribute as shown in the following image:



Attribute Name	Value
Node ID	2
Virtual IP(VIP)	192.168.10.81
Active IP	192.168.10.77
Active Heart Beat IP	172.25.20.222
Active Host Name	pikannan-77.cisco.com
Standby IP	192.168.10.76
Standby Heart Beat IP	172.25.20.221
Standby Host Name	pikannan-76.cisco.com
HA Status	OK



Note It may take some time for both hosts to be recognized. Once the OK status is displayed, AMQP notifications can begin. Check the AMQP server status below.

- The AMQP Server returns a status of OK:
 1. In Cisco DCNM, click **Dashboard**.
 2. In the **Server Status** tile, click **Health Check**.
 3. Check the status of the **AMQP Server** component as shown in the following image:

Server Status		
DCNM	Health Check	
Component	Status	Details
AMQP Server 10.126.216.240	✔ Ok	Service responds to user access.
DHCP Server Local	✔ Ok	Service is running.
TFTP Server Local	✔ Ok	Service is running.
EPLS Local	✘ Down	Service is stopped.
EPLC Local	✘ Down	Service is stopped.

- Network Timing Protocol (NTP) must be configured on all switches you want to support with Cisco NIR. If the fabric is managed, Cisco DCNM will push respective NTP configurations to the switches. If fabric is in monitor mode, the user must ensure NTP is correctly configured on all switches in the fabric. To ensure Telemetry Network and NTP is setup correctly, refer to the following:
 - For details about Lan Fabric Deployments, refer to [Telemetry Network and NTP Requirements](#).
 - For details about Lan Classic Deployments, refer to [Telemetry Network and NTP Requirements](#).

Step 1 On the welcome dialog, click **Begin Set Up**.

The **Data Collection Set Up** window appears.

Step 2 In the list of available fabrics, choose a fabric you want to monitor with Cisco NIR.

Step 3 In the **VXLAN / Classic** column, choose the fabric type:

- **VXLAN**: Identifies the fabric as a VXLAN fabric type.

Note If your network is a VXLAN fabric and you want to see VXLAN-specific information in the Cisco NIR application, you must select this option.

- **Classic**: Identifies the fabric as a Classic LAN fabric.

Step 4 In the **Mode** column, choose the mode you want to use for the fabric selected:

- **Managed**: Cisco DCNM monitors and manages the configuration of the switches in the selected fabric. This option allows Cisco NIR app to push the telemetry configuration to the switches in the chosen fabric.
- **Monitored**: Cisco DCNM does not deploy configuration to the switches. Cisco DCNM discovers the switches and displays them in the topology (read-only). Cisco NIR app will not send telemetry configuration to the switches.

Note If this option is chosen, telemetry must be configured directly on the switches in order for Cisco NIR app to received data. The following configuration must be added on the NX-OS switches to stream telemetry data to Cisco NIR app when the fabric is configured to be in *Monitored* mode:

Example:

```
configure terminal

feature nxapi
feature icam
```

```

feature telemetry

telemetry
  destination-group 500
    ip address <IP address of port 57500 protocol gRPC encoding GPB
  sensor-group 500
    data-source NX-API
    path "show environment temperature" depth unbounded
    path "show interface" depth unbounded
    path "show interface transceiver details" depth unbounded
    path "show ip mroute summary vrf all" depth unbounded
    path "show ip route summary vrf all" depth unbounded
    path "show ipv6 mroute summary vrf all" depth unbounded
    path "show ipv6 route summary vrf all" depth unbounded
    path "show lacp counters detail" depth unbounded
    path "show lacp interface" depth unbounded
    path "show lldp all" depth unbounded
    path "show lldp neighbors" depth unbounded
    path "show lldp traffic interface all" depth unbounded
    path "show mac address-table count" depth unbounded
    path "show module" depth unbounded
    path "show nve peers detail" depth unbounded
    path "show nve vni" depth unbounded
    path "show nve vrf" depth unbounded
    path "show processes cpu" depth unbounded
    path "show processes log" depth unbounded
    path "show processes memory physical" depth unbounded
    path "show system internal icam app hardware internal forwarding table utilization"
      depth unbounded query-condition show-output-format=json
    path "show system internal icam app system internal access-list resource utilization"
      depth unbounded query-condition show-output-format=json
    path "show system resources" depth unbounded
    path "show vlan summary" depth unbounded
    path "show vrf all" depth unbounded
  sensor-group 501
    data-source NX-API
    path dir depth unbounded
    path "show clock" depth unbounded
    path "show environment fan detail" depth unbounded
    path "show environment power" depth unbounded sensor-group 502 data-source NX-API
    path "show system routing mode" depth unbounded subscription 500 dst-grp 500
  snsr-grp 500 sample-interval 60000
  snsr-grp 501 sample-interval 300000
  snsr-grp 502 sample-interval 3600000

```

Step 5 Click **Save**.


Step 6 Click **Done**.

Cisco NIR App Settings

Once Cisco NIR app is installed, the following need to be checked off for the application to be fully set up:

- NTP and Time Zone Configuration

If there are Faults present in the application, they will show on the Faults tab. In the **Settings** menu click **Collection Status**, you should see the green circles in the table indicating the nodes where information is being transmitted.

Property	Description
Time Range	Specify a time range and the tables below display the data that is collected during the specified interval.
Fabric	Choose a fabric containing the switches from which to collect telemetry data.
	<p>Clicking on this icon allows you to alter the following:</p> <ul style="list-style-type: none"> • System Status—Displays software, hardware, operational, and other capacity usage of the Cisco NIR application on the compute cluster. • Collection Status—Displays data collection of System Metrics, and Events information per node. • Rerun Set Up—Allows you to go back to the Data Collection Set Up check list. • About Network Insights—Resources—Displays the application version number.

Cisco NIR Service Instance Status

To view Cisco NIR app service instance status, exit the Cisco NIR app and click the gear image in the lower left corner of the Cisco NIR app icon in the Cisco DCNM application work pane.

The **Application Specifications** dialog lists all running service instances for the Cisco NIR app:

Application Specifications ✕

Info Spec

Running Instance Info ↻

Container Name	Compute	East-West IP	Fabric IP
eventcollector_Cis...	pikannan-78.cisc...	192.168.2.30	
eventcollector_Cis...	pikannan-79.cisc...	192.168.2.31	
eventcollector_Cis...	pikannan-80.cisc...	192.168.2.32	
postprocessor_Cis...	pikannan-78.cisc...	192.168.2.34	
postprocessor_Cis...	pikannan-79.cisc...	192.168.2.35	
postprocessor_Cis...	pikannan-80.cisc...	192.168.2.36	
utr_Cisco_afw.3	pikannan-80.cisc...	192.168.2.40	172.25.20.243
utr_Cisco_afw.2	pikannan-79.cisc...	192.168.2.39	172.25.20.242
utr_Cisco_afw.1	pikannan-78.cisc...	192.168.2.38	172.25.20.241
apiserver_Cisco_a...	pikannan-79.cisc...	192.168.2.43	
apiserver_Cisco_a...	pikannan-78.cisc...	192.168.2.42	
apiserver_Cisco_a...	pikannan-80.cisc...	192.168.2.44	
scheduler_Cisco_...	pikannan-79.cisc...	192.168.2.46	
predictor_Cisco_af...	pikannan-80.cisc...	192.168.2.48	
correlator_Cisco_...	pikannan-79.cisc...	192.168.2.50	

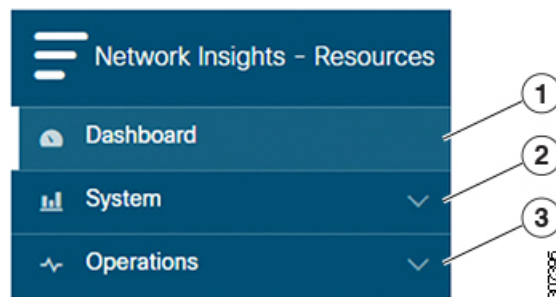
307447

Navigating Cisco NIR

The Cisco NIR app window is divided into two parts: the Navigation pane and the Work pane.

Navigation Pane

The Cisco NIR app navigation pane divides the collected data into three categories:

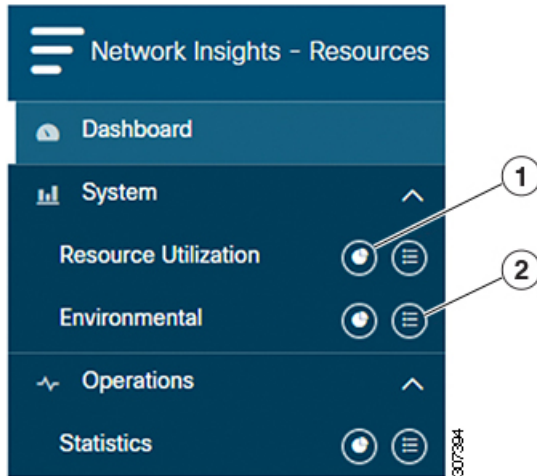


1 Dashboard: The main dashboard for the Cisco NIR app providing immediate access to anomalies.

2 System: Resource and environmental utilization.

3 Operations: Statistics information for interfaces and protocols.

Expanding System and/or Operations reveals additional functions:



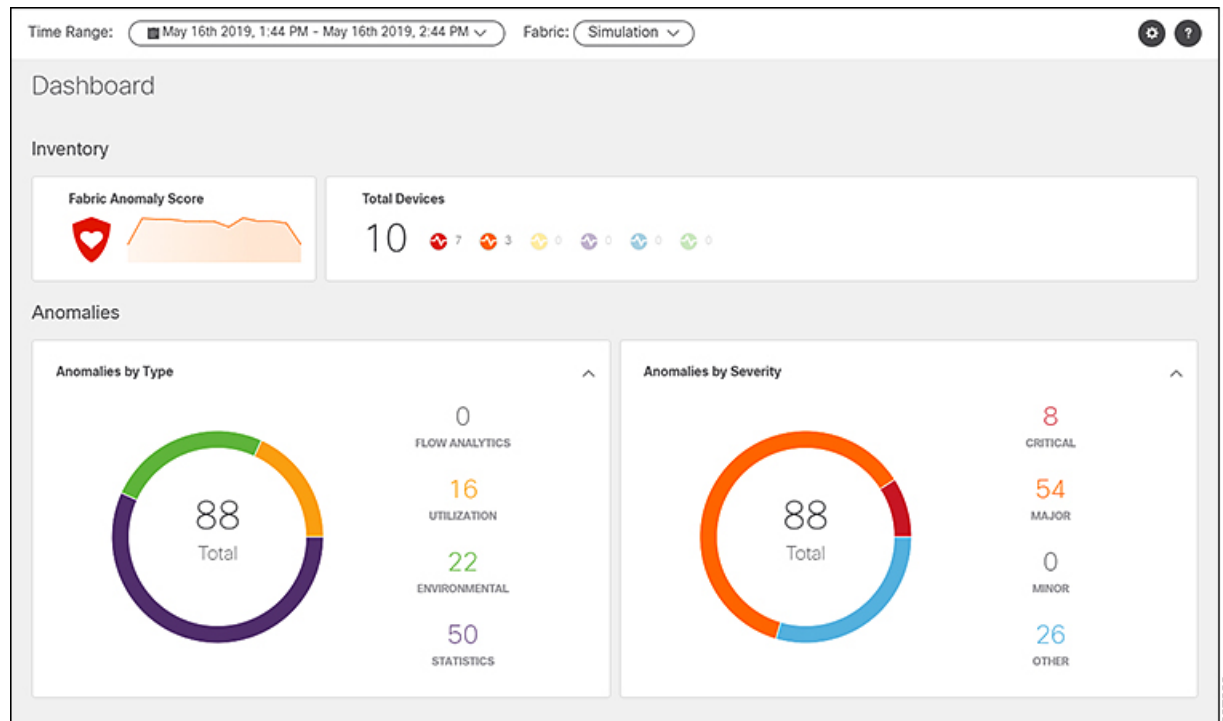
1 Dashboard View icon: Provides immediate access to top usage or issues for the selected telemetry type.

2 Browse View icon: Provides a detailed view of returned data for the selected telemetry type and allows for filtering to further isolate problem areas.

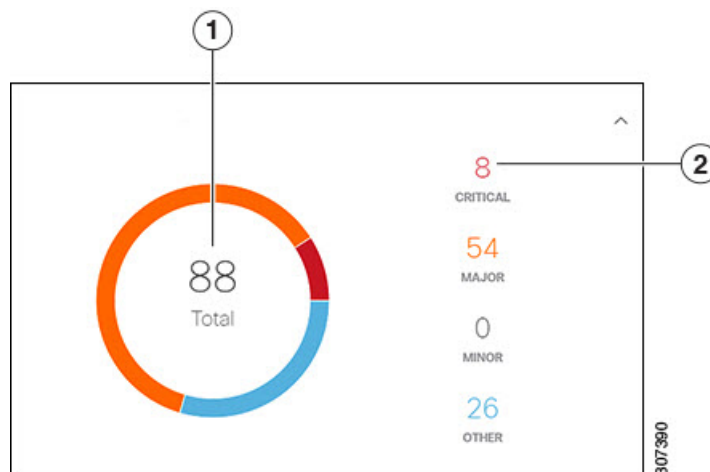
Work Pane

The work pane is the main viewing location in the Cisco NIR app. All information tiles, graphs, charts, and lists appear in the work pane.

Dashboard Work Pane



In an information tile, you can usually click on a numeric value to switch to the Browse work pane:






1 Launches the Browse work pane with all of the items displayed from the graph in the information tile.

2 Launches the Browse work pane with only the selected items displayed from the number in the information tile.

Browse Work Pane

The Browse work pane isolates the data for the parameter chosen on the Dashboard. The Browse work pane displays a top node lists, graphs over time, and lists all the nodes in an order defined by the anomaly score:

Start Time	End Time	Severity ^	Resource Type	Nodes	Description
May 16 2019 12:14:25pm	May 16 2019 07:54:37pm	 Critical	config	N9Kv-2	Number of VRFs is above critical threshold (Usage : 991, Critical-Threshold : 900)
May 16 2019 12:14:53pm	May 16 2019 07:55:08pm	 Critical	environmental	N9Kv-7	[Outlet Sensor] : Temperature is above critical threshold (Current Value : 75 C, Critical-Threshold : 72 C)
May 16 2019 12:14:17pm	May 16 2019 07:54:28pm	 Critical	environmental	N9Kv-1	[Outlet Sensor] : Temperature is above critical threshold (Current Value : 75 C, Critical-Threshold : 72 C)

307381

Clicking on one of the nodes in the list opens the Details work pane for that selection.

Details Work Pane

The Details work pane provides resource details about the item selected in the event list on the Browse work pane. The Details work pane consists of:

- General Information: Includes the anomaly score and the node name.
- Resource Trends: Includes operational resources, configuration resources, and hardware resources.
- Anomalies: Includes all anomalies for the node resource.

Using the Cisco Network Insights for Resources App

Each Cisco switch streams telemetry events from the fabric to an external service that analyzes the events and proactively detects issues in the fabric behavior. Use the dashboards in the Cisco NIR app to view relevant information and select specific items to view details.

Main Dashboard

The Cisco Network Insights for Resources (Cisco NIR) application main dashboard provides immediate access to anomalies occurring in the network. Anomalies are learned deviations from the last known "good" state of a switch and are displayed by type and severity. Anomalies include resource utilization, environmental, and interface-level errors, and are color coded based on severity: Critical: Red, Major: Orange, Minor: Yellow, Warning: Green, Information: Blue.

Some factors that contribute to the presence of an anomalies are exceeded thresholds and excessive rates of change.

Inventory

Property	Description
Fabric Anomaly Score	Displays the health of the fabric through color.
Total Devices	Displays the top number of devices with anomalies. The types are Leaf or Spine.

Anomalies

Click on any number to access the Browse Anomalies work pane.

Property	Description
Anomalies by Type	Displays the number of Anomalies by their type.

Property	Description
Anomalies by Severity	Displays the number of Anomalies (internal Fabric failures) and their severity level. Clicking on the area shows detail fault information, such as Node and Anomaly Score .

Browse Anomalies

View, sort, and filter anomalies through the Browse Anomalies work pane.

Filters

You can refine the displayed anomalies by the following filters:

- Start Time - display only anomalies with a specific start time.
- End Time - display only anomalies with a specific end time.
- Description -
- Node - display only anomalies for specific nodes.
- Category - display only anomalies from a specific category.
- Resource Type - display only anomalies of a specific resource type.
- Severity - display only anomalies of a specific severity.

As a secondary filter refinement, use the following operators:

- == - with the initial filter type, this operator, and a subsequent value, returns an exact match.
- != - with the initial filter type, this operator, and a subsequent value, returns all that do not have the same value.
- contains - with the initial filter type, this operator, and a subsequent value, returns all that contain the value.
- !contains - with the initial filter type, this operator, and a subsequent value, returns all that do not contain the value.

Property	Description
Start Time	The start time stamp for the anomaly detection.
End Time	The end time stamp for the anomaly detection.

Property	Description
Severity	<p>The current severity level of the event. The levels are:</p> <ul style="list-style-type: none"> • Critical—A service-affecting condition that requires immediate corrective action. For example, this severity could indicate that the managed object is out of service and its capability must be restored. • Major—Serious problems exist with one or more components. These issues should be researched and fixed immediately. • Minor—Problems exist with one or more components that might adversely affect system performance. These issues should be researched and fixed as soon as possible before they become a critical problem. • Other—Potential problems exist with one or more components that might adversely affect system performance if they are allowed to continue. These issues should be researched and fixed as soon as possible before they become a critical problem.
Resource Type	The resource type such as Flow , configuration, or operational.
Nodes	The node where the anomaly occurred.
Description	Additional information about the anomaly.

System

The System section of the NIR application contains two areas of data collection:

- **Resource Utilization**—Fabric component capacity information.
- **Environmental**—Hardware component capacity information.

Resource Utilization Dashboard

The Resource Utilization dashboard displays utilization, rate of change, trends, and resource anomalies over time for operational, configuration and hardware resources. Top leaf and spine nodes are displayed based on the factors that produced the high utilization.

Property	Description
Top Nodes by Capacity	The leaf node observations search can be more refined by filtering the information by the top leaf nodes.
Node Details	<p>Displays the node trend observations by resource type:</p> <ul style="list-style-type: none"> • Operational Resources • Configuration Resources • Hardware Resources

Browse Resource Utilization

View, sort, and filter statistics through the Browse Resource Utilization work pane.

Filters

You can refine the displayed statistics by the following filters:

- Node - display only nodes.

As a secondary filter refinement, use the following operators:

- = - with the initial filter type, this operator, and a subsequent value, returns an exact match.
- != - with the initial filter type, this operator, and a subsequent value, returns all that do not have the same value.
- contains - with the initial filter type, this operator, and a subsequent value, returns all that contain the value.
- !contains - with the initial filter type, this operator, and a subsequent value, returns all that do not contain the value.

Property	Description
Top Nodes by	Displays the top nodes by: <ul style="list-style-type: none"> • MAC • IPv4 Host Routes • IPv6 Host Routes • IPv4 Prefix Routes • IPv6 Prefix Routes • Multicast Routes • VLAN • VRF • Port Usage • Ingress Port Bandwidth • Egress Port Bandwidth • CoPP • LPM • HRT • L2 QoS TCAM • L3 QoS TCAM • VTEP • VNI L2 • VNI L3 • VLAN • Ingress VLAN ACL • Egress VLAN ACL • Ingress Port ACL • Ingress Routed ACL • Egress Routed ACL

Property	Description
Operational Resources	Displays a list of operational resources based on anomaly score. List information includes: <ul style="list-style-type: none">• Anomaly Score• Node• MAC• IPv4 Host Routes• IPv6 Host Routes• IPv4 Prefix Routes• IPv6 Prefix Routes• Multicast Routes
Configuration Resources	Displays a list of configuration resources based on anomaly score. List information includes: <ul style="list-style-type: none">• Anomaly Score• Node• VLAN• VTEP• VNI<ul style="list-style-type: none">• L2• L3• VRF

Property	Description
Hardware Resources	<p>Displays a list of configuration resources based on anomaly score. List information includes:</p> <ul style="list-style-type: none"> • Anomaly Score • Node • Port Usage • Port Bandwidth • CoPP • LPM • HRT • QoS TCAM <ul style="list-style-type: none"> • L2 • L3 • VLAN ACL <ul style="list-style-type: none"> • Ingress • Egress • Port ACL <ul style="list-style-type: none"> • Ingress • Egress • Routed ACL <ul style="list-style-type: none"> • Ingress • Egress

Environmental Dashboard

The Environmental Dashboard displays utilization, rate of change, trends, and anomalies over time for switch environmental resources such as fans, power, CPU, and memory.

Property	Description
Top Nodes by Utilization	Displays the percentage utilized per component: <ul style="list-style-type: none"> • CPU • Memory • Temperature • Fan Utilization • Power Supply • Storage
Node Details	Displays the node trend observations by environmental resource type:

Browse Environmental Resources

View, sort, and filter statistics through the Browse Environmental Resources work pane.

Filters

You can refine the displayed statistics by the following filters:

- Node - display only nodes.

As a secondary filter refinement, use the following operators:

- == - with the initial filter type, this operator, and a subsequent value, returns an exact match.
- != - with the initial filter type, this operator, and a subsequent value, returns all that do not have the same value.
- contains - with the initial filter type, this operator, and a subsequent value, returns all that contain the value.
- !contains - with the initial filter type, this operator, and a subsequent value, returns all that do not contain the value.

Property	Description
Top Nodes by	Displays the top nodes by: <ul style="list-style-type: none"> • CPU (percent utilization) • Memory (percent utilization) • Temperature • Fan Utilization • Power Supply • Storage

Operations

The Operations section of the NIR application contains statistic information.

Statistics Dashboard

The Statistics Dashboard displays top switch nodes by interface errors or usage, and protocol statistics.

Property	Description
Top Nodes by Interface Utilization	Displays the top nodes based on the combined bandwidth utilization of it's interfaces.
Top Nodes by Interfaces	Displays the top nodes and lists the transmit and receive bandwidth utilization of each of it's interfaces.

Browse Statistics

View, sort, and filter statistics through the Browse Statistics work pane.

Filters

You can refine the displayed statistics by the following filters:

- Node - display only nodes.
- Interface - display only interfaces.
- Protocol - display only protocols.

As a secondary filter refinement, use the following operators:

- == - with the initial filter type, this operator, and a subsequent value, returns an exact match.
- != - with the initial filter type, this operator, and a subsequent value, returns all that do not have the same value.
- contains - with the initial filter type, this operator, and a subsequent value, returns all that contain the value.
- !contains - with the initial filter type, this operator, and a subsequent value, returns all that do not contain the value.

Property	Description
Top 10 Interfaces by	Displays the top interfaces by: <ul style="list-style-type: none"> • Transmit Utilization • Receive Utilization • Error

Property	Description
Interface Statistics	Displays a list of interface statistics based on anomaly score. List information includes: <ul style="list-style-type: none">• Anomaly Score• Interface• Node• Receive Utilization• Transmit Utilization• Errors
Protocol Statistics	Displays a list of protocol statistics based on anomaly score. List information includes: <ul style="list-style-type: none">• Node• Protocol• Number of Interfaces• Errors



Note In order for the Cisco NIR app to receive data from the switches, confirm that all the devices in the fabric are synced.



CHAPTER 4

Cisco NIR DCNM REST API Examples

This chapter contains the following sections:

- [all_resources\(\)](#), on page 27
- [anomalies_details\(\)](#), on page 28
- [anomalies_summary\(\)](#), on page 29
- [get_fabrics_anomaly_summary\(\)](#), on page 29
- [get_fabrics_list\(\)](#), on page 30
- [get_nodes_list\(\)](#), on page 30
- [get_protocols_details\(\)](#), on page 31
- [get_protocols_resources\(\)](#), on page 32
- [get_protocols_topentities\(\)](#), on page 33
- [get_protocols_topnodes\(\)](#), on page 34
- [health_diagnostics\(\)](#), on page 35
- [service_health\(\)](#), on page 35
- [utilization_node_details\(\)](#), on page 36
- [utilization_top_nodes\(\)](#), on page 37

all_resources()

```
Get all resources .
REST URL   :
            GET /api/telemetry/utilization/resources.json
Parameters :
            None
Example    :
            curl -k -i -XGET
            'https://<ip:port>/appcenter/Cisco/NIR/apiserver-api/api/telemetry/utilization/resources.json'
Response   :
            {
              "totalResultsCount": 5,
              "totalItemsCount":5,
              "entries": [
                {
                  "categoryName": "",
                  "resourceName": "EndPoints",
                }
              <-- SNIP LIST OF ALL OTHER RESOURCES -->
            }
```

```
    ]
  }
```

anomalies_details()

Get the anomalies in the system

REST URL :

```
GET /api/telemetry/anomalies/details.json
```

Parameters :

```
startTs (optional) => Start timestamp, default:now-1h
endTs (optional) => End timestamp, default:current-time
count (optional) => Num.of nodes in response, default:10
orderBy (optional) => Sort per the given field
```

Example :

```
curl -ksb -XGET
```

```
'https://<ip:port>/appcenter/Cisco/NIR/apiserver-api/api/telemetry/anomalies/details.json'
```

Response :

```
{
  "totalItemsCount": 90,
  "totalResultsCount": 90,
  "offset": 0,
  "entries": [
    {
      "anomalyId": "QUE0000000000018",
      "category": "System Resource",
      "startTs": "2018-09-19T16:45:05.679Z",
      "endTs": "2018-09-19T16:58:05.778Z",
      "entityName": "svc_ifc_policyelem",
      "severity": "critical",
      "anomalyType": "build-up",
      "nodeNames": [
        "leaf2"
      ],
      "resourceType": "queue",
      "resourceName": "recvQ",
      "anomalyStr": "[svc_ifc_policyelem] : Unexpected build-up of 7487 message[s]
in recvQ",
      "anomalyScore": 83
    },
    {
      "anomalyId": "QUE0000000000007",
      "category": "System Resource",
      "startTs": "2018-09-19T15:16:10.420Z",
      "endTs": "2018-09-19T16:49:01.289Z",
      "entityName": "svc_ifc_policyelem",
      "severity": "critical",
      "anomalyType": "build-up",
      "nodeNames": [
        "leaf1"
      ],
      "resourceType": "queue",
      "resourceName": "recvQ",
      "anomalyStr": "[svc_ifc_policyelem] : Unexpected build-up of 7502 message[s]
in recvQ",
      "anomalyScore": 83
    }
  ]
}
```


anomalies_summary()

```

Get summary of the anomalies in the system
REST URL   :
            GET /api/telemetry/anomalies/summary.json
Parameters :
            startTs (optional) => Start timestamp, default:now-1h
            endTs   (optional) => End timestamp, default:current-time
Example    :
            curl -ksb -XGET
            'https://<ip:port>/appcenter/Cisco/NIR/apiserver-api/api/telemetry/anomalies/summary.json'
Response   :
            {
              "totalAnomalyCount": 2,
              "totalAnomalyScore": 120.0,
              "entries": [
                {
                  "severity": "warning",
                  "anomalyCount": 1,
                  "anomalyScore": 40.0
                },
                {
                  "severity": "major",
                  "anomalyCount": 1,
                  "anomalyScore": 80.0
                }
              ]
            }

```

get_fabrics_anomaly_summary()

```

Get fabric anomaly summary.
REST URL   :
            GET /api/telemetry/fabricsSummary.json
Parameters :
            fabricName (mandatory) => Name of the Fabric
            startTs     => Start timestamp, default:current-time - 1 hour
            endTs       => End timestamp, default:current-time
            include="anomalyScore" => Requires the Latest Maximum anomaly scores of the fabric,
            default:'no'
            history     => Requires the timeseries data of sum(anomaly scores, default:'no'

            granularity => applicable if history = "yes" , granulairy of the timeseries
            data, default=5m
Example    :
            curl -k -i -XGET
            'https://<ip:port>/appcenter/Cisco/NIR/apiserver-api/api/telemetry/fabricsSummary.json'
Response   :
            {
              "anomalyScore" : "X"
              "entries": [
                {
                  totalAnomalyScore ; X
                  ts : now
                }
                .....
                {
                  totalAnomalyScore ; X
                  ts : now
                }
              ]
            }

```

```

    },
    "totalResultsCount": N,
    "totalItemsCount": N
  }
}

```

get_fabrics_list()

```

Get fabrics list.
REST URL   :
  GET /api/telemetry/fabrics.json
Parameters :
  filter           => Lucene format filter, default:null
Example     :
  curl -k -i -XGET
'https://<ip:port>/appcenter/Cisco/NIR/apiserver-api/api/telemetry/fabrics.json'
Response    :
  {
    "entries": [
      {
        "fabricName": "FABRIC1",
        "fabricId": "1",
        "vendor": "CISCO_N9K_STANDALONE",
        "fabricType": "VXLAN",
        "configStatus": "ENABLED",
        "switchCount": 2,
        "controllerCount": 0
      },
      {
        "fabricName": "FABRIC2",
        "fabricId": "2",
        "vendor": "CISCO_ACI",
        "fabricType": "VXLAN",
        "configStatus": "ENABLED",
        "switchCount": 4,
        "controllerCount": 3
      }
    ],
    <--snip-->
    "totalResultsCount": 11,
    "totalItemsCount": 11
  }

```

get_nodes_list()

```

Get nodes list.
REST URL   :
  GET /api/telemetry/nodes.json
Parameters :
  startTs (mandatory) => Start timestamp
  endTs           => End timestamp, default:current-time
  count           => Num.of nodes in response, default:1000
  filter           => Lucene format filter, default:null
Example     :
  curl -k -i -XGET
'https://<ip:port>/appcenter/Cisco/NIR/apiserver-api/api/telemetry/nodes.json'
Response    :
  {
    "entries": [

```

```

    {
      "nodeRole": "leaf",
      "nodeId": "302",
      "nodeName": "rleaf-scrimshaw2",
      "nodeMgmtIp": "1.2.3.4"
    },
    {
      "nodeRole": "spine",
      "nodeId": "205",
      "nodeName": "swmp14-dopplebock",
      "nodeMgmtIp": "1.2.3.4"
    },
    <--snip-->
  ],
  "totalResultsCount": 11,
  "offset": 0,
  "totalItemsCount": 11
}

```

get_protocols_details()

Get Telemetry Protocol Stats details.

REST URL :
GET /api/telemetry/protocols/details.json

Parameters :

- startTs (mandatory) => Start timestamp
- endTs => End timestamp, default:current-time
- fabricName => limit the records pertaining to this fabricName
- nodeName => Name of node
- statName => <protocol[:counter[:qualifier]], protocol[:counter[:qualifier]]...>
- history => '1' or '0', default is '0', indicates time-series request
- granularity => Granularity of time period, default:5m
- orderBy => One statName of the format <protocol[:counter[:qualifier]]>
- filter => Lucene format filter to query for specific nodeName or sourceName, default:null

Example :
curl -k -i -XGET 'https://<ip:port>/appcenter/Cisco/NIR/apiserver-api/api/telemetry/protocols/details.json'

Response :

```

{
  "totalResultsCount": 6,
  "totalItemsCount": 6,
  "offset": 0,
  "description": "Protocol statistical counters",
  "entries": [
    {
      "nodeName": "leaf-103",
      "entries": [
        {
          "sourceName": "phys-[eth1/14]",
          "entries": [
            {
              "counterName": "InterfaceUtilisationIngress",
              "value": 60.625,
              "trending": "up",
              "stats": [
                {
                  "ts": "2018-10-24T05:05:00.000Z",
                  "value": 60.625
                },
                {

```

get_protocols_resources()

```

        "ts": "2018-10-24T05:00:00.000Z",
        "value": 59.827586206896555
      },
      {
        "ts": "2018-10-24T04:55:00.000Z",
        "value": 59.57142857142857
      }
    ]
  }
},
<--snip-->
{
  "sourceName": "phys-[eth1/11]",
  "entries": [
    {
      "counterName": "LldpPktsEgress",
      "value": 111.0,
      "trending": "up",
      "stats": [
        {
          "ts": "2018-10-24T05:05:00.000Z",
          "value": 111.0
        },
        {
          "ts": "2018-10-24T05:00:00.000Z",
          "value": 110.10344827586206
        },
        {
          "ts": "2018-10-24T04:55:00.000Z",
          "value": 109.61904761904762
        }
      ]
    }
  ]
}

```

get_protocols_resources()

```

Get Telemetry Protocol Stats resources.
REST URL   :
  GET /api/telemetry/protocols/resources.json
Parameters :
  filter           => Lucene format filter, default:null
  fabricName      => limit the records pertaining to this fabricName
Example    :
  curl -k -i -XGET
'https://<ip:port>/appcenter/Cisco/NIR/apiserver-api/api/telemetry/protocols/resources.json'
Response   :
  [
    {
      "protocol": "interface",
      "counter": "utilisation",
      "qualifiers": [
        "ingress",
        "egress"
      ]
    }
  ],

```

```

    {
      "protocol": "interface",
      "counter": "bytes",
      "qualifiers": [
        "ingress",
        "egress"
      ]
    },
<--snip-->
    {
      "protocol": "lldp",
      "counter": "pkts",
      "qualifiers": [
        "ingress",
        "egress"
      ]
    },
    {
      "protocol": "lldp",
      "counter": "errors"
    }
  ]
}

```

get_protocols_topentities()

```

Get Telemetry Protocol Stats topEntities.
REST URL   :
            GET /api/telemetry/protocols/topEntities.json
Parameters :
  startTs   (mandatory) => Start timestamp
  endTs     => End timestamp, default:current-time
  fabricName => limit the records pertaining to this fabricName
  statName  => parameter to find topEntities protocol[:counter[:qualifier]]
  granularity => Granularity of time period, default:5m
  filter    => Lucene format filter to query for specific nodeName or sourceName,
            default:null
Example    :
            curl -k -i -XGET
            'https://<ip:port>/appcenter/Cisco/NIR/apiserver-api/api/telemetry/protocols/topEntities.json'
Response   :
  {
    "totalResultsCount": 6,
    "totalItemsCount": 6,
    "offset": 0,
    "description": "Protocol statistical counters",
    "entries": [
      {
        "nodeName": "leaf-103",
        "entries": [
          {
            "sourceName": "phys-[eth1/4]",
            "entries": [
              {
                "counterName": "InterfaceUtilisationIngress",
                "value": 65.53333333333333,
                "trending": "down",
                "stats": [
                  {
                    "ts": "2018-10-24T05:20:00.000Z",
                    "value": 65.53333333333333
                  },
                  {

```

get_protocols_topnodes()

```

        "ts": "2018-10-24T05:15:00.000Z",
        "value": 65.78571428571429
      }
    ]
  },
  {
    "sourceName": "phys-[eth1/14]",
    "entries": [
      {
        "counterName": "InterfaceUtilisationIngress",
        "value": 59.666666666666664,
        "trending": "up",
        "stats": [
          {
            "ts": "2018-10-24T05:20:00.000Z",
            "value": 59.666666666666664
          },
          {
            "ts": "2018-10-24T05:15:00.000Z",
            "value": 59.5
          }
        ]
      }
    ]
  }
],
},
<--snip-->
]
}
}

```

get_protocols_topnodes()

Get Telemetry Protocol Stats topNodes.

REST URL :
GET /api/telemetry/protocols/topNodes.json

Parameters :

- startTs (mandatory) => Start timestamp
- endTs => End timestamp, default:current-time
- fabricName => limit the records pertaining to this fabricName
- nodeName => Name of node
- statName => interface:utilization
- summarize => '1' or '0', default is '0', summarizes across protocols

Example :

```
curl -k -i -XGET 'https://<ip:port>/appcenter/Cisco/NIR/apiserver-api/api/telemetry/protocols/topNodes.json'
```

Response :

```

{
  "totalResultsCount": 6,
  "totalItemsCount": 6,
  "offset": 0,
  "description": "Protocol top nodes by score",
  "entries": [
    {
      "nodeName": "leaf-103",
      "entries": [
        {
          "counterName": "protocol|utilization",
          "stats": [

```

```

        "ts": "2019-02-08T13:50:00.000Z",
        "value": 62.333333333333336
      },
      {
        "ts": "2019-02-08T13:45:00.000Z",
        "value": 62.833333333333336
      }
    ],
    "value": 62.333333333333336,
    "trending": "down"
  }
},
]
....
}

```

health_diagnostics()

```

Get health dianostics.
REST URL   :
            GET /api/telemetry/health/collectionStats.json
Parameters :
            None
Example    :
            curl -k -i -XGET
            'https://<ip:port>/appcenter/Cisco/NIR/apiserver-api/api/telemetry/health/collectionStats.json'
Response   :
            {
              "totalItemsCount": 11,
              "entries": [
                {
                  "nodeName": "pod20-leaf3",
                  "stats": [
                    {
                      "resource": "sysStats",
                      "totalItemsCount": 9600,
                      "lastUpdatedTs": "2018-06-13T10:25:52.468Z",
                      "state": "HEALTHY"
                    }
                  ]
                },
                <---snip-->
              ]
            }

```

service_health()

```

Get the health of the services
REST URL   :
            GET /api/telemetry/health/serviceHealth.json
Parameters :
            None
Example    :
            curl -k -i -XGET
            'https://<ip:port>/appcenter/Cisco/NIR/apiserver-api/api/telemetry/health/serviceHealth.json'
Response   :
            {
              "entries": [
                {

```

```

        "serviceType": "THIRD_PARTY_SERVICE",
        "serviceName": "elastic",
        "state": "HEALTHY",
        "displayName": "Data Store"
    },
    {
        "serviceType": "CISCO_SERVICE",
        "serviceName": "correlator",
        "state": "HEALTHY",
        "displayName": "Correlator"
    },
    <--snip-->
]
}

```

utilization_node_details()

```

Get node details .
REST URL      :
    GET /api/telemetry/utilization/nodeDetails.json
Parameters    :
    None
Example       :
    curl -k -i -XGET
'https://<ip:port>/appcenter/Cisco/NIR/apiserver-api/api/telemetry/utilizationnodeDetails.json'
Response      :
    {
        "totalResultsCount": 157,
        "totalItemsCount":157,
        "entries": [
            {
                "nodeName": "node-1",
                "entries": [
                    {
                        "resourceName":"cpu",
                        "latestValue":"85",
                        "maxValue":"100",
                        "resourceCategory":"","
                        "trending":"down",
                        "values":[
                            { "value":"85", "ts":"2018-02-21T20:21:03.109Z" },
                            {},
                            <--snip-->
                            {}
                        ]
                    },
                    {
                        "resourceName":"memory",
                        "latestValue":"84",
                        "maxValue":"100",
                        "resourceCategory":"","
                        "trending":"up",
                        "values":[
                            { "value":"84", "ts":"2018-02-21T20:21:03.109Z" },
                            {},
                            <--snip-->
                            {}
                        ]
                    },
                    <-- snip , LIST OF ALL OTHER RESOURCES -->
                    {
                        "resourceName":"ports",

```



```

        "latestValue": "83",
        "maxValue": "100",
        "resourceCategory": "",
        "trending": "up",
        "values": [
            { "value": "83", "ts": "2018-02-21T20:21:03.109Z" },
            {},
            <--snip-->
            {}
        ]
    }
}
],
{
    "nodeName": "node-2"
    <-- same as in node-1 -->
}
<----snip LIST OF ALL OTHER NODES ---->
{
    "nodeName": "node-10"
    <-- same as in node-1 -->
}
]
}

```

utilization_top_nodes()

Get top nodes by utilization .

REST URL :

GET /api/telemetry/utilization/topNodes.json

Parameters :

None

Example :

curl -k -i -XGET

'https://<ip:port>/appcenter/Cisco/NIR/apiserver-api/api/telemetry/utilization/topNodes.json'

Response :

```

{
    "totalResultsCount": 10,
    "totalItemsCount": 10,
    "entries": [
        {
            "nodeName": "node-1",
            "entries": [
                {
                    "resourceName": "cpu",
                    "latestValue": "85",
                    "maxValue": "100",
                    "resourceCategory": "",
                    "trending": "down",
                    "values": [
                        { "value": "85", "ts": "2018-02-21T20:21:03.109Z" },
                        {},
                        <--snip-->
                        {}
                    ]
                },
                {
                    "resourceName": "memory",
                    "latestValue": "84",
                    "maxValue": "100",
                    "resourceCategory": "",
                    "trending": "up",

```

```

        "values":[
          { "value":"84", "ts":"2018-02-21T20:21:03.109Z" },
          {},
          <--snip-->
          {}
        ]
      },
      {
        "resourceName":"ports",
        "latestValue":"83",
        "maxValue":"100",
        "resourceCategory":"",
        "trending":"up",
        "values":[
          { "value":"83", "ts":"2018-02-21T20:21:03.109Z" },
          {},
          <--snip-->
          {}
        ]
      }
    ]
  },
  {
    "nodeName": "node-2"
    <-- same as in node-1 -->
  }
  <----snip---->
  {
    "nodeName": "node-10"
    <-- same as in node-1 -->
  }
]
}

```