

Cisco Ultra Cloud Core CPC AAA- Metrics Reference, Release 2026.04

First Published: 2026-11-05

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

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED “AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: https://www.cisco.com/c/en/us/about/legal/trademarks.html. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2026 Cisco Systems, Inc. All rights reserved.



CONTENTS

Full Cisco Trademarks with Software License ?

PREFACE

About this Guide vii

Conventions Used vii

Contacting Customer Support viii

CHAPTER 1

Promethus and Grafana 1

Feature Summary 1

Summary Data 1

Feature Description 1

How it Works 2

Automatic site failover KPIs 2

Managing the cnAAA Statistics 3

Viewing the Statistics 3

Accessing the Grafana dashboard 4

Viewing the cnAAA Dashboard 4

Running a Query in Grafana 4

Configuring Autorefresh 5

Exporting and Importing Dashboards 5

Exporting Dashboards 5

Importing Dashboards 5

Exporting the Graph Data to CSV 6

Filtering the Graphs 6

Bulk Statistics 6

Configuring the Bulk Statistics collection 7

Sample Queries for Bulk Statistics 8

Sample Configuration 8
Sample Bulk Statistics Configuration 8
Policy configuration counters 13
SVN repo monitoring counters 13

CHAPTER 2 **Statistics and KPI Reference** 15
cnAAA Statistics 15
SOAP API rate-limit KPI 27
SPR subscriber record count 27
System KPIs 27
System Health Monitoring KPIs 27
System Status KPIs 29
System Configuration KPIs 30
CPU Category 30
Disk Category 32
File System Category 35
Load Category 37
Memory Category 38
Network Category 39
Radius Endpoint Requests Category 40
Additional KPI support for performance benchmarking 66
Feature History 66
Overview 66
Enhanced KPI Support for 66
Bulk Stats 70
Bulk Stats configuration for RADIUS requests 70
Bulk Stats Sample Query Configuration 71
Subscriber migration from CPS 7.5 to cnAAA 72
Feature History 72
Subscriber migration KPIs 72
Subscriber migration from CPS 7.5 to cnAAA SOAP Kafka Relay 78
Feature History 78
Subscriber migration SOAP Kafka Relay KPIs 78

CHAPTER 3 MIB Reference 85

CISCO-CNEE-MIB 85

CISCO-SMI 85



About this Guide

- Conventions Used, on page vii
- Contacting Customer Support, on page viii

Conventions Used

The following tables describe the conventions used throughout this documentation.

Notice Type	Description
Information Note	Provides information about important features or instructions.
Caution	Alerts you of potential damage to a program, device, or system.
Warning	Alerts you of potential personal injury or fatality. May also alert you of potential electrical hazards.

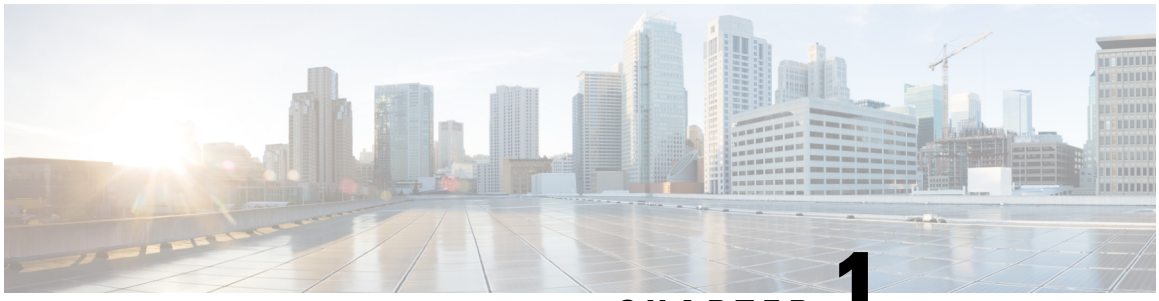
Typeface Conventions	Description
Text represented as a `screen display`	This typeface represents displays that appear on your terminal screen, for example: `Login:`
Text represented as **commands**	This typeface represents commands that you enter, for example: **show ip access-list** This document always gives the full form of a command in lowercase letters. Commands are not case sensitive.

Typeface Conventions	Description
Text represented as a **command** *variable*	This typeface represents a variable that is part of a command, for example: **show card** *slot_number* *slot_number* is a variable representing the applicable chassis slot number.
Text represented as menu or sub-menu names	This typeface represents menus and sub-menus that you access within a software application, for example: Click the **File** menu, then click **New**

Contacting Customer Support

Use the information in this section to contact customer support.

Refer to the support area of http://www.cisco.com for up-to-date product documentation or to submit a service request. A valid username and password are required to access this site. Please contact your Cisco sales or service representative for additional information.



CHAPTER 1

Promethus and Grafana

- Feature Summary, on page 1
- Feature Description, on page 1
- Automatic site failover KPIs, on page 2
- Managing the cnAAA Statistics, on page 3
- Bulk Statistics, on page 6
- Configuring the Bulk Statistics collection, on page 7
- Sample Queries for Bulk Statistics, on page 8
- Sample Configuration, on page 8
- Sample Bulk Statistics Configuration, on page 8
- Policy configuration counters, on page 13
- SVN repo monitoring counters, on page 13

Feature Summary

Summary Data

Table 1: Summary Data

Applicable Product(s) or Functional Area	cnAAA
Applicable Platform(s)	SMI
Feature Default Setting	Enabled – Always-on
Related Changes in this Release	Not Applicable
Related Documentation	Not Applicable

Feature Description

You can a monitor wide range of application and system statistics, and key performance indicators (KPI) within the cnAAA infrastructure. KPIs are useful to gain insight into the overall health of the cnAAA

environment. Statistics offer a simplified representation of the cnAAA configurations and utilization-specific data.

The cnAAA integrates with Prometheus, a third-party monitoring and alerting solution to capture and preserve the performance data. This data is reported as statistics and can be viewed in the web-based dashboard. Grafana provides a graphical or text-based representation of statistics and counters, which the Prometheus database collects. The Grafana dashboard projects a comprehensive set of quantitative and qualitative data that encourages you to analyze cnAAA metrics in the reporting tool of your choice and take informed decisions.

By default, the monitoring solution is enabled, which indicates that Prometheus continually monitors your cnAAA environment and the Prometheus data source is associated with Grafana. You must have the administrative privileges to access Grafana. However, to view a specific dashboard, run the Prometheus queries. The queries are available in the built-in and custom format.

How it Works

KPIs constitute of metrics such as statistics and counters. These metrics represent the performance improvement or degradation. By default, Prometheus is enabled on the system where cnAAA is deployed, and configured with Grafana. Prometheus dynamically starts monitoring the data sources that are available on the system. For new dashboard panels, execute queries in Prometheus.

For more information about Prometheus, consult the Prometheus documentation at https://prometheus.io/docs/introduction/overview/.

Automatic site failover KPIs

RADIUS endpoint pod availability

Trigger condition: At least 50% of the `radius-ep` StatefulSet replicas are unavailable.

Rule name: `auto-failover-radius-endpoint-pods-unavailable`

Alert duration: 3 minutes.

Severity: Critical.

PromQL expression:

```
((kube_statefulset_replicas{statefulset="radius-ep"} -
kube_statefulset_status_replicas_ready{statefulset="radius-ep"}) /
kube_statefulset_replicas{statefulset="radius-ep"} >= 0.5) and
(kube_statefulset_replicas{statefulset="radius-ep"} > 0)
```

RADIUS authentication failure

Trigger condition: At least 50% of RADIUS pods produce no responses for Access-Requests while inbound requests continue.

Rule name: `auto-failover-radius-pod-access-request-missing-terminal-response`

Alert duration: 1 minute.

Severity: Critical.

PromQL expression:

```
(((sum by (pod) (rate(inbound_request_total{message_type="AccessRequest"}[5m])) - sum by
(pod) (rate(radius_responses_total{message_type=~"AccessAccept|AccessReject"}[5m]))) / on
(pod) sum by (pod) (rate(inbound_request_total{message_type="AccessRequest"}[5m]))) >= 0.5)
 and on (pod) (sum by (pod) (rate(inbound_request_total{message_type="AccessRequest"}[5m]))
 > 0)
```

Policy Engine pod availability

Trigger condition: At least 50% of configured production Policy Engine replicas are unavailable.

Rule name: `auto-failover-policy-engine-pods-unavailable`

Alert duration: 1 minute.

Severity: Critical.

PromQL expression:

```
(((sum(kube_deployment_spec_replicas{deployment="pcf-engine-pcf-gamma-cncps-pcf-engine-app-pcf01production"})
 -
sum(kube_deployment_status_replicas_available{deployment="pcf-engine-pcf-gamma-cncps-pcf-engine-app-pcf01production"}))
 /
sum(kube_deployment_spec_replicas{deployment="pcf-engine-pcf-gamma-cncps-pcf-engine-app-pcf01production"}))
 >= 0.5) and
(sum(kube_deployment_spec_replicas{deployment="pcf-engine-pcf-gamma-cncps-pcf-engine-app-pcf01production"})
 > 0)
```

Expression behavior

Use the configured replica count for availability calculations. Exclude completed or terminating pods. Require positive inbound traffic for the RADIUS access response failure expression.

Managing the cnAAA Statistics

This section describes how to view statistics within cnAAA.

Managing the cnAAA statistics involves the following:

1. Viewing the Statistics
2. Accessing the Grafana Dashboard
3. Viewing the cnAAA Dashboard, on page 4
4. Running a Query in Grafana
5. Configuring Autorefresh
6. Exporting and Importing Dashboards

Viewing the Statistics

This section describes how to view the statistics information.

1. On the system where cnAAA is deployed, navigate to the following URL:

   `https://docs.`*`namespace`*`-product-documentation.`*`IP_address`*`.nip.io/`

All the cnAAA-specific statistics and other generic statistics such as system-statistics derived from the SMI deployer get displayed on the HTML page.

Accessing the Grafana dashboard

This section describes how to access Grafana to view the visual representation of KPIs.

1. On the system where cnAAA is deployed, navigate to the following URL to view the dashboard:

   `https://grafana.smi-cnat-monitoring.IP_address.nip.io`

2. Enter the administrative user's username and password.

For more information on Grafana's capabilities, consult the Grafana documentation available at http://docs.grafana.org.

Viewing the cnAAA Dashboard

This section how to view the cnAAA dashboard.

1. On the cnAAA Application dashboard, in the left pane, click the dashboard icon to open the menu and select **Manage**.
2. In the **Manage** tab, click the *namespace* folder.

   The folder name resembles the namespace in which cnAAA is installed. The available dashboards are listed.
3. Click **cnAAA Application**.

   The cnAAA Application dashboard displays the graphs. You can shuffle the location of the graphs by dragging the panels.

Note Cisco recommends configuring the panel options in the Grafana dashboard. With this option, you can view only the required graphs when the dashboard is loaded.

Running a Query in Grafana

This section describes how to execute a query in Grafana.

The cnAAA Dashboard creates a panel containing the graph that is based on the query that it ingests. Grafana brings up a panel to visualize data that is retrieved for one or more queries. You can run canned and custom queries from the dashboard. The canned queries are preexisting in the dashboard with the define syntax. Custom queries permit you to formulate queries that return specific information.

1. On the cnAAA Application Dashboard, in the left pane, click the explore icon to open the menu. On hovering over the icon, the tooltip text appears as **Explore**.
2. In the **Explore** pane, click the drop-down to choose the data source as **Prometheus**.
3. Do one of the following:

- To execute a built-in query, click the **Metrics** drop-down and choose the query that you want to run.
- To execute a custom query, enter the query in the corresponding field next to **Metrics**.

4. Click **Run Query**.

   The query retrieves the information from Prometheus and displays it in a graphical representation.

Configuring Autorefresh

This section describes how to configure autorefresh to ensure that you view the recent information on Grafana.

1. On the cnAAA Application dashboard, click the gear on the top-right corner to open the **Settings**. On hovering over this icon, the tooltip text displays **Dashboard settings**.
2. In the **General** pane, navigate to the **Time Options** section and enter the time range in the **Autorefresh** field. You can specify the range in seconds, minutes, hours, and days format.



Note If you opt not to specify range, then the dashboard gets refreshed at the default interval.

Exporting and Importing Dashboards

This section describes how to export and import Grafana dashboards between environments and share them.

Exporting Dashboards

To export a dashboard configuration to a file:

1. Log in as an administrative user.
2. Open the dashboard that you want to export.
3. Click the gear icon at the top of the page, and then click **Export** to save the dashboard configuration on your local system.
4. If prompted, browse to the location on your local system to save the dashboard template, then click OK.

Importing Dashboards

To import a dashboard from a file:

1. Log in as an administrative user.
2. In the left pane, click the Dashboard icon to open the menu and click **Home**. The home pane opens.
3. Click the Home drop-down and click **Import dashboard**.
4. Specify the Grafana dashboard URL or ID that you want to import, provide the JSON details, or click **Upload.json File** and browse to the JSON file that you want to import.

5. Click **Load**.

   Make sure to save the dashboard to protect the changes that you made to the dashboard.

Exporting the Graph Data to CSV

This section describes how to export a dashboard in a CSV format.

1. On the Grafana dashboard, click the title of the graph to open the graph controls.
2. Click the rows button to open the menu.
3. To view the export option, click **More** and then click **Export CSV**.

   Your web browser downloads the *grafana_data_export.csv* file.

Filtering the Graphs

This section describes how to filter graphs on a dashboard.

You can narrow down the visualizations appearing on a dashboard by filtering them based on the specific time range.

1. On the cnAAA Application dashboard, in the top-right corner, click the clock icon.
2. Choose the range for which you want to view the graphs. Quick ranges provide the commonly used ranges that retrieve data in the shortest time. For specific range, provide the range under the Custom range heading.

Bulk Statistics

Bulk statistics are the statistics that are collected periodically and written to a set of CSV files. These statistics can be used by external analytic processes and/or network management systems. Bulk stats allows you to combine different KPIs into a unified query that fetches the custom statistical data.

The SMI component handles the collection of the bulk stats from the nodes and cnAAA consumes these stats. The bulk stats are generated for the following components:

- Container: Includes the raw and rate of the change statistics.
- Pod: Includes the raw and rate of the change statistics.
- System: Consists of the system level KPIs.



Note The container and Pod statistics contain the predefined infrastructure outputs such as CPU and memory. You can also customize the query to fetch specific outputs as per your requirement.

There are two types of bulk statistics:

- Gauge - A snapshot value that shows the statistic at that reporting moment (for example, the number of current PDP contexts, simultaneous Active EPS Bearers). Gauge statistics can increment or decrement continuously.
- Counter - A historic value that shows the statistic that accumulated over time (for example, the total number of CSR requests received). Counter values can only increment except in two cases: rollover, where a counter exceeds its maximum value and rolls over to zero, and reset, where a counter is manually reset to zero.

Configuring the Bulk Statistics collection

This section describes how to configure the bulk statistics collection feature.

You can optimize and control the bulk statistics collection by creating the Prometheus query that you configure on the SMI Ops-Center. The cumulative result of the statistics query is available in a CSV file which is created on the node where you run the query.

1. Log in to the SMI Ops Center and run the following:

```
configure
  bulk-stats enable true
  bulk-stats query  kpi_name
  expression "sum(irate(kpi_name
{exported_application=~\".*\",command_code=~\".*\"}[1m])>0) by
(exported_application,command_code)"

  label  operation_name
  exit
  bulk-stats query   kpi_name
  expression "(sum(rate(kpi_name[duration])) by (operation_name))"
  label  operation_name
  exit
```

NOTES:

- **bulk-stats query** *kpi_name*: Specify the statistics name for which you want to generate stats in bulk. For example, inbound_request_total and radius_requests_total

  See Statistics and KPI reference for the list of KPIs.

- **expression** *"sum(irate(kpi_name {exported_application=~\".*\",command_code=~\".*\"}[1m])>0) by (exported_application,command_code)"*: Indicates the query format following which SMI collects the stats. For example:
  "sum(irate(radius_requests_total{exported_application=~\".\",command_code=~\".*\"}[1m])>0) by (exported_application,command_code)"*

Note Based on the KPI tha you specify, manipulate the query. For instance, in case of the inbound_request_total KPI, add a parameter for specifying the duration as [5m]. This means that the SMI collects the stats for the total inbound requests that are processed in 5 minutes.

- **label** *operation_name*: Specify the operation that processes the KPI.

Sample Queries for Bulk Statistics

For more information about Sample Bulk Statistics queries, refer to the cnAAA Statistics section.

Sample Configuration

The following is a sample bulk statistic:

```
 cee(config)#
bulk-stats enable true
bulk-stats query inbound_request_total
expression "(sum(rate(inbound_request_total[5m])) by (operation_name))"
label operation_name
exit
bulk-stats query radius_requests_total
expression "sum(radius_requests_total{message_type=~\".*\"}) by (message_type)"
labels [ message_type  ]
alias radius_inbound
exit
bulk-stats query outgoing_request_total
default-value 0
expression "(sum(rate(outgoing_request_total[5m])) by (operation_name))"
label operation_name
exit
```

Sample Bulk Statistics Configuration

This section provides sample bulk statistics configurations that are defined in cnAAA.

active-alerts

```
bulk-stats query active-alerts
expression sum(ALERTS{alertstate=\"firing\"})"
label (alertname)
exit
```

config-query-memory-used

```
bulk-stats query config-query-memory-used
expression sum(node_memory_MemTotal_bytes)-sum(node_memory_MemFree_bytes
label (hostname)
exit
```

query cpu-idle

```
bulk-stats query query cpu-idle
expression avg(rate(node_cpu_seconds_total{mode=\"idle\"}[1m]))
label (hostname)
exit
```

cpu-iowait

```
bulk-stats query cpu-iowait
expression avg(rate(node_cpu_seconds_total{mode=\"iowait\"}[1m]))*100.00
label (hostname)
exit
```

cpu-softirq

```
bulk-stats query cpu-softirq
expression avg(rate(node_cpu_seconds_total{mode=\"softirq\"}[1m]))*100.00
label (hostname)
exit
```

cpu-steal

```
bulk-stats query cpu-steal
expression avg(rate(node_cpu_seconds_total{mode=\"steal\"}[1m]))*100.00
label (hostname)
exit
```

cpu-system

```
bulk-stats query cpu-system
expression avg(rate(node_cpu_seconds_total{mode=\"system\"}[1m]))*100.00
label (hostname)
exit
```

cpu-user

```
bulk-stats query cpu-user
expression avg(rate(node_cpu_seconds_total{mode=\"user\"}[1m]))*100.00
label (hostname)
exit
```

daemonset-ready-percent

```
bulk-stats query daemonset-ready-percent
expression
kube_daemonset_status_number_ready/kube_daemonset_status_desired_number_scheduled*100
label (daemonset)
exit
```

datastore_failures

```
bulk-stats query datastore_failures
expression sum(datastore_request_total{error_code!~\"0|409\"})
label (error_code)
exit
```

deployment-ready-percent

```
bulk-stats query deployment-ready-percent
expression kube_deployment_status_replicas_available/kube_deployment_status_replicas*100
```

```
label (deployment)
exit
```

entitlement-status

```
bulk-stats query entitlement-status
expression entitlement_status{enforce_mode!=\"InCompliance\"}
label (hostname)
exit
```

filesystem-data-avail-bytes

```
bulk-stats query filesystem-data-avail-bytes
expression avg(node_filesystem_avail_bytes{device=\"/dev/vda1\"})
label (hostname)
exit
```

filesystem-root-avail-bytes

```
bulk-stats query filesystem-root-avail-bytes
expression avg(node_filesystem_avail_bytes{device=\"/dev/sda1\"})
label (hostname)
exit
```

k8s-pods-status

```
bulk-stats query  k8s-pods-status
expression sum(kube_pod_status_phase)
label (phase)
exit
```

kubelet-node-status

```
bulk-stats query kubelet-node-status
expression sum(kube_node_status_condition{status=\"true\"})
label (condition)
exit
```

kublet-running-pod-count

```
bulk-stats query kublet-running-pod-count
expression kubelet_running_pod_count
label (hostname)
exit
```

memory-used

```
bulk-stats query query memory-used
expression sum(node_memory_MemTotal_bytes)
label (hostname)
exit
```

network-carrier-bond-changes-total

```
bulk-stats query network-carrier-bond-changes-total
expression sum(node_network_carrier_changes_total{device=~\"bond[0-9]\"} OR on ()
vector (0)) by (namespace)
label (hostname)
exit
```

network-carrier-ens-changes-total

```
bulk-stats query network-carrier-ens-changes-total
expression sum(node_network_carrier_changes_total{device=~\"ens.*\"} OR on () vector
(0)) by (namespace)
label (hostname)
exit
```

network-errors-total

```
bulk-stats query network-errors-total
expression sum(node_network_receive_errs_total)
label (hostname)
exit
```

network-receive-bond-bytes-total

```
bulk-stats query network-receive-bond-bytes-total
expression sum(node_network_receive_bytes_total{device=~\"bond[0-9]\"})
label (hostname)
exit
```

network-receive-ens-bytes-total

```
bulk-stats query network-receive-ens-bytes-total
expression sum(node_network_receive_bytes_total{device=~\"ens.*\"})
label (hostname)
exit
```

network-transmit-bond-bytes-total

```
bulk-stats query network-transmit-bond-bytes-total
expression sum(node_network_transmit_bytes_total{device=~\"bond[0-9]\"})
label (hostname)
exit
```

network-transmit-ens-bytes-total

```
bulk-stats query network-transmit-ens-bytes-total
expression sum(node_network_transmit_bytes_total{device=~\"ens.*\"})
label (hostname)
exit
```

node-disk-rate-read-bytes

```
bulk-stats query node-disk-rate-read-bytes
expression sum(rate(node_disk_read_bytes_total[5m]))
label (hostname)
exit
```

node-disk-write-read-bytes

```
bulk-stats query node-disk-write-read-bytes
expression sum(rate(node_disk_written_bytes_total[5m]))
label (hostname)
exit
```

node-load-15

```
bulk-stats query node-load-15
expression node_load15
label (hostname)
exit
```

node-memory-free-bytes

```
bulk-stats query node-memory-free-bytes
expression sum(node_memory_MemTotal_bytes)
label (hostname)
exit
```

record_conflict

```
bulk-stats query record_conflict
expression sum(datastore_notify_total{notification_type=\"RECORD_CONFLICT\"} OR on
() vector (0)) by (namespace)
label (notification_type)
exit
```

statefulset-ready-percent

```
bulk-stats query statefulset-ready-percent
expression kube_statefulset_status_replicas_ready/kube_statefulset_status_replicas*100
label (statefulset)
exit
```

timer_expiry

```
bulk-stats query timer_expiry
expression sum(datastore_notify_total{notification_type=\"TIMER_EPIRED\"} OR on ()
vector (0)) by (namespace)
label (notification_type)
exit
```

version_mismatch_retries

```
bulk-stats query version_mismatch_retries
expression sum(datastore_request_total{error_code=\"409\"} OR on () vector (0)) by
(namespace)
label (error_code)
exit
```

Policy configuration counters

Policy configurations during system startup and policy structure changes are monitored using engine counters to ensure system stability.

To view the counters, use this command:

```
kubectl exec -it <engine-pod> -n <namespace> -- curl -G http://127.0.0.1:8080/metrics |
grep poli
```

- **Set up policies_not_configured_total Counter**

  This counter tracks the success or failure of policy configurations during system startup. It increments when policy configuration fails and resets when successful.

  Example:

  ```
  # HELP policies_not_configured_total Total of policies_not_configured
  # TYPE policies_not_configured_total counter
  policies_not_configured_total{node_type="unknown", message_type="Policies Not
  Configured",}
  ```

- **last_policy_configuration_failed_total**

  This counter monitors changes to the system policy structure. It increments if the last policy configuration attempt fails and resets upon success.

  Example:

  ```
  # HELP last_policy_configuration_failed_total Total of last_policy_configuration_failed
  # TYPE last_policy_configuration_failed_total counter
  last_policy_configuration_failed_total{node_type="unknown", message_type="Last Policy
  Configuration Failed due to java.lang.ArithmeticException / by zero",}
  ```

SVN repo monitoring counters

This section provides an overview of KPIs designed to monitor various activities and states within SVN repositories, focusing on critical counter metrics.

- **svn_repo_deleted_total**

  This counter tracks the total number of SVN repositories that have been deleted. It increments each time an SVN repository deletion event occurs.

  Example:

  ```
  # HELP svn_repo_deleted_total Total of svn_repo_deleted
              # TYPE svn_repo_deleted_total counter
              svn_repo_deleted_total{node_type="unknown",message_type="new_import3",}
  20.0
  ```

- **config_file_deleted_from_svn_repo_total**

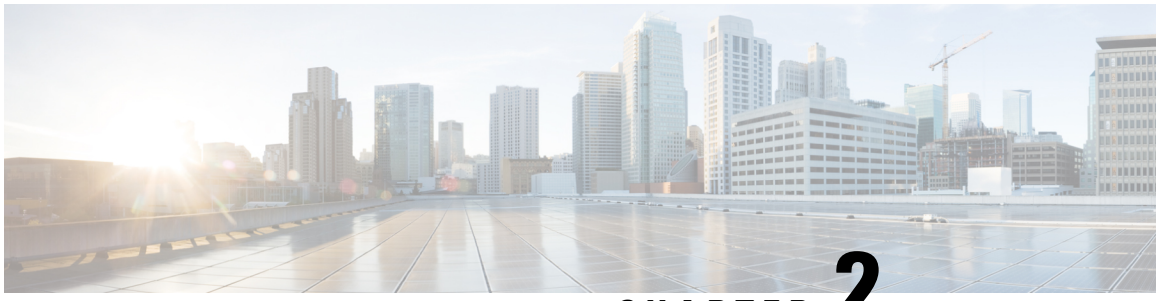
  This counter tracks the total number of configuration files that have been deleted from an SVN repository. It increments each time a configuration file is removed from an SVN repository.

  Example:

```
# HELP config_file_deleted_from_svn_repo_total Total of config_file_deleted_from_svn_repo

            # TYPE config_file_deleted_from_svn_repo_total counter

config_file_deleted_from_svn_repo_total{node_type="unknown",message_type="new_import2",}
 1.0
```



CHAPTER 2

Statistics and KPI Reference

- cnAAA Statistics, on page 15
- SOAP API rate-limit KPI, on page 27
- SPR subscriber record count, on page 27
- System KPIs, on page 27

cnAAA Statistics

action_duration_seconds

Duration for the <type> action to complete from policy engine.

Sample Query: action_duration_seconds{node_type="unknown",type="update-subscriber-service"}

- **Label: node_type**

  Description:: Node Type mentioned in Kubernetes configuration

  Example: unknown

- **Label: type**

  Description:: Type of action

  Example: create-subscriber, delete-subscriber etc.

action_total

Count of actions of <type> as defined in the label from policy engine.

Sample Query: `action_total{node_type="unknown",type="add-subscriber-service",status="success"}`

- **Label: node**

  Description:: Node type

  Example: unknown

- **Label: type**

  Description:: Action type

  Example: add-subscriber-service, update-subscriber-service, delete-subscriber-service, etc.

- **Label: status**

  Description:: Status of the action

  Example: success, error

dispatch_error_seconds_total

Total processing duration for error scenario of dispatched <message_type> response from radius endpoint to the engine.

Sample Query: `dispatch_error_seconds_total{message_type="AsyncCoARequest",replyto_address="<IP address>"}`

- **Label: message_type**

  Description:: Type of CoA Message

  Example: AsyncCoARequest, BundledCoARequest, ProxyAccountingRequest

- **Label: replyto_address**

  Description:: BNG IP Address from which radius message (associated with this radius session id) was sent to CPC

dispatch_error_total

Total error count of dispatched <message_type> response from radius endpoint to the engine.

Sample Query: `dispatch_error_total{message_type="BundledCoARequest",replyto_address="<IP address>"}`

- **Label: message_type**

  Description:: Type of CoA Message

  Example: AsyncCoARequest, BundledCoARequest, ProxyAccountingRequest

- **Label: replyto_address**

  Description:: BNG IP Address from which radius message (associated with this radius session id) was sent to CPC

dispatch_message_seconds_total

Sample Query:
`dispatch_message_seconds_total{message_type="AsyncCoARequest",replyto_address="<IP address>"}`

- **Label: message_type**

  Description:: Type of CoA Message

  Example: AsyncCoARequest, BundledCoARequest, ProxyAccountingRequest

- **Label: replyto_address**

  Description:: BNG IP Address from which radius message (associated with this radius session id) was sent to CPC

dispatch_message_total

Total count of dispatched <message_type> response from radius endpoint to the engine.

Sample Query: `dispatch_message_total{message_type="AsyncCoARequest",replyto_address="<IP address>"}`

- **Label: message_type**

  Description:: Type of CoA Message

  Example: AsyncCoARequest, BundledCoARequest, ProxyAccountingRequest

- **Label: replyto_address**

  Description:: BNG IP Address from which radius message (associated with this radius session id) was sent to CPC

etcd_registry_lookup_total

Calculates the total number of etcd registry lookups.

Sample Query: `etcd_registry_lookup_total{node_type="unknown"}`

- **Label: node**

  Description:: Node type

  Example: unknown

grpc_message_send_total

Total no. of messages sent to engine, from radius EP, via gRPC.

Sample Query: `grpc_message_send_total{message_type="AccountingRequest",replyto_address="<IP address>"}`

- **Label: message_type**

  Description:: Type of Message

  Example: AccountingRequest, AccessRequest

- **Label: replyto_address**

  Description:: BNG IP Address from which radius message was sent to CPC

inbound_request_total

Total count of inbound messages <message_type>, received at radius EP from client.

Sample Query: `inbound_request_total{message_type="AccountingRequest",client_ip="<IP address>"}`

- **Label: message_type**

  Description:: Type of Message

  Example: AccountingRequest, AccessRequest

- **Label: client_ip**

  Description:: BNG IP Address from which radius message was sent to CPC

input_queue_result_total

Total count of input messages, received at engine queue.

Sample Query: input_queue_result_total{node_type="unknown"}

- **Label: node_type**

  Description:: Node Type

  Example: unknown

message_total

Total number of successful or failed execution of various actions in Policy engine.

Sample Query:
`message_total{node_type="unknown",type="radius-access-request-message",status="success"}`

- **Label: node_type**

  Description:: Node Type

  Example: unknown

- **Label: type**

  Description:: Indicates message type

  Example: radius-access-request-message, radius-accounting-message, remove-session-imp, etc

- Label: status

  Description:: indicates operation completion status

  Example: success, error

outbound_request_total

Total count of outbound messages <message_type>, sent from radius EP to client.

Sample Query: `outbound_request_total{message_type="CoARequest",client_ip="<IP address>",ocs_server="NA"}`

- **Label: message_type**

  Description:: type of message that request that is been handled.

  Example: CoARequest, ProxyAccounting

- **Label: client_ip**

  Description:: IP Address to which radius message was sent from radius EP.

- **Label: ocs_server**

  Description:: OCS server detail

policy_engine_message_seconds_total

Total round trip response time in seconds, for processing accounting/access request from engine via gRPC.

Sample Query:
`policy_engine_message_seconds_total{message_type="AccountingRequest",replyto_address="<IP address>"}`

- **Label: message_type**

  Description:: Indicates type of message

  Example: AccountingRequest, AccessRequest

- **Label: replyto_address**

  Description:: BNG IP Address from which radius message was sent to CPC

policy_engine_message_total

Total number of responses received, for processing accounting/access request from engine via gRPC.

Sample Query: `policy_engine_message_total{message_type="AccessRequest",replyto_address="<IP address>"}`

- **Label: message_type**

  Description:: Indicates type of message

  Example: AccountingRequest, AccessRequest

- **Label: replyto_address**

  Description:: BNG IP Address from which radius message was sent to CPC

process_message_seconds_total

Total time taken for processing messages <message_type>.

Sample Query:
`process_message_seconds_total{message_type="AccountingResponse",replyto_address="<IP address>"}`

- **Label: message_type**

  Description:: Type of the message

  Example: AccessReject, AccessAccept and AccountingResponse

- **Label: replyto_address**

  Description:: BNG IP Address from which radius message (associated with this radius session id) was sent to CPC

process_message_total

Total count of messages <message_type> processed.

Sample Query: process_message_total{message_type="AccessReject",replyto_address="<IP address>"}

- **Label: message_type**

  Description:: Type of the message

  Example: AccessReject, AccessAccept and AccountingResponse

- **Label: replyto_address**

  Description:: BNG IP Address from which radius message (associated with this radius session id) was sent to CPC

radius_accounting_request_total

Total count of accounting requests <accounting_type> against status <status_type>

Sample Query:`radius_accounting_request_total{accounting_type="ServiceAccounting",status_type="Interim-Update", clientIp="<IP address>",endPointIp="<IP address>",result="SUCCESS"}`

- **Label: accounting_type**

  Description:: Indicates that the message pertains to service-level or session accounting activities.

  Example: SessionAccounting , ServiceAccounting

- **Label: status_type**

  Description:: Specifies that the request is part of Start, interim update or stop.

  Example: start, Interim-Update or stop

- **Label: clientIp**

  Description:: BNG IP Address from which radius message was sent to CPC.

- **Label: endPointIp**

  Description:: Radius Pod Endpoint IP address (can change upon restart).

- **Label: result**

  Description:: Indicates that the status of request.

  Example: SUCCESS, DROP, etc

radius_accounting_response_seconds_total

Total processing time taken for accounting response <accounting_type> against status <status_type>

Sample Query:
`radius_accounting_response_seconds_total{accountingType="ServiceAccounting",statusType="Interim-Update", clientIp="<IP address>",endPointIp="<IP address>"}`

- **Label: accounting_type**

  Description:: Indicates that the message pertains to service-level or session accounting activities.

  Example: SessionAccounting , ServiceAccounting

- **Label: status_type**

  Description:: Specifies that the response is part of Start, interim update or stop.

  Example: start, Interim-Update or stop

- **Label: clientIp**

Description:: BNG IP Address from which radius message was sent to CPC.

- **Label: endPointIp**

  Description:: Radius Pod Endpoint IP address (can change upon restart).

radius_accounting_response_total

Total count of accounting response <accounting_type> against status <status_type>

Sample Query:
```
radius_accounting_response_total{accountingType="SessionAccounting",statusType="Stop",
clientIp="<IP address>",endPointIp="<IP address>"}
```

- **Label: accounting_type**

  Description:: Indicates that the message pertains to service-level or session accounting activities.

  Example: SessionAccounting , ServiceAccounting

- **Label: status_type**

  Description:: Specifies that the response is part of Start, interim update or stop.

  Example: start, Interim-Update or stop

- **Label: clientIp**

  Description:: BNG IP Address from which radius message was sent to CPC.

- **Label: endPointIp**

  Description:: Radius Pod Endpoint IP address (can change upon restart).

radius_discard_requests_total

Number of request messages <message_type> with late response comes from engine, via gRPC.

Sample Query: `radius_discard_requests_total{message_type="AccessRequest",nas_ip_address="<IP address>",`

`client_ip_address="<IP address>",endpoint_ip_address="<IP address>"}`

- **Label: message_type**

  Description:: Type of message

  Example: AccessRequest, AccountingRequest

- **Label: nas_ip_address**

  Description:: NAS IP Address

- **Label: client_ip_address**

  Description:: BNG IP Address from which radius message was sent to CPC

- **Label: endpoint_ip_address**

  Description:: Radius Pod Endpoint IP address (can change upon restart)

radius_engine_total

Number of messages dropped or skipped during overload condition.

Sample Query:
`radius_engine_total{node_type="unknown",message_type="Session-Accounting-Stop_REQ-DROP_Engine-Overload"}`

- **Label: endpoint_ip_address**

  Description:: Type of overload action

  Example: Session-Accounting-Stop_REQ-DROP_Engine-Overload, Service-Accounting-Start_REQ_in-queue-drop, etc

radius_late_responses_total

Total number of late Responses received in engine, via gRPC.

Sample Query: `radius_late_responses_total{message_type="AccessRequest",nas_ip_address="<IP address>",`

`client_ip_address="<IP address>",endpoint_ip_address="<IP address>"}`

- **Label: endpoint_ip_address**

  Description:: Message's type

  Example: AccountingRequest/ AccessRequest

- **Label: nas_ip_address**

  Description:: NAS IP Address

- **Label: client_ip_address**

  Description:: BNG IP Address from which radius message was sent to CPC

- **Label: endpoint_ip_address**

  Description:: Radius Endpoint IP address from which Radius Proxy request messages were initiated.

radius_proxy_accounting_request_total

Total count of service accounting requests, forwarded to AAA server.

Sample Query:
`radius_proxy_accounting_request_total{accounting_type="ServiceAccounting",status_type="Interim-Update",`

`AAAServer="<AAA Server>",endpoint_ip="<IP address>"}`

- **Label: accounting_type**

  Description:: Type of accounting request

  Example: ServiceAccounting

- **Label: status_type**

  Description:: Specifies that the response is part of Start, interim update or stop.

  Example: start, Interim-Update or stop

- **Label: AAAServer**

Description:: The AAA server name configured in Policy Builder, and OPs center.

- **Label: endpoint_ip**

  Description:: Radius Endpoint IP address from which Radius Proxy request messages were initiated.

radius_proxy_accounting_response_second_total

Time taken for service accounting response, from AAA server.

Sample Query:
```
radius_proxy_accounting_response_second_total{accounting_type="ServiceAccounting",status_type="Start",
AAAServer="<AAAServer>",result="DROP",server_ip="NA",endpoint_ip="<IP address>",retries="0",}
```

- **Label: accounting_type**

  Description:: Type of accounting request

  Example: ServiceAccounting

- **Label: status_type**

  Description:: Specifies that the response is part of Start, interim update or stop.

  Example: start, Interim-Update or stop

- **Label: AAAServer**

  Description:: The AAA server name configured in Policy Builder and OPs Center

- **Label: result**

  Description:: Radius Proxy message response handling at Proxy Server/AAA end.

  Example: Success /Timeout/ ERROR

- **Label: server_ip**

  Description:: AAA Server IP

- **Label: endpoint_ip**

  Description:: Radius Endpoint IP address.

- **Label: retries**

  Description:: The Number of retries on which Radius Proxy message was attempted towards AAA server.

  Example: 2

radius_proxy_accounting_response_total

Total count of service accounting response, from AAA server.

Sample Query:
radius_proxy_accounting_response_total{accounting_type="ServiceAccounting",status_type="Start",
AAAServer="<AAAServer>",result="Timeout",server_ip="NA",endpoint_ip="<IP address>", retries="NA"}

- **Label: accounting_type**

  Description:: Type of Accounting message

Example: ServiceAccounting

- **Label: status_type**

  Description:: Radius Accounting status type

  Example: Start Interim-Update, Stop

- **Label: AAAServer**

  Description:: The AAA server name configured in Policy Builder and OPs Centre

- **Label: result**

  Description:: State of the proxy response

  Example: Success, Timeout, ERROR

- Label: server_ip

  Description:: AAA Server IP

- **Label: endpoint_ip**

  Description:: Radius Endpoint IP address

- **Label: retries**

  Description:: The Number of retries on which Radius Proxy messages were attempted by AAA server.

  Example: 2

radius_request_timeout_total

Total number of request timeouts happened.

Sample Query: `radius_request_timeout_total{message_type="CoaRequest",nas_ip_address="<IP address>",`

`client_ip_address="<IP address>",endpoint_address="<IP address>"}`

- **Label: endpoint_ip_address**

  Description:: Radius Message type

  Example: CoARequest/AccessRequest

- **Label: nas_ip_address**

  Description:: NAS IP Address

- **Label: client_ip_address**

  Description:: BNG IP Address from which radius message was sent to CPC

- **Label: endpoint_address**

  Description:: Radius Endpoint IP address

radius_requests_total

Total number of Radius messages <message_type> received by Radius EP.

Sample Query: `radius_requests_total{message_type="AccessRequest",nas_ip_address="<IP address>",client_ip_address="<IP address>",endpoint_address="<IP address>",result="SUCCESS"}`

- **Label: endpoint_ip_address**

  Description:: Radius Message type

  Example: AccessRequest / CoARequest

- **Label: nas_ip_address**

  Description:: NAS IP Address

- Label: client_ip_address

  Description:: BNG IP Address from which radius message was sent to CPC

- **Label: endpoint_address**

  Description:: Radius Endpoint IP address

- **Label: result**

  Description:: SUCCESS, DROP, etc

  Example: SUCCESS/DROP

radius_responses_seconds_total

Total time taken for Radius messages <message_type> responses at Radius EP.

Sample Query: `radius_responses_seconds_total{message_type="AccessAccept",nas_ip_address="<IP address>",client_ip_address="<IP address>",endpoint_address="<IP address>"}`

- **Label: endpoint_ip_address**

  Description:: Radius Message type

  Example: AccessAccept AccessReject CoAACKResponse, etc

- **Label: nas_ip_address**

  Description:: NAS IP Address

- **Label: client_ip_address**

  Description:: BNG IP Address from which radius message was sent to CPC

- **Label: endpoint_address**

  Description:: Radius Endpoint IP address

radius_responses_total

Total number of Radius messages <message_type> responses at Radius EP.

Sample Query: `radius_responses_total{message_type="AccessAccept",nas_ip_address="<IP address>",client_ip_address="<IP address>",endpoint_address="<IP address>"}`

- **Label: endpoint_ip_address**

  Description:: Radius Message type

  Example: AccessAccept , AccessReject , CoAACKResponse , etc

- **Label: nas_ip_address**

  Description:: NAS IP Address

- **Label: client_ip_address**

  Description:: BNG IP Address from which radius message was sent to CPC

record conflict merges

Description: Total count of record conflict merges.

Sample Query: `record_conflict_merge_total`

- **Label: endpoint_address**

  Description:: Radius Endpoint IP address

total_radius_auth_messages_overload_rejected

Total number of reject radius messages discarded due to overload protection.

Sample Query:
`total_radius_auth_messages_overload_rejected{message_type="AccessReject",nas_ip_address="<IP address>",`

`client_ip_address="<IP address>",endpoint_address="<IP address>"}`

- **Label: endpoint_ip_address**

  Description:: Radius Message Type

  Example: AccessReject

- **Label: nas_ip_address**

  Description:: NAS IP Address

- **Label: client_ip_address**

  Description:: BNG IP Address from which radius message was sent to CPC

- **Label: endpoint_address**

  Description:: Radius Endpoint IP address

total_radius_messages_overload_dropped

Number of radius accounting messages "'<<message_type>"'> dropped due to overload protection.

Sample Query: `total_radius_messages_overload_dropped{message_type="AccessAccept",`

`status_type="Start",nas_ip_address="<IP address>",client_ip_address="<IP address>",endpoint_address="<IP address>"}`

- **Label: endpoint_ip_address**

  Description:: Radius Message Type

  Example: SessionAccounting/ ServiceAccounting

- **Label: status_type**

  Description:: Radius Acct_status_type

  Example: Start/ Interim-Update /Stop

- **Label: nas_ip_address**

  Description:: NAS IP Address

- **Label: client_ip_address**

  Description:: BNG IP Address from which radius message was sent to CPC

- **Label: endpoint_address**

  Description:: Radius Pod Endpoint IP address (can change upon restart)

SOAP API rate-limit KPI

SOAP API rate-limit rejections

KPI: `api-ratelimiting-reject-<error-code>`

Description: The KPI tracks SOAP API requests rejected after they exceed the configured transactions-per-second threshold.

The rate-limiting feature applies to SOAP API POST requests. REST GET requests are not rate-limited by this feature.

SPR subscriber record count

mongo_record_collection_count

Description: Total count of records in the `subscriber` collection for each SPR replica set.

Formula: `sum by(replica_set) (mongo_record_collection_count{dbname="spr", collection="subscriber"}) > 0`

System KPIs

System Health Monitoring KPIs

The following table lists the KPIs and thresholds to track the overall performance of the cnAAA deployment, including information about the underlying hardware.

CPU Utilization

Description: CPU is a critical system resource. When the demand increases and CPU utilization exceeds 80% utilization, the efficiency of the CPU is reduced. When CPU utilization exceeds 80%, the application processing time will increase, message response will increase, and drops and timeouts will be seen.

Statistics/Formula: (avg without (cpu,mode)(irate(node_cpu_seconds_total{component="node-exporter",mode!="idle"}[1m])))

Warning Threshold: > 60% utilization over 60 second period (assuming that idle is less than 40%)

Major Threshold: > 80% utilization over 60 second period (assuming idle is less than 20%)

CPU Steal

Description: If multiple VMs on the same hypervisor and same hardware have concurrent CPU demands, the hypervisor will “steal” CPU from one VM to satisfy another VM CPU needs. If the CPU Steal statistic is non-zero, there is not enough CPU allocated for the VMs.

Statistics/Formula: (avg without (cpu,mode)(irate(node_cpu_seconds_total{component="node-exporter",mode="steal"}[1m])))

Warning Threshold: NA

Major Threshold: > 2% over 60 second period

CPU I/O Wait

Description: This monitors CPU I/O wait time. High CPU wait times may indicate CPUs waiting on disk access.

Statistics/Formula: (avg without (cpu,mode)(irate(node_cpu_seconds_total{component="node-exporter",mode="wait"}[1m])))

Warning Threshold: > 30 for more than 5 min

Major Threshold: > 50 for more than 10 min

Memory utilization

Description: Memory is a system resource, which needs to be less than 80%. The swap threshold has been reduced, and swapping should occur when the system resources are exhausted and memory utilization hits 99%.

Statistics/Formula: 100 - ((node_memory_MemAvailable_bytes * 100) / node_memory_MemTotal_bytes)

Warning Threshold: > 70% utilization over 60 second period

Major Threshold: > 80% utilization over 60 second period

Disk Utilization

Description: Disk storage is a critical system resource, and when file system utilization exceeds 90% utilization the system can become less efficient. When the file system utilization hits 100%, then application can stop functioning.

Statistics/Formula:

100 - ((node_filesystem_avail_bytes{mountpoint="/",fstype!="rootfs"} * 100) / node_filesystem_size_bytes{mountpoint="/",fstype!="rootfs"})

Warning Threshold: > 80% utilization

Major Threshold: > 90% utilization

In Queue

Description: These statistics monitors how long a message waits in the application queue, waiting to be serviced. The value should be 10ms all the time. higher values indicate the application is too slow, short of resources, or overwhelmed.

Statistics/Formula: sum(irate(input_queue_duration_seconds[1m])) / sum(irate(input_queue_total[1m]))

Warning Threshold: NA

Major Threshold: More than 10 ms over 60 seconds

Active Session Count

Description: Number of total sessions currently active.

Statistics/Formula: avg(db_records_total{session_type="total"})

Warning Threshold:

>80% of the lessor of the dimensioned or licensed capacity for more than 1 hour

or

= 0 for more than 5 minutes

Major Threshold:

>80% of the lessor of the dimensioned or licensed capacity for more than 10 minutes

or

= 0 for more than 10 minutes

System Status KPIs

system_mode

Description: Indicates the current mode the system is running on.

Statistics/Formula: system_mode

Labels:

- Label: `0`

  Label Description: The system is in shutdown mode.

- Label: `1`

  Label Description: The system is running.

- Label: `2`

  Label Description: The system is under maintenance.

- Label: `-1`

Label Description: The system mode is unknown.

system_synch_running

Description: Specifies whether the system configuration synch process is running or not.

Statistics/Formula: system_synch_running

Labels:

- Label: `1`

  Label Description: The system configuration sync process is running.

- Label: `0`

  Label Description: The system configuration sync process is not running.

system_running_percent

Description: Captures the percentage of the system currently in use.

Statistics/Formula: system_running_percent

System Configuration KPIs

system_configuration_bac kup_total

Description: Captures the total number of system configuration backups that are executed.

Statistics/Formula: irate(system_configuration_backup_total [1m])

Labels:

- Label: `status`

  Label Description: The status of the executed backups. For example, success or error.

configuration_change_total

Description: Captures the total number of configuration changes that are executed.

Statistics/Formula: sum(irate(configuration_change_total[1m]))

CPU Category

node_cpu_seconds_total

Description: Seconds the cpus spent in each mode

Metric Type:

Data Type:

Sample Query: `avg(irate(node_cpu_seconds_total{mode=~\"irq|softirq\"}[1m])) by (instance) * 100`

Labels:

- Label: `instance`

  Label Description: the virtual machine/instance

  Example: master-0, control-0, dra-director-1, etc

- Label: `job`

  Label Description: the name of job

  Example: node_exporter

- Label: `cpu`

  Label Description: the cpu number

  Example: cpu0, cpu1, etc

- Label: `mode`

  Label Description: the cpu mode

  Example: system, user, sotirq, irq, idle, iowait, etc

CPU Utilization

Description: CPU is a critical system resource. When the demand increases and CPU utilization exceeds 80% utilization, the efficiency of the CPU is reduced. When CPU utilization exceeds 80%, the application processing time will increase, message response will increase, and drops and timeouts will be seen.

Metric Type:

Data Type:

Sample Query: `100 - cpu.<cpuid>.idle`

Warning Threshold:

- > 60% utilization over 60 second period (assuming that idle is less than 40%).

Major Threshold:

- > 80% utilization over 60 second period (assuming idle is less than 20%).

CPU Steal

Description: If multiple VMs on the same hypervisor and same hardware have concurrent CPU demands, the hypervisor will “steal” CPU from one VM to satisfy another VM CPU needs. If the CPU Steal statistic is non-zero, there is not enough CPU allocated for the VMs.

Metric Type:

Data Type:

Sample Query: `cpu.<cpuid>.steal`

Major Threshold:

- > 2% over 60 second period.

CPU I/O Wait

Description: This monitors CPU I/O wait time. High CPU wait times may indicate CPUs waiting on disk access.

Metric Type:

Data Type:

Sample Query: `cpu.<cpuid>.wait`

Warning Threshold:

- > 30 for more than 5 min.

Major Threshold:

- > 50 for more than 10 min.

Disk Category

node_disk_bytes_read

Description: This metrics gives the total number of bytes read successfully.

Metric Type:

Data Type:

Sample Query: `sum(irate(node_disk_bytes_read[1m])) by (instance)`

Labels:

- Label: `instance`

  Label Description: the virtual machine/instance

  Example: master-0, control-0, dra-director-1, etc

- Label: `job`

  Label Description: the name of job

  Example: node_exporter

- Label: `device`

  Label Description: the name of the disk device

  Example: vdb, vdd, sr0

node_disk_read_time_seconds_total

Description: This metrics gives the total number of seconds spent by all reads

Metric Type:

Data Type:

Sample Query: `sum(irate(node_disk_read_time_seconds_total[1m])) by (instance) / sum(irate(node_disk_reads_completed_total[1m])) by (instance)`

Labels:

- Label: `instance`

  Label Description: the virtual machine/instance

  Example: master-0, control-0, dra-director-1, etc

- Label: `job`

  Label Description: the name of job

  Example: node_exporter

- Label: `device`

  Label Description: the name of the disk device

  Example: vdb, vdd, sr0

node_disk_reads_completed_total

Description: This metrics gives the total number of reads completed successfully.

Metric Type:

Data Type:

Sample Query: `sum(irate(node_disk_reads_completed_total[1m])) by (instance)`

Labels:

- Label: `instance`

  Label Description: the virtual machine/instance

  Example: master-0, control-0, dra-director-1, etc

Labels:

- Label: `job`

  Label Description: the name of job

  Example: node_exporter

- Label: `device`

  Label Description: the name of the disk device

  Example: vdb, vdd, sr0

node_disk_write_time_seconds_total

Description: This metrics gives the total number of seconds spent by all writes

Metric Type:

Data Type:

Sample Query: `sum(irate(node_disk_write_time_seconds_total[1m])) by (instance) / sum(irate(node_disk_writes_completed_total[1m])) by (instance)`

Labels:

- Label: `instance`

  Label Description: the virtual machine/instance

  Example: master-0, control-0, dra-director-1, etc

Labels:

- Label: `job`

  Label Description: the name of job

  Example: node_exporter

Labels:

- Label: `device`

  Label Description: the name of the disk device

  Example: vdb, vdd, sr0

node_disk_writes_completed_total

Description: This metrics gives the total number of writes completed successfully.

Metric Type:

Data Type:

Sample Query: `sum(irate(node_disk_writes_completed[1m])) by (instance)`

Labels:

- Label: `instance`

  Label Description: the virtual machine/instance

  Example: master-0, control-0, dra-director-1, etc

Labels:

- Label: `job`

  Label Description: the name of job

  Example: node_exporter

Labels:

- Label: `device`

  Label Description: the name of the disk device

  Example: vdb, vdd, sr0

node_disk_written_bytes_total

Description: This metrics gives the total number of bytes written successfully.

Metric Type:

Data Type:

Sample Query: `sum(irate(node_disk_written_bytes_total[1m])) by (instance)`

Labels:

- Label: `instance`

  Label Description: the virtual machine/instance

  Example: master-0, control-0, dra-director-1, etc

Labels:

- Label: `job`

  Label Description: the name of job

  Example: node_exporter

Labels:

- Label: `device`

  Label Description: the name of the disk device

  Example: vdb, vdd, sr0

Disk Utilization

Description: Disk storage is a critical system resource, and when file system utilization exceeds 90% utilization the system can become less efficient. When the file system utilization hits 100%, then application can stop functioning.

Metric Type:

Data Type:

Sample Query: `df.<fs>.df_complex.free - df.<fs>.df_complex.used`

Warning Threshold:

- > 80% utilization.

Major Threshold:

- > 90% utilization

File System Category

node_filesystem_free_bytes

Description: This metrics gives the total number of bytes of the free disk space available on the instance

Metric Type:

Data Type:

Sample Query: `sum(node_filesystem_free_bytes{mountpoint=\"/data\"}) by (device, instance)`

Labels:

- Label: `instance`

  Label Description: the virtual machine/instance

  Example: master-0, control-0, dra-director-1, etc

- Label: `job`

  Label Description: the name of job

  Example: node_exporter

- Label: `device`

  Label Description: the name of the disk device

  Example: /dev/vda3, /dev/vdb

- Label: `fstype`

  Label Description: the file system type

  Example: ext4

- Label: `mountpoint`

  Label Description: the file system mount directory

  Example: /data, /tootfs

node_filesystem_size_bytes

Description: This metrics gives the total number of bytes of the total disk space provisioned on the instance

Metric Type:

Data Type:

Sample Query: `sum(node_filesystem_size_bytes{mountpoint=\"/data\"}) by (device, instance)`

Labels:

- Label: `instance`

  Label Description: the virtual machine/instance

  Example: master-0, control-0, dra-director-1, etc

- Label: `job`

  Label Description: the name of job

  Example: node_exporter

- Label: `device`

  Label Description: the name of the disk device

  Example: /dev/vda3, /dev/vdb

- Label: `fstype`

Label Description: the file system type

Example: ext4

- Label: `mountpoint`

  Label Description: the file system mount directory

  Example: /data, /tootfs

Load Category

node_load1

Description: This metrics gives the 1m load average.

Metric Type: Gauge

Data Type: Float

Sample Query: `avg(irate(node_load1[1m])) by (instance)`

Labels:

- Label: `instance`

  Label Description: the virtual machine/instance

  Example: master-0, control-0, dra-director-1, etc

- Label: `job`

  Label Description: the name of job

  Example: node_exporter

node_load15

Description: This metrics gives the 15m load average.

Metric Type: Gauge

Data Type: Float

Sample Query: `avg(irate(node_load15[1m])) by (instance)`

Labels:

- Label: `instance`

  Label Description: the virtual machine/instance

  Example: master-0, control-0, dra-director-1, etc

- Label: `job`

  Label Description: the name of job

  Example: node_exporter

node_load5

Description: This metrics gives the 5m load average.

Metric Type: Gauge

Data Type: Float

Sample Query: `avg(irate(node_load5[1m])) by (instance)`

Labels:

- Label: `instance`

  Label Description: the virtual machine/instance

  Example: master-0, control-0, dra-director-1, etc

Labels:

- Label: `job`

  Label Description: the name of job

  Example: node_exporter

Memory Category

node_memory_MemFree_bytes

Description: This metrics gives the total number of bytes of the free memory available on the node

Metric Type:

Data Type:

Sample Query: `sum(node_memory_MemFree_bytes) by (instance)`

Labels:

- Label: `instance`

  Label Description: the virtual machine/instance

  Example: master-0, control-0, dra-director-1, etc

- Label: `job`

  Label Description: the name of job

  Example: node_exporter

node_memory_MemTotal_bytes

Description: This metrics gives the total number of bytes of the total memory provisioned on the node

Metric Type:

Data Type:

Sample Query: `sum(node_memory_MemTotal_bytes) by (instance)`

Labels:

- Label: `instance`

  Label Description: the virtual machine/instance

  Example: master-0, control-0, dra-director-1, etc

- Label: `job`

  Label Description: the name of job

  Example: node_exporter

Memory Utilization

Description: Memory is a system resource, which needs to be less than 80%. The swap threshold has been reduced for cnAAA, and swapping should occur when the system resources are exhausted and memory utilization hits 99%.

Metric Type:

Data Type:

Sample Query: `memory.free - memory.used`

Warning Threshold:

- > 70% utilization over 60 second period.

Major Threshold:

- > 80% utilization over 60 second period.

Network Category

node_network_receive_bytes_total

Description: This metrics gives the total number of bytes received over the network device

Metric Type:

Data Type:

Sample Query: `sum(irate(node_network_receive_bytes_total[1m])) by (device)`

Labels:

- Label: `instance`

  Label Description: the virtual machine/instance

  Example: master-0, control-0, dra-director-1, etc

- Label: `job`

  Label Description: the name of job

  Example: node_exporter

- Label: `device`

Label Description: the name of the network device/interface

Example: ens3, ens4

node_network_transmit_bytes_total

Description: This metrics gives the total number of bytes sent over the network device

Metric Type:

Data Type:

Sample Query: `sum(irate(node_network_transmit_bytes_total[1m])) by (device)`

Labels:

- Label: `instance`

  Label Description: the virtual machine/instance

  Example: master-0, control-0, dra-director-1, etc

- Label: `job`

  Label Description: the name of job

  Example: node_exporter

- Label: `device`

  Label Description: the name of the network device/interface

  Example: ens3, ens4

Radius Endpoint Requests Category

AddSubscriberService

Description: Total count of subscribers successfully added to a service.

Formula: `action_total{node_type="unknown",type="add-subscriber-service",status="success"}`

UpdateSubscriberService

Description: Total count of subscribers successfully updated in a service.

Formula: `action_total{node_type="unknown",type="update-subscriber-service",status="success"}`

DeleteSubscriberService

Description: Total count of subscribers successfully removed from a service.

Formula: `action_total{node_type="unknown",type="delete-subscriber-service",status="success"}`

GetSubscriberService

Description: Total count of successful retrieval operations for subscriber actions within the service.

Formula: `action_total{node_type="unknown",type="get-subscriber-action-impl",status="success"}`

createBulkSubscribers

Description: Total count of bulk subscriber creation actions successfully executed.

Formula: `action_total{node_type="unknown",type="create-bulk-subscribers",status="success"}`

getBulkSubscribers

Description: Total count of successful retrieval operations for bulk subscriber details.

Formula: `action_total{node_type="unknown",type="get-bulk-subscribers",status="success"}`

updateBulkSubscribers

Description: Total count of bulk subscriber update actions successfully executed.

Formula: `action_total{node_type="unknown",type="update-bulk-subscribers",status="success"}`

deleteBulkSubscribers

Description: Total count of bulk subscriber deletion actions successfully executed.

Formula: `action_total{node_type="unknown",type="delete-bulk-subscribers",status="success"}`

ProvisionedSubscriberCount

Description: Total count of provisioned subscribers accurately tallied in the system.

Formula: `action_total{node_type="unknown",type="provisioned-subscriber-count",status="success"}`

UpdateSubscriberServiceInSeconds

Description: Duration (in seconds) for the update-subscriber-service action to complete.

Formula: `action_duration_seconds{node_type="unknown",type="update-subscriber-service"}`

GetSubscriberServiceInSeconds

Description: Duration (in seconds) for the get-subscriber-action-impl action to complete.

Formula: `action_duration_seconds{node_type="unknown",type="get-subscriber-action-impl"}`

DeleteSubscriberServiceInSeconds

Description: Duration (in seconds) for the delete-subscriber-service action to complete.

Formula: `action_duration_seconds{node_type="unknown",type="delete-subscriber-service"}`

ProvisionedSubscriberCountInSeconds

Description: Duration (in seconds) for the provisioned-subscriber-count action to complete.

Formula: `action_duration_seconds{node_type="unknown",type="provisioned-subscriber-count"}`

BlockedSubscriberForAccessRequest

Description: Total count of subscribers blocked for an access request.

Formula: `action_total{node_type="unknown",type="blocked-subscriber-for-access-request ",status="success"}`

BlockedSubscriberForSPRUnaviliavility

Description: Total count of subscribers blocked due to service provider resource (SPR) unavailability.

Formula:
`action_total{node_type="unknown",type="blocked-subscriber-for-s-p-r-unavailability",status="success"}`

ProxyAccountingStartRequest

Description: Total count of session accounting requests with a start status sent to the proxy.

Formula:
`radius_proxy_accounting_request_total{accounting_type="ServiceAccounting",status_type="Start", AAAServer="PassiveMZ-12997",endpoint_ip="192.168.74.76"}`

ProxyAccountingStartResponse

Description: Total count of service accounting responses with a start status and result timeout received from the proxy.

Formula: `radius_proxy_accounting_response_total{accounting_type="ServiceAccounting", status_type="Start",AAAServer="kolkata_12100",result="Timeout",server_ip="NA",endpoint_ip="192.168.202.251", timeout_ip1="11.11.11.11",timeout_ip2="12.12.12.12",retries="NA"}`

ProxyAccountingStartRetransmit

Description: Total count of retransmissions for session accounting, interim-update, and stop accounting response types, with result timeout.

Formula:
radius_proxy_accounting_response_total{accounting_type="SessionAccounting",status_type="Start",AAAServer="kolkata_12100",result="TIMEOUT",server_ip="NA",endpoint_ip="192.168.117.210",timeout_ip1="10.197.98.181",timeout_ip2="10.197.98.182",retries="6"}
`radius_proxy_accounting_response_total{accounting_type="SessionAccounting", status_type="Interim-Update",AAAServer="kolkata_12100",result="TIMEOUT",server_ip="NA", endpoint_ip="192.168.117.210",timeout_ip1="10.197.98.181",timeout_ip2="10.197.98.182",retries="6"}`
`radius_proxy_accounting_response_total{accounting_type="SessionAccounting", status_type="Stop",AAAServer="kolkata_12100",result="TIMEOUT",server_ip="NA", endpoint_ip="192.168.117.210",timeout_ip1="10.197.98.181",timeout_ip2="10.197.98.182",retries="6"}`

ProxyAccountingStartResponse (Failure case)

Description: Total count of service accounting responses with a start status and result error received from the proxy.

Formula: `radius_proxy_accounting_response_total{accounting_type="ServiceAccounting", status_type="Start",AAAServer="kolkata_12100",result="ERROR",server_ip="NA",endpoint_ip="192.168.202.251", timeout_ip1="11.11.11.11",timeout_ip2="12.12.12.12",retries="NA"}`

ProxyAccountingStartTimeout

Description: Total count of service accounting responses with a start status and result timeout received from the proxy.

Formula: `radius_proxy_accounting_response_total{accounting_type="ServiceAccounting", status_type="Start",AAAServer="kolkata_12100",result="Timeout",server_ip="NA",endpoint_ip="192.168.202.251", timeout_ip1="11.11.11.11",timeout_ip2="12.12.12.12",retries="NA"}`

ProxyAccountingInterim-UpdateRequest

Description: Total count of service accounting requests with an interim-update status sent to the proxy.

Formula: `radius_proxy_accounting_request_total{accounting_type="ServiceAccounting", status_type="Interim-Update",AAAServer="DEL_OCS",endpoint_ip="192.168.74.76"}`

ProxyAccountingInterim-UpdateResponse

Description: Total count of session accounting responses with an interim-update status and result success received from the proxy.

Formula: `radius_proxy_accounting_response_total{accounting_type="SessionAccounting", status_type="Interim-Update",AAAServer="kolkata_12100",result="Success",server_ip="10.197.98.180", endpoint_ip="192.168.116.13",timeout_ip1="NA",timeout_ip2="NA",retries="0"}`

ProxyAccountingInterim-UpdateRetransmit N

Description: Total count of retransmissions for session accounting responses with an interim-update status and result timeout.

Formula: `radius_proxy_accounting_response_total{accounting_type="SessionAccounting", status_type="Interim-Update",AAAServer="kolkata_12100",result="TIMEOUT",server_ip="NA", endpoint_ip="192.168.117.210",timeout_ip1="10.197.98.181",timeout_ip2="10.197.98.182",retries="6"}`

ProxyAccountingInterim-UpdateResponse (Failure case)

Description: Total count of service accounting responses with an interim-update status and result error received from the proxy.

Formula: `radius_proxy_accounting_response_total{accounting_type="ServiceAccounting", status_type="Interim-Update",AAAServer="kolkata_12100",result="ERROR",successIP="NA", endpoint_ip="192.168.202.251",timeout_ip1="11.11.11.11",timeout_ip2="12.12.12.12",retries="NA"}`

ProxyAccountingInterim-UpdateTimeout

Description: Total count of session accounting responses with an interim-update status and result success received from the proxy.

Formula:

ProxyAccountingStopRequest

Description: Total count of service accounting requests with a stop status sent to the proxy.

Formula:
radius_proxy_accounting_request_total{accounting_type="ServiceAccounting",status_type="Stop",AAAServer="DEL_OCS",endpoint_ip="192.168.74.76"}

ProxyAccountingStopResponse

Description: Total count of session accounting responses with a stop status and result success received from the proxy.

Formula: `radius_proxy_accounting_response_total{accounting_type="SessionAccounting", status_type="Stop",AAAServer="kolkata_12100",result="Success",server_ip="10.197.98.180", endpoint_ip="192.168.116.13",timeout_ip1="NA",timeout_ip2="NA",retries="0"}`

ProxyAccountingStopRetransmit N

Description: Total count of retransmissions for session accounting responses with a stop status and result timeout.

Formula:
`radius_proxy_accounting_response_total{accounting_type="SessionAccounting",status_type="Stop", AAAServer="kolkata_12100",result="TIMEOUT",server_ip="NA",endpoint_ip="192.168.117.210", timeout_ip1="10.197.98.181",timeout_ip2="10.197.98.182",retries="6"}`

ProxyAccountingStopResponse (Failure case)

Description: Total count of session accounting responses with a stop status and result error received from the proxy.

Formula: `radius_proxy_accounting_response_total{accounting_type="ServiceAccounting", status_type="Stop",AAAServer="kolkata_12100",result="ERROR",successIP="NA",endpoint_ip="192.168.202.251", timeout_ip1="11.11.11.11",timeout_ip2="12.12.12.12",retries="NA"}`

ProxyAccountingStopTimeout

Description: Total count of service accounting responses with a stop status and result timeout received from the proxy.

Formula: `radius_proxy_accounting_response_total{accounting_type="ServiceAccounting", status_type="Interim-Update",AAAServer="kolkata_12100",result="ERROR",successIP="NA", endpoint_ip="192.168.202.251",timeout_ip1="11.11.11.11",timeout_ip2="12.12.12.12",retries="NA"}`

ProxyAccountingInterim-UpdateResponsedrop

Description: Total count of service accounting responses with an interim-update status and result drop received from the proxy.

Formula: `radius_proxy_accounting_response_total{accounting_type="ServiceAccounting", status_type="Start",AAAServer="kolkata_12100",result="ERROR",successIP="NA",endpoint_ip="192.168.202.251",`

timeout_ip1="11.11.11.11",timeout_ip2="12.12.12.12",retries="NA"}

ProxyAccountingStartResponsedrop

Description: Total count of service accounting responses with a start status and result drop received from the proxy.

Formula: radius_proxy_accounting_response_total{accounting_type="ServiceAccounting", status_type="Start",AAAServer="DEL_OCS",result="DROP",server_ip="NA",endpoint_ip="192.168.195.127", timeout_ip1="NA",timeout_ip2="NA",retries="0"}

ProxyAccountingStopResponsedrop

Description: Total count of service accounting responses with a stop status and result drop received from the proxy.

Formula: radius_proxy_accounting_response_total{accounting_type="ServiceAccounting", status_type="Stop",AAAServer="DEL_OCS",result="DROP",server_ip="NA",endpoint_ip="192.168.195.127", timeout_ip1="NA",timeout_ip2="NA",retries="0"}

SubscribersPerService_Service_Name

Description: Total count of subscribers for a specified service, where Service_Name denotes a particular service.

Formula: action_total{node_type="unknown",type="subscribersperservice_a0f0002m002m000005mq",status="success"}

MaxAvailableRadiusSession

Description: Maximum available RADIUS session count.

Formula: action_total{node_type="unknown",type="max-available-radius-session",status="success"}

ProvisionedSubscriberCount

Description: Total count of provisioned subscribers.

Formula: action_total{node_type="unknown",type="provisioned-subscriber-count",status="success"}

ActiveSubscriberCount

Description: Total count of active subscribers stored in the session database.

Formula: db_records_total{appInstanceId="0",app_name="datastore-ep",cdl_slice="session", cluster="session",data_center="test",db="session",instance_id="3232300165",service_name="datastore-ep",session_type="total",systemId="1"} 0

ServiceAccountingStartRequest

Description: Total count of service accounting requests with a start status sent to the proxy.

Formula: radius_accounting_request_total{accounting_type="ServiceAccounting", status_type="Start",clientIp="192.168.68.192",endPointIp="192.168.74.76",result="SUCCESS"}

SessionAccountingStartRequest

Description: Total count of session accounting requests with a start status sent to the proxy.

Formula:
`radius_accounting_request_total{accounting_type="SessionAccounting",status_type="Start",clientIp="192.168.68.192",endPointIp="192.168.74.76",result="SUCCESS"}`

ServiceAccountingStartResponse

Description: Total count of service accounting responses with a start status sent from the proxy.

Formula: `radius_accounting_response_total{accountingType="ServiceAccounting",statusType="Start",clientIp="192.168.41.128",endPointIp="192.168.109.183"}`

SessionAccountingStartResponse

Description: Total count of session accounting responses with a start status sent from the proxy.

Formula: `radius_accounting_response_total{accountingType="SessionAccounting",statusType="Start",clientIp="192.168.41.128",endPointIp="192.168.109.183"}`

ServiceAccountingInterim-UpdateRequest

Description: Total count of service accounting requests with an interim-update status sent to the proxy.

Formula:
`radius_accounting_request_total{accounting_type="ServiceAccounting",status_type="Interim-Update",clientIp="192.168.68.192",endPointIp="192.168.74.76",result="SUCCESS"}`

SessionAccountingInterim-UpdateRequest

Description: Total count of session accounting requests with an interim-update status sent to the proxy.

Formula: `radius_accounting_request_total{accounting_type="SessionAccounting",status_type="Interim-Update",clientIp="192.168.68.192",endPointIp="192.168.74.76",result="SUCCESS"}`

ServiceAccountingInterim-UpdateResponse

Description: Total count of service accounting responses with an interim-update status received from the proxy.

Formula: `radius_accounting_response_total{accountingType="ServiceAccounting",statusType="Interim-Update",clientIp="192.168.41.128",endPointIp="192.168.109.183"`

SessionAccountingInterim-UpdateResponse

Description: Total count of session accounting responses with an interim-update status received from the proxy.

Formula: `radius_accounting_response_total{accountingType="SessionAccounting",statusType="Interim-Update",clientIp="192.168.41.128",endPointIp="192.168.109.183"}`

ServiceAccountingStopRequest_terminationcause

Description: Total count of service accounting requests with a stop status sent to the proxy, including termination cause details.

Formula: `radius_accounting_request_total{accounting_type="ServiceAccounting", status_type="Stop",clientIp="192.168.68.192",endPointIp="192.168.74.76",result="SUCCESS"}`

SessionAccountingStopRequest_terminationcause

Description: Total count of session accounting requests with a stop status sent to the proxy.

Formula: `radius_accounting_request_total{accounting_type="SessionAccounting", status_type="Stop",clientIp="192.168.68.192",endPointIp="192.168.74.76",result="SUCCESS"}`

ServiceAccountingStopResponse_terminationcause

Description: Total count of service accounting responses with a stop status received from the proxy.

Formula: `radius_accounting_response_total{accountingType="ServiceAccounting", statusType="Stop",clientIp="192.168.41.128",endPointIp="192.168.109.183"}`

SessionAccountingStopResponse_terminationcause

Description: Total count of session accounting responses with a stop status received from the proxy.

Formula: `radius_accounting_response_total{accountingType="SessionAccounting", statusType="Stop",clientIp="192.168.41.128",endPointIp="192.168.109.183"}`

ServiceAccountingStartTimeInSeconds

Description: Total duration in seconds for service accounting responses with a "start" status.

Formula: `radius_accounting_response_seconds_total{accountingType="ServiceAccounting",statusType="Start", clientIp="192.168.41.128",endPointIp="192.168.109.183"}`

ServiceAccountingInterim-UpdateTimeInSeconds

Description: Total duration in seconds for service accounting responses with an "interim-update" status.

Formula: `radius_accounting_response_seconds_total{accountingType="ServiceAccounting", statusType="Interim-Update",clientIp="192.168.41.128",endPointIp="192.168.109.183"}`

ServiceAccountingStopTimeInSeconds

Description: Total duration in seconds for service accounting responses with a "stop" status.

Formula: `radius_accounting_response_seconds_total{accountingType="ServiceAccounting", statusType="Stop",clientIp="192.168.41.128",endPointIp="192.168.109.183"}`

SessionAccountingStartTimeInSeconds

Description: Total duration in seconds for session accounting responses with a "start" status.

Formula: `radius_accounting_response_seconds_total{accountingType="SessionAccounting", statusType="Start",clientIp="192.168.41.128",endPointIp="192.168.109.183"}`

SessionAccountingInterim-UpdateTimeInSeconds

Description: Total duration in seconds for session accounting responses with an "interim-update" status.

Formula: `radius_accounting_response_seconds_total{accountingType="SessionAccounting", statusType="Interim-Update",clientIp="192.168.41.128",endPointIp="192.168.109.183"}`

SessionAccountingStopTimeInSeconds

Description: Total duration in seconds for session accounting responses with a "stop" status.

Formula: `radius_accounting_response_seconds_total{accountingType="SessionAccounting", statusType="Stop",clientIp="192.168.41.128",endPointIp="192.168.109.183"}`

CoARequest

Description: Total number of CoA request messages sent.

Formula: `radius_requests_total{message_type="CoARequest",nas_ip_address="192.168.23.123", client_ip_address="192.168.23.123",endpoint_address="192.168.219.201",result="SUCCESS"}`

CoANAK

Description: Total number of CoA NAK response messages received.

Formula: `radius_responses_total{message_type="CoANAKResponse",nas_ip_address="10.197.98.180", client_ip_address="10.197.98.180",endpoint_address="192.168.159.32"}`

CoATimeout

Description: Total number of CoA request timeouts.

Formula: `radius_request_timeout_total{message_type="CoaRequest",nas_ip_address="192.168.104.50", client_ip_address="192.168.104.50",endpoint_address="192.102.11.126"}`

CoAResponse

Description: Total number of CoA ACK response messages received.

Formula: `radius_responses_total{message_type="CoAACKResponse",nas_ip_address="192.168.41.137", client_ip_address="192.168.41.137",endpoint_address="192.168.109.183"}`

AccessRequest

Description: Total number of Access-Request messages with a successful result.

Formula: `radius_requests_total{message_type="AccessRequest",nas_ip_address="192.168.23.123", client_ip_address="192.168.23.64",endpoint_address="192.168.219.201",result="SUCCESS"}`

AccessRequest

Description: Total number of Access-Request messages that were dropped.

Formula: `radius_requests_total{message_type="AccessRequest",nas_ip_address="192.168.23.123",client_ip_address="192.168.23.64",endpoint_address="192.168.219.201",result="DROP"}`

AccessAccept

Description: Total number of Access-Accept responses received.

Formula: `radius_responses_total{message_type="AccessAccept",nas_ip_address="192.168.41.137",client_ip_address="192.168.41.128",endpoint_address="192.168.109.183"}`

AccessReject

Description: Total number of Access-Reject responses received.

Formula: `radius_responses_total{message_type="AccessReject",nas_ip_address="192.168.41.137",client_ip_address="192.168.41.128",endpoint_address="192.168.159.32"}`

spr.getSubscriberAuthAttmpts.qns_stat.error

Description: Total number of errors for the "get-subscriber-auth-attempts-action-impl."

spr.getSubscriberAuthAttmpts.qns_stat.success

Description: Total number of successful attempts for "get-subscriber-auth-attempts-action-impl."

Formula: `action_total{node_type="unknown",type="get-subscriber-auth-attmpts-action-impl",status="success"}`

spr.getSubscriber.qns_stat.error

Description: Total number of errors for the "get-subscriber-action-impl."

Formula: `action_total{node_type="unknown",type="get-subscriber-action-impl",status="error"}`

spr.getSubscriber.qns_stat.success

Description: Total number of successful attempts for "get-subscriber-action-impl."

Formula: `action_total{node_type="unknown",type="get-subscriber-action-impl",status="success"}`

spr.removeSubscriberAuthAttmpts.qns_stat.error

Description: Total number of errors for the "remove-subscriber-action-impl."

spr.removeSubscriberAuthAttmpts.qns_stat.success

Description: Total number of successful attempts for "remove-subscriber-action-impl."

spr.searchSubscribers.qns_stat.error

Description: Total count of "search-subscriber" actions with an error status.

spr.searchSubscribers.qns_stat.success

Description: Total count of "search-subscriber" actions with a success status.

spr.validate.qns_stat.error

Description: Total count of "validate" actions with an error status.

spr.validate.qns_stat.success

Description: Total count of "validate" actions with a success status.

actions.AddSubscriberService.qns_stat.error

Description: Total count of "add-subscriber-service" actions with an error status.

actions.AddSubscriberService.qns_stat.success

Description: Total count of "add-subscriber-service" actions with a success status.

Formula: `action_total{node_type="unknown",type="add-subscriber-service",status="success"}`

actions.UpdateSubscriberService.qns_stat.error

Description: Total count of "update-subscriber-service" actions with an error status.

Formula:

actions.UpdateSubscriberService.qns_stat.success

Description: Total count of "update-subscriber-service" actions with a success status.

Formula: `action_total{node_type="unknown",type="update-subscriber-service",status="success"}`

actions.DeleteSubscriberService.qns_stat.error

Description: Total count of "delete-subscriber-service" actions with an error status.

actions.DeleteSubscriberService.qns_stat.success

Description: Total count of "delete-subscriber-service" actions with a success status.

Formula: `action_total{node_type="unknown",type="delete-subscriber-service",status="success"}`

spr.createSubscriber.qns_stat.total_time_in_ms

Description: Total time in milliseconds for "create-subscriber" actions.

Formula: `action_duration_seconds{node_type="unknown",type="create-subscriber"}`

spr.deleteSubscriber.qns_stat.total_time_in_ms

Description: Total time in milliseconds for "delete-subscriber" actions.

Formula: `action_duration_seconds{node_type="unknown",type="delete-subscriber"}`

spr.getSubscriberAuthAttmpts.qns_stat.total_time_in_ms

Description: Total time in milliseconds for "get-subscriber-auth-attempts" actions.

Formula:
`action_duration_seconds{node_type="unknown",type="get-subscriber-auth-attmpts-action-impl"}`

spr.getSubscriber.qns_stat.total_time_in_ms

Description: Total time in milliseconds for "get-subscriber" actions.

Formula: `action_duration_seconds{node_type="unknown",type="get-subscriber-action-impl"}`

spr.removeSubscriberAuthAttmpts.qns_stat.total_time_in_ms

Description: Total time in milliseconds for "remove-subscriber-auth-attempts" actions.

spr.searchSubscribers.qns_stat.total_time_in_ms

Description: Total time in milliseconds for "search-subscriber" actions.

spr.updateSubscriber.qns_stat.total_time_in_ms

Description: Total time in milliseconds for "update-subscriber" actions.

Formula: `action_duration_seconds{node_type="unknown",type="update-subscriber"}`

spr.validate.qns_stat.total_time_in_ms

Description: Total time in milliseconds for "validate" actions.

Formula:

actions.AddSubscriberService.qns_stat.total_time_in_ms

Description: Total time in milliseconds for "add-subscriber-service" actions.

Formula: `action_duration_seconds{node_type="unknown",type="add-subscriber-service"}`

actions.IAsyncCoARequest.qns_stat.error

Description: Tracks errors in the "i-async-co-a-request" process, indicating failures in asynchronous CoA requests.

Formula:

actions.IAsyncCoARequest.qns_stat.success

Description: Tracks successful "i-async-co-a-request" actions.

Formula: `action_total{node_type="unknown",type="i-async-co-a-request",status="success"}`

actions.ISendAccessAccept.qns_stat.error

Description: Captures errors in the "i-send-access-accept" action, indicating failure in sending Access-Accept messages.

actions.ISendAccessAccept.qns_stat.success

Description: Tracks successful "i-send-access-accept" actions.

Formula: `action_total{node_type="unknown",type="i-send-access-accept",status="success"}`

actions.ISendAccessReject.qns_stat.error

Description: Captures errors in the "i-send-access-reject" action, indicating failure in sending Access-Reject messages.

Formula:

actions.ISendAccessReject.qns_stat.success

Description: Tracks successful "i-send-access-reject" actions.

Formula: `action_total{node_type="unknown",type="i-send-access-reject",status="success"}`

actions.ISendAccountingResponse.qns_stat.error

Description: Captures errors in the "i-send-accounting-response" action, indicating failure in sending accounting responses.

actions.ISendAccountingResponse.qns_stat.success

Description: Tracks successful "i-send-accounting-response" actions.

Formula: `action_total{node_type="unknown",type="i-send-accounting-response",status="success"}`

actions.ISendBundledCoA.qns_stat.error

Description: Captures errors in the "i-send-bundled-co-a" action, indicating failures in sending bundled CoA requests.

Formula:

actions.ISendBundledCoA.qns_stat.success

Description: Tracks successful "i-send-bundled-co-a" actions.

Formula: `action_total{node_type="unknown",type="i-send-bundled-co-a",status="success"}`

actions.ISendBundledProxyAccounting.qns_stat.error

Description: Captures errors in the "i-send-bundled-proxy-accounting" action, indicating failures in sending proxy accounting messages.

actions.ISendBundledProxyAccounting.qns_stat.success

Description: Records successful executions of the "i-send-bundled-proxy-accounting" action.

Formula:
`action_total{node_type="unknown",type="i-send-bundled-proxy-accounting",status="success"}`

actions.IRemoveSessionAction.qns_stat.error

Description: Logs errors in the "i-remove-session-action" process, indicating failures in session removal.

Formula:

actions.IRemoveSessionAction.qns_stat.success

Description: Records successful executions of the "i-remove-session-action" action.

Formula: `action_total{node_type="unknown",type="i-remove-session-action",status="success"}`

actions.GetRadiusDeviceInformation.qns_stat.error

Description: Logs errors in the "get-radius-device-information" action, indicating failures in retrieving RADIUS device information.

actions.GetRadiusDeviceInformation.qns_stat.success

Description: Records successful executions of the "get-radius-device-information" action.

Formula:
`action_total{node_type="unknown",type="get-radius-device-information",status="success"}`

messages.AsynchCoAResponse.qns_stat.error

Description: Logs errors in "asynch-co-a-response" messages, indicating failures in the CoA call flow.

Formula: `message_total{node_type="unknown",type="asynch-co-a-response",status="error"}`

messages.AsynchCoAResponse.qns_stat.success

Description: Records successful executions of "asynch-co-a-response" messages.

Formula: `message_total{node_type="unknown",type="asynch-co-a-response",status="success"}`

messages.RadiusAccessRequestMessage.qns_stat.error

Description: Logs errors in the "radius-access-request-message," indicating failures.

Formula: `message_total{node_type="unknown",type="radius-access-request-message",status="error"}`

messages.RadiusAccessRequestMessage.qns_stat.success

Description: Records successful executions of "radius-access-request-message" actions, part of the basic call flow with incorrect configuration.

Formula:
`message_total{node_type="unknown",type="radius-access-request-message",status="success"}`

messages.RadiusAccountingMessage.qns_stat.error

Description: Logs errors in the "radius-accounting-message," indicating failures.

Formula: `message_total{node_type="unknown",type="radius-accounting-message",status="error"}`

messages.RadiusAccountingMessage.qns_stat.success

Description: Records successful executions of "radius-accounting-message" actions.

Formula: `message_total{node_type="unknown",type="radius-accounting-message",status="success"}`

messages.RefreshSPRProfile.qns_stat.error

Description: Logs errors in "refresh-SPR-profile" messages, typically related to subscriber updates.

messages.RefreshSPRProfile.qns_stat.success

Description: Records successful executions of "refresh-SPR-profile" messages, typically involving live session updates.

messages.RemoveSessionImpl.qns_stat.error

Description: Logs errors in "remove-session-impl" messages, indicating failures in session removal.

messages.RemoveSessionImpl.qns_stat.success

Description: Records successful executions of "remove-session-impl" actions.

Formula: `message_total{node_type="unknown",type="remove-session-impl",status="success"}`

actions.GetRadiusDeviceInformation.qns_stat.total_time_in_ms

Description: Calculates the total time in milliseconds for the "get-radius-device-information" action.

Formula: `action_duration_seconds{node_type="unknown",type="get-radius-device-information"}`

actions.IAsyncCoARequest.qns_stat.total_time_in_ms

Description: Calculates the total time in milliseconds for the "i-async-co-a-request" action.

Formula: `action_duration_seconds{node_type="unknown",type="i-async-co-a-request"}`

actions.IRemoveSessionAction.qns_stat.total_time_in_ms

Description: Calculates the total time in milliseconds for the "i-remove-session-action" action.

Formula: `action_duration_seconds{node_type="unknown",type="i-remove-session-action"}`

actions.ISendAccessAccept.qns_stat.total_time_in_ms

Description: Calculates the total time in milliseconds for the "i-send-access-accept" action.

Formula: `action_duration_seconds{node_type="unknown",type="i-send-access-accept"}`

actions.ISendAccessReject.qns_stat.total_time_in_ms

Description: Calculates the total time in milliseconds for the "i-send-access-reject" action.

Formula: `action_duration_seconds{node_type="unknown",type="i-send-access-reject"}`

actions.ISendAccountingResponse.qns_stat.total_time_in_ms

Description: Records the time taken to send an accounting response.

Formula: `action_duration_seconds{node_type="unknown",type="i-send-accounting-response"}`

actions.ISendBundledCoA.qns_stat.total_time_in_ms

Description: Records the time taken to send a bundled CoA.

Formula: `action_duration_seconds{node_type="unknown",type="i-send-bundled-co-a"}`

actions.ISendBundledProxyAccounting.qns_stat.total_time_in_ms

Description: Calculates the time taken for sending bundled proxy accounting in the proxy call flow.

Formula: `action_duration_seconds{node_type="unknown",type="i-send-bundled-proxy-accounting"}`

messages.AsynchCoAResponse.qns_stat.total_time_in_ms

Description: Records the response time for asynchronous CoA requests.

Formula: `message_duration_seconds{node_type="unknown",type="asynch-co-a-response"}`

messages.RadiusAccessRequestMessage.qns_stat.total_time_in_ms

Description: Calculates the time taken for sending a RADIUS access request message.

Formula: `message_duration_seconds{node_type="unknown",type="radius-access-request-message"}`

messages.RadiusAccountingMessage.qns_stat.total_time_in_ms

Description: Records the time taken for sending a RADIUS accounting message.

Formula: `message_duration_seconds{node_type="unknown",type="radius-accounting-message"}`

messages.RefreshSPRProfile.qns_stat.total_time_in_ms

Description: Calculates the time taken for refreshing the SPR profile.

messages.RemoveSessionImpl.qns_stat.total_time_in_ms

Description: Records the time taken for the Remove Session implementation message.

Formula: `message_total{node_type="unknown",type="remove-session-impl",status="success"}`

input_queue_result

Description: Records the total number of items in the input queue.

Formula: `input_queue_result_total{node_type="unknown"}`

etcd_registry_lookup

Description: Calculates the total number of etcd registry lookups.

Formula: `etcd_registry_lookup_total{node_type="unknown"}`

record_conflict_merge

Description: Total count of record conflict merges.

Formula: `record_conflict_merge_total 0.0`

INBOUND_REQUEST_TOTAL

Description: Total count of inbound AccountingRequest messages.

Formula:
inbound_request_total{message_type="AccountingRequest",client_ip="192.168.21.0",client_port="38995"}

INBOUND_REQUEST_TOTAL

Description: Total count of inbound AccessRequest messages.

Formula:
`inbound_request_total{message_type="AccessRequest",client_ip="192.168.21.0",client_port="25322"}`

OUTGOING_REQUEST_TOTAL

Description: Total count of outbound ProxyAccounting messages.

Formula: `outbound_request_total{message_type="ProxyAccounting",client_ip="192.168.21.0",`

`ocs_server="PassiveMZ-12997",client_port="38995"}`
`outbound_request_total{message_type="ProxyAccounting",client_ip="192.168.21.0",ocs_server="DEL_OCS",`

`client_port="38995"}`

OUTGOING_REQUEST_TOTAL

Description: Total count of outbound CoARequest messages.

Formula:
`outbound_request_total{message_type="CoARequest",client_ip="192.168.21.6",ocs_server="NA",client_port="1700"}`

RADIUS_REQUESTS

Description: Total count of CoARequest messages.

Formula: `radius_requests_total{message_type="CoARequest",nas_ip_address="10.110.196.244",`

`client_ip_address="10.110.196.244",endpoint_address="192.168.117.204"}`

RADIUS_REQUESTS

Description: Total count of AccessRequest messages.

Formula:
`radius_requests_total{message_type="AccessRequest",nas_ip_address="10.110.196.244",client_ip_address="192.168.202.192",endpoint_address="192.168.117.204"}`

RADIUS_RESPONSES

Description: Total count of responses to CoARequest messages.

Formula: `radius_responses_total{message_type="AccessAccept",nas_ip_address="192.168.41.`

`137",client_ip_address="10.1.0.80",endpoint_address="192.168.41.133"}`

RADIUS_RESPONSES

Description: Total count of AccessAccept responses to AccessRequest messages.

Formula: `radius_responses_total{message_type="CoAACKResponse",nas_ip_address="192.168.41.137",`

`client_ip_address="192.168.41.137",endpoint_address="192.168.41.133"}`

RADIUS_RESPONSES

Description: Total count of AccessReject responses to AccessRequest messages.

Formula: `radius_responses_total{message_type="AccessReject",nas_ip_address="192.168.68.198",client_ip_address="192.168.68.192",endpoint_address="192.168.74.76"}`

RADIUS_RESPONSES_SECONDS

Description: Records the time taken for AccessAccept responses to AccessRequest messages.

Formula:
`radius_responses_seconds_total{message_type="AccessAccept",nas_ip_address="192.168.41.137",client_ip_address="10.1.0.80",endpoint_address="192.168.41.133"}`

RADIUS_RESPONSES_SECONDS

Description: Records the time taken for responses to CoARequest messages.

Formula:
`radius_responses_seconds_total{message_type="CoAACKResponse",nas_ip_address="192.168.41.137",client_ip_address="192.168.41.137",endpoint_address="192.168.41.133"}`

RADIUS_RESPONSES_SECONDS

Description: Records the time taken for AccessReject responses to AccessRequest messages.

Formula:
`radius_responses_seconds_total{message_type="AccessReject",nas_ip_address="192.168.68.198",client_ip_address="192.168.68.192",endpoint_address="192.168.74.76"}`

PROCESS_MESSAGE

Description: Total count of AccountingResponse messages.

Formula:
`process_message_total{message_type="AccountingResponse",replyto_address="192.168.202.192"}`

PROCESS_MESSAGE

Description: Total count of AccessAccept messages processed.

Formula: `process_message_total{message_type="AccessAccept",replyto_address="192.168.202.192"}`

PROCESS_MESSAGE

Description: Total count of AccessReject messages processed.

Formula: `process_message_total{message_type="AccessReject",replyto_address="192.168.68.192"}`

PROCESS_MESSAGE_SECONDS

Description: Total processing time in seconds for AccountingResponse messages.

Formula:
`process_message_seconds_total{command_code="AccountingResponse",application="192.168.202.192"}`

PROCESS_MESSAGE_SECONDS

Description: Total processing time in seconds for AccessAccept messages.

Formula:
```
process_message_seconds_total{command_code="AccessAccept",application="192.168.202.192"}
```

PROCESS_MESSAGE_SECONDS

Description: Total processing time in seconds for AccessReject messages.

Formula:
```
process_message_seconds_total{message_type="AccessReject",replyto_address="192.168.68.192"}
```

DISPATCH_MESSAGE

Description: Total count of dispatched AsyncCoARequest messages.

Formula:
```
dispatch_message_total{message_type="AsyncCoARequest",replyto_address="192.168.41.137"}
```

DISPATCH_MESSAGE

Description: Total processing time for BundledCoARequest messages.

Formula:
```
dispatch_message_total{message_type="BundledCoARequest",replyto_address="192.168.41.137"}
```

DISPATCH_MESSAGE_SECONDS

Description: Total processing duration for asynchronous CoA request dispatches in the CoA call flow.

Formula:
```
dispatch_message_seconds_total{message_type="AsyncCoARequest",replyto_address="192.168.41.137"}
```

DISPATCH_MESSAGE_SECONDS

Description: Total processing duration for bundled CoA request dispatches in the CoA call flow.

Formula:
```
dispatch_message_seconds_total{message_type="BundledCoARequest",replyto_address="192.168.41.137"}
```

DISPATCH_ERROR_MESSAGE

Description: Number of errors encountered during asynchronous CoA request dispatch.

Formula: `dispatch_error_total{message_type="AsyncCoARequest",replyto_address="192.168.41.137"}`

DISPATCH_ERROR_MESSAGE

Description: Number of errors encountered during bundled CoA request dispatch.

Formula:
```
dispatch_error_total{message_type="BundledCoARequest",replyto_address="192.168.41.137"}
```

DISPATCH_ERROR_SECONDS

Description: Total time associated with errors during asynchronous CoA request dispatch.

Formula:
`dispatch_error_seconds_total{message_type="AsyncCoARequest",replyto_address="192.168.41.137"}`

DISPATCH_ERROR_SECONDS

Description: Total time associated with errors during bundled CoA request dispatch.

Formula:
`dispatch_error_seconds_total{message_type="BundledCoARequest",replyto_address="192.168.41.137"}`

POLICY_ENGINE_MESSAGE

Description: Number of accounting request messages processed.

Formula:
`policy_engine_message_total{message_type="AccountingRequest",replyto_address="192.168.202.192"}`

POLICY_ENGINE_MESSAGE

Description: Number of access request messages processed.

Formula:
`policy_engine_message_total{message_type="AccessRequest",replyto_address="192.168.202.192"}`

POLICY_ENGINE_MESSAGE_SECONDS

Description: Total processing time for access request messages.

Formula:
`policy_engine_message_seconds_total{message_type="AccessRequest",replyto_address="10.1.0.80"}`

POLICY_ENGINE_MESSAGE_SECONDS

Description: Total processing time for accounting request messages.

Formula:
`policy_engine_message_seconds_total{message_type="AccountingRequest",replyto_address="10.1.0.80"}`

POLICY_ENGINE_TIMEOUT_MESSAGE

Description: Total number of gRPC messages sent for accounting requests.

POLICY_ENGINE_TIMEOUT_MESSAGE

Description: Total number of gRPC messages sent for access requests.

GRPC_MESSAGE_SEND_TOTAL

Description: Number of access reject messages discarded due to overload protection mechanisms.

Formula:
`grpc_message_send_total{message_type="AccountingRequest",replyto_address="192.168.68.192"}`

GRPC_MESSAGE_SEND_TOTAL

Description: Total number of session accounting start messages discarded due to overload protection.

Formula:
```
grpc_message_send_total{message_type="AccessRequest",replyto_address="192.168.202.192"}
```

Total radius auth messages overload rejected

Description: Total number of service accounting start messages discarded due to overload protection.

Formula:
```
total_radius_auth_messages_overload_rejected{message_type="AccessReject",nas_ip_address="10.225.106.50",

client_ip_address="192.168.216.64",endpoint_address="192.168.221.51"}
```

total radius messeages overload dropped

Description: Number of access request messages discarded in the discard or late response flow.

Formula:
```
total_radius_messages_overload_dropped{message_type="SessionAccounting",status_type="Start",

nas_ip_address="10.225.106.50",client_ip_address="192.168.21
6.64",endpoint_address="192.168.221.56"}
```

total radius messeages overload dropped

Description: Number of accounting request messages with late responses in the discard or late response flow.

Formula:
```
total_radius_messages_overload_dropped{message_type="ServiceAccounting",status_type="Start",

nas_ip_address="10.225.106.50",client_ip_address="192.168.21
6.64",endpoint_address="192.168.221.56"}
```

radius_discard_requests_total

Description: Number of access request messages with late responses in the discard or late response flow.

Formula:
```
radius_discard_requests_total{message_type="AccessRequest",nas_ip_address="/192.168.160.69",

client_ip_address="192.168.160.64",endpoint_ip_address="192.168.242.196"}
```

radius_late_responses_total

Description: Time taken for ServiceAccounting responses with status type Stop and result Timeout received for Proxy.

Formula:
```
radius_late_responses_total{message_type="AccountingRequest",nas_ip_address="/192.168.160.69",

client_ip_address="192.168.160.64",endpoint_ip_address="192.168.242.233"}
```

radius_late_responses_total

Description: Time taken for ServiceAccounting responses with status type Interim-Update and result Timeout received for Proxy.

Formula:
```
radius_late_responses_total{message_type="AccessRequest",nas_ip_address="/192.168.160.69",
```

```
client_ip_address="192.168.160.64",endpoint_ip_address="192.168.242.233"}
```

radius_proxy_accounting_response_second_total TIMEOUT Stop

Description: Time taken for ServiceAccounting responses with status type Start and result Timeout received for Proxy.

Formula:

```
radius_proxy_accounting_response_second_total{accounting_type="ServiceAccounting",status_type="Stop",AAAServer="PassiveMZ-12997",
result="TIMEOUT",server_ip="NA",endpoint_ip="192.168.74.76",timeout_ip1="10.1.34.69",timeout_ip2="10.1.34.69",retries="6"}
```

radius_proxy_accounting_response_second_total TIMEOUT Interim-update

Description: Time taken for ServiceAccounting responses with status type Stop and result Error received for Proxy.

Formula:

```
radius_proxy_accounting_response_second_total{accounting_type="ServiceAccounting",status_type="Interim-Update",
AAAServer="PassiveMZ-12997",result="TIMEOUT",server_ip="NA",endpoint_ip="192.168.74.76",
timeout_ip1="10.1.34.69",timeout_ip="10.1.34.69",retries="6"}
```

radius_proxy_accounting_response_second_total TIMEOUT Start

Description: Time taken for ServiceAccounting responses with status type Interim-Update and result Error received for Proxy.

Formula:

```
radius_proxy_accounting_response_second_total{accounting_type="ServiceAccounting",status_type="Start",
AAAServer="PassiveMZ-12997",result="TIMEOUT",server_ip="NA",endpoint_ip="192.168.74.76",
timeout_ip1="10.1.34.69",timeout_ip2="10.1.34.69",retries="6"}
```

radius_proxy_accounting_response_second_total ERROR Stop

Description: Time taken for ServiceAccounting responses with status type Start and result Error received for Proxy.

Formula:

```
radius_proxy_accounting_response_second_total{accounting_type="ServiceAccounting",status_type="Stop",
AAAServer="kolkata_12100",result="ERROR",successIP="NA",endpoint_ip="192.168.202.251",
timeout_ip1="11.11.11.11",timeout_ip2="12.12.12.12",retries="NA"}
```

radius_proxy_accounting_response_second_total ERROR Interim-update

Description: Time taken for ServiceAccounting responses with status type Interim-Update and result Drop received for Proxy.

Formula:

```
radius_proxy_accounting_response_second_total{accounting_type="ServiceAccounting",status_type="Interim-Update",
AAAServer="kolkata_12100",result="ERROR",successIP="NA",endpoint_ip="192.168.202.251",
timeout_ip1="11.11.11.11",timeout_ip2="12.12.12.12",retries="NA"}
```

radius_proxy_accounting_response_second_total ERROR Start

Description: Time taken for ServiceAccounting responses with status type Stop and result Drop received for Proxy.

Formula:
```
radius_proxy_accounting_response_second_total{accounting_type="ServiceAccounting",status_type="Start",
AAAServer="kolkata_12100",result="ERROR",successIP="NA",endpoint_ip="192.168.202.251",
timeout_ip1="11.11.11.11",timeout_ip2="12.12.12.12",retries="NA"}
```

radius_proxy_accounting_response_second_total DROP Interim-Update

Description: Number of access accept messages discarded due to overload protection mechanisms.

Formula:
```
radius_proxy_accounting_response_second_total{accounting_type="ServiceAccounting",status_type="Interim-Update",
AAAServer="DEL_OCS",result="DROP",server_ip="NA",endpoint_ip="192.168.195.127",timeout_ip1="NA",timeout_ip2="NA",retries="0"}
```

radius_proxy_accounting_response_second_total DROP Stop

Description: Number of service accounting interim update messages discarded due to overload protection.

Formula:
```
radius_proxy_accounting_response_second_total{accounting_type="ServiceAccounting",status_type="Stop",AAAServer="DEL_OCS",
result="DROP",server_ip="NA",endpoint_ip="192.168.195.127",timeout_ip1="NA",timeout_ip2="NA",retries="0"}
```

total_radius_messages_overload_dropped

Description: Number of session accounting interim update messages discarded due to overload protection.

Formula: `total_radius_messages_overload_dropped{message_type="AccessAccept",status_type="",
nas_ip_address="10.225.106.50",client_ip_address="192.168.216.64",endpoint_address="192.168.31.180"}`

total_radius_messages_overload_dropped

Description: Number of session accounting stop messages discarded due to overload protection.

Formula:
```
total_radius_messages_overload_dropped{message_type="ServiceAccounting",status_type="Interim-Update",
nas_ip_address="10.225.106.50",client_ip_address="192.168.216.64",endpoint_address="192.168.221.33"}
```

total_radius_messages_overload_dropped

Description: Total number of CoA request messages processed successfully.

Formula:
```
total_radius_messages_overload_dropped{message_type="SessionAccounting",status_type="Interim-Update",
nas_ip_address="10.225.106.50",client_ip_address="192.168.216.64",endpoint_address="192.168.31.180"}
```

total_radius_messages_overload_dropped

Description: Total number of access request messages processed successfully.

Formula:
`total_radius_messages_overload_dropped{message_type="SessionAccounting",status_type="Stop",`
`nas_ip_address="10.225.106.50",client_ip_address="192.168.216.64",endpoint_address="192.168.31.180"}`

radius_requests_total

Description: Total number of access request messages discarded.

Formula: `radius_requests_total{message_type="CoARequest",nas_ip_address="192.168.23.123",`
`client_ip_address="192.168.23.123",endpoint_address="192.168.219.201",result="SUCCESS"}`

radius_requests_total

Description: Response time for service accounting start messages that were discarded.

Formula: `radius_requests_total{message_type="AccessRequest",nas_ip_address="192.168.23.123",`
`client_ip_address="192.168.23.64",endpoint_address="192.168.219.201",result="SUCCESS"}`

radius_proxy_accounting_response_second_total

Description: Number of session accounting start messages discarded due to engine overload.

Formula:
`radius_proxy_accounting_response_second_total{accounting_type="ServiceAccounting",status_type="Start",`
`AAAServer="DEL_OCS",result="DROP",server_ip="NA",endpoint_ip="192.168.195.127",timeout_ip1="NA",timeout_ip2="NA",retries="0"}`

radius_engine_total

Description: Number of session accounting stop messages discarded due to engine overload.

Formula:
`radius_engine_total{node_type="unknown",message_type="Access-Request_REQ-DROP_Engine-Overload"}`

radius_engine_total

Description: Number of accounting update request messages discarded due to engine overload.

Formula:
`radius_engine_total{node_type="unknown",message_type="Session-Accounting-Start_REQ-DROP_Engine-Overload"}`

radius_engine_total

Description: Number of service accounting start messages skipped due to engine overload.

Formula:
`radius_engine_total{node_type="unknown",message_type="Session-Accounting-Stop_REQ-DROP_Engine-Overload"}`

radius_engine_total

Description: Number of service accounting stop messages skipped due to engine overload.

Formula:
`radius_engine_total{node_type="unknown",message_type="Accounting-Update-Request_REQ-DROP_Engine-Overload"}`

radius_engine_total

Description: Number of access request messages discarded due to message overload.

Formula:
```
radius_engine_total{node_type="unknown",message_type="Service-Accounting-Start_REQ-SKIP_Engine-Overload"}
```

radius_engine_total

Description: Number of session accounting start messages discarded due to message overload.

Formula:
```
radius_engine_total{node_type="unknown",message_type="Service-Accounting-Stop_REQ-SKIP_Engine-Overload"}
```

radius_engine_total

Description: Number of session accounting stop messages discarded due to message overload.

Formula:
```
radius_engine_total{node_type="unknown",message_type="Access-Request_REQ-DROP_Message-Overload"}
```

radius_engine_total

Description: Number of accounting update request messages discarded due to message overload.

Formula:
```
radius_engine_total{node_type="unknown",message_type="Session-Accounting-Start_REQ-DROP_Message-Overload"}
```

radius_engine_total

Description: Number of service accounting start messages discarded due to message overload.

Formula:
```
radius_engine_total{node_type="unknown",message_type="Session-Accounting-Stop_REQ-DROP_Message-Overload"}
```

radius_engine_total

Description: Number of service accounting stop messages discarded due to message overload.

Formula:
```
radius_engine_total{node_type="unknown",message_type="Accounting-Update-Request_REQ-DROP_Message-Overload"}
```

radius_engine_total

Description: Number of access request messages removed from the queue due to overload.

Formula:
```
radius_engine_total{node_type="unknown",message_type="Service-Accounting-Start_REQ-DROP_Message-Overload"}
```

radius_engine_total

Description: Number of session accounting start requests dropped in the processing queue due to overload protection.

Formula:
```
radius_engine_total{node_type="unknown",message_type="Service-Accounting-Stop_REQ-DROP_Message-Overload"}
```

radius_engine_total

Description: Number of session accounting stop requests dropped in the processing queue due to engine overload protection.

Formula:
```
radius_engine_total{node_type="unknown",message_type="Access-Request_REQ_in-queue-drop"}
```

radius_engine_total

Description: Number of accounting update requests dropped in the processing queue due to overload protection.

Formula:
```
radius_engine_total{node_type="unknown",message_type="Session-Accounting-Start_REQ_in-queue-drop"}
```

radius_engine_total

Description: Number of service accounting start requests dropped in the queue due to overload protection.

Formula:
```
radius_engine_total{node_type="unknown",message_type="Session-Accounting-Stop_REQ_in-queue-drop"}
```

radius_engine_total

Description: Number of service accounting stop requests dropped in the queue due to engine overload protection.

Formula:
```
radius_engine_total{node_type="unknown",message_type="Accounting-Update-Request_REQ_in-queue-drop"}
```

radius_engine_total

Description: Total number of session accounting interim update requests discarded.

Formula:
```
radius_engine_total{node_type="unknown",message_type="Service-Accounting-Start_REQ_in-queue-drop"}
```

radius_engine_total

Description: Total number of session accounting stop requests discarded.

Formula:
```
radius_engine_total{node_type="unknown",message_type="Service-Accounting-Stop_REQ_in-queue-drop"}
```

radius_accounting_request_total

Description: Total number of session accounting start requests discarded.

Formula:
```
radius_accounting_request_total{accounting_type="SessionAccounting",status_type="Interim-Update",
clientIp="192.168.23.64",endPointIp="192.168.219.201",result="DROP"}
```

radius_accounting_request_total

Description: Total count of session accounting stop requests dropped

Formula:
```
radius_accounting_request_total{accounting_type="SessionAccounting",status_type="Stop",
```

```
clientIp="192.168.23.64",endPointIp="192.168.219.201",result="DROP"}
```

radius_accounting_request_total

Description: Total count of session accounting start requests dropped

Formula:

```
radius_accounting_request_total{accounting_type="SessionAccounting",status_type="Start",
clientIp="192.168.23.64",endPointIp="192.168.219.201",result="DROP"}
```

Additional KPI support for performance benchmarking

Feature History

Feature Name	Release Information	Description
Additional KPI Support for Performance Benchmarking	2025.02.0	The additional KPIs for cnAAA enhance performance monitoring and troubleshooting by providing insights into CoA, Proxy Accounting, Engine, and MongoDB operations.

Overview

The additional KPIs support for cnAAA enhances performance monitoring and troubleshooting capabilities by offering insights into system operations, focusing on components like CoA (Change of Authorization), Proxy Accounting, Engine, and MongoDB operations.

Enhanced KPI Support for

CoA KPIs

These KPIs track Change of Authorization (CoA) operations, providing information on request handling, throttling, timeouts, and NAK responses to enhance network performance and support troubleshooting.

- **CoA Requests**

  Description: Tracks CoA Requests distinctly from Access and Accounting Requests.

  Formula:

  ```
  radius_coa_request_total{message_type="CoARequest",nas_ip_address="192.0.2.1",
  endpoint_address="192.0.2.4",retry_type="NA",result="SUCCESS",}
  ```

- **CoA Throttling**

  Description: Tracks the number of CoA requests throttled.

  Formula:

  ```
  radius_coa_requests_throttled_total{message_type="CoARequestThrottle",nas_ip_address="192.0.2.1",
  endpoint_address="192.0.2.18",}
  ```

- **CoA Request Timeout KPI**

Description: Tracks the number of CoA requests that have timed out from BNG.

Formula:

```
radius_coa_request_timeout_total{message_type="CoaRequest",nas_ip_address="192.0.2.1",
client_ip_address="192.0.2.1",endpoint_address="192.0.2.18",}
```

- **CoA NAK Response KPI**

  Description: Provides insights into CoA NAK responses by adding error causes as labels.

  Formula:

```
radius_coa_responses_total{message_type="CoANAKResponse",
nas_ip_address="192.0.2.1",client_ip_address="192.0.2.1",
endpoint_address="198.51.100.1",nak_error_cause="405",}
```

Proxy Accounting KPIs

These KPIs monitor errors and mismatches, offering metrics on queue full errors and retry outcomes to maintain accounting and support troubleshooting.

- **Proxy Accounting Queue Full Error**

  Monitor errors in proxy accounting where the thread pool queue is full, causing "Start" requests to be rejected for endpoint.

  Formula:

```
radius_proxy_accounting_response_second_total{accounting_type="ServiceAccounting",
status_type="Start",AAAServer="PassiveMZ-12997",result="QUEUE_FULL_ERROR",server_ip="NA",
endpoint_ip="192.168.253.16",timeout_ip1="NA",timeout_ip2="NA",retries="0",}
```

- **Accounting Mismatch Protection KPI**

  Protects against accounting mismatches with specific retry outcome labels.

  Formula:

```
radius_proxy_accounting_response_second_total{accounting_type="ServiceAccounting",
status_type="Start",AAAServer="PassiveMZ-12997",result="QUEUE_FULL_ERROR",server_ip="NA",
endpoint_ip="198.51.100.1",timeout_ip1="NA",timeout_ip2="NA",retries="0"}
```

- **Proxy Accounting Queue Full Error (Stop)**

  Track errors in proxy accounting where the thread pool queue is full, causing "Stop" requests to be rejected for endpoint

  Formula:

```
radius_proxy_accounting_response_second_total{accounting_type="ServiceAccounting",
status_type="Stop",AAAServer="PassiveMZ-12997",result="QUEUE_FULL_ERROR",
server_ip="NA",endpoint_ip="198.51.100.1",timeout_ip1="NA",timeout_ip2="NA",
retries="0"}
```

Accounting Mismatch Protection

These KPIs monitor retry attempts under success or failure conditions, offering insights to prevent accounting mismatches.

- **Start Retry on Failure:**

Track retries on failure for "Start" accounting requests to endpoint, identifying mismatches in proxy accounting.

Formula:

```
radius_proxy_accounting_retry_on_failure_total{accounting_type="ServiceAccounting",
status_type="Start",AAAServer="PassiveMZ-12997",endpoint_ip="198.51.100.2",timeout_ip1="1.1.1.1",
timeout_ip2="203.0.113.1",circle_code="DL",retries="0"}
```

- **Interim Update Retry on Failure**

  Monitor retries on failure for "Interim-Update" accounting requests to endpoint, aiding in mismatch detection.

  Formula:

  ```
  radius_proxy_accounting_retry_on_failure_total{accounting_type="ServiceAccounting",
  status_type="Interim-Update",AAAServer="PassiveMZ-12997",endpoint_ip="198.51.100.2",
  timeout_ip1="203.0.113.1",timeout_ip2="203.0.113.1",circle_code="DL",retries="0"}
  ```

- **Stop Retry on Failure**

  Monitor retries on failure for "Stop" accounting requests to endpoint 198.51.100.2, enhancing protection against accounting mismatches.

  Formula:

  ```
  radius_proxy_accounting_retry_on_failure_total{accounting_type="ServiceAccounting",
  status_type="Stop",AAAServer="PassiveMZ-12997",endpoint_ip="198.51.100.2",timeout_ip1="1.1.1.1",
  timeout_ip2="203.0.113.1",circle_code="DL",retries="0"}
  ```

- **Start Retry on Success**

  Track retries on success for "Start" accounting requests to endpoint, supporting mismatch protection strategies.

  Formula:

  ```
  radius_proxy_accounting_retry_on_success_total{accounting_type="ServiceAccounting",
  status_type="Start",AAAServer="PassiveMZ-12997",server_ip="192.0.2.1",endpoint_ip="198.51.100.2",
  timeout_ip1="1.1.1.1",timeout_ip2="203.0.113.1",circle_code="DL",retries="0"}
  ```

- **Interim Update Retry on Success**

  Monitor retries on success for "Interim-Update" accounting requests to endpoint, aiding in effective mismatch prevention.

  Formula:

  ```
  radius_proxy_accounting_retry_on_success_total{accounting_type="ServiceAccounting",
  status_type="Interim-Update",AAAServer="PassiveMZ-12997",server_ip="192.0.2.1",endpoint_ip="198.51.100.2",
  timeout_ip1="203.0.113.1",timeout_ip2="203.0.113.1",circle_code="DL",retries="0"}
  ```

- **Stop Retry on Success**

  Track retries on success for "Stop" accounting requests to endpoint, contributing to accounting mismatch protection.

  Formula:

  ```
  radius_proxy_accounting_retry_on_success_total{accounting_type="ServiceAccounting",
  status_type="Stop",AAAServer="PassiveMZ-12997",server_ip="192.0.2.1",endpoint_ip="198.51.100.2",
  timeout_ip1="1.1.1.1",timeout_ip2="203.0.113.1",circle_code="DL",retries="0"}
  ```

Engine KPIs

These KPIs monitor cache data status and GRPC requests and responses, providing insights into data integrity and communication within the policy engine.

- **Cache Data Out of Date**

  Monitor instances where cache data is outdated to ensure timely updates and maintain data integrity within the policy engine.

  Formula: `radius_engine_cache_total{node_type="unknown",name="cache_data_outofdate"}`
- **Outbound GRPC Request**

  Track outbound GRPC requests for operations such as Bundle CoA Request, Async CoA Request, and Proxy Accounting Request.

  Formula:
  `radius_engine_total{node_type="unknown",message_type="proxy_service-accounting_stop0_request"}`
- **Outbound GRPC Response**

  Monitor outbound GRPC responses for operations, capturing response types such as timeout, success, and unprocessed.

  Formula:

  ```
  radius_engine_total{node_type="unknown",response_type="TIMEOUT",
  message_type="CoAResponse",coa_type="bundled_coa"}
  ```
- **Inbound GRPC Response**

  Track inbound GRPC responses for Access and Accounting Requests to assess acceptance and rejection rates.

  Formula: `radius_engine_total{node_type="unknown",message_type="accounting_response"}`

RADIUS Endpoint KPIs

These KPIs track CoA request outcomes, such as successful requests with or without retries, and timeouts, providing insights into network performance and potential issues.

- **Back Off Retry Success**

  Track successful CoA requests using back-off retry from NAS IP to endpoint.

  Formula:

  ```
  radius_coa_requests_total{message_type="CoARequest",nas_ip_address="192.0.2.1",
  endpoint_address="198.51.100.1",retry_type="BACK_OFF_RETRY",result="SUCCESS"}
  ```
- **CoA Request Success**

  Track successful CoA requests without retry from NAS IP to endpoint.

  Formula:

  ```
  radius_coa_requests_total{message_type="CoARequest",nas_ip_address="192.0.2.1",
  endpoint_address="198.51.100.1",retry_type="NA",result="SUCCESS"}
  ```
- **CoA Request Timeout**

  Track the CoA request timeouts from NAS IP to endpoint.

  Formula:

```
radius_coa_request_timeout_total{message_type="CoaRequest",nas_ip_address="192.0.2.1",
client_ip_address="192.0.2.1",endpoint_address="198.51.100.1"}
```

Mongo KPIs

This KPI monitors MongoDB performance by tracking operation times for read, write, and total activities on specified collections, such as "subscriber," and databases.



Note Ensure that `k8s single-node` is set to `false`.

- **Mongo Operation Time**

  Tracks time a MongoDB instance spends reading and writing data for each collection, measured in milliseconds (ms).

  Formula:

  ```
  mongo_operation_time{host="203.0.113.1",port="65001",replica_set="sdb-subscriber1",
  member_name="sdb-rs1-s1-m2",type="mongo",dbname="spr",collection="subscriber",op="read"}
  ```

Bulk Stats

Bulk stats refers to statistics or data collected and analyzed in large volumes, applicable in contexts such as network management, data analysis, and performance monitoring. These combined statistics are used in performance analysis, such as traffic reports, to monitor the overall health and performance of nodes. They help in taking appropriate actions, optimizing the packet core network for better use, and reducing overall expenses.

Explanation of RADIUS Request Query

This query explanation breaks down the components of a RADIUS request tracking metric:

- **radius_requests_total**: This is the metric name, likely tracking the total number of requests related to the RADIUS protocol, used for Authentication, Authorization, and Accounting (AAA) services in networking or security systems
- **sum(radius_requests_total) by (message_type)**: The `sum\(\)` function aggregates the `radius_requests_total` metric, grouping the sum by the `message_type` label. This provides the total number of requests for each `message_type` (e.g., Access-Request, Access-Accept) in your RADIUS data.
- **labels [message_type]**: This indicates that the result should include the `message_type` label in the output, providing context for each total.
- **alias radius_requests_total**: The alias assigns a custom name to the metric for easier reference in visualization or further querying, allowing you to refer to the result as `radius_requests_total` in the output.
- **default-value 0**: This sets a default value of 0 for any `message_type` with no values, ensuring the query returns 0 instead of an empty or missing value.

Bulk Stats configuration for RADIUS requests

To configure bulk stats and analyze RADIUS request metrics, follow these steps:

Procedure

Step 1 Login to Master Node and the CEE Ops-Center IP using the following command:

```
kubectl get pod -n cee -o wide |grep ops
```

Step 2 Log into the CEE Ops-Center.

Step 3 Verify Bulk Stats Pods using the following command:

```
kubectl get pod -n cee |grep bulk-stat
```

Step 4 Enter CEE Ops configuration mode and add this configuration:

```
bulk-stats query radius_requests_total
  expression    "sum(radius_requests_total) by (message_type)"
  labels        [ message_type ]
  alias         radius_requests_total
  default-value 0
exit
```

Step 5 Commit changes and verify if the system status is at 100%.

Step 6 Execute Commands on master node:

```
kubectl exec -it bulk-stats-0 -n <namespace> -- bash
```

Step 7 Navigate to the log file directory

```
cd /var/stats/bulk
```

```
cat delta-bulk-stats-1742968920000.csv
```

Bulk Stats Sample Query Configuration

This section shows a sample configuration for bulk stats queries in a cnAAA environment:

- **General Configuration**:

```
[unknown] cee# show running-config bulk-stats
bulk-stats enable true
bulk-stats user admin
bulk-stats external-port 2222
bulk-stats vnf-name cnaaa
bulk-stats vnf-alias cee-global
alias cee-global
exit
```

- **Query Configuration**:

```
bulk-stats query action_total
expression "sum(action_total) by (type,status)"
labels [ status type ]
alias action_total
default-value 0
exit
```

Accessing Bulk Statistics files

- The bulk statistics are generated and stored in the `/var/stats/bulk directory within the bulk-stats` pod.
- The recommended command for offloading files, such as `kubectl cp`:

```
kubectl cp -n <namespace> bulk-stats-0:/var/stats/bulk/<filename>.csv
      ./local-path/<filename>.csv
```

Subscriber migration from CPS 7.5 to cnAAA

Feature History

Feature Name	Release Information	Description
Subscriber migration from CPS 7.5 to cnAAA	2025.04.0	This feature migrates subscriber management and policy enforcement from CPS 7.5 to the cnAAA platform to ensure service continuity during the transition. The migration is performed in phased, circle-by-circle upgrades with a SOAP proxy managing traffic and enabling temporary shared SPR access.

This feature migrates subscriber management and policy enforcement from CPS 7.5 to the cnAAA platform. It ensures service continuity during migration, intelligent API routing between platforms, and reliable rollback mechanism. It addresses key migration challenges, including staggered BNG transitions, single IP address constraints, and session data inconsistencies between platforms. A SOAP proxy manages traffic and allows temporary shared SPR access. This phased, circle-by-circle upgrade minimizes service disruption.

Subscriber migration KPIs

SOAP proxy KPIs

The SOAP proxy generates these KPIs to provide operational insights and facilitate monitoring of the subscriber migration process. These metrics track various aspects of request handling, response times, errors, and resource utilization for both Cisco Policy Suite (CPS) 7.5 and cnAAA interactions. Each KPI is categorized by message type and source IP address.

- **Cps75_incoming_requests_total**

  Description: Tracks the total number of incoming requests received by the SOAP Proxy that are destined for CPS 7.5, categorized by message type and source IP address.

  Formula:

  ```
  Cps75_incoming_requests_total
  {message_type="GET_SUBSCRIBER",SourceIP="10.189.154.31",} 5.0
  ```

- **cnaaa_incoming_requests_total**

Description: Tracks the total number of incoming requests received by the SOAP Proxy that are destined for cnAAA, categorized by message type and source IP address.

Formula:

```
cnaaa_incoming_requests_total
{message_type="GET_SUBSCRIBER",SourceIP="10.189.154.31",} 5.0
```

- **outgoing_responses_success_total**

  Description: Tracks the total number of successful outgoing responses generated by the SOAP Proxy, categorized by message type and source IP address.

  Formula:

  ```
  outgoing_responses_success_total
  {message_type="GET_SUBSCRIBER_FOR_DELETE",SourceIP="192.168.64.64",} 1.0
  ```

- **outgoing_response_success_duration_seconds**

  Description: Tracks the duration for successful outgoing responses from the SOAP Proxy, categorized by message type and source IP address.

  Formula:

  ```
  outgoing_response_success__duration_seconds{message_
  type="GET_SUBSCRIBER_FOR_DELETE",SourceIP="192.168.64.64",} 1.756229297503E9
  ```

- **outgoing_response_error_total**

  Description: Tracks the total number of failed outgoing responses generated by the SOAP Proxy, categorized by message type and source IP address.

  Formula:

  ```
  outgoing_response_error_total
  {message_type="DELETE_SUBSCRIBER",SourceIP="192.168.64.64",} 1.0
  ```

- **outgoing_response_error_duration_seconds**

  Description: Tracks the duration for failed outgoing responses from the SOAP Proxy, categorized by message type and source IP address.

  Formula:

  ```
  outgoing_response_error_duration_seconds
  {message_type="DELETE_SUBSCRIBER",SourceIP="192.168.64.64",} 1.756229305042E9
  ```

- **Cps75_fwd_requests_total**

  Description: Tracks the total number of requests forwarded by the SOAP Proxy to CPS 7.5, categorized by message type and source IP address.

  Formula:

  ```
  Cps75_fwd_requests_total{message_type="GET_SUBSCRIBER_FOR_DELETE",SourceIP="192.168.64.64",}
   2.0
  ```

- **Cps75_fwd_response_success_total**

  Description: Tracks the total number of successful responses received by the SOAP Proxy from CPS 7.5, categorized by message type and source IP address.

  Formula:

```
Cps75_fwd_response_success_total{message_type="GET_SUBSCRIBER_FOR_DELETE",SourceIP="192.168.64.64",}
 2.0
```

- **Cps75_fwd_response_error_total**

  Description: Tracks the total number of error responses received by the SOAP Proxy from CPS 7.5, categorized by message type and source IP address.

  Formula:

```
Cps75_fwd_response_error_total{message_type="DELETE_SUBSCRIBER",SourceIP="192.168.64.64",}
 1.0
```

- **Cps75_fwd_response_error_duration_seconds**

  Description: Tracks the total number of error responses received by the SOAP Proxy from CPS 7.5, categorized by message type and source IP address.

  Formula:

```
Cps75_fwd_response_error_total{message_type="DELETE_SUBSCRIBER",SourceIP="192.168.64.64",}
 1.0
```

- **cps_response_success_duration_seconds**

  Description: Tracks the duration for successful responses received from CPS 7.5, categorized by message type and source IP address.

  Formula:

```
cps_response_success_duration_seconds{message_type="GET_SUBSCRIBER_FOR_DELETE",SourceIP="192.168.64.64",}
 1.756229297503E9
```

- **cps_response_error_duration_seconds**

  Description: Tracks the duration for error responses received from CPS 7.5, categorized by message type and source IP address.

  Formula:

```
cps_response_error_duration_seconds_total{message_type="DELETE_SUBSCRIBER",SourceIP="192.168.64.64",}
 1.197E-6
```

- **Cps75_fwd_connection_error_total**

  Description: Tracks the total number of connection errors encountered when forwarding requests to CPS 7.5, categorized by message type and source IP address.

  Formula:

```
Cps75_fwd_connection_error_total{message_type="GET_SUBSCRIBER",SourceIP="192.168.64.64",}
 1.0
```

- **Cps75_proxy_requests_error_total**

  Description: Tracks the total number of errors during the processing of requests by the SOAP proxy related to CPS 7.5 operations, categorized by message type and source IP address.

  Formula:

```
Cps75_proxy_requests_error_total{message_type="GET_SUBSCRIBER_FOR_DELETE",SourceIP="192.168.64.64",}
 1.0
```

- **Cps75_request_duration_seconds**

Description: Tracks the duration of proxy request processing for CPS 7.5 operations, categorized by message type and source IP address.

Formula:

```
Cps75_request_duration_seconds{message_type="GET_SUBSCRIBER",SourceIP="192.168.64.64",}
 1.197E-6
```

- **cnaaa_refresh_profile_success_total**

  Description: Tracks the total number of successful Refresh Profile requests sent to cnAAA, categorized by message type and source IP address.

  Formula:

```
cnaaa_refresh_profile_success_total{message_type="refresh_profile",SourceIP="192.168.64.64"}
 1.0
```

- **cnaaa_refresh_profile_error_total**

  Description: Tracks the total number of failed `Refresh Profile` requests sent to cnAAA, categorized by message type and source IP address.

  Formula:

```
cnaaa_refresh_profile_error_total{message_type="refresh_profile",SourceIP="192.168.64.64"}
 1.0
```

- **cnaaa_refresh_profile_duration_seconds**

  Description: Tracks the duration for Refresh Profile operations with cnAAA, categorized by message type and source IP address.

  Formula:

```
cnaaa_refresh_profile_duration_seconds{message_type="refresh_profile",SourceIP="192.168.64.64",}

 1.756229266433E9
```

- **cnaaa_delete_session_success_total**

  Description: Tracks the total number of successful Delete Session requests sent to cnAAA, categorized by message type and source IP address.

  Formula:

```
cnaaa_delete_session_success_total{message_type="delete_session",SourceIP="192.168.64.64",}
 1.0
```

- **cnaaa_delete_session_error_total**

  Description: Tracks the total number of failed Delete Session requests sent to cnAAA, categorized by message type and source IP address.

  Formula:

```
cnaaa_delete_session_error_total{message_type="delete_session",SourceIP="192.168.64.64",}
 1.0
```

- **cnaaa_delete_session_duration_seconds**

  Description: Tracks the duration for Delete Session operations with cnAAA, categorized by message type and source IP address.

  Formula:

```
cnaaa_delete_session_duration_seconds{message_type="delete_session",SourceIP="192.168.64.64",}
 1.756229298911E9
```

- **cnaaa_processing_error_total**

  Description: Tracks the total number of processing errors encountered within cnAAA, categorized by message type and source IP address.

  Formula:

```
cnaaa_processing_error_total{message_type="UPDATE_SUBSCRIBER",SourceIP="192.168.64.64",}
 1.0
```

- **request_duration_seconds**

  Description: Provides general duration tracking for all requests processed by the SOAP Proxy, categorized by message type and source IP address.

  Formula:

```
request_duration_seconds{message_type="UPDATE_SUBSCRIBER",SourceIP="192.168.64.64",}
1.197E-6
```

- **cnaaa_retry_attempt_total**

  Description: Tracks the total number of retry attempts made for cnAAA operations, including the attempt number, categorized by message type and source IP address.

  Formula:

```
cnaaa_retry_attempt_total{message_type="refresh_profile",SourceIP="192.168.64.64",attempt="3",}
 1.0
```

- **cnaaa_retry_success_total**

  Description: Tracks the total number of successful retries for cnAAA operations, including the attempt number, categorized by message type and source IP address.

  Formula:

```
cnaaa_retry_success_total{message_type="delete_session",SourceIP="192.168.64.64",attempt="1",}
 1.0
```

- **cnaaa_retry_exhausted_total**
- Description: Tracks the total number of cnAAA operations that ultimately failed after all configured retry attempts were exhausted, including the last attempt number, categorized by message type and source IP address.

  Formula:

```
cnaaa_retry_exhausted_total{message_type="refresh_profile",SourceIP="192.168.64.64",attempt="3",}
 1.0
```

- **threadpool_queue_size**

  Description: Indicates the current number of tasks waiting in the thread pool queue for processing, categorized by pool name.

  Formula:

```
threadpool_queue_size{pool_name="cnaaa",} 0.0
```

- **threadpool_active_threads**

Description: Indicates the number of threads currently executing tasks within the thread pool, categorized by pool name.

Formula:

```
threadpool_active_threads{pool_name="cnaaa",} 0.0
```

- **threadpool_idle_threads**

  Description: Indicates the number of threads available but not currently executing tasks within the thread pool, categorized by pool name.

  Formula:

  ```
  threadpool_idle_threads{pool_name="cnaaa",} 2.0
  ```

- **threadpool_core_pool_size**

  Description: Indicates the minimum number of threads maintained in the thread pool, categorized by pool name.

  Formula:

  ```
  threadpool_core_pool_size{pool_name="cnaaa",} 30.0
  ```

- **threadpool_maximum_pool_size**

  Description: Indicates the maximum number of threads allowed in the thread pool, categorized by pool name.

  Formula:

  ```
  threadpool_maximum_pool_size{pool_name="cnaaa",} 30.0
  ```

- **threadpool_current_pool_size**

  Description: Indicates the current total number of threads in the thread pool, categorized by pool name.

  Formula:

  ```
  threadpool_current_pool_size{pool_name="cnaaa",} 2.0
  ```

- **threadpool_remaining_thread_capacity**

  Description: Indicates the remaining capacity for new tasks in the thread pool before reaching the maximum pool size, categorized by pool name.

  Formula:

  ```
  threadpool_remaining_thread_capacity{pool_name="cnaaa",} 28.0
  ```

Subscriber migration from CPS 7.5 to cnAAA SOAP Kafka Relay

Feature History

Feature Name	Release Information	Description
Subscriber migration from CPS 7.5 to cnAAA (SOAP Kafka Relay)	2026.01.0	The SOAP Kafka Relay solution for subscriber migration from CPS 7.5 to cnAAA, is an advanced solution which addresses the limitations of subscriber migration solution that uses the common CPS 7.5 SPR until the migration is completed. It addresses • performance and stability issues from shared database usage in CPS 7.5 and cnAAA, • introduces an asynchronous, Kafka-based replication mechanism to decouple database operations, and • ensures data consistency, system stability, and reduced operational risk during subscriber migration.

The SOAP Kafka Relay migrates subscriber data by functioning as a dual-write proxy. It synchronously services the primary legacy system (CPS 7.5) and asynchronously replicates changes to the new system (cnAAA). This decoupling prevents performance issues associated with shared database access.

Subscriber migration SOAP Kafka Relay KPIs

SOAP proxy KPIs

The SOAP Kafka Relay generates these KPIs to provide operational insights and facilitate monitoring of SOAP-to-Kafka data flows. These metrics track various aspects of request handling, response times, error conditions, Kafka interactions, upstream/downstream communications, and resource utilization. Each KPI is categorized by relevant labels such as message type, source IP address, topic, endpoint, or attempt number.

Producer KPIs (unified-api-proxy-ep-producer)

The following KPIs are generated by the Producer component of the Unified API Proxy.

- **soap_incoming_requests_total**

  Description: Total number of incoming SOAP requests received by the producer.

  Formula:

```
soap_incoming_requests_total{message_type="CreateSubscriberRequest",source_ip="10.84.117.121",}
 10001.0
```

- **soap_response_duration_seconds**

  Description: Total duration of all SOAP responses in seconds.

  Formula:

```
soap_response_duration_seconds_total{message_type="CreateSubscriberRequest",source_ip="192.168.205.0",}
 670.3258733479987
```

- **soap_response_success_total**

  Description: Total number of successful SOAP responses returned to clients.

  Formula:

```
soap_response_success_total{message_type="CreateSubscriberRequest",source_ip="10.84.117.121",}
 10001.0
```

- **soap_response_error_total**

  Description: Total number of failed SOAP responses returned to clients.

  Formula:

```
soap_response_error_total{message_type="CreateSubscriberRequest",source_ip="192.168.205.0",}
 15.0
```

- **soap_response_success_duration_seconds**

  Description: Total duration of successful responses in seconds.

  Formula:

```
soap_response_success_duration_seconds_total{message_type="CreateSubscriberRequest",source_ip="10.1.33.115",}
 30.07539027100001
soap_response_success_duration_seconds_total{message_type="UpdateSubscriberRequest",source_ip="192.168.205.0",}
 0.074180371
soap_response_success_duration_seconds_total{message_type="CreateSubscriberRequest",source_ip="192.168.205.0",}
 33.00786633000007
```

- **soap_response_error_duration_seconds**

  Description: Total duration of error responses in seconds.

  Formula:

```
soap_response_error_duration_seconds_total{message_type="CreateSubscriberRequest",source_ip="192.168.205.0",}
 45.8374291
```

- **kafka_publish_requests_total**

  Description: Total number of Kafka publish requests attempted.

  Formula:

```
kafka_publish_requests_total{message_type="CreateSubscriberRequest",source_ip="10.84.117.121",}
 5000.0
```

- **kafka_publish_success_total**

  Description: Total number of successful Kafka message publishes.

  Formula:

```
kafka_publish_success_total{message_type="CreateSubscriberRequest",source_ip="10.84.117.121",}
 5000.0
```

- **kafka_publish_error_total**

Description: Total number of failed Kafka message publishes.

Formula:

```
kafka_publish_error_total{message_type="CreateSubscriberRequest",source_ip="192.168.52.128",}
 184.0
```

- **kafka_publish_success_duration_seconds**

  Description: Total duration of successful Kafka publishes in seconds.

  Formula:

```
kafka_publish_success_duration_seconds_total{message_type="CreateSubscriberRequest",source_ip="192.168.205.0",}
 22.81055704900002
```

- **kafka_publish_error_duration_seconds**

  Description: Total duration of failed Kafka publishes in seconds.

  Formula:

```
kafka_publish_error_duration_seconds_total{message_type="CreateSubscriberRequest",source_ip="192.168.52.128",}
 368.2182095640003
```

- **upstream_forward_requests_total**

  Description: Total number of requests forwarded to upstream service.

  Formula:

```
upstream_forward_requests_total{message_type="CreateSubscriberRequest",source_ip="10.84.117.121",}
 10001.0
```

- **upstream_response_success_total**

  Description: Total number of successful upstream service responses.

  Formula:

```
upstream_response_success_total{message_type="CreateSubscriberRequest",source_ip="10.84.117.121",}
 5000.0
```

- **upstream_response_error_total**

  Description: Total number of failed upstream service responses.

  Formula:

```
upstream_response_error_total{message_type="CreateSubscriberRequest",source_ip="10.84.117.121",}
 5001.0
```

- **upstream_response_success_duration_seconds**

  Description: Total duration of successful upstream calls in seconds.

  Formula:

```
upstream_response_success_duration_seconds_total{message_type="CreateSubscriberRequest",source_ip="192.168.205.0",}
 8.973665204000001
```

- **upstream_response_error_duration_seconds**

  Description: Total duration of failed upstream calls in seconds.

  Formula:

```
upstream_response_error_duration_seconds_total{message_type="CreateSubscriberRequest",source_ip="192.168.205.0",}
 0.011748717
```

- **upstream_connection_error_total**

  Description: Total number of upstream connection errors.

  Formula:

  ```
  upstream_connection_error_total{message_type="CreateSubscriberRequest",source_ip="192.168.205.0",}
   8.0
  ```

- **upstream_connection_timeout_total**

  Description: Total number of upstream connection timeout errors.

  Formula:

  ```
  upstream_connection_timeout_total{message_type="CreateSubscriberRequest",source_ip="192.168.205.0",}
   3.0
  ```

- **upstream_retry_attempt_total**

  Description: Total number of retry attempts for upstream calls.

  Formula:

  ```
  upstream_retry_attempt_total{message_type="CreateSubscriberRequest",source_ip="10.84.117.121",attempt="1",}
   5001.0
  ```

- **upstream_retry_success_total**

  Description: Total number of successful retry attempts.

  Formula:

  ```
  upstream_retry_success_total{message_type="CreateSubscriberRequest",source_ip="192.168.205.0",attempt="1",}
   12.0
  ```

- **upstream_retry_exhausted_total**

  Description: Total number of requests where all retry attempts were exhausted.

  Formula:

  ```
  upstream_retry_exhausted_total{message_type="CreateSubscriberRequest",source_ip="192.168.205.0",}
   5.0
  ```

- **upstream_failover_events_total**

  Description: Total number of failover/failback events (primary/secondary switching).

  Formula:

  ```
  upstream_failover_events_total{event_type="failover",} 7.0
  ```

- **upstream_active_endpoint**

  Description: Currently active endpoint (1=PRIMARY, 0=SECONDARY).

  Formula:

  ```
  upstream_active_endpoint 0.0
  ```

- **upstream_primary_endpoint_healthy**

  Description: Health status of primary endpoint (1=HEALTHY, 0=UNHEALTHY).

  Formula:

  ```
  upstream_primary_endpoint_healthy 0.0
  ```

- **soap_processing_error_total**

  Description: Total number of request processing errors.

  Formula:

  ```
  soap_processing_error_total{message_type="CreateSubscriberRequest",source_ip="192.168.205.0",}
   4.0
  ```

- **soap_processing_error_duration_seconds**

  Description: Total duration of processing errors in seconds.

  Formula:

  ```
  soap_processing_error_duration_seconds_total{message_type="CreateSubscriberRequest",source_ip="192.168.205.0",}
   23.4561892
  ```

- **rate_limit_rejection_total**

  Description: Total number of requests rejected due to rate limiting.

  Formula:

  ```
  rate_limit_rejection_total{message_type="CreateSubscriberRequest",source_ip="192.168.205.0",}
   47.0
  ```

Consumer Component (unified-api-proxy-ep-consumer)

- **consumer_messages_consumed_total**

  Description: Total number of messages consumed from Kafka topic.

  Formula:

  ```
  consumer_messages_consumed_total{message_type="CreateSubscriberRequest",topic="SoapAPI-details",}
   10000.0
  ```

- **consumer_processing_success_total**

  Description: Total number of messages processed successfully.

  Formula:

  ```
  consumer_processing_success_total{message_type="CreateSubscriberRequest",} 4931.0
  ```

- **consumer_processing_error_total**

  Description: Total number of message processing errors.

  Formula:

  ```
  consumer_processing_error_total{message_type="CreateSubscriberRequest",error_type="timeout",}
   6.0
  ```

- **consumer_processing_duration_seconds_total**

  Description: Total time spent processing messages in seconds.

  Formula:

  ```
  consumer_processing_duration_seconds_total{message_type="CreateSubscriberRequest",status="success",}
   98.93288240799995
  ```

- **consumer_rate_limit_rejections_total**

  Description: Total number of rate limit rejections in consumer.

Formula:

```
consumer_rate_limit_rejections_total{message_type="CreateSubscriberRequest",} 25843.0
```

- **consumer_downstream_calls_total**

  Description: Total number of downstream cnAAA calls attempted.

  Formula:

```
consumer_downstream_calls_total{message_type="CreateSubscriberRequest",endpoint="PRIMARY",}
 30000.0
```

- **consumer_downstream_response_success_total**

  Description: Total number of successful downstream cnAAA responses.

  Formula:

```
consumer_downstream_response_success_total{message_type="CreateSubscriberRequest",endpoint="PRIMARY",}
 4737.0
```

- **consumer_downstream_response_error_total**

  Description: Total number of downstream response errors.

  Formula:

```
consumer_downstream_response_error_total{message_type="CreateSubscriberRequest",endpoint="PRIMARY",error_type="application_error",}
 29975.0
```

- **consumer_downstream_duration_seconds_total**

  Description: Total time spent on downstream calls in seconds.

  Formula:

```
consumer_downstream_duration_seconds_total{message_type="CreateSubscriberRequest",endpoint="PRIMARY",status="error",}
 282.66106414900014
```

- **consumer_downstream_retry_attempt_total**

  Description: Total number of downstream retry attempts.

  Formula:

```
consumer_downstream_retry_attempt_total{message_type="CreateSubscriberRequest",endpoint="PRIMARY",attempt="1",}
 10000.0
```

- **consumer_downstream_retry_success_total**

  Description: Total number of successful retries.

  Formula:

```
consumer_downstream_retry_success_total{message_type="CreateSubscriberRequest",endpoint="PRIMARY",attempt="2",}
 5.0
```

- **consumer_downstream_retry_exhausted_total**

  Description: Total number of times all downstream retries were exhausted.

  Formula:

```
consumer_downstream_retry_exhausted_total{message_type="CreateSubscriberRequest",endpoint="PRIMARY",}
 10000.0
```

- **consumer_downstream_failover_events_total**

Description: Total number of downstream failover/failback events.

Formula:

```
consumer_downstream_failover_events_total{event_type="failback",endpoint="PRIMARY",}
1.0
```

- **consumer_downstream_failover_events_created**

  Description: Timestamp of the last downstream failover event creation.

  Formula:

  ```
  consumer_downstream_failover_events_created{event_type="failback",endpoint="PRIMARY",}
   1.768285725213E9
  ```

- **consumer_downstream_active_endpoint**

  Description: Currently active downstream endpoint (1=PRIMARY, 0=SECONDARY).

  Formula:

  ```
  consumer_downstream_active_endpoint 1.0
  ```

- **consumer_downstream_primary_endpoint_healthy**

  Description: Primary downstream endpoint health (1=HEALTHY, 0=UNHEALTHY).

  Formula:

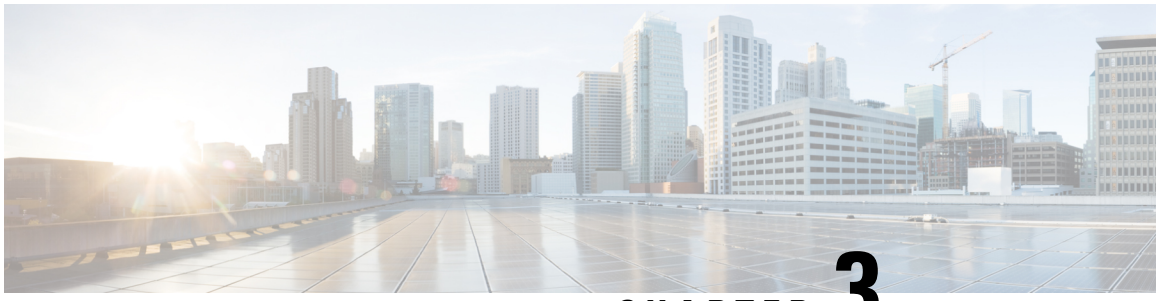
  ```
  consumer_downstream_primary_endpoint_healthy 1.0
  ```

- **publish_to_dlq_total**

  Description: This counter tracks the total number of messages published to the Dead Letter Queue (DLQ).

  Formula:

  ```
  publish_to_dlq_total{message_type="CreateSubscriber",} 10000.0
  ```



CHAPTER 3

MIB Reference

•
• CISCO-CNEE-MIB, on page 85
• CISCO-SMI, on page 85

CISCO-CNEE-MIB

This is the MIB module for the Cisco Cloud Native Execution Environment (CNEE) platform. This MIB only handles notifications from the CNEE.

Note The Cisco Cloud Native Execution Environment (CNEE) MIB (CISCO-CNEE-MIB.my) uses definitions that are defined in the Cisco Enterprise Structure of Management Information (SMI) MIB (CISCO-SMI.my).

For more information, see the "*UCC Subscriber Microservice Infrastructure - Operations Guide*" > "SMI MIB Reference" chapter.

CISCO-SMI

This is the Structure of Management Information for the Cisco Enterprise.

Note The Cisco Cloud Native Execution Environment (CNEE) MIB (CISCO-CNEE-MIB.my) uses definitions that are defined in the Cisco Enterprise Structure of Management Information (SMI) MIB (CISCO-SMI.my).

For more information, see the "*UCC Subscriber Microservice Infrastructure - Operations Guide*" > "SMI MIB Reference" chapter.