

# Troubleshooting Information


- Feature Summary and Revision History, on page 1
- Description, on page 3
- Using CLI Data, on page 3
- Alerts, on page 37
- Metrics, on page 60
- Logs, on page 64
- Dispatcher, on page 75


# Feature Summary and Revision History

## Summary Data

***Table 1: Summary Data***

| | |
|---|---|
| Applicable Product(s) or FunctionalArea | SMF |
| Applicable Platform(s) | SMI |
| Feature Default Setting | Not Applicable |
| Related Changes in this Release | Not Applicable |
| Related Documentation | Not Applicable |

## Revision History

***Table 2: Revision History***

| Revision Details | Release |
|---|---|
| Added the wait time display for ongoing bulk clear subscriber CLI and blocking the consecutive CLI. | 2023.04.0 |

| Revision Details | Release |
|---|---|
| Added the following support:<br>• Enabling UPF Monitor Subscriber from SMF.<br>• Session Count per slice and NSSAI. | 2023.03.0 |
| As part of the IP pool allocation per slice and DNN feature, added example configuration to configure NSSAI labels of smf_service_stats metrics. | 2022.04.0 |
| Introduced support for classification and configuration of application metrics | 2021.02.3 |
| Added support for the following enhancements:<br>• The **show subscriber nf-service smf** *smf_url* command to show subscriber details based on the IP address value of the vSMF or hSMF.<br>• The **clear subscriber nf-service smf** *smf_url* command to clear subscriber details based on the IP address value of the vSMF or hSMF.<br>• The **clear subscriber nf-service smf** *smf_url* command to clear subscriber details based on the IP address value of the vSMF or hSMF.<br>• The **show subscriber supi** *supi_id* **nf-service smf psid** *psid_value* **full** command to show detailed subscriber information for roaming-specific use case as hSMF and vSMF.<br>• The **show subscriber supi** *supi_id* **nf-service smf psid** *psid_value* **summary** command to show detailed information about subscriber sessions for roaming-specific use case as hSMF and vSMF. | 2021.02.2 |
| Added support for the following enhancements:<br>• The **show subscriber supi** *supi_value* **nf-service smf psid** *psid_value* **summary** command to provide detailed information about subscriber sessions.<br>• The **clear subscriber nf-service smf** and **show subscriber nf-service smf** commands with supported keywords and filters.<br>• The **clear subscriber** and **clear subscriber nf-service smf** commands to support the **reactivation** keyword to clear sessions when release cause as reactivation-required is configured. This enhancement also supports disconnect and release reasons.<br>• The **imei** keyword for **monitor subscriber**, **clear subscriber**, and **show subscriber** CLI commands. | 2021.02.0 |
| First introduced. | Pre-2020.02.0 |

# Description

This chapter provides information on using the command line interface (CLI) commands, alerts, metrics, monitor tools, and logs for troubleshooting any issues that may arise during system operation.

# Using CLI Data

This section describes the show and clear commands and the monitor commands that are used for troubleshooting.

## Show and Clear Commands

### show Commands

This section lists some of the key show commands that are available for troubleshooting the issues. The output of these show commands provides specific configuration and status information.

#### show config-error

Use this command to display the configuration error-related information for all pods in the cluster. The following sample output is for the **show config-error** command:

```
[smf] smf# show config-error
ERROR
COMPONENT       ERROR DESCRIPTION
------------------------------------------------------------------------------
RuleBase        Default bandwidth policy does not exist in rulebase <rba1> for charging
action <ca1> .Dropping ruleDef <rda1>
RuleBase        Default bandwidth policy does not exist in rulebase <rba6> for charging
action <ca1>.Dropping ruleDef <rda60>
RuleBase        Default bandwidth policy does not exist in rulebase <rba6> for charging
action <ca1>.Dropping ruleDef <rda61>
ChargingAction  Packet filter <pkt1234> configured for charging action <ca4> associated
with rulebase <rb1> does not exist
BandWidthPolicy Uplink peak data rate less than commited data rate in charging action
<ca6>Dropping ruleDef <rd6>
```

*Table 3: Output Field Descriptions for the **show config-error** Command*

| Field | Description |
|---|---|
| Error Component | Specifies the error component. |
| Error Description | Specifies the description of the Error. |

## show diagnostics

Use this command to display the diagnostics information. The following sample output is for the **show diagnostics** command:

```
[smf] smf# show diagnostics
POD INSTANCE       DIAGNOSTIC           COMPONENT  START TIME               STATUS   RETRIES
-------------------------------------------------------------------------------------------
bgpspeaker-pod-1   Topology             AppInfra   2022/03/08 20:36:24.674  Success  0
bgpspeaker-pod-1   System Topology      AppInfra   2022/03/08 20:36:24.676  Success  0
sgw-service-0      Topology             AppInfra   2022/03/08 20:36:17.152  Success  0
sgw-service-0      System Topology      AppInfra   2022/03/08 20:36:17.154  Success  0
sgw-service-0      Cache Pod            AppInfra   2022/03/08 20:36:27.223  Success  0
sgw-service-0      SESSION_DB Datastore AppInfra   2022/03/08 20:36:17.155  Success  0
li-ep-0            Topology             AppInfra   2022/03/08 20:36:20.743  Success  0
li-ep-0            System Topology      AppInfra   2022/03/08 20:36:20.741  Success  0
smf-service-1      Topology             AppInfra   2022/03/08 20:36:19.216  Success  0
smf-service-1      System Topology      AppInfra   2022/03/08 20:36:19.218  Success  0
smf-service-1      Cache Pod            AppInfra   2022/03/08 20:36:26.276  Success  0
smf-service-1      SESSION_DB Datastore AppInfra   2022/03/08 20:36:19.220  Success  0
dns-proxy-0        Topology             AppInfra   2022/03/08 20:36:21.885  Success  0
dns-proxy-0        System Topology      AppInfra   2022/03/08 20:36:21.887  Success  0
protocol2-1        System Topology      AppInfra   2022/03/08 20:36:24.858  Success  0
protocol2-1        Cache Pod            AppInfra   2022/03/08 20:36:25.937  Success  0
protocol2-1        Topology             AppInfra   2022/03/08 20:36:24.856  Success  0
nodemgr-0          System Topology      AppInfra   2022/03/08 20:36:14.831  Success  0
nodemgr-0          Cache Pod            AppInfra   2022/03/08 20:36:26.485  Success  0
nodemgr-0          SESSION_DB Datastore AppInfra   2022/03/08 20:36:14.833  Success  0
nodemgr-0          Topology             AppInfra   2022/03/08 20:36:14.835  Success  0
nodemgr-1          Topology             AppInfra   2022/03/08 20:36:23.068  Success  0
nodemgr-1          System Topology      AppInfra   2022/03/08 20:36:23.071  Success  0
nodemgr-1          Cache Pod            AppInfra   2022/03/08 20:36:26.690  Success  0
nodemgr-1          SESSION_DB Datastore AppInfra   2022/03/08 20:36:23.066  Success  0
```

***Table 4: Output Field Descriptions for the show diagnostics Command***

| Field | Description |
|---|---|
| Component | Specifies the component name. |
| Diagnostics | Specifies the diagnostics details. |
| Pod Instance | Specifies the instance information of the pod. |
| Retries | Specifies the retry count. |
| Start Time | Specifies the start time of the application. |
| Status | Specifies if the diagnostics status is successful or not. |

## show endpoint all

Use this command to display the list of all internal and external endpoints running on all pods in the cluster. The following sample output is for the **show endpoint all** command:

```
[smf] smf# show endpoint all

                        START    STOPPED
GR INSTANCE               ENDPOINT            ADDRESS  TYPE     STATUS
   INTERFACE  INTERNAL  TIME     TIME
```

```
cache-pod                  xx.xx.xx.xx:0000     Grpc     Started  cache-pod
   true       4 weeks   <none>   0
cache-pod                  xx.xx.xx.xx:0000     Grpc     Started  cache-pod
   true       4 weeks   <none>   0
internal-admin-ep          xx.xx.xx.xx:0000     Rest     Started  internal-admin-ep
   true       4 weeks   4 weeks  0
internal-admin-ep          xx.xx.xx.xx:0000     Rest     Started  internal-admin-ep
   true       4 weeks   <none>   0
internal-admin-ep          xx.xx.xx.xx:0000     Rest     Started  internal-admin-ep
   true       4 weeks   <none>   0
:
:
keep-alived-ep             xx.xx.xx.xx:0000     Tcp      Started  keep-alived-ep
   true       2 weeks   <none>   0
keep-alived-ep             xx.xx.xx.xx:0000     Tcp      Started  keep-alived-ep
   true       2 weeks   <none>   0
oam-grpc-ep                xx.xx.xx.xx:0000     Grpc     Started  oam-grpc-ep
   true       4 weeks   <none>   0
oam-rest-ep                xx.xx.xx.xx:0000     Rest     Started  oam-rest-ep
   true       4 weeks   <none>   0
```

***Table 5: Output Field Descriptions for the show endpoint all Command***

| Field | Description |
|---|---|
| Address | Specifies the host and port of the endpoint. |
| Endpoint | Specifies the name of the endpoint. |
| GR Instance | Specifies the GR instance. |
| Interface | Specifies the interface name of the endpoint. |
| Internal | Specifies the type of the endpoint (Internal or External). |
| Start Time | Specifies the start time of the endpoint. |
| Status | Specifies current status of the endpoint. |
| Stopped Time | Specifies the end time of the endpoint. |
| Type | Specifies the type of the endpoint. |

## show endpoint info

Use this command to display the list of endpoints running on all pods in the cluster. The following sample output is for the **show endpoint info** command:

```
[smf] smf# show endpoint info
                                                                                                    START
 STOPPED  GR
ENDPOINT                  ADDRESS              TYPE  STATUS   INTERFACE  INTERNAL  TIME
 TIME     INSTANCE
-------------------------------------------------------------------------------------------------------
sbi                     xxx.xxx.xxx.xxx:0000  Rest  Started  rest       false     2 weeks
 <none>   0
sbi                     xxx.xxx.xxx.xxx:0000  Rest  Started  rest       false     2 weeks
 <none>   0
```

*Table 6: Output Field Descriptions for the **show endpoint all** Command*

| Field | Description |
|---|---|
| Address | Specifies the host and port of the endpoint. |
| Endpoint | Specifies the name of the endpoint. |
| GR Instance | Specifies the GR instance. |
| Interface | Specifies the interface name of the endpoint. |
| Internal | Specifies the type of the endpoint (Internal or External). |
| Start Time | Specifies the start time of the endpoint. |
| Status | Specifies current status of the endpoint. |
| Stopped Time | Specifies the end time of the endpoint. |
| Type | Specifies the type of the endpoint. |

### show geo-maintenance-mode

Use this command to display whether the maintenance mode is enabled or disabled. The following sample output is for the **show geo-maintenance-mode** command:

```
[smf] smf# show geo-maintenance-mode
result "geo-maintenance-mode is disabled"
```

```
[smf] smf# show geo-maintenance-mode
result "geo-maintenance-mode is enabled"
```

### show georeplication checksum instance-id

Use this command to display replication details for etcd and cache-pod data. The following sample output is for the **show georeplication checksum instance-id** command:

```
[smf] smf# show georeplication checksum instance-id
Value for 'instance-id' (<string>): 1
checksum-details
--      ----  --------
ID      Type  Checksum
--      ----  --------
1       ETCD  1646812528
IPAM    CACHE 1646812528
NRFMgmt CACHE 1646812528
```

### show georeplication-status

Use this command to display the replication status between two racks in a Geo setup.

The following sample output displays, if the connection is successful:

```
[smf] smf# show georeplication-status
result "pass"
```

The following sample output displays, if there is an error:

```
[smf] smf# show georeplication-status
result "fail: [424] checksum missmatch"
```

### show helm

The **show helm** command displays the version information for the SMF system image.

### show ipam pool

| Field | Description |
|---|---|
| PoolName | Name of the Address Pool. |
| Ipv4Utilization | Utilization percentage for IPv4 address for this pool. |
| Ipv6AddrUtilization | Utilization percentage for IPv6 address for this pool. |
| Ipv6PrefixUtilization | Utilization percentage for IPv6 prefix address for this pool. |

### show ipam pool <pool-name>

| Field | Description |
|---|---|
| Ipv4Addr [Total/Used/Utilization] | Total IPv4 address available(configured for this pool) / Number of used address / Utilization percentage for IPv4 address. |
| Ipv6Addr [Total/Used/Utilization] | Total IPv6 address available(configured for this pool) / Number of used address / Utilization percentage for IPv6 address. |
| Ipv6Prefix [Total/Used/Utilization] | Total IPv6 prefix address available(configured for this pool) / Number of used address / Utilization percentage for IPv6 prefix |

### show ipam pool <pool-name> ipv4-addr

| Field | Description |
|---|---|
| StartAddress | Start address of the range. |
| EndAddress | End address of the range. |
| AllocContext | Name of data plane to which this address range is allocated. |
| Flag | Flag Indicate weather pool is Static or if it is offline. |

### show ipam pool <pool-name> ipv6-addr

| Field | Description |
|---|---|
| StartAddress | Start address of the range. |

| Field | Description |
|---|---|
| EndAddress | End address of the range. |
| AllocContext | Name of data plane to which this address range is allocated. |
| Flag | Flag Indicate weather pool is Static or if it is offline. |

**show ipam pool <pool-name> ipv6-prefix**

| Field | Description |
|---|---|
| StartAddress | Start address of the range. |
| EndAddress | End address of the range. |
| AllocContext | Name of data plane this address range is allocated. |
| Flag | Flag Indicates whether pool is Static or if it is offline, S(Static) and O(Offline). |

**show ipam dp**

| Field | Description |
|---|---|
| DpName | Name of the data plane which is registered. |
| Ipv4Utilization | Utilization percentage for IPv4 by this data plane. |
| Ipv6AddrUtilization | Utilization percentage for Ipv6 address by this data plane. |
| Ipv6PrefixUtilization | Utilization percentage for Ipv6 prefix by this data plane. |

**show ipam dp <dataplane-name>**

| Field | Description |
|---|---|
| Ipv4Addr [Total/Used/Utilization] | Total IPv4 address available(configured for this data plane) / Number of used address / Utilization percentage for IPv4. |
| Ipv6Addr [Total/Used/Utilization] | Total IPv6 address available(configured for this data plane) / Number of used address / Utilization percentage for IPv6. |
| Ipv6Prefix [Total/Used/Utilization] | Total IPv6 prefix address available(configured for this data plane) / Number of used address / Utilization percentage for IPv6 prefix. |

### show ipam dp <dataplane-name> ipv4-address

| Field | Description |
|---|---|
| StartAddress | Start address of the range. |
| EndAddress | End address of the range. |
| Route | Route allocated for this data plane. |
| N/P | Display the NodeMgr instance IDs from which it received routes Flag Indication S(Static) and O(Offline). |

### show ipam pool <pool-name> ipv6-addr

| Field | Description |
|---|---|
| StartAddress | Start address of the range. |
| EndAddress | End address of the range. |
| AllocContext | Name of data plane to which this address range is allocated. |
| Flag | Flag Indicate weather pool is Static or if it is offline. |

### show ipam

| Field | Description |
|---|---|
| PoolName | Displays Ipv4Utilization, Ipv6AddrUtilization, and Ipv6PrefixUtilization. |
| DpName | Displays Ipv4Utilization, Ipv6AddrUtlization, and Ipv6PrefixUtlization. |

### show nrf registration-info

*Table 7: show nrf registration-info Command Output Description*

| Field | Description |
|---|---|
| NF Status | Displays the NRF registration information. |
| Registration Time | Displays the time of registration with NRF. |
| Active MgmtEP Name | Displays the active NRF management endpoint name. |
| Heartbeat Duration | Displays the heartbeat duration. |
| Uri | Displays the Uri information. |

| Field | Description |
|---|---|
| Host Type | Displays the NRF host type information. |
| GR Instance ID | Displays the GR instance ID. |

## show nrf subscription-info

*Table 8: show nrf subscription-info Command Output Description*

| Field | Description |
|---|---|
| NF Instance Id | Displays the NF instance identity. |
| SubscriptionID | Displays the subscription identity information. |
| Actual Validity Time | Displays the actual validity time received from NRF server. |
| Requested Validity Time | Displays NF requested validity subscription time. |
| GR Instance ID | Displays the GR instance ID. |

## show nrf discovery info

*Table 9: show nrf discovery info Command Output Description*

| Field | Description |
|---|---|
| NF Type | Displays the NF type information. |
| Number of Discovery Filters | Displays the number of discovery filters. |
| Number of NF Profiles | Displays the number of NF profiles. |
| GR Instance ID | Displays the GR instance ID. |

## show nrf discovery-info AMF discovery-filter

*Table 10: show nrf discovery-info AMF discovery-filter Command Output Description*

| Step | Description |
|---|---|
| Discovery Filter | Displays the discovery filter information. |
| Expiry Time | Displays the expiry time for discovery filter. |
| GR Instance ID | Displays the GR instance ID. |

### show nrf discovery-info AMF discovery-filter <discovery_filter>

*Table 11: show nrf discovery-info AMF discovery-filter <discovery_filter> Command Description*

| Field | Description |
|---|---|
| NF InstanceId | Displays the NF Instance Identity. |
| NF Type | Displays the NF Type Information. |
| Discovery Filter | Displays the Discovery Filter Information. |
| NF Status | Displays the NF Status Information. |
| Priority | Displays the Priority Information. |
| Capacity | Displays the NF Profile Capacity Information. |
| Load | Displays the Load Information. |
| Locality | Displays the Locality Information. |
| ipv4 address | Displays IPv4 Address received from the discovery response for this NF profile. |
| ipv6 address | Displays the IPv6 Address received from the discovery response for this NF profile. |

### show nrf discovery-info AMF discovery-filter <discovery_name> nf-discovery-profile <nf_discovery_profile> nf-service

*Table 12: show nrf discovery-info AMF discovery-filter <discovery_name> nf-discovery-profile <nf_discovery_profile> nf-service Command Output Description*

| Field | Description |
|---|---|
| ServiceInstanceId | Displays the NF Service Instance ID. |
| ServiceName | Displays the NF Service Name. |
| UriScheme | Displays the Uri Scheme Information. |

### show peers all

Use this command to display the list all external inbound and outbound connections that are established by SMF. Only the key information is displayed. The following sample output is for the **show peers all** command:

```
[smf] smf# show peers all
GR                                                                    POD
CONNECTED                                    INTERFACE
INSTANCE  ENDPOINT      LOCAL ADDRESS   PEER ADDRESS     DIRECTION  INSTANCE     TYPE  TIME
       RPC     ADDITIONAL DETAILS       NAME       VRF
-------------------------------------------------------------------------------------------
1         <none>        xx.xx.xx.xx    xx.xx.xx.xx:0000  Outbound   rest-ep-0    Rest  25
hours   UDM     <none>                     n10
1         <none>        xx.xx.xx.xx    xx.xx.xx.xx:0000  Outbound   rest-ep-0    Rest  25
hours   UDM     <none>                     n10
1         <none>        xx.xx.xx.xx    xx.xx.xx.xx:0000  Outbound   rest-ep-0    Rest  25
hours   CHF     <none>                     n40
1         <none>        xx.xx.xx.xx    xx.xx.xx.xx:0000  Outbound   rest-ep-0    Rest  25
```

```
hours   PCF     <none>                      n7
1         <none>        xx.xx.xx.xx    xx.xx.xx.xx:0000  Outbound   rest-ep-0    Rest  25
hours   PCF     <none>                      n7
1         <none>        xx.xx.xx.xx    xx.xx.xx.xx:0000  Outbound   rest-ep-0    Rest  25
hours   AMF     <none>                      n11
```

*Table 13: Output Fields Description for the **show peers all** Command*

| Field | Description |
|---|---|
| Additional Details | Specifies the additional details for the peer such as status or type. |
| Connected Time | Specifies the duration of the connected peer. |
| Direction | Specifies if peer connection direction is inbound or outbound. |
| Endpoint | Specifies the name of the endpoint. |
| GR Instance | Specifies the GR instance. |
| Interface Name | Specifies the interface name for the endpoint. |
| Local Address | Specifies the local IP address and port of the instance. For endpoint, it is the endpoint address and port. For RPC, it is the instance IP. |
| Peer Address\| | Specifies the host and port of peer address. |
| Pod Instance | Specifies the pod for the peer. |
| RPC | Specifies the rpc of the specific peer. |
| Type | Specifies the type of peer. |

## show resources

Use this command to display the list of resource information for all pods in the cluster. The following sample output is for the **show resources** command:

```
[smf] show resources
                          TOTAL   USED    DISK
                          NODE    POD     USAGE  GO               GC
                    CPU   MEMORY  MEMORY  IN     ROUTINES  GC     PAUSE
POD INSTANCE        USAGE IN MB   IN MB   KBPS   COUNT     COUNT  IN NS
-------------------------------------------------------------------------
bfdmgr-1            0     32117   56      0      56        1950   56
bfdmgr-2            0     32117   55      0      56        1935   55
bfdmgr-3            1     32117   54      0      56        2636   54
bfdmgr-4            0     32117   55      0      56        1946   55
bgpspeaker-pod-1    1     32117   104     0      94        9315   104
bgpspeaker-pod-2    1     32117   102     0      78        9300   102
cache-pod-1         4     7962    96      0      325       778    96
cache-pod-2         10    32117   91      0      325       778    91
gtpc-ep-0           2     32117   82      0      160       777    82
internal-gr-pod-1   2     32117   124     0      317       63     124
internal-gr-pod-2   1     32117   93      0      182       63     93
li-ep-0             0     32117   64      0      68        2723   64
nodemgr-0           3     32117   113     0      270       784    113
nodemgr-1           2     32117   115     0      252       784    115
oam-pod-0           3     7962    121     0      249       2110   121
```

```
protocol-0          2       32117   82      0       159         777     82
radius-ep-0         5       32117   76      0       145         782     76
rest-ep-0           3       32117   105     0       298         779     105
sgw-service-0       9       32117   138     0       262         779     138
smf-service-0       3       32117   228     0       347         2645    228
udp-proxy-0         0       32117   72      0       112         778     72
udp-proxy-1         0       32117   72      0       112         778     72
```

*Table 14: Output Field Descriptions for the **show resources** Command*

| Field | Description |
|---|---|
| CPU Usage | Specifies CPU Usage In Percentage. |
| Disk Usage In Kbps | Specifies disk usage in Kbps. |
| GC Count | Specifies garbage collection cycle count. |
| GC Pause In NS | Specifies garbage collection pause in nanoseconds. |
| Go Routines Count | Specifies count of go routines. |
| Pod Instance | Specifies the instance info of the pod. |
| Total Node Memory In MB | Specifies total node memory usage in MB. |
| Used Pod Memory In MB | Specifies the consumption of pod memory in MB. |

## show rpc all

Use this **show rpc all** command to display the RPC configuration and remote host information for maximum of 200,000 RPCs.

The number of RPCs received per pod instance is limited to 50,000.

The following sample output is for the **show rpc all** command:

```
[smf] smf# show rpc all | tab | nomore

PROCESSING

                                                        CONNECTED  DISCONNECTED  MONITOR
INSTANCE
POD INSTANCE       NAME                        REMOTE ADDRESS             REMOTE HOST
    TYPE                    SET NAME          STATUS    TIME       TIME          RPCHOST
INFO                                              VERSION
-------------------------------------------------------------------------------------------
cache-pod-1        cache-pod-affinity           xx.xx.xx.xx:0000          cache-pod_20
    Grpc                    cache-pod_2       Started   4 weeks    <none>        false
<none>                                            <none>
```

```
cache-pod-1        cache-pod-affinity          xx.xx.xx.xx:0000          cache-pod_10
    Grpc                   cache-pod_1      Started   4 weeks    <none>        false
<none>                                           <none>
cache-pod-1        stream_cache-pod-affinity   xx.xx.xx.xx:0000          cache-pod_10
    GrpcServerClientStream  cache-pod_1      Started   4 weeks    <none>        false
<none>                                           <none>
cache-pod-1        stream_cache-pod-affinity   xx.xx.xx.xx:0000          cache-pod_20
    GrpcServerClientStream  cache-pod_2      Started   4 weeks    <none>        false
<none>                                           <none>
cache-pod-1        oam-pod                     xx.xx.xx.xx:0000          oam-pod
    GrpcStream             <none>           Started   4 weeks    <none>        false
<none>                                           <none>
cache-pod-1        Replication                 xx.xx.xx.xx:0000          cachepod_1
    GrpcStream             <none>           Started   3 weeks    <none>        false
<none>                                           <none>
cache-pod-1        Replication                 xx.xx.xx.xx:0000          cachepod_2
    GrpcStream             <none>           Started   3 weeks    <none>        false
<none>                                           <none>
cache-pod-2        cache-pod-affinity          xx.xx.xx.xx:0000          cache-pod_10
    Grpc                   cache-pod_1      Started   4 weeks    <none>        false
<none>                                           <none>
:
:
cache-pod-2        cache-pod-affinity          xx.xx.xx.xx:0000          cache-pod_20
    Grpc                   cache-pod_2      Started   4 weeks    <none>        false
<none>                                           <none>
cache-pod-2        stream_cache-pod-affinity   xx.xx.xx.xx:0000          cache-pod_20
    GrpcServerClientStream  cache-pod_2      Started   4 weeks    <none>        false
<none>                                           <none>
cache-pod-2        stream_cache-pod-affinity   xx.xx.xx.xx:0000          cache-pod_10
    GrpcServerClientStream  cache-pod_1      Started   4 weeks    <none>        false
<none>                                           <none>
cache-pod-2        oam-pod                     xx.xx.xx.xx:0000          oam-pod
    GrpcStream             <none>           Started   4 weeks    <none>        false
<none>                                           <none>
cache-pod-2        Replication                 xx.xx.xx.xx:0000          cachepod_1
    GrpcStream             <none>           Started   3 weeks    <none>        false
<none>                                           <none>
cache-pod-2        Replication                 xx.xx.xx.xx:0000          cachepod_2
    GrpcStream             <none>           Started   3 weeks    <none>        false
<none>                                           <none>
example-rest-ep-1  example-service             example-service:0000      example-service0
    Grpc                   example-service  Started   2 weeks    <none>        true
example.example-service.cluster1.example-data.12  <none>
```

***Table 15: Output Field Descriptions for the show rpc Command***

| Field | Description |
|---|---|
| Connected Time | Specifies the duration when the RPC host is connected. |
| Disconnected Time | Specifies the duration when the RPC host is disconnected. |
| Monitor RPC Host | Indicates whether the RPC host is being monitored for connection status. |
| Name | Displays the name of the RPC registered in pod. |
| Pod Instance | Displays the instance information of the pod. |

| Field | Description |
|---|---|
| Processing Instance Info | Indicates the processing instance name, if available. |
| Remote Address | Displays IP address and port of remote endpoint. |
| Remote Host | Displays the name of the RPC host. |
| Set Name | Displays the RPC set name for a group of RPC hosts. |
| Status | Displays the current status of the RPC host. The status values are Started, Starting, and Stopped. |
| Type | Displays the type of connection such as Rest, Grpc, and GrpcStream. |
| Version | Displays the version of the RPC host API, if available. |

## show running-status

Use this command to display the running status related information for all the pods in system. The following sample output is for the **show running-status** command:

```
[smf] smf# show running-status
                    RUNNING  SYSTEM  START
POD INSTANCE        STATUS   HEALTH  TIME
-----------------------------------------------
bfdmgr-1            Started  Normal  2 hours
bfdmgr-2            Started  Normal  2 hours
bgpspeaker-pod-1    Started  Normal  2 hours
bgpspeaker-pod-2    Started  Normal  2 hours
cache-pod-1         Started  Normal  2 hours
cache-pod-2         Started  Normal  2 hours
dns-proxy-0         Started  Normal  2 hours
dns-proxy-1         Started  Normal  2 hours
gtpc-ep1-1          Started  Normal  2 hours
gtpc-ep1-2          Started  Normal  2 hours
gtpc-ep2-1          Started  Normal  2 hours
gtpc-ep2-2          Started  Normal  2 hours
internal-gr-pod-1   Started  Normal  2 hours
internal-gr-pod-2   Started  Normal  2 hours
li-ep-0             Started  Normal  2 hours
li-ep-1             Started  Normal  2 hours
nodemgr-0           Started  Normal  2 hours
nodemgr-1           Started  Normal  2 hours
oam-pod-0           Started  Normal  2 hours
protocol1-1         Started  Normal  2 hours
protocol1-2         Started  Normal  2 hours
protocol2-1         Started  Normal  2 hours
protocol2-2         Started  Normal  2 hours
radius-ep-0         Started  Normal  2 hours
radius-ep-1         Started  Normal  2 hours
rest-ep-0           Started  Normal  2 hours
rest-ep-1           Started  Normal  2 hours
sgw-service-0       Started  Normal  2 hours
sgw-service-1       Started  Normal  2 hours
sgw-service-2       Started  Normal  2 hours
sgw-service-3       Started  Normal  2 hours
sgw-service-4       Started  Normal  2 hours
sgw-service-5       Started  Normal  2 hours
```

```
smf-service-0      Started  Normal  2 hours
smf-service-1      Started  Normal  2 hours
smf-service-2      Started  Normal  2 hours
smf-service-3      Started  Normal  2 hours
smf-service-4      Started  Normal  2 hours
smf-service-5      Started  Normal  2 hours
udp-proxy-0        Started  Normal  2 hours
udp-proxy-1        Started  Normal  2 hours
```

*Table 16: Output Field Descriptions for the **show running-status** Command*

| Field | Description |
|---|---|
| Pod Instance | Specifies the instance info of the pod. |
| Running Status | Specifies the system running status (Starting, Started, Stopping, or Stopped). |
| Start Time | Specifies the start time of the application. |
| System Health | Specifies the health status of the application. |

## show sessions affinity

Use this command to display affinity count, pod instance wise. This affinity count defines the affinity of sessions toward the pod. The following sample output is for the **show sessions affinity** command:

```
[smf] smf# show sessions affinity
POD
INSTANCE    COUNT
-------------------
service-1   10
service-11  12
service-12  15
service-13  12
service-14  15
service-2   15
service-3   14
service-4   19
```

*Table 17: Output Field Descriptions for the **show sessions affinity** Command*

| Field | Description |
|---|---|
| Count | Specifies the affinity count. |
| Pod Instance | Specifies the instance info of the pod. |

## show sessions commit-pending

Use this command to display the current number of sessions per pod along with the sessions that are pending commit in the database. The following sample output is for the **show sessions commit-pending** command:

```
[smf] smf# show sessions commit-pending
                                        DB
               GR                PENDING  BINARY  LAST DB SYNC
POD INSTANCE   INSTANCE  COUNT  COMMIT   SIZE    TIME
-------------------------------------------------------------------------
sgw-service-1  1         0      0        0       Less than a second
sgw-service-1  2         0      0        0       Less than a second
```

```
sgw-service-2  1          0       0         0       Less than a second
sgw-service-2  2          0       0         0       Less than a second
sgw-service-4  1          0       0         0       Less than a second
sgw-service-4  2          0       0         0       Less than a second
sgw-service-5  1          0       0         0       Less than a second
sgw-service-5  2          0       0         0       Less than a second
smf-service-0  1          0       0         0       Less than a second
smf-service-0  2          0       0         0       Less than a second
smf-service-1  1          0       0         0       Less than a second
smf-service-1  2          0       0         0       Less than a second
smf-service-2  1          0       0         0       Less than a second
smf-service-2  2          0       0         0       Less than a second
smf-service-4  1          0       0         0       Less than a second
smf-service-4  2          0       0         0       Less than a second
smf-service-5  1          0       0         0       Less than a second
smf-service-5  2          0       0         0       Less than a second
```

***Table 18: Output Field Descriptions for the show sessions commit-pending Command***

| Field | Description |
|---|---|
| Count | Specifies the count. |
| DB Binary Size | Specifies the DB binary Size. |
| GR Instance | Specifies the GR Instance ID. |
| Last DB Sync Time | Specifies the previous DB sync time. |
| Pod Instance | Specifies the instance info of the pod. |

## show subscriber

This commands displays the existing show subscriber CLI output with the newly added CLI output.

***Table 19: show subscriber command output description***

| Field | Description |
|---|---|
| all | Displays the information for all SUPIs or IMEIs. |
| amf | Displays the AMF address. |
| chf | Displays the CHF address. |
| count | Displays the number of sessions. |
| debug | Displays the debugging information. |
| dnn | Displays the DNN value. |
| gr-instance | Displays the Geographic Redundancy (GR) instance. |
| gtp-peer | Displays the GTP-peer address. |
| imei | Displays the IMEI containing 15 or 16 digits. |

| Field | Description |
|---|---|
| namespace | **Important**<br>This keyword is deprecated in release 2021.02.0 and replaced with the **nf-service** keyword.<br><br>Displays the product namespace under which to search. Default: none. |
| nf-service { none \| sgw \| smf } | Displays the network function service under which to search. Default: none. |
| pcf | Displays the PCF address. |
| rat | Displays the RAT type as 4G or 5G. |
| roaming-status | Displays the UE roaming status—homer, visitor-lbo, visitor-hr, roamer. |
| supi | Displays the SUPI value. |
| udm | Displays the UDM address. |
| upf | Displays the UPF address. |
| rulebase | Displays the subscriber using the rulebase. |
| \| | The output modifiers. |

## show subscriber all

Use this command to display all the sessions for all the SUPIs and NF services. The following sample output is for the **show subscriber all** command:

```
[smf] smf# show subscriber all
subscriber-details
{
  "subResponses": [
    [
      ""
    ],
    [
      "id-index:1:0:32768",
      "id-value:16777505",
      "imsi:imsi-123456123456123",
      "msisdn:msisdn-123456123456123",
      "imei:imei-310220000000000",
      "upf:xx.xx.xx.xx",
      "upfEpKey: xx.xx.xx.xx: xx.xx.xx.xx ",
      "s5s8Ipv4: xx.xx.xx.xx ",
      "s11Ipv4: xx.xx.xx.xx",
      "namespace:sgw",
      "nf-service:sgw"
    ],
    [
      "roaming-status:roamer",
      "ue-type:4g-only",
      "supi:imsi-123456123456123",
      "gpsi:msisdn-123456123456123",
      "pei:imei-310220000000000",
      "psid:69",
```

```
      "dnn:papn1.com",
      "emergency:false",
      "rat:e-utran",
      "access:3gpp access",
      "connectivity:4g",
      "auth-status:authenticated",
      "pcfGroupId:PCF-*",
      "policy:2",
      "pcf: xx.xx.xx.xx",
      "ipv4-addr:pool-static1-v4/xx.xx.xx.xx",
      "ipv4-pool:pool-static1-v4",
      "ipv4-range:pool-static1-v4/xx.xx.xx.xx",
      "ipv4-startrange:pool-static1-v4/",
      "id-index:1:0:32768",
      "id-value:8/310",
      "upf:xx.xx.xx.xx",
      "chfGroupId:CHF-*",
      "chf:209.165.202.133",
      "gtp-peer:xx.xx.xx.xx",
      "peerGtpuEpKey:xx.xx.xx.xx:xx.xx.xx.xx",
      "namespace:smf",
      "nf-service:smf"
    ],
    [
      ""
    ]
  ]
}
```


***Table 20: show subscriber Command Output Description***

| Field | Description |
|---|---|
| subscriber-details | Displays the details for all subscribers in JSON format. |

## show subscriber count

This command displays the CLI options for the count CLI command.

***Table 21: show subscriber count Command Output Description***

| Field | Description |
|---|---|
| all | Displays all the SUPIs. |
| amf | Displays the AMF address. |
| apn | Displays the APN value. |
| auth-status | Displays the RADIUS Authentication Status - authenticated or unauth status. |
| chf | Displays the CHF address. |
| connectivity | Displays the connectivity - 4g or 5g. |
| dnn | Displays the DNN value. |
| emergency | Displays the Emergency Session indication - true or false. |

| Field | Description |
|---|---|
| gpsi | Displays the GPSI value. |
| gr-instance | Displays the subscriber's from the provided GR Instance. |
| gtp-peer | Displays the GTP peer address. |
| ipv4-addr | Displays IPv4 address in the format:- <poolName> or <ipv4-addr>. |
| ipv4-pool | Displays the IPv4 pool name. |
| ipv4-range | Displays the IPv4 address range. |
| ipv6-pfx | Displays IPv6 prefix in the format <poolName> or <ipv6-pfx> |
| ipv6-pool | Displays the IPv6 pool name. |
| ipv6-range | Displays the IPv6 prefix range. |
| msid | Displays the MSID value. |
| msisdn | Displays the MSISDN value |
| namespace | Displays the deprecated option, use nf-service instead (default: none). |
| nf-service | Displays the network function service (SMF, S-GW) under which to search (default: none).<br>This parameter can be used with the slice name or the NSSAI filter for SMF in the following format:<br>**nf-service smf** *slice-name* or **nf-service smf** *nssai*. |
| pcf | Displays the PCF address. |
| peerGtpuEpKey | Displays the GTPU peer address in <upf_addr:gptu-peer-addr> format. |
| pei | Displays the PEI - Permanent Equipment Identifier. |
| policy | Displays the Subscriber Policy Information. |
| rat | Displays the RAT type as 4G or 5G. |
| roaming-status | Displays the UE roaming status – homer/roamer/visitor-hr/lbo-visitor. |
| smf | Displays the SMF address. |
| supi | Displays the specific SUPI value. |
| udm-sdm | Displays the UDM-SDM Address. |
| udm-uecm | Displays the UDM-UECM Address. |
| ue-type | Displays the device capability - 4g-only or nr-capable. |
| upf | Displays the UPF address. |

| Field | Description |
|---|---|
| rulebase | Displays the subscriber using the rulebase. |
| \| | Displays the output modifiers. |

## show subscriber count all

Use this command to display the total number of sessions for all the SUPIs. The following sample output is for the **show subscriber count all** command:

```
[smf] smf# show subscriber count all
subscriber-details
{
  "sessionCount": 20
}
```

*Table 22: Output Field Descriptions for the show subscriber count all Command*

| Field | Description |
|---|---|
| subscriber-details | Displays the count for all subscribers in JSON format. |

## show subscriber debug-info

This command displays the debug information for the specific SUPI value where the PSID value is optional.

*Table 23: show subscriber debug-info Command Output Description*

| Field | Description |
|---|---|
| gpsi | Displays GPSI value. |
| gr-instance | Displays the subscriber's from the provided GR Instance. |
| imsi | Displays the IMSI value. |
| msid | Displays the MSID value. |
| msisdn | Displays the MSISDN value. |
| namespace | Deprecated option, Use nf-service instead (default: none) |
| nf-service | Displays the network function service (SMF, SGW) under which to search (default: none). |
| pei | Displays the PEI or IMEI value. |
| supi | Displays the SUPI value, value must include the imsi- prefix. |
| \| | Displays the output modifiers. |

## show subscriber gpsi

*Table 24: show subscriber gpsi*

| Field | Description |
|---|---|
| policy | Displays the policy information. |
| ipv4-addr | Displays the IPv4 pool name. |
| dnn | Displays the DNN value. |
| pcf | Displays the PCF Address. |
| rat | Displays the RAT Type—nr, e-utran, or wlan information. |
| connectivity | Displays the connectivity—4G or 5G. |
| ipv4-range | Displays the IPv4 address range. |
| chf | Displays the CHF address. |
| pei | Displays the Permanent Equipment Identifier (PEI). |
| udm | Displays the UDM address. |
| upfEpKey | Displays the UPF address EP key information. |
| ipv6-pfx | Displays the IPv6 prefix information. |
| ipv6-pool | Displays the IPv6 pool name. |
| chfGroupId | Displays the CHF address group ID information. |
| gpsi | Displays the Generic Public Subscription Identifier (GPSI). |
| pcfGroupId | Specifies PCF Address group ID. |
| upf | Displays the UPF address. |
| ipv4-pool | Displays the IPv4 pool name. |
| ipv6-range | Displays the IPv4 address range. |
| amf | Displays the AMF address. |
| supi | Displays the SUPI value. |
| access | Displays the access information. |
| gr-instance | Displays the GR instance. |

## show subscriber nf-service smf

**Important** The wildcard input is not supported with the listed filters.

*Table 25: show subscriber nf-service smf Command Output Description*

| Field | Description |
|---|---|
| apn | Displays the APN value. |
| msid | Displays the MSID value. |
| msisdn | Displays the MSISDN value. |
| roaming-status | Displays the UE roaming status—homer, visitor-lbo, visitor-hr, roamer. |
| smf | Displays the subscriber details based on the IP address value of the vSMF or hSMF.<br>For example:<br>`[smf] smf# show subscriber nf-service smf smf <smf_url> subscriber-details`<br>`{}` |
| rulebase | Displays the subscriber using the rulebase. |

## show subscriber pei

*Table 26: show subscriber pei*

| Field | Description |
|---|---|
| policy | Displays the policy information. |
| ipv4-addr | Displays the IPv4 pool name. |
| dnn | Displays the DNN value. |
| pcf | Displays the PCF Address. |
| rat | Displays the RAT Type—nr, e-utran, or wlan information. |
| connectivity | Displays the connectivity—4G or 5G. |
| ipv4-range | Displays the IPv4 address range. |
| chf | Displays the CHF address. |
| pei | Displays the Permanent Equipment Identifier (PEI). |
| udm | Displays the UDM address. |
| upfEpKey | Displays the UPF address EP key information. |

| Field | Description |
|---|---|
| ipv6-pfx | Displays the IPv6 prefix information. |
| ipv6-pool | Displays the IPv6 Pool name. |
| chfGroupId | Displays the CHF address group ID information. |
| gpsi | Displays the Generic Public Subscription Identifier (GPSI). |
| pcfGroupId | Displays the PCF address group ID. |
| upf | Displays the UPF address. |
| ipv4-pool | Displays the IPv4 pool name. |
| ipv6-range | Displays the IPv4 address range. |
| amf | Displays the AMF address. |
| supi | Displays the SUPI value. |
| access | Displays the access information. |
| gr-instance | Displays the GR instance. |
| rulebase | Displays the subscriber using the rulebase. |

### show subscriber supi <supi_value> nf-service smf psid <psid_value> full

This command displays detailed subscriber information.

*Table 27: show subscriber supi <supi_value> nf-service smfpsid <psid_value> full Command Output Description*

| Field | Description |
|---|---|
| sessTimeStamp | Displays the connected time of the session. |
| callDuration | Displays the call duration. |
| commonId | Displays the call ID equivalent for the session (common ID). |
| ipPool, ipv6Pool | Displays the IP pool from which the address has been allocated. |
| linkedEbi | Displays the linked EBI for a session. |
| snssai | Displays the sNssai details. |
| smfIwkEpsInd | Displays the SMF EPS IWK decision based on AMF and UDM data. |
| TotalNumberOfPdrs | Displays the number of associated PDRs. |
| TotalNumberOfFars | Displays the number of associated FARs. |
| TotalNumberOfQers | Displays the number of associated QERs. |

| Field | Description |
|---|---|
| TotalNumberOfUrrs | Displays the number of associated URRs. |
| upfSeid | Displays the remote SEID for a particular UPF session. |
| epsInterworking Indication | Displays the EPS interworking indication status of AMF. |
| ebi | Displays the ERAB ID allocated for each flow. |
| revalidationTime | Displays the revalidation timer information for a session. |

### show subscriber supi <supi_value> nf-service smfpsid <psid_value> summary

This command displays detailed information about subscriber sessions. This command improves usability and can be used for debugging purposes.

*Table 28: show subscriber supi <supi_value> nf-service smf psid <psid_value> summary Command Output Description*

| Field | Description |
|---|---|
| supi | Displays the 5G Subscription Permanent Identifier. |
| pduSessionId | Displays the PDU session identifier. |
| pduSesstype | Displays the PDU session type. |
| accessType | Displays the access type. |
| dnn | Displays the DNN profile name. |
| allocatedIp/ allocatedIpv6 | Displays the allocated IP address details. |
| ratType | Displays the RAT type. |
| sessTimeStamp | Displays the connected Time of the session. |
| TotalDynamicRules/ TotalStaticRules/ TotalPredefinedRules | Displays the number of Dynamic rules or Static rules or Predefined rules. |
| TotalGBRFlows/ TotalNonGBRFlows | Displays the number of GBR flows or non-GBR flows. |
| pcfInteraction | Displays the PCF interaction status. |
| ruleBase | Displays the rulebase name. |
| chargingId | Displays the charging descriptor name. |
| offlineConverted | Displays the online charging parameters converted to offline. |
| chargingDisabled | Displays the charging parameters when charging is disabled. |
| dropTraffic | Displays the charging parameters when traffic is dropped. |

| Field | Description |
|---|---|
| gtppGrp | Displays the EGCDR configuration for GTPP name. |
| profileName | Displays the charging profile name. |
| deferredUsageCount | Displays the number of deferred multi-unit usages. |
| smfSeid | Displays the local SEID for a particular UPF session. |
| upfSeid | Displays the remote SEID for a particular UPF session. |
| TunnelID | Displays the GTPU peer tunnel ID. |
| TunnelName | Displays the GTPU peer tunnel name. |
| RemoteTeid (teid/ipAddr) | Displays the GTPU peer TEID and IP address. |
| TotalNumberOfPdrs | Displays the number of associated PDRs. |
| TotalNumberOfFars | Displays the number of associated FARs. |
| TotalNumberOfQers | Displays the number of associated QERs. |
| TotalNumberOfUrrs | Displays the number of associated URRs. |

## clear Commands

This section lists some of the key clear commands that are available for troubleshooting the issues.

**Important** The SMF Ops center allows you to issue only one **clear subscriber all** command at a time. The Ops center restricts the subsequent **clear subscriber all** and other variants of **clear subscriber** commands until the ongoing **clear subscriber all** command is complete.

**Note** The Ops Center displays the expected waiting time for an ongoing bulk **clear subscriber** CLI command. In addition, the clear subscriber CLI gets blocked while the processing of the earlier CLI is in progress.

### clear subscriber

"`clear subscriber`" command displays the list of subscriber SMF fields.

***Table 29: clear subscriber command output description***

| Field | Description |
|---|---|
| all | Clears all the session information. |
| amf | Clears subscriber based on AMF address information. |
| chf | Clears subscriber based on CHF address information. |

| Field | Description |
|---|---|
| dnn | Clears subscriber based on DNN value. |
| gr-instance | Clears subscriber based on the specified Geographic Redundancy (GR) instance information. |
| gtp-peer | Clears subscriber based on GTP peer address information. |
| ipv4-pool | Clears subscriber based on IPv4 pool name. |
| ipv4-range | Clears subscriber based on IPv4 address-range value. |
| ipv6-pool | Clears subscriber based on IPv6 pool name information. |
| ipv4-range | Clears subscriber based on IPv4 prefix-range value. |
| ipv6-range | Clears subscriber based on IPv6 prefix-range value. |
| namespace | **Important**<br>This keyword is deprecated in release 2021.02.0 and is replaced with the **nf-service** keyword.<br><br>Clears subscriber based on the respective namespace. Default: none. |
| nf-service { none \| sgw \| smf } | Clears subscriber based on the specified network function service. Default: none. |
| pcf | Clears subscriber based on PCF address information. |
| policy | Clears subscriber information based on policy. |
| purge | Clears subscriber information, if purged locally. |
| reactivation [ true \| false ] | Clears subscriber based on the Reactivation Required cause value. This option is set to true if reactivation is requested. |
| roaming-status | Clears subscriber based on the UE roaming status—homer, visitor-lbo, visitor-hr, roamer values. |
| sgw | Clears subscriber information based on the S-GW address information. |
| smf | Clears subscriber information based on the SMF address information. |
| supi | Clears subscriber based on the SUPI value. |
| rulebase | Clears subscriber using the rulebase. |
| \| | The output modifiers. |

## clear subscriber nf-service smf



**Important** The wildcard input is not supported with the listed filters.

"`clear subscriber nf-service smf` " command displays the list of nf-service SMF fields.

***Table 30: clear subscriber nf-service smf Command Output Description***

| Field | Description |
|---|---|
| apn | Clears subscriber based on the APN value. |
| connected - duration* | Clears subscriber sessions based on connected duration with conditions: greater-than, lesser-than, equals (in seconds). Supported with specified option combinations.<br>The valid range is 1–604800 seconds.<br>For example:<br>`clear subscriber nf-service smf rat nr connected-duration greater-than 10` |
| dnn | Clears subscriber based on the DNN value. |
| max-count | Clears a random number of sessions less than the specified max-count value. Supported with specified option combinations.<br>For example:<br>`clear subscriber nf-service smf rat nr max-count { lesser-than 6 }` |
| msid | Clears subscriber based on the MSID value. |
| msisdn | Clears subscriber based on the MSISDN value. |
| non-emergency | Clears subscriber sessions matching the specified options for non-emergency sessions. |
| non-volte | Clears subscriber sessions matching the specified options for non-VoLTE sessions. |
| rate | Clears subscriber based on the rate value. The maximum value for the rate is 1500. |
| reactivation [ true \| false ] | Clears subscriber based on the Reactivation Required cause. This option is set to true if reactivation is requested. |
| roaming-status | Clears subscriber based on the UE roaming status—homer, visitor-lbo, visitor-hr, roamer. |
| rulebase | Clears subscriber based on the rulebase value.<br>This keyword is used as a secondary filter. Ensure that the rulebase value includes the rulebase prefix.<br>For example:<br>`[smf] smf# clear subscriber nf-service smf dnn <dnn_val> rulebase <rulebase_value>`<br>`result`<br>`ClearSubscriber Request submitted` |

| Field | Description |
|---|---|
| rulename | Modifies session based on the rulename value.<br>This keyword is used as a secondary filter. Ensure that the rulename value includes the rulename prefix.<br>For example:<br>`[smf] smf# clear subscriber nf-service smf dnn <dnn_val> rulename <rulename_value>`<br>`result`<br>`ClearSubscriber Request submitted` |
| SMF | Clears subscriber based on the IP address value of the vSMF or hSMF.<br>For example:<br>`[smf] smf# clear subscriber nf-service smf smf <smf_url>`<br>`result`<br>`ClearSubscriber Request submitted` |
| x5qi | Modifies session based on 5QI for 5G sessions, and QCI for 4G and WLAN sessions. This keyword is used as a secondary filter.<br>For example:<br>`smf] smf# clear subscriber supi <supi_val> x5qi <x5qi_value>`<br>`result`<br>`ClearSubscriber Request submitted`<br><br>`[smf] smf# clear subscriber apn <apn_val> x5qi <x5qi_value>`<br>`result`<br>`ClearSubscriber Request submitted` |

**Note**

- Combination of **connected-duration** and **max-count** option is not supported.
- When using the **non-volte** and **max-count** filters together in the clear subscriber command, the number of sessions actually cleared may be lower than the specified **max-count**.

### clear subscriber supi imsi <imsi_value>

"`clear subscriber supi imsi` *`imsi_value`*" command displays the list of subscriber SUPI IMSI value SMF fields.

***Table 31: clear subscriber supi imsi <imsi_value> Command Output Description***

| Field | Description |
|---|---|
| ebi | Clears subscriber based on EPS bearer ID value. |
| imsi | Clears subscriber based on IMSI information. |
| purge | Clears true, if purged locally. |

| Field | Description |
|---|---|
| \| | Output modifier. |

### clear subscriber supi imsi <imsi_value> psid <psid_value>

"`clear subscriber supi imsi` *`imsi_value`* `psid` *`psid_value`*" command displays the list of subscriber SUPI IMSI and PSID value SMF fields.

***Table 32: clear subscriber supi imsi <imsi_value> psid <psid_value> Command Output Description***

| Field | Description |
|---|---|
| ebi | Clears subscriber based on EPS bearer ID value. |
| imsi | Clears subscriber based on IMSI information. |
| psid | Clears subscriber based on Service ID value. |
| purge | Clears true, if purged locally. |
| \| | Output modifier. |

# Monitor Subscriber and Monitor Protocol

## Feature Description

The SMF supports the Monitor Subscriber and Monitor Protocol on the Kubernetes environment. The monitor tools allow you to capture messages of subscribers and protocols.

This section provides information on CLI commands for monitoring the health of SMF.

## Configuring the Monitor Subscriber and Monitor Protocol Feature

### Monitoring the Subscriber Session

To monitor the subscriber in the SMF, use the following CLI command:

**`monitor subscriber [ capture-duration`** *`duration`* **`| gr-instance`** *`gr_instance_id`* **`| imei`** *`imei_id`* **`| imsi`** *`imsi_value`* **`| internal-messages [ yes ] | namespace [ sgw | smf ] | nf-service [ sgw | smf ] | supi`** *`supi_id`* **`| transaction-logs [ yes ] ]`**

**NOTES**:

- **capture-duration** *duration*: Specify the duration in seconds during which monitor subscriber is enabled. The default value is 300 seconds (5 minutes). This is an optional parameter.
- **gr-instance** *gr_instance_id*: Specify the GR instance ID. The instance ID 1 denotes the local instance ID.
- **imei** *imei_id*: Specify the subscriber IMEI. For example: 123456789012345, *
- **imsi** *imsi_value*: Specify the subscriber IMSI. For example: 123456789, *

- **internal-messages [ yes ]**: Enable internal messages when set to **yes**. By default, it is disabled. This is an optional parameter.
- **namespace [ sgw | smf ]**: Enable the specified namespace. By default, namespace is set to none. This is an optional parameter.

**Important** This keyword is deprecated in release 2021.02.0 and replaced with **nf-service** keyword.

- **nf-service [ sgw | smf ]** : Enable the specified NF service. By default, nf-service is set to none.

**Important** The **nf-service** keyword replaces the **namespace** keyword in release 2021.02 and beyond.

- **supi** *supi_id*: Specify the subscriber identifier. For example: imsi-123456789, imsi-123*
- **transaction-logs [ yes ]**: Enable transaction logs when set to **yes**. By default, it is disabled. This is an optional parameter.

  To view the transaction history logs, use the **dump transactionhistory** command.

**Note** The most recent transaction logs are stored in a circular queue of size 1024 transaction logs.

The **monitor subscriber** CLI command can be run simultaneously on multiple terminals. For example, run the CLI simultaneously in two SMF Ops Center terminals for two subscribers (for example, imsi-123456789012345 and imsi-456780123456789) to implement the following:

- Monitor the duration when the monitor subscriber is enabled
- View internal messages for the specified subscriber
- View transaction logs for the specified subscriber

Terminal 1: The following command monitors and displays subscriber messages for the specified subscriber.

```
monitor subscriber supi imsi-123456789012345 capture-duration 1000 internal-messages yes
```

Terminal 2: The following command monitors and displays transaction logs for the specified subscriber.

```
monitor subscriber supi imsi-456780123456789 capture-duration 500 internal-messages yes
transaction-logs yes
```

After the capture duration is completed, stop the CLI by using the **Ctrl+C** keys. The captured messages are reordered and stored in a file. To retrieve the list of stored files, use the **monitor subscriber list** CLI command.

For example:

```
monitor subscriber list
RELEASE_NAMESPACE: 'smf'
'monsublogs/subscriberID_imsi-*_AT_2019-10-22T09:19:05.586237087.txt.sorted'
monsublogs/subscriberID_imsi-123456789012345_AT_2019-10-22T09:20:11.122225534.txt.sorted
```

## Monitoring Subscriber Dump

To view the sorted file on the SMF Ops Center screen, use the following CLI command:

**monitor subscriber dump filename** *filename*

For example:

```
monitor subscriber dump filename
monsublogs/subscriberID_imsi-123456789012345_AT_2019-10-22T09:20:11.122225534.txt.sorted
```

## Monitoring the Interface Protocol

To monitor the interface protocol on the SMF, use the following CLI command:

**monitor protocol { interface** *interface_name* **[ capture-duration** *duration* **| gr-instance***gr_instance* **| pcacp yes | | ] | list [ | ] }**

**Warning** Use the **monitor protocol interface** command cautiously, as it may disrupt user services. When used in a live network, it can cause session delays, call drops, and degrade system performance.

**Note**

- **interface** *interface_name*—Specify the interface name on which PCAP is captured. This CLI allows the configuration of multiple interface names in a single CLI command.
- **capture-duration** *duration*—Specify the duration in seconds during which pcap is captured. The default is 300 seconds (5 minutes).
- The configured interface names can be retrieved using the **show endpoint** CLI command.
- **gr-instance** *gr_instance_id*—Specify the GR instance ID. The instance ID 1 denotes the local instance ID.
- **pcap yes**—Configure this option to enable PCAP file generation. By default, this option is disabled.
- **list**—Monitor protocol list files.

**Important** The **monitor protocol** command in Exec mode is restricted based on pod's CPU utilization configured through **monitor protocol cpu-limit** *threshold_percentage* command in the Global Configuration mode.

The **monitor protocol** CLI can be run simultaneously on multiple terminals. Also, the **interface** *interface_name* CLI allows the configuration of multiple endpoint names in a single CLI command.

For example:

```
monitor protocol interface sbi,N4:209.165.200.241:8805,gtpc
capture-duration 1000
```

# UPF Monitor Subscriber from SMF

## Feature Description

SMF sends the tracing trigger to the selected UPF automatically. Sending the trigger from SMF facilitates in parsing minimum files for troubleshooting across Control Plane and User Plane.

**Note**

- The operator can configure a maximum of five subscribers at a time on SMF for monitoring.
- When the data tracing is enabled, the available VPP, FCAP, and MEH are captured on the UPF PCAP.
- When the protocol tracing is enabled, the following options are enabled on UPF:
  - Rulematch Events
  - L3 Data
  - PFCP Events Tracing
  - EDR
  - SessMGR
  - Subscriber Summary After Call Disconnect
- After SMF GR switchover, the existing sessions of the monitored subscribers aren't impacted on UPF. As monitor nf CLI exists in the config mode, an operator needs to configure the CLI on both the SMF instances.

For UPF reload and switchover scenarios, see the UPF troubleshooting guide for monitor NF.

This section provides information on CLI commands for configuring the UPF monitor subscriber from SMF.

## Configuring UPF Monitor Subscriber from SMF

Use the following configuration to enable or disable the UPF Monitor Subscriber protocol from SMF.

- Enabling UPF Monitor Subscriber from SMF
- Disabling UPF Monitor Subscriber from SMF

### Enabling UPF Monitor Subscriber from SMF

Use the following configuration to enable UPF Monitor Subscriber from SMF.

```
config
   monitor nf subscriber  [ gpsi gpsi_value |imei imei_id|imsiimsi_value control
 { true | false }  data { true | false } target-nf { sgwu | upf } ]
   end
```

**NOTES**:

- **monitor nf** —Specify the NF that you want to monitor.
- **gpsi** *gpsi_value*—Specify the subscriber GPSI.

- **imsi** *imsi_value*—Specify the subscriber IMSI. For example: 123456789, *
- **imei** *imei_id*—Specify the subscriber IMEI. For example: 123456789012345, *
- **control { true | false }**—Specify whether to enable or disable the control event tracing.
- **data { true | false }** —Specify whether to enable or disable the data event tracing.
- **target-nf { sgwu | upf }** —Specify the target NF as SGW-U or UPF. **target-nf** is an optional parameter and if an operator doesn't configure this parameter, monitor subscriber is enabled on both the UPF and SGW-U.

### Configuration Example

The following is an example configuration.

```
[smf] smf(config)# monitor nf subscriber
   imsi 234150999999999
      control true
      data true
      target-nf upf
exit
```

## Disabling UPF Monitor Subscriber from SMF

Use the following configuration to disable UPF Monitor Subscriber from SMF.

```
config
   [ no ]monitor nf subscriber  [ gpsi gpsi_value |imei imei_id | imsiimsi_value
 ]
   end
```

**NOTES**:

- **monitor nf** —Specify the NF that you want to monitor.
- **gpsi** *gpsi_value*—Specify the subscriber GPSI.
- **imsi** *imsi_value*—Specify the subscriber IMSI. For example: 123456789, *
- **imei** *imei_id*—Specify the subscriber IMEI. For example: 123456789012345, *

# Monitor and Troubleshoot GTP-C Services

*Table 33: Feature History*

| Feature Name | Release Information | Description |
|---|---|---|
| Troubleshooting GTP-C Services | 2024.02.0 | This feature allow the cnSGW+SMF+PGW-C to send GTPC test echo command to peer nodes to:<br>• Troubleshoot and monitor the connectivity of peer nodes.<br>• Test a Round Trip Time (RTT) of the peer.<br>• Supports S11, S5E, S5, S2B and S8 GTPC interfaces, and IPv4 and IPv6 transport types.<br>**Default Setting**: Enabled – Always-on |

## Feature Description

The GTP-C test echo CLI commands on services allows you to troubleshoot and monitor the connectivity of peer nodes, and provides a round trip time (RTT) of the peer during system operation.

## Using the GTP-C Test Echo Command

The system uses the following test command to check the connectivity with GTP-C peer nodes.

**test gtpc echo instance-id** *id* **interface** *interface_type* **peer-address** *ip_address*

**NOTES**:

- **instance-id** *id*: Specifies the Instance ID of a GR instance that is configured on the system.
- **interface** *interface_type*: Specifies the GTP-C interface type such as s11, s5e, s5, s2b, or s8.
- **peer address** *ip_address*: Specifies specific peer IPv4 or IPv6 address to which GTP-C echo request is sent.

The following example displays sample output.

```
Success Case
[sgw] smf# test gtpc echo instance-id 1 interface s11 peer-address 1.1.1.1
 result
{
  "testGtpcEchoResponse": {
    "rx": 1,
    "tx": 1,
    "rtt(ms)": 3,
    "recovery": "10 (0x0A)",
    "status": {
      "success": true
```

```
    }
  }
}

Timeout Case:
[sgw] smf# test gtpc echo instance-id 1 interface s5 peer-address 2.2.2.2

Wed Jan  10 13:38:51.415 UTC+00:00
result
{
  "testGtpcEchoResponse": {
    "tx": 4,
    "recovery": "NA",
    "status": {
      "errorMsg": "No Response Received, Timeout"
    }
  }
}
```

## Using the GTPP Test Echo Command

The system uses the following test command to check the new CGF server connectivity that is not configured in a GTPP profile.

**test gtpp echo instance-id** *id* **ca-address** *ip_address* **[ ca-port** *port* **] [ cgf-address** *ip_address* **] [ cgf-port** *port* **]**

**NOTES**:

- **instance-id** *id* : Specifies the Instance ID of a GR instance that is configured on the system.
- **ca-address** *ip_address*: Specifies the charging agent IPv4 address configured within gtpp profile.
- **ca-port** *port*: Specifies the charging agent port configured within gtpp profile, ranging 1–65535. By default port is 49999.
- **cgf-address** *ip_address*: Specifies the specific CGF server IPv4 address to which GTPP echo request is sent.
- **cgf-port** *port*: Specifies specific CGF Server port, ranging from 1– 65535. By default port is 3386.

The following example displays a sample of the GTPP Echo command output.

```
Success Case:
[sgw] smf# test gtpp echo instance-id 1 ca-address 10.0.0.1 ca-port 2222 cgf-address 11.0.0.1


Tue Jan  23 12:26:40.164 UTC+00:00
result
{
  "testGtppEchoResponse": {
    "rx": 1,
    "tx": 1,
    "rtt(ms)": 3,
    "recovery": "10 (0x0A)",
    "status": {
      "success": true
    }
  }
}

Timeout Case:
```

```
[sgw] smf# test gtpp echo instance-id 1 ca-address 10.0.0.1 ca-port 2222 cgf-address 11.0.0.2


Tue Jan  24 07:43:22.164 UTC+00:00
result
{
  "testGtppEchoResponse": {
    "tx": 4,
    "recovery": "NA",
    "status": {
      "errorMsg": "No Response Received, Timeout"
    }
  }
}
```

# Alerts

## Feature Description

When the system detects an anomaly, CEE Ops Center generates an alert notification. The system statistics are the cause for these alert notifications. You can set an expression to trigger an alert when the expression becomes true.

## How it Works

The Common Execution Environment (CEE) uses the Prometheus Alert Manager for alerting operations. The CEE YANG model - either through CLI or API - allows users to view the active alerts, silenced alerts, and alert history. Also, the applications can call the alert API directly to add or clear alerts. The Prometheus Alert Manager API (v2) is the standard API used.

The Prometheus Alerts Manager includes the following options:

- **Defining Alert Rules**: This option defines the types of alerts that the Alert Manager should trigger. Use the Prometheus Query Language (PromQL) to define the alerts.
- **Defining Alert Routing**: This option defines the action the Alert Manager should take after receiving the alerts. At present, the SNMP Trapper is supported as the outbound alerting. Also, the CEE provides an Alert Logger for storing the generated alerts.

## Configuring Alert Rules

Use the following sample configuration to configure the alert rules:

```
config
   alerts rules group alert_group_name
   interval-seconds seconds
   rule rule_name
      expression promql_expression
      duration duration
```

```
      severity severity_level
      type alert-type
      annotation annotation_name
      value annotation_value
      exit
   exit
```

**NOTES**:

- **alerts rules**: Specifies the Prometheus alerting rules.
- **group** *alert_group_name*: Specifies the Prometheus alerting rule group. One alert group can have multiple lists of rules. *alert-group-name* is the name of the alert group. The alert-group-name must be a string in the range of 0–64 characters.
- **interval-seconds** *seconds*: Specifies the evaluation interval of the rule group in seconds.
- **rule** *rule_name*: Specifies the alerting rule definition. *rule_name* is the name of the rule.
- **expression** *promql_expression*: Specifies the PromQL alerting rule expression. *promql_expression* is the alert rule query expressed in PromQL syntax.
- **duration** *duration*: Specifies the duration of a true condition before it is considered true. *duration* is the time interval before the alert is triggered.
- **severity** *severity_level*: Specifies the severity of the alert. *severity-level* is the severity level of the alert. The severity levels are critical, major, minor, and warning.
- **type** *alert_type*: Specifies the type of the alert. *alert_type* is the user-defined alert type. For example, Communications Alarm, Environmental Alarm, Equipment Alarm, Indeterminate Integrity Violation Alarm, Operational Violation Alarm, Physical Violation Alarm, Processing Error Alarm, Quality of Service Alarm, Security Service Alarm, Mechanism Violation Alarm, or Time Domain Violation Alarm.
- **annotation** *annotation_name*: Specifies the annotation to attach to the alerts. *annotation_name* is the name of the annotation.
- **value** *annotation_value*: Specifies the annotation value. *annotation_value* is the value of the annotation.

The following example configures an alert, which is triggered when the percentage of Unified Data Management (UDM) responses is less than the specified threshold limit.

**Example:**

```
 config terminal
  alerts rules group SMFUDMchk_incr
  interval-seconds 300
  rule SMFUDMchk_incr
  expression "sum(increase(smf_restep_http_msg_total{nf_type=\"udm\",
message_direction=\"outbound\", response_status=~\"2..\"}[3m])) /
sum(increase(smf_restep_http_msg_total{nf_type=\"udm\", message_direction=\"outbound\"}[3m]))
 < 0.95"
  severity major
  type "Communications Alarm"
  annotation summary
  value "This alert is fired when the percentage of UDM responses is less than threshold"
  exit
 exit
exit
```

You can view the configured alert using the **show running-config alerts** command.

**Example:**

The following example displays the alerts configured in the running configuration:

```
 show running-config alerts
  interval-seconds 300
  rule SMFUDMchk_incr
  expression "sum(increase(smf_restep_http_msg_total{nf_type=\"udm\",
message_direction=\"outbound\", response_status=~\"2..\"}[3m])) /
sum(increase(smf_restep_http_msg_total{nf_type=\"udm\", message_direction=\"outbound\"}[3m]))
 < 0.95"
  severity major
  type "Communications Alarm"
  annotation summary
  value "This alert is fired when the percentage of UDM responses is less than threshold‟

  exit
 exit
exit
```

## Viewing Alert Logger

The Alert Logger stores all the generated alerts by default. You can view the stored alerts using the following command.

**show alert history [ filtering ]**

You can narrow down the result using the following filtering options:

- **annotations**: Specifies the annotations of the alert.
- **endsAt**: Specifies the end time of the alert.
- **labels**: Specifies the additional labels of the alert.
- **severity**: Specifies the severity of the alert.
- **source**: Specifies the source of the alert.
- **startsAt**: Specifies the start time of the alert.
- **type**: Specifies the type of the alert.

The following example history of the alerts configured in the system appears:

**Example:**

```
show alerts history
alerts active SMFUDMchk_incr ac2a970ab621
state active
severity major
type "Communications Alarm"
startsAt 2019-11-15T08:26:48.283Z
source System
annotations [ "summary:This alert is fired when the percentage of UDM responses is less
than threshold.” ]
```

You can view the active and silenced alerts with the **show alerts active** command.

The following active alerts example appears. The alerts remain active as long as the evaluated expression is true.

**Example:**

```
show alerts active
alerts active SMFUDMchk_incr ac2a970ab621
state active
severity major
type "Communications Alarm"
startsAt 2019-11-15T08:26:48.283Z
source System
annotations [ "summary:This alert is fired when the percentage of UDM responses is less
than threshold.” ]
```

# Call Flow Procedure Alerts

This section provides detail of commands that are required to configure alerts related to various call flow procedures.

The alerts, which are specific to SMF, are configured on the Common Execution Environment (CEE). The expressions are developed and new counters are created. Based on the user requirements, the call flow procedure alerts are configured in CEE. These alerts are triggered when the conditions, as specified by users, are met.

## 4G PDN Modify

Use the following sample configuration to configure alerts related to the 4G PDN Modify procedure:

```
alerts rules group SMFPDN
   interval-seconds 300
   rule SMFPDNModify
   expression “sum(smf_service_stats{procedure_type=~\”pdn_ho_location_changed|
pdn_ho_rat_type_changed|pdn_inter_sgw_handover|pdn_mbr\” ,
status=\“success\”})/sum(smf_service_stats{procedure_type=~
\”pdn_ho_location_changed|pdn_ho_rat_type_changed |pdn_inter_sgw_handover|pdn_mbr\” ,
status=\“attempted\”}) < 0.95”
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage successful 4G PDN Modify is below
threshold”
   exit
exit
```

## 4G PDN Release Success

Use the following sample configuration to configure alerts related to the 4G PDN Release Success procedure:

```
alerts rules group SMFPDN
   interval-seconds 300
   rule SMFPDNRelease
   expression “sum(smf_service_stats{procedure_type=~\“.*pdn_sess_rel\” ,
status=\“success\”}) / sum(smf_service_stats{procedure_type=~\“.*pdn_sess_rel\” ,
status=\“attempted\”} < 0.95 “
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage successful 4G PDN Release is below
threshold.”
   exit
exit
```

### 4G PDN Setup Success

Use the following sample configuration to configure alerts related to the 4G PDN Setup Success procedure:

```
alerts rules group SMFPDN
   interval-seconds 300
   rule SMFPDNSetup
   expression "sum(smf_service_stats{procedure_type=\"pdn_sess_create\" ,
status=\"success\"}) / sum(smf_service_stats{procedure_type=\"pdn_sess_create\" ,
status=\"attempted\"} < 0.95 "
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage successful 4G PDN Setup is below
threshold."
   exit
exit
```

### 4G to 5G HO Success

Use the following configuration to configure alerts related to the 4G to 5G HO Success procedure:

```
alerts rules group Handover
   interval-seconds 300
   rule 4gTo5gHOSuccess
   expression
"sum(smf_service_stats{procedure_type=~\"n26_4g_to_5g_handover|n26_4g_to_5g_im_mobility\"
, status=\"success\"}) /
sum(smf_service_stats{procedure_type=~\"n26_4g_to_5g_handover|n26_4g_to_5g_im_mobility\" ,
 status=\"attempted\"}) < 0.95 "
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage successful 4G to 5G HO is below
threshold."
   exit
exit
```

### 4G To WiFi HO Success

Use the following configuration to configure alerts related to the 4G to WiFi HO Success procedure:

```
alerts rules group Handover
   interval-seconds 300
   rule 4GtoWifiHOSuccess
   expression "sum(smf_service_stats{procedure_type=\"enb_to_untrusted_wifi_handover\"
, status=\"success\"}) /
sum(smf_service_stats{procedure_type=\"enb_to_untrusted_wifi_handover\" ,
status=\"attempted\"}) < 0.95 "
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of N4 responses sent is lesser than 95
 %."
   exit
exit
```

### 5G N2 HO Success

Use the following configuration to configure alerts related to the 5G N2 HO Success procedure:

```
alerts rules group Handover
   interval-seconds 300
   rule N2HOSuccess
   expression "sum(smf_service_stats{procedure_type=\"n2_handover\" , status=\"success\"})
 / sum(smf_service_stats{procedure_type=\"n2_handover\" , status=\"attempted\"}) < 0.95 "
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage successful 5G N2 HO is below threshold."

   exit
exit
```

### 5G PDU Idle Success

Use the following configuration to configure alerts related to the 5G PDU Idle Success procedure:

```
alerts rules group SMFPDU
   interval-seconds 300
   rule SMFPDUIdleSuccess
   expression "sum(smf_service_stats{procedure_type=~\".*idle\" , status=\"success\"})
/ sum(smf_service_stats{procedure_type=~\".*idle\" , status=\"attempted\"}) < 0.95 "
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage successful 5G PDU Idle is below threshold"

   exit
exit
```

### 5G PDU Modify Success

Use the following configuration to configure alerts related to the 5G PDU Modify Success procedure:

```
alerts rules group SMFPDU
   interval-seconds 300
   rule SMFSessionModifySuccess
   expression "sum(smf_service_stats{procedure_type=~\".*pdu_sess_mod\" ,
status=\"success\"}) / sum(smf_service_stats{procedure_type=~\".*pdu_sess_mod\" ,
status=\"attempted\"}) < 0.95 "
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage successful 5G PDU Modify is below
threshold"
   exit
exit
```

### 5G PDU Release Success

Use the following configuration to configure alerts related to the 5G PDU Release Success procedure.

```
alerts rules group SMFPDU
   interval-seconds 300
   rule SMFSessionReleaseFailure
   expression "sum(smf_service_stats{procedure_type=~\".*pdu_sess_rel\" ,
status=\"success\"}) / sum(smf_service_stats{procedure_type=~\".*pdu_sess_rel\" ,
status=\"attempted\"}) < 0.95 "
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage successful 5G PDU Setup is below
threshold"
   exit
exit
```

### 5G PDU Setup Success

Use the following configuration to configure alerts related to the 5G PDU Setup Success procedure:

```
alerts rules group SMFPDU
   interval-seconds 300
   rule SMFSessionSetupFailure
   expression "sum(smf_service_stats{procedure_type=\"pdu_sess_create\" ,
status=\"success\"}) / sum(smf_service_stats{procedure_type=\"pdu_sess_create\" ,
status=\"attempted\"} < 0.95 "
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when failed to setup sessions is more than 5%"
   exit
exit
```

### 5G to 4G HO Success

Use the following configuration to configure alerts related to the 5G to 4G HO Success procedure:

```
alerts rules group Handover
   interval-seconds 300
   rule 5gTo4gHOSuccess
   expression "sum(smf_service_stats{procedure_type=~\"pdn_5g_4g_handover
|pdn_5g_4g_handover_dft|eps_fb_5g_4g_handover_dft|eps_fb_5g_4g_handover_idft
|pdn_5g_4g_handover_idft\" , status=\"success\"}) /
sum(smf_service_stats{procedure_type=~\"pdn_5g_4g_handover
|pdn_5g_4g_handover_dft|eps_fb_5g_4g_handover_dft|
eps_fb_5g_4g_handover_idft|pdn_5g_4g_handover_idft\" , status=\"attempted\"}) < 0.95 "
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage successful 5G to 4G HO is below
threshold."
   exit
exit
```

### 5G To WiFi HO Success

Use the following sample configuration to configure alerts related to the 5G to WiFi HO Success procedure:

```
alerts rules group Handover
  interval-seconds 300
  rule 5GtoWifiHOSuccess
  expression "sum(smf_service_stats{procedure_type=\"nr_to_untrusted_wifi_handover\" ,
 status=\"success\"}) / sum(smf_service_stats{procedure_type=\"nr_to_untrusted_wifi_handover\"
 , status=\"attempted\"}) < 0.95 "
  severity major
  type "Communications Alarm"
  annotation summary
  value "This alert is fired when the percentage of N4 responses sent is lesser than 95
 %."
  exit
exit
```

### 5G Xn HO Success

Use the following sample configuration to configure alerts related to the 5G Xn HO Success procedure:

```
alerts rules group Handover
  interval-seconds 300
  rule XnHOSuccess
  expression "sum(smf_service_stats{procedure_type=\"xn_handover\" , status=\"success\"})
 / sum(smf_service_stats{procedure_type=\"xn_handover\" , status=\"attempted\"}) < 0.95 "
  severity major
  type "Communications Alarm"
  annotation summary
  value "This alert is fired when the percentage successful 5G Xn HO is below threshold."

  exit
exit
```

### PDN Session Create

Use the following sample configuration to configure alerts related to the PDN Session Create procedure.

```
alerts rules group SMFProcStatus
  interval-seconds 300
  rule PDNSessCreate
  expression "sum(increase(smf_service_stats{app_name=\"SMF\",procedure_type=
/\"pdn_sess_create\",status=\"success\"}[5m])) /
sum(increase(smf_service_stats{app_name=\"SMF\
/",procedure_type=\"pdn_sess_create\",status=\"  /attempted\"}[5m])) < 0.95"
  severity major
  type "Communications Alarm"
  annotation summary
  value "This alert is fired when the success percentage of pdn_sess_create procedure is
 lesser threshold."
  exit
exit
```

### PDU Session Create

Use the following sample configuration to configure alerts related to the PDU Session Create procedure.

```
alerts rules group SMFProcStatus
  interval-seconds 300
```

```
   rule PDUSessCreate
   expression "sum(increase(smf_service_stats{app_name=\"SMF\",procedure_type=
/\"pdu_sess_create\",status=\"success\"}[5m]))sum
/(increase(smf_service_stats{app_name=\"SMF\",  /procedure_type=\"pdu_sess_create\",status=\
  /"attempted\"}[5m])) < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the success percentage of pdu_sess_create procedure is
 lesser threshold."
   exit
exit
```

### PDU Session Modify

Use the following sample configuration to configure alerts related to the PDU Session Modify procedure.

```
alerts rules group SMFProcStatus
   interval-seconds 300
   rule PDUSessModify
   expression "sum(increase(smf_service_stats{app_name=\"SMF\",procedure_type=~\".
/*req_pdu_sess_mod\",status=\"success\"}[5m]))sum(increase
/(smf_service_stats{app_name=\"SMF\",procedure_type=~
/\".*req_pdu_sess_mod\",status=\"attempted\"}[5m]))  / < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the success percentage of req_pdu_sess_mod procedure
is lesser threshold."
   exit
exit
```

### PDU Session Release

Use the following sample configuration to configure alerts related to the PDU Session Release procedure:

```
alerts rules group SMFProcStatus
   interval-seconds 300
   rule PDUSessRelease
   expression
"sum(increase(smf_service_stats{app_name=\"SMF\",procedure_type=~\".*req_pdu_sess_rel\",status=\
  /"success\"}[5m]))sum(increase(smf_service_stats{app_name=\"SMF
/\",procedure_type=~\".*req_pdu_sess_rel\",status=\  /"attempted\"}[5m])) < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the success percentage of req_pdu_sess_rel procedure
is lesser threshold."
   exit
exit
```

## Interface Specific Alerts

This section provides detail of commands that are required to configure alerts related to various interfaces.

### GTPC Peer Down

Use the following commands to configure alerts related to the GTPC Peer Down procedure.

```
alerts rules group GTPCPeerDown
   interval-seconds 300
   rule GTPCPeerDown
   expression nodemgr_gtpc_peer_status{gtpc_peer_status=\"gtpc_peer_path_down\"})
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the GTPC Path failure detected for peer crosses
threshold"
   exit
exit
```

### N4 Message Success

Use the following commands to configure alerts related to the N4 Message Success procedure.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN4MessageSuccess
   expression "sum(protocol_udp_res_msg_total{message_direction=\"inbound\",
status=\"accepted\"}) / sum(protocol_udp_res_msg_total{message_direction=\"inbound\",
status=~\"accepted|denied\"}) < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of N4 responses sent is lesser than 95
 %."
   exit
exit
```

### N4 UPF Association Down

Use the following commands to configure alerts related to the N4 UPF Association Down query by N4 address.

```
alerts rules group N4Association
   interval-seconds 300
   rule SMFAssociationRelease
   expression "proto_udp_res_msg_total{procedure_type=\"n4_association_release_res\",
message_direction= \"inbound\", status=\"accepted\"}) "
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the N4 Association with UPF is released"
   exit
exit
```

### N4 UPF Association Up

Use the following commands to configure alerts related to the N4 UPF Association Up query by N4 address.

```
alerts rules group N4Association
   interval-seconds 300
```

```
   rule N4AssociationUP
   expression ”proto_udp_res_msg_total{procedure_type=\“n4_association_setup_res\”,
message_direction= \“inbound\”, status=\“accepted\”}”
   severity major
   type "Communications Alarm"
   annotation summary
   value “This alert is fired when the N4 Association with UPF is established”
   exit
exit
```

### N7 Interface Outbound

Use the following commands to configure alerts related to an outbound N7 interface.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN7Outbound
   expression "sum(increase(smf_restep_http_msg_total{nf_type=\"pcf\",
message_direction=\"outbound\", response_status=~\”2..\”}[5m])) /
sum(increase(smf_restep_http_msg_total{nf_type=\"pcf\", message_direction=\"outbound\”}[5m]))
 < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of N7 responses received is lesser
threshold.”
   exit
exit
```

### N7 Interface Inbound

Use the following commands to configure alerts related to an inbound N7 interface.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN7Inbound
   expression "sum(increase(smf_restep_http_msg_total{nf_type=\"pcf\",
message_direction=\"inbound\", response_status=~\”2..\”}[5m])) /
sum(increase(smf_restep_http_msg_total{nf_type=\"pcf\", message_direction=\"inbound\”}[5m]))
 < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
  value "This alert is fired when the percentage of N7 responses sent is lesser threshold.”

   exit
exit
```

### N7 Message Timed Out

Use the following commands to configure alerts related to the N7 Message Timed Out procedure.

```
alerts rules group MessageTimeout
   interval-seconds 300
   rule SMFN7Timeout
   expression ”sum(irate(smf_restep_http_msg_total{nf_type=\“pcf\”,
```

```
message_direction=\"inbound\", response_status=\"504\"}[5m])) > 5"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the increase in timeout for N7 messages toward PCF
crosses threshold"
   exit
exit
```

### N10 Interface

Use the following commands to configure alerts related to the N10 interface.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN10
   expression "sum(increase(smf_restep_http_msg_total{nf_type=\"udm\",
message_direction=\"outbound\", response_status=~\"2..\"}[5m])) /
sum(increase(smf_restep_http_msg_total{nf_type=\"udm\", message_direction=\"outbound\"}[5m]))
 < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of N10 responses received is lesser
threshold."
   exit
exit
```

### N11 Interface Inbound

Use the following commands to configure alerts related to an inbound N11 interface.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN11Inbound
   expression "sum(increase(smf_restep_http_msg_total{nf_type=\"amf\",
message_direction=\"inbound\", response_status=~\"2..\"}[5m])) /
sum(increase(smf_restep_http_msg_total{nf_type=\"amf\", message_direction=\"inbound\"}[5m]))
 < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of N11 responses sent is lesser
threshold."
   exit
exit
```

### N11 Interface Outbound

Use the following commands to configure alerts related to an outbound N11 interface.

```
alerts rules group SMFSvcStatus
   interval-seconds 60
   rule SMFN11Outbound
   expression "sum(increase(smf_restep_http_msg_total{nf_type=\"amf\",
message_direction=\"outbound\", response_status=~\"2..\"}[5m])) /
sum(increase(smf_restep_http_msg_total{nf_type=\"amf\", message_direction=\"outbound\"}[5m]))
```

```
 < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of N11 responses received is lesser
threshold."
   exit
exit
```

### N11 Message Timed Out

Use the following commands to configure alerts related to the N11 Message Timed Out procedure.

```
alerts rules group MessageTimeout
   interval-seconds 300
   rule SMFN40Timeout
   expression "sum(irate(smf_restep_http_msg_total{nf_type=\"chf\",
message_direction=\"inbound\", response_status=\"504\"}[5m])) > 5"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the increase in timeout for N11 messages toward AMF
crosses threshold"
   exit
exit
```

### N40 Interface Inbound

Use the following commands to configure alerts related to an inbound N40 interface.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN40Inbound
   expression "sum(increase(smf_restep_http_msg_total{nf_type=\"chf\",
message_direction=\"inbound\", response_status=~\"2..\"}[5m])) /
sum(increase(smf_restep_http_msg_total{nf_type=\"chf\", message_direction=\"inbound\"}[5m]))
 < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of N40 responses sent is lesser
threshold."
   exit
exit
```

### N40 Interface Outbound

Use the following commands to configure alerts related to an outbound N40 interface.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN40Outbound
   expression "sum(increase(smf_restep_http_msg_total{nf_type=\"chf\",
message_direction=\"outbound\", response_status=~\"2..\"}[5m])) /
sum(increase(smf_restep_http_msg_total{nf_type=\"chf\", message_direction=\"outbound\"}[5m]))
 < 0.95"
   severity major
```

```
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of N40 responses received is lesser
threshold."
   exit
exit
```

### N40 Message Timed Out

Use the following commands to configure alerts related to the N40 Message Timed Out procedure.

```
alerts rules group MessageTimeout
   interval-seconds 300
   rule SMFN11Timeout
   expression "sum(irate(smf_restep_http_msg_total{nf_type=\"chf\",
message_direction=\"inbound\", response_status=\"504\"}[5m])) > 5"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired the increase in timeout for N40 messages toward CHF crosses
 threshold"
   exit
exit
```

### NRF Discovery

Use the following commands to configure alerts related to the NRF Discovery procedure.

```
alerts rules group NRF
   interval-seconds 300
   rule NRFDiscovery
   expression
"sum(nf_discover_messages_total{result=~\"success|failure\",svc_name=\"nnrf-disc\",
service_name=\"smf-rest-ep\"}) < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of N4 responses sent is lesser than 95
 %."
   exit
exit
```

### SMF Service Start

Use the following commands to configure alerts related to the SMF Service Start procedure.

```
alerts rules group SMFService
   interval-seconds 300
   rule SMFServiceStart
   expression "irate(outgoing_response_msg_total{msg_type=\"NrfNfmRegistration\"}[5m])"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when SMF-Service starts upon registration with NRF"
   exit
exit
```

# IP Pool

This section provides detail of commands that are required to configure alerts related to IP Pool.

### IP Pool Used

Use the following commands to configure alerts related to the IP Pool used procedure.

```
alerts rules group IPPool
   interval-seconds 300
   rule IPPool
   expression "sum(IPAM_address_allocations_current) > THRESHOLD"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage IP pool addresses used is above the
threshold"
   exit
exit
```

# Message Level Alerts

This section provides detail of commands that are required to configure alerts related to various messages.

### N11 SM Create

Use the following commands to configure alerts related to N11 SM Create.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN11Success
  expression "sum(increase(smf_restep_http_msg_total{api_name=\"amf_create_sm_context\",
 message_direction=\"inbound\", response_status=\"201\"}[5m])) /
sum(increase(smf_restep_http_msg_total{api_name=\"amf_create_sm_context\",
message_direction=\"inbound\"}[5m])) < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of amf_create_sm_context responses sent
 is lesser threshold."
   exit
exit
```

### N11 SM Update

Use the following commands to configure alerts related to N11 SM Update.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN11Update
  expression "sum(increase(smf_restep_http_msg_total{api_name=\"amf_update_sm_context\",
 message_direction=\"inbound\", response_status=\"200\"}[5m])) /
sum(increase(smf_restep_http_msg_total{api_name=\"amf_update_sm_context\",
message_direction=\"inbound\"}[5m])) < 0.95"
   severity major
   type "Communications Alarm"
```

```
   annotation summary
   value "This alert is fired when the percentage of amf_update_sm_context responses sent
 is lesser threshold."
   exit
exit
```

### N11 SM Release

Use the following commands to configure alerts related to N11 SM Release.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN11Release
   expression "sum(increase(smf_restep_http_msg_total{api_name=\"amf_release_sm_context\",
 message_direction=\"inbound\", response_status=\"204\"}[5m])) /
sum(increase(smf_restep_http_msg_total{api_name=\"amf_release_sm_context\",
message_direction=\"inbound\"}[5m])) < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of amf_release_sm_context responses sent
 is lesser threshold."
   exit
exit
```

### N1 N2 Message Transfer

Use the following commands to configure alerts related to N1 N2 Message Transfer.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN1N2Transfer
   expression "sum(increase(smf_restep_http_msg_total{api_name=\"amf_n1_n2_transfer\",
message_direction=\"outbound\", response_status=\"200\"}[5m])) /
sum(increase(smf_restep_http_msg_total{api_name=\"amf_n1_n2_transfer\",
message_direction=\"outbound\"}[5m])) < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of amf_n1_n2_transfer responses received
 is lesser threshold."
   exit
exit
```

### N11 EBI Assignment

Use the following commands to configure alerts related to N11 EBI Assignment.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN11EBI
   expression "sum(increase(smf_restep_http_msg_total{api_name=\"amf_assign_ebi\",
message_direction=\"outbound\", response_status=\"200\"}[5m])) /
sum(increase(smf_restep_http_msg_total{api_name=\"amf_assign_ebi\",
message_direction=\"outbound\"}[5m])) < 0.95"
   severity major
   type "Communications Alarm"
```

```
   annotation summary
   value "This alert is fired when the percentage of amf_assign_ebi responses received is
 lesser threshold."
   exit
exit
```

### N11 SM Status Notify

Use the following commands to configure alerts related to N11 SM Status Notify.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN11StatusNotify
   expression "sum(increase(smf_restep_http_msg_total{api_name=\"amf_status_notify\",
message_direction=\"outbound\", response_status=\"201\"}[5m])) /
sum(increase(smf_restep_http_msg_total{api_name=\"amf_status_notify\",
message_direction=\"outbound\"}[5m])) < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of amf_status_notify responses received
 is lesser threshold."
   exit
exit
```

### N11 SM Context Retrieve

Use the following commands to configure alerts related to N11 SM Context Retrieve.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN11ContextRetrieve
  expression "sum(increase(smf_restep_http_msg_total{api_name=\"amf_retrieve_sm_context\",
 message_direction=\"inbound\", response_status=\"201\"}[5m])) /
sum(increase(smf_restep_http_msg_total{api_name=\"amf_retrieve_sm_context\",
message_direction=\"inbound\"}[5m])) < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of amf_retrieve_sm_context responses
sent is lesser threshold."
   exit
exit
```

### N7 SM Policy Create

Use the following commands to configure alerts related to N7 SM Policy Create.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN7PolicyCreate
   expression
"sum(increase(smf_restep_http_msg_total{api_name=\"pcf_sm_policy_control_create\",
message_direction=\"outbound\", response_status=\"201\"}[5m])) /
sum(increase(smf_restep_http_msg_total{api_name=\"pcf_sm_policy_control_create\",
message_direction=\"outbound\"}[5m])) < 0.95"
   severity major
```

```
   type "Communications Alarm"
   annotation summary
  value "This alert is fired when the percentage of pcf_sm_policy_control_create responses
 received is lesser threshold."
   exit
exit
```

### N7 SM Policy Update

Use the following commands to configure alerts related to N7 SM Policy Update.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN7PolicyUpdate
   expression
"sum(increase(smf_restep_http_msg_total{api_name=\"pcf_sm_policy_control_update\",
message_direction=\"outbound\", response_status=\"200\"}[5m])) /
sum(increase(smf_restep_http_msg_total{api_name=\"pcf_sm_policy_control_update\",
message_direction=\"outbound\"}[5m])) < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
  value "This alert is fired when the percentage of pcf_sm_policy_control_update responses
 received is lesser threshold."
   exit
exit
```

### N7 SM Policy Delete

Use the following commands to configure alerts related to N7 SM Policy Delete.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN7PolicyDelete
   expression
"sum(increase(smf_restep_http_msg_total{api_name=\"pcf_sm_policy_control_delete\",
message_direction=\"outbound\", response_status=\"204\"}[5m])) /
sum(increase(smf_restep_http_msg_total{api_name=\"pcf_sm_policy_control_delete\",
message_direction=\"outbound\"}[5m])) < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
  value "This alert is fired when the percentage of pcf_sm_policy_control_delete responses
 received is lesser threshold."
   exit
exit
```

### N7 SM Policy Notify Update

Use the following commands to configure alerts related to N7 SM Policy Notify Update.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN7PolicyUpdateNotify
   expression
"sum(increase(smf_restep_http_msg_total{api_name=\"pcf_sm_policy_control_update_notify\",
message_direction=\"inbound\", response_status=\"201\"}[5m])) /
```

```
sum(increase(smf_restep_http_msg_total{api_name=\"pcf_sm_policy_control_update_notify\",
message_direction=\"inbound\”}[5m])) < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of pcf_sm_policy_control_update_notify
 responses sent is lesser threshold.”
   exit
exit
```

### N7 SM Policy Notify Terminate

Use the following commands to configure alerts related to N7 SM Policy Terminate.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN7PolicyTerminateNotify
   expression
"sum(increase(smf_restep_http_msg_total{api_name=\"pcf_sm_policy_control_terminate_notify\",
 message_direction=\"inbound\", response_status=\"201\”}[5m])) /
sum(increase(smf_restep_http_msg_total{api_name=\"pcf_sm_policy_control_terminate_notify\",
 message_direction=\"inbound\”}[5m])) < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
  value "This alert is fired when the percentage of pcf_sm_policy_control_terminate_notify
 responses sent is lesser threshold.”
   exit
exit
```

### N10 UE Register

Use the following commands to configure alerts related to N10 UE Register.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN10UERegister
   expression "sum(increase(smf_restep_http_msg_total{api_name=\"register_ue\",
message_direction=\"outbound\", response_status=\"201\”}[5m])) /
sum(increase(smf_restep_http_msg_total{api_name=\"register_ue\",
message_direction=\"outbound\”}[5m])) < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of register_ue responses received is
lesser threshold.”
   exit
exit
```

### N10 UE DeRegister

Use the following commands to configure alerts related to N10 UE DeRegister.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN10UEDeRegister
   expression "sum(increase(smf_restep_http_msg_total{api_name=\"deregister_ue\",
```

```
message_direction=\"outbound\", response_status=\"201\”}[5m])) /
sum(increase(smf_restep_http_msg_total{api_name=\"deregister_ue\",
message_direction=\"outbound\”}[5m])) < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of deregister_ue responses received is
 lesser threshold.”
   exit
exit
```

### N10 SM Subscription Fetch

Use the following commands to configure alerts related to N10 Subscription Fetch.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN10SubscriptionFetch
   expression "sum(increase(smf_restep_http_msg_total{api_name=\"subscription_req\",
message_direction=\"outbound\", response_status=\"200\”}[5m])) /
sum(increase(smf_restep_http_msg_total{api_name=\"subscription_req\",
message_direction=\"outbound\”}[5m])) < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of subscription_req responses received
 is lesser threshold.”
   exit
exit
```

### N10 SM Subscribe for Notification

Use the following commands to configure alerts related to N10 Subscribe for Notification.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN10SubscriptionNotification
  expression "sum(increase(smf_restep_http_msg_total{api_name=\"sdm_subscription_req\",
 message_direction=\"outbound\", response_status=\"201\”}[5m])) /
sum(increase(smf_restep_http_msg_total{api_name=\"sdm_subscription_req\",
message_direction=\"outbound\”}[5m])) < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
  value "This alert is fired when the percentage of sdm_subscription_req responses received
 is lesser threshold.”
   exit
exit
```

### N10 Charging Data Request

Use the following commands to configure alerts related to N10 Charging Data Request.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN10ChargingRequest
   expression
```

```
"sum(increase(smf_restep_http_msg_total{api_name=\"chf_charging_data_request\",
message_direction=\"outbound\", response_status=\"201\"}[5m])) /
sum(increase(smf_restep_http_msg_total{api_name=\"chf_charging_data_request\",
message_direction=\"outbound\"}[5m])) < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of chf_charging_data_request responses
 received is lesser threshold."
   exit
exit
```

### N10 Charging Data Notify

Use the following commands to configure alerts related to N10 Charging Data Notify.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFN10ChargingDataNotify
   expression "sum(increase(smf_restep_http_msg_total{api_name=\"chf_abort_notify\",
message_direction=\"inbound\", response_status=\"201\"}[5m])) /
sum(increase(smf_restep_http_msg_total{api_name=\"chf_abort_notify\",
message_direction=\"inbound\"}[5m])) < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of chf_abort_notify responses sent is
lesser threshold."
   exit
exit
```

## Policy Rule Alerts

This section provides detail of commands that are required to configure alerts related to various policy rules.

### Addition of Dynamic PCC Rules

Use the following commands to configure alerts related to addition of dynamic PCC rules.

```
alerts rules group SMFPolicyStatus
   interval-seconds 300
   rule AddPCCRule
   expression
"sum(increase(policy_dynamic_pcc_rules_total{app_name=\"SMF\",event=\"success\",operation=\"install\"}[5m]))
  /
sum(increase(policy_dynamic_pcc_rules_total{app_name=\"SMF\",event=\"attempted\",operation=\"install\"}[5m]))
 < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of successful addition of dynamic pcc
rules is lesser threshold."
   exit
exit
```

### Modification of Dynamic PCC Rules

Use the following commands to configure alerts related to modification of dynamic PCC rules.

```
alerts rules group SMFPolicyStatus
   interval-seconds 300
   rule ModifyPCCRule
   expression
"sum(increase(policy_dynamic_pcc_rules_total{app_name=\"SMF\",event=\"success\",operation=\"modify\"}[5m]))
  /
sum(increase(policy_dynamic_pcc_rules_total{app_name=\"SMF\",event=\"attempted\",operation=\"modify\"}[5m]))
 < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of successful modification of dynamic
pcc rules is lesser threshold."
   exit
exit
```

### Removal of Dynamic PCC Rules

Use the following commands to configure alerts related to removal of dynamic PCC rules.

```
alerts rules group SMFPolicyStatus
   interval-seconds 300
   rule RemovePCCRule
   expression
"sum(increase(policy_dynamic_pcc_rules_total{app_name=\"SMF\",event=\"success\",operation=\"remove\"}[5m]))
  /
sum(increase(policy_dynamic_pcc_rules_total{app_name=\"SMF\",event=\"attempted\",operation=\"remove\"}[5m]))
 < 0.95"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the percentage of successful removal of dynamic pcc
rules is lesser threshold."
   exit
exit
```

## SMF Overload/Congestion

This section provides detail of commands that are required to configure alerts related to various SMF Overload/Congestion.

### SMF Overload

Use the following commands to configure alerts related to the SMF Overload procedure.

```
alerts rules group SMFSvcStatus
   interval-seconds 300
   rule SMFOverload
   expression "sum by (component) (system_overload_status) == true"
   severity major
   type "Communications Alarm"
   annotation summary
```

```
  value "This alert is fired when increase in events not processed due to system overload"

  exit
exit
```

## SMF Sessions

This section provides detail of commands that are required to configure alerts related to various SMF sessions.

### Session Release Rate

Use the following commands to configure alerts related to the Session Release Rate procedure.

```
alerts rules group SMFSession
  interval-seconds 300
  rule SMFSessionReleaseRate
  expression "sum(rate(smf_service_stats{procedure_type=~\".*pdu_sess_rel|.*pdn_sess_rel\"
 , status=\"attempted\"}[5m])) >  THRESHOLD "
  severity major
  type "Communications Alarm"
  annotation summary
  value "This alert is fired when the session release rate exceeds the threshold"
  exit
exit
```

### Session Setup Failure

Use the following commands to configure alerts related to the Session Setup Failure procedure.

```
alerts rules group SMFSession
  interval-seconds 300
  rule SMFSessionSetupFailure
  expression "sum(smf_service_stats{procedure_type=~\"pdu_sess_create|pdn_sess_create\"
 , status=\"failures\"}) /
sum(smf_service_stats{procedure_type=\"pdu_sess_create|pdn_sess_create\" ,
status=\"attempted\"}) >  0.05 "
  severity major
  type "Communications Alarm"
  annotation summary
  value "This alert is fired when failed to setup sessions is more than 5%"
  exit
exit
```

### Session Setup Rate

Use the following commands to configure alerts related to the Session Setup Rate procedure.

```
alerts rules group SMFSession
  interval-seconds 300
  rule SMFSessionSetupRate
  expression
"sum(rate(smf_service_stats{procedure_type=~\"pdu_sess_create|pdn_sess_create\" ,
status=\"attempted\"}[5m]))>  THRESHOLD "
  severity major
  type "Communications Alarm"
  annotation summary
```

```
   value "This alert is fired when the session setup rate exceeds the threshold"
   exit
exit
```

### Subscriber Limit

Use the following commands to configure alerts related to the Subscriber Limit procedure.

```
alerts rules group SMFSession
   interval-seconds 300
   rule SMFSubscriberLimit
   expression "sum(smf_session_counters{pdu_type=~\"ipv4v6|ipv4|ipv6\"}) > THRESHOLD"
   severity major
   type "Communications Alarm"
   annotation summary
   value "This alert is fired when the max number of subscribers is more
   than the threshold"
   exit
exit
```

# Metrics

## Feature Description

You can a monitor wide range of application and system statistics, and key performance indicators (KPI) within the SMF infrastructure. KPIs are useful to gain insight into the overall health of the SMF environment. Statistics offer a simplified representation of the SMF configurations and utilization-specific data.

The SMF 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 SMF 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 SMF 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.

The following snapshot is a sample of the Grafana dashboard.

*Figure 1: Grafana Dashboard*




# 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 SMF 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/.

# Configuring Metrics Collection

The labels of each SMF metrics are classified into the following three categories:

- Production
- Debug
- Granular

All the SMF application metrics are controlled through the CLI command for performance optimization.

To collect the necessary SMF metrics and labels, use the following sample configuration:

```
config
  infra metrics verbose { service | protocol | load-balancer | application
 } [ level { debug | off | production | trace } | metrics metrics_name [
granular-labels label_name| level { debug | off | production | trace } |
pod pod_name | level { debug | off | production | trace }  ] ]
  end
```

**NOTES**:

- If the metrics verbosity is not configured, then the default verbosity level for pod type is as follows.
  - LoadBalancer = Production
  - Protocol = Trace
  - Service = Trace
  - Application = Debug
- The order of the level for verbose metrics is in the following priority order:
  - **metrics [ [metrics_name] level [production|debug|trace|off]**: [Priority 1]
  - **pod [[pod_Name]] level [ production | debug | trace | off]]** [Priority 2]
  - **level [production | debug | trace | off]** [Priority 3]
- **infra metrics verbose { service | protocol | load-balancer | application }**: Enable the metric collection. This configuration helps to collect the required application metrics and labels. By default, this command captures the debug labels of metrics.
- **level { debug | off | production | trace }**: Specify the application metrics category to capture the required application metrics and labels.
  - **debug**: Capture all the labels that are classified as production and debug categories. This option is the default configuration.
  - **off**: Disable the application level metrics collection.

    For example, configuring the **infra metrics verbose application smf_service_stats level off** command disables the smf_service_stats application metrics.
  - **production**: Capture the labels that are classified as production category.
  - **trace**: This option is not supported for SMF application metrics. If this option is configured, the SMF treats this option as **debug**.
- If production and debug classification is empty for a metrics, then all the labels except granular-labels (if configured) are classified as debug.
- **metrics** *metrics_name*: Specify the metrics name to capture only the labels that correspond to the given metrics. The metric-level configuration takes precedence over the application-level configuration. If the metrics level is not configured, the labels are captured at the application level.
- **granular-labels**: Capture only the granular labels. By default, this option is disabled.

  If a granular label is required for KPI, then that label must be configured. For example, to capture dnn labels of smf_service_stats metrics, you must configure the following CLI command:

  ```
  infra metrics verbose application metrics smf_service_stats level debug
   granular-labels [ dnn ]
  ```

## Configuration Example

The following is an example configuration to enable only production level for all the application metrics.

```
infra metrics verbose application level production
```

The following is an example configuration to enable production level for smf_service_stats application metrics and debug level for all other application metrics.

```
infra metrics verbose application smf_service_stats level production
```

The following is an example configuration to enable debug level for smf_service_stats application metrics along with granular labels and production level for all other application metrics.

```
infra metrics verbose application level production smf_service_stats level
 debug granular-labels [ dnn ]
```

The following is an example configuration to enable production level for smf_service_stats application metrics along with granular labels and debug level for all other application metrics.

```
infra metrics verbose application smf_service_stats level production
granular-labels [ dnn ]
```

The following is an example configuration to disable smf_service_stats application metrics and debug level for all other application metrics.

```
infra metrics verbose application smf_service_stats level off
```

The following is an example configuration to configure NSSAI labels of smf_service_stats metrics.

```
infra metrics verbose application metrics smf_service_stats level debug
granular-labels [ snssai ]
```



**Note** The NSSAI statistics are not pegged without configuring the NSSAI label in the granular-labels configuration.

## Configuration Verification

To verify the configuration, use the following show command:

```
show running-config infra metrics verbose application
```

The following are example outputs of the **show running-config infra metrics verbose application** command.

```
[smf] smf# show running-config infra metrics verbose application
infra metrics verbose application
 metrics smf_service_stats
  level production
  granular-labels [ dnn ]
 exit
exit
```

The preceding output indicates that the configuration to capture production labels for smf_service_stats application metrics along with granular labels and debug levels of all other application metrics is enabled.

```
[smf] smf# show running-config infra metrics verbose application
 infra metrics verbose application
  level production
  metrics smf_service_stats
   level debug
   granular-labels [ [dnn] ]
  exit
exit
```

The preceding output indicates that the configuration to capture debug labels for smf_service_stats application metrics along with granular labels and production level of all other application metrics is enabled.

To verify the slice information on procedure and session statistics, use the following show command:

```
show running-config infra metrics verbose application
infra metrics verbose application
 metrics smf_service_stats
  level debug
  granular-labels [ snssai ]
 exit
```

# Bulk Statistics and Key Performance Indicators

## Feature Description

This section provides details of bulk statistics, and Key Performance Indicators (KPIs) used for performance analysis on SMF.

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.
  - Reset - where a counter is manually reset to zero.



**Important** For the complete list of supported bulk statistics and KPIs, see the *UCC 5G SMF Metrics Reference* applicable for this release.

# Logs

# Feature Description

The system logging feature provides a common way to log the log messages across applications. Each log consists of the following components:

- Timestamp—Shows the date and time of the log creation.
- Log message—Shows the message of a specific log.
- Log level—Shows the level of importance of log message.
- Log tag—Shows the details of module name, component name, and interface name. A log tag is pre-created and passes during logging.

SMF provides various types of logging to log the messages. These logging types are application logging, transaction logging, monitor subscriber logging, and trace logging.

The SMF maintains various logs, such as trace logs and event logs. Use the **kubectl get pods -n** *namespace* CLI command to check all the pods and the services that are currently running. Then, use the **kubectl logs** *podname* **-n** *namespace* CLI command to display the log in a pod.

If you encounter any error during the operation of this feature, use the SMF service logs for a particular subscriber session to identify the issues and determine the solution to your problem.

## Download OAM and EDR Monitor Pod Files

### Feature Description

Files that are generated using the **monitor subscriber** command, **monitor protocol** command, and transaction logs are stored in the OAM pod. The files that are generated in OAM pod are collected and stored in an internal Apache server. You can view and download the files by using a web browser, after user authentication.

**Note** Use the same credentials as ops-center to authenticate user access to the files present in the oam-pod and edr-monitor pod using a browser.

The files are created in separate folders, as and when their respective commands are executed. You can download the following OAM and EDR pod files:

- **Monitor subscriber files**: These files are generated using the **monitor subscriber** CLI option to trace messages that are related to a specified subscriber. The files that are generated for the **monitor protocol** command are present in the `monsublogs/` directory.
- **Monitor protocol files**: These files are using the **monitor protocol** CLI option to capture packets on a specific interface provided under the CLI command. The files that are generated for the **monitor protocol** command are present in `monprologs/` directory.
- **Transaction logs**: When transaction logging is enabled, the transaction logs are sent to oam-pod and can be downloaded from there. The files generated for transaction logging when enabled and are present in the `transactionlogs/` directory.
- **EDR files**: These files are generated in smf service pod and periodically copied to edr-monitor pod. The files are available in /edr directory.

# How it Works

This section describes how to view and download the log files in the oam-pod and edr-monitor pod.

### Downloading OAM Pod Files

Open a browser and log on to the Apache server using the *https://oam-files.*<*ReleaseName*>*.*<*Ingress-host-name*>*.nip.io/* URL. Use the ops-center user credentials. Replace <*ReleaseName*> and <*Ingress-host-name*> with the release name and ingress host name respectively.

The oam-pod directory comprises folders to archive the monitor protocol logs, monitor subscriber logs, and transaction logs.

The directory folders are visible as per the commands executed.

To download the monitor protocol files, use the following URL:

*https://oam-files.<ReleaseName>.<Ingress-host-name>.nip.io/ monprologs/*

In the preceding URL, replace monprologs with monsublogs for monitor subscriber files and with transactionlogs for the transaction log files.

#### Downloading EDR Files

To access the EDR files in the persistent volume of EDR monitor pod, log on to the Ops center with required credentials, and use the edr-monitor pod ingress URL.

To determine the ingress URL, use the following command:

**kubectl get ingress -n** *namespace* **| grep edr**

**Example:**

*cloud-user@svi-cndp-tb41-gr-setup-smf-cluster-2-cndp-server-1:~$* **kubectl get ingress -n smf-smf | grep edr**

# Configuring the Logs

This section describes how to configure the logs.

## Enabling or Disabling the Transaction Messages

To enable or disable the presence of request response messages in the transaction logs, use the following sample configuration:

```
config
  logging transaction message { disable | enable }
  commit
end
```

**NOTES**:

- **logging transaction message { disable | enable }**: Specify whether to enable or disable messages in transaction logging.

### Viewing Transaction History Logs

To view the transaction history on an OAM pod shell, use the following CLI command in the SMF Ops Center:

```
dump transactionhistory
```

**Note** The most recent transaction logs are stored in a circular queue of size 1024 transaction logs.

To display the logs in a pod, use the following command on the Kubernetes master node:

**kubectl logs -n** *<SMF namespace> podname*

### Sample Transaction Log

The following is an example of transaction log collected in Monitor Subscriber during SMF PDU session establishment.

```
Transaction Log received from Instance: smf.smf-rest-ep.unknown.smf.0
************* TRANSACTION: 00010 *************
TRANSACTION SUCCESS:
    Txn Type             : N10RegistrationRequest(33)
    Priority             : 1
    Session State        : No_Session
LOG MESSAGES:
    2020/03/03 05:31:39.345 [DEBUG] [infra.transaction.core] Processing transaction Id: 10
 Type: 33 SubscriberID: imsi-123456789012345 Keys: []
    2020/03/03 05:31:39.345 [DEBUG] [infra.transaction.core] Trace is disabled
    2020/03/03 05:31:39.346 [TRACE] [infra.message_log.core] >>>>>>>
IPC message
Name: N10RegistrationRequest
MessageType: N10RegistrationRequest
Key:
--body--
{"regInfo":{"ueId":"imsi-123456789012345","pduSessionId":5},"regReq":{"dnn":"intershat",

"pduSessionId":5,"pgwFqdn":"cisco.com.apn.epc.mnc456.mcc123","plmnId":{"mcc":"123","mnc":"456"},


"smfInstanceId":"c388eec5-e2ff-4bda-8154-b5dd9f10ad97","supportedFeatures":"0","singleNssai":{"sd":"Abf123","sst":2}},


"msgReq":{"Type":2,"ServiceName":4,"Versions":["v1"],"ProfileName":"UP1","FailureProfile":"FH1","SvcMsgType":3,

  "Filter":{"Bitmapfeilds":2,"Dnn":"intershat"}}}
   2020/03/03 05:31:39.346 [DEBUG] [nrfClient.Discovery.nrf] Message send Metadata [Type:UDM
 ServiceName:nudm-uecm
      ..
      ..
Request
Name: UdmRegistrationRequest
Host:
http://209.165.200.229:9020/nudm-uecm/v1/imsi-123456789012345/registrations/smf-registrations/5
Method: PUT
RequestURI:
--- Headers ---
Content-Type: application/json
Body:{"dnn":"intershat","pduSessionId":5,"pgwFqdn":"cisco.com.apn.epc.mnc456.mcc123",
"plmnId":{"mcc":"123","mnc":"456"},
"singleNssai":{"sd":"Abf123","sst":2},"smfInstanceId":"c388eec5-e2ff-4bda-8154-b5dd9f10ad97","supportedFeatures":"0"}

    2020/03/03 05:31:39.376 [TRACE] [infra.message_log.core] >>>>>>>
Response
Name:
Response Status 201
--- Headers ---
Location:
http://209.165.200.229:9020/nudm-uecm/v1/imsi-123456789012345/registrations/smf-registrations/5
Content-Length: 225
Content-Type: application/json
Body:{"pgwFqdn": "cisco.com.apn.epc.mnc456.mcc123", "plmnId": {"mcc": "123", "mnc": "456"},
 "dnn": "intershat",
  "smfInstanceId": "524f5f8a-b584-47b8-86f5-a5292eabcdef", "pduSessionId": 5, "singleNssai":
 {"sd": "Abf123", "sst": 2}}
    ..
    ..
    ..
--body--
{"regRes":{"dnn":"intershat","pduSessionId":5,"pgwFqdn":"cisco.com.apn.epc.mnc456.mcc123",
"plmnId":{"mcc":"123","mnc":"456"},
  ..
  ..
```

```
*********************************************
Transaction Log received from Instance: smf.smf-rest-ep.unknown.smf.0
************* TRANSACTION: 00011 *************
TRANSACTION SUCCESS:
    Txn Type             : N10SubscriptionFetchReq(36)
    Priority             : 1
    Session State        : No_Session
LOG MESSAGES:
    2020/03/03 05:31:39.384 [DEBUG] [infra.transaction.core] Processing transaction Id: 11
 Type: 36 SubscriberID: imsi-123456789012345 Keys: []
    2020/03/03 05:31:39.384 [DEBUG] [infra.transaction.core] Trace is disabled
    2020/03/03 05:31:39.384 [TRACE] [infra.message_log.core] >>>>>>>
IPC message
Name: N10SubscriptionFetchReq
MessageType: N10SubscriptionFetchReq
Key:
--body--
  ..
  ..
Request
Name: UdmSubscriptionRequest
Host:
http://209.165.200.229:9020/nudm-sdm/v1/imsi-123456789012345/sm-data?dnn=intershat&plmn-id=%7B%22mcc%22%3A%22123%22%2C%22mnc%22%3A%2

   2456%22%7D&single-nssai=%7B%22sd%22%3A%22Abf123%22%2C%22sst%22%3A2%7D&supported-features=0
Method: GET
RequestURI:
--- Headers ---
IPC message
Name: N10SubscriptionFetchSuccess
MessageType: N10SubscriptionFetchSuccess
Key:
  ..
  ..
--body--
  ..
  ..
```

## Configuring the Logging Levels

This section describes how to configure the logging level parameters.

Use the following sample configuration to configure the logging level:

```
config
  logging level { application | monitor-subscriber | tracing |transaction
 }
  end
```

**NOTES:**

- **logging level { application | monitor-subscriber | tracing | transaction }**– Enter the transaction log configuration mode.
  - **application** – Configures the option application logging level.
  - **monitor-subscriber** – Configures the option monitor subscriber logging level.
  - **tracing** – Configures the option logging level tracing
  - **transaction** – Configures the option transaction logging level.

# Configuring Persistent Transaction Logs

This section describes how to configure the persistent transaction log parameters.

The transaction logs are saved in the transaction log file that resides in the transaction logs directory of OAM pod.

Use the following sample configuration to configure the persistent transaction logs:

```
config
  logging transaction persist enable { max-file-size | max-rotation }
  end
```

**NOTES:**

- **logging transaction**– Enter the transaction log configuration mode.
- **persist enable { max-file-size | max-rotation }** – Configure the option to enable writing of transaction logs to the transaction log file.
  - **max-file-size** *max_filesize*– Specify the maximum size (in MB) of the transaction logs that must be preserved in the file. The default size is 50 MB. The accepted range is 1-10000 MB.
  - **max-rotation** *max_rotation*– Specify the maximum number of files that must be stored in the folder. After reaching the specified number, the file rotation begins. With this rotation, the oldest file is deleted and the latest log file is added to the folder. For example, if the folder has files a1.txt–a.10.txt and when the a.11.txt is added, then a1.txt is deleted. The default number is 10. The accepted range is 2 -1000.
- **persist enable** – Disables writing of transaction logs to the transaction log file.

## Viewing Persistent Transaction Logs

This section describes how to view the transaction logs that are stored on the OAM pod.

To view the persistent transaction logs, use the following configuration through the SMF Ops Center:

**transaction file dump filename** *file_path*

You can use the **transaction log list** command to view the list of log files and their paths.

The following is a sample output of the transaction logs:

```
RELEASE_NAMESPACE: 'example-data'
Dumping file 'transactionlogs/transaction.log.20200907033433.4.gz'
InstanceInfo: example.example-rest-ep.cluster1.example-data.1
TimeStamp: 2020-09-09 00:25:18.379439773 +0000 UTC
************* TRANSACTION: 01371 *************
TRANSACTION SUCCESS:
    Txn Type             : MessageTypeExampleCreate(1)
    Priority             : 1
    Session Namespace    : none(0)
LOG MESSAGES:
    2020/09/09 00:25:18.339 [INFO] [rest_ep.app.n7] Message Example_Create decod
ed
    2020/09/09 00:25:18.339 [INFO] [rest_ep.app.n7] Process init
    2020/09/09 00:25:18.339 [DEBUG] [rest_ep.app.n7] Config from GetConfig is Ve
rsion: 783da2fc038c6bc961a95e2bf3dd6d93f282e36b30e0362698a1de369a2fd15c Services
: [Name: restServer Type: Rest Endpoint: sbi Name: tcpServer Type: Tcp Endpoint:
 tcp-protocol Name: udpServer Type: Udp Endpoint: udp-protocol]
    2020/09/09 00:25:18.339 [INFO] [rest_ep.app.n7] Process continue
```

```
    2020/09/09 00:25:18.339 [DEBUG] [rest_ep.app.n7] DerivedConfig from GetConfi
g is DerivedNameToBeTested_cb3383b95927a434d42cd9d5687ccf1b13e2de4b2faf4543287a3
4afb32518fe
    2020/09/09 00:25:18.339 [DEBUG] [rest_ep.udp.n5] Sending message Example_Cre
ate to example-service
    2020/09/09 00:25:18.342 [INFO] [infra.transaction.core] Calling RPC example-
service_ipc_stream on host example-service_1 proc-name example-service_ipc_strea
```

## Enabling or Disabling JSON Logging

To enable or disable the JSON logging in pod logs, use the following sample configuration:

```
config
  [no] logging json-logging [ application | monitor-subscriber |
transaction ]
end
```

**NOTES**:

- **no**: Disables the JSON logging in the pod logs.
- **logging json-logging [ application | monitor-subscriber | transaction ]**: Enables the SMF-based application pods to start logging in the JSON format for the **application**, **monitor-subscriber**, and **transaction** log types.

# Logs for Event Failures

***Table 34: Feature History***

| Feature Name | Release Information | Description |
|---|---|---|
| Event Failure Logs for N2HO procedure | 2025.03.0 | With this feature, the consistent event failure logs are enhanced to support the N2HO procedure for the SMF and vSMF interfaces. |
| Event Failure Logs for Session Report procedure | 2025.01.0 | With this feature, the consistent event failure logs are enhanced to support the Session Report procedure for the SMF, PGW, hSMF, and vSMF interfaces. |
| Event Failure Logs for vSMF interface | 2024.04.0 | With this feature, the consistent event failure logs are enhanced to support the Create, Insert, Release, and Update procedures for the vSMF interface. |
| Event Failure Logs for Release procedure | 2024.03.0 | With this feature, the consistent event failure logs are enhanced to support the Release procedure for SMF, 4G, and hSMF interfaces.<br>**Default Setting:** Not Applicable |

| Feature Name | Release Information | Description |
|---|---|---|
| Event Failure Logs | 2024.02.0 | SMF provides the following support:<br>• Consistent event failure logs for Create, and Idle or Active procedures across pods<br>• Configurable logs at pod type<br>• Inclusion of request and response details in a single-line format<br><br>The significant volume of unnecessary system-generated logs resulted in increased memory consumption, performance impact, and ineffective management and utilization of logs. To prevent these issues, the consistent error log message format across various pods is introduced. The enhanced error logging for SMF procedures provides significant improvements, such as reduced memory consumption, minimized number of log generations by the system, detailed, consistent, and configurable logging that help in effective debugging and system monitoring.<br><br>**Default Setting:** Not Applicable |

The error logging capabilities for SMF procedures are enhanced for providing detailed, consistent, and configurable logging. These enhancements help in effective debugging and system monitoring. These enhancements are:

- Consistent log format—The single-line log format is standardized across different pods, such as REST endpoint and service pods, to ensure uniformity in how logs are recorded and interpreted.
- Enabling and disabling logging—An option to enable or disable logging is available at specific pod types. This option provides flexibility in managing log storage.
- Detailed log content—Logs include comprehensive details, such as primary key, interface, procedure details, message requests, and responses. This level of detail helps in thorough debugging and analysis.
- Log level management—By default, logs are written at the INFO level. You can enable the logs, as required. The log enablement helps in controlling the amount of log data generated and stored.

  By default, the log level is set to WARN, which ensures that logging is disabled by default. You can enable the logging, as required.

- Common logging interface—A common interface is implemented for event logging. All components use this interface for the same logging standards and formats.
- Log tags—Log tags are enhanced to allow you to enable or disable logs for specific pod types or services and provides granular control over logging.
- Supported logs—Logging is supported for various procedures, pods, and interfaces.

## How Event Failure Logging Works

To have the consistent log format across each pod, each component uses a common interface for event logging. A log uses the JSON format so that all the data appears in a single line. Logs are written at INFO level so that this level can be disabled by default and enabled, as required.

A log tag has the following format:

**transaction.event.<pod-type>**, where **<pod-type>** is the service name that a pod uses. For example, rest-ep and smf-service.

You can enable or disable logs for a specific pod type or service using log tags.

The logs are written when the corresponding log level is set to INFO, DEBUG, or TRACE. However, the message request or response fields are populated only when the log level is set to DEBUG or TRACE.

Example of a log:

### Sample Log


```
smf-service-1 [INFO] [Transaction.go:1576] [transaction.event.smf-service]
 {"TxnId":2,"StartTime":"2024-03-01T08:16:54Z","GRInstanceId":1,"TxnType":
"N7SmPolicyCreateReq","Priority": 33,"SessionNamespace":"smf(1)",
"CdlSliceName":"1","SubscriberId": "imsi-123456789012345", "SessionPrimaryKey":
"imsi-123456789012345:5","SessionKeys":"imsi-123456789012345:5 (pk)
roaming-status:homer (nuk) ue-type:nr-capable (nuk) supi:imsi-123456789012345 (nuk)
gpsi:msisdn-223310101010101 (nuk) pei:imei-123456786666660 (nuk) psid:5 (nuk) snssai:002abf123
 (nuk)
dnn:intershat (nuk) emergency:false (nuk) rat:e-utran (nuk) access:3gpp access (nuk)
connectivity:4g (nuk) udm-sdm:10.1.10.69 (nuk) auth-status:unauthenticated (nuk)
rulebase:starent (nuk)
policy:1(nuk)","SessionState":"Create_Session","ErrorMessage":"
{\"ErrType\":3,\"StatusCode\":500,
\"N1SmCause\":0,\"Detail\":\"\",\"InvalidParams\":null,\
"Title\":\"\",\"SourceDetails\":\"/opt
/workspace/smf-service/src/smf-service/procedures/4g/pdnsetup/idlestate.go:1318\"}",
"MessageRequest":{"smPolicyRequestPB": {"PolicyRequestPB":{"smPolicyCreateRequest":
{"smContextCreateData":{"accessType":1,"chargingcharacteristics":"1","dnn":
"intershat","gpsi":"msisdn-223310101010101","pduSessionId":5,"pduSessionType":1,
"pei":"imei-123456786666660","ratType":2,"servingNetwork":{"mcc":"123","mnc":
"456"},"sliceInfo": {"sd":"Abf123","sst":2},"subsDefQos":{"_5qi":5,"arp":
{"preemptCap":1,"preemptVuln":1,"priorityLevel":15},"priorityLevel":15},
"subsSessAmbr":{"downlink":"4096 Kbps","uplink":"4096 Kbps"},"supi":
"imsi-123456789012345","suppFeat":"407B","ueTimeZone":"+00:15+1",
"userLocationInfo":{"eutraLocation":{"ecgi":{"eutraCellId":"1234567","plmnId":
{"mcc":"123","mnc":"456"}},"tai":{"plmnId":{"mcc":"123","mnc":"456"},
"tac":"1820"},"ueLocationTimestamp":"2024-03-01T08:16:54Z"}},
"smPoliciesUpdateNotificationUrl":"http://10.1.8.54:8090/callbacks/v1/
smPoliciesUpdateNotification/imsi-123456789012345:5:1709281014408558981",
"recoveryTime":"2024-03-01T07:25:54Z","smfId":"94c78abd-ed68-49bb-b382-fd879504b702","servNfId":
{"anGwAddr":{"anGwIpv4Addr":"10.1.10.69"}}}}}, "commonApplMetaData":{"NFType":4,
"ApplData":{"smfApplData":{"Dnn":"intershat","ProcType":"PDN Connect [LTE]",
"RatType":"EUTRA","reason":1,"SessType":"ipv4","PduSessionId":"imsi-123456789012345:5"}}}},
```

```
"msgReq": "CAcYEkoCdjFiA1BQMWoNCAISCWludGVyc2hhdLoBA0ZIMcABCIgCApoCBAgCEAGaAgQIARACsAIg"},

"MessageResponse":{"PolicyResponsePB":null,"locationHeader": "http://10.1.10.69:9013
/npcf-smpolicycontrol/v1/sm-policies/imsi-123456789012345_5", "msgRsp":
"CAcYlAMwEmIDUFAxalFodHRwOi8vMTAuMS4xMC42OTo5MDEz L25wY2Ytc21wb2xpY3ljb250cm9s
L3YxL3NtLXBvbGljaWVzL2ltc2ktMTIzNDU2Nzg5MDEyMzQ1XzVwgxq
AAQKiAQNGSDGqARJQQ0YtZG5uPWludGVyc2hhdDs="}}
```


## Supported Log Levels

SMF supports the following log levels:

- Debug—This level is used to get detailed messages about the application operations and events, such as variable values, function calls, and interactions between components. This information is useful during the development and debugging of an application.
- Info—This level is used for providing a high-level information on the progress of an application. These messages are less detailed than the debug messages and are used to confirm if an application is functioning as expected. This level is used to log application milestones, such as the start and completion of significant tasks, configuration settings, or the successful completion of operations.

## Supported Logs

The following table lists the procedures, pods, and interfaces for the supported logging on SMF.

***Table 35: Supported Logs***

| SMF Procedures with Supported Logging | Pod Involved in Logging | SMF Interfaces Associated with Pod and Procedure | P-GW Interfaces (4G/Wi-Fi) Associated with Pod and Procedure | hSMF Interface Associated with Pod and Procedure | vSMF Interface Associated with Pod and Procedure |
|---|---|---|---|---|---|
| Create | REST-EP | N10, N40, N7, N11 | N10, N40, N7 | — | N16, N40, N11 |
| | SERVICE | N10, N40, N7, N11, N4, RMGR | N10, N40, N7, N4, RMGR | — | N40, N11, N4, RMGR |
| | PROTOCOL | N4 | N4, GTP | — | N4 |
| Idle/Active | REST-EP | N11 | — | — | — |
| | SERVICE | N11, N4 | — | — | — |
| | PROTOCOL | N4 | — | — | — |
| N2HO | REST-EP | N11 | — | — | N11 |
| | SERVICE | N11, N4 | — | — | N11, N4 |
| | PROTOCOL | N4 | — | — | N4 |

| SMF Procedures with Supported Logging | Pod Involved in Logging | SMF Interfaces Associated with Pod and Procedure | P-GW Interfaces (4G/Wi-Fi) Associated with Pod and Procedure | hSMF Interface Associated with Pod and Procedure | vSMF Interface Associated with Pod and Procedure |
|---|---|---|---|---|---|
| Release | REST-EP | N10, N40, N7, N11 | N10, N40,N7, N16 | N10, N40, N7, N16 | N16, N40, N11 |
| | SERVICE | N10, N40, N7, N11, N4, RMGR | N10, N40,N7, N4, RMGR | N10, N40, N7, N16, N4, RMGR | N16, N40, N11, N4, RMGR |
| | PROTOCOL | N4 | N4, GTP | N4 | N4 |
| Session Report | REST-EP | N40 | N40 | N40 | N40 |
| | SERVICE | N40, N4 | N40, N4 | N40, N4 | N40, N4 |
| | PROTOCOL | N4 | N4 | N4 | N4 |
| Update | PROTOCOL | — | — | — | N4 |

## Enable Event Logging

Use the following procedure to enable or disable the event logs of the SMF service. The appropriate log level configuration using the CLI command allows you to control the amount and type of log data generated. Hence, this configuration helps in effective monitoring, troubleshooting, and performance management.

### Procedure

**Step 1** Specify the log level for event logging.

**logging name transaction.event.**<*podname*> **level application** *application_log_level*

The <pod-type> is the service name that a pod uses.

**Example:**

```
[smf] smf# config
[smf] smf(config)#  logging name transaction.event.smf-service level application [ debug
 | error | info | off | trace | warn ]
```

**Example:**

**Note**
You can enable or disable the log for the supported debug and info levels only.

**Step 2** [*Optional*] Use **show running-config logging** command to verify the configured application event logging level.

**Example:**

```
[[smf] smf# show running-config logging
Thu Jun  20 18:50:12.888 UTC+00:00
logging level tracing debug
```

```
logging name infra.config.core level application trace
logging name infra.config.core level transaction trace
logging name infra.config.core level tracing off
logging name infra.message_log.core level transaction trace
logging name transaction.event.smf-service level application debug
```

# Dispatcher

## Feature Description

Dispatcher is an ingress throttling configuration of rate of admission with number of queues and size of queue configuration at GTP and PFCP.

**Configuration Limit for Dispatcher on S11 Interface**

When impacted, an IMS bearer results in voice call failures due to S11 interface and GTPv2 message drops at the ingress protocol dispatcher.

The S11 protocol dispatcher configuration at cnSGW is static and requires tuning based on the production requirements. If the configuration isn't appropriate for the MME pool that is being served, then in the case of high call events per seconds (CEPS), the GTPv2 message drops.

Any GTPv2 -S11 is discarded or dropped due to dispatcher queue limits at the ingress, which includes request or response messages, such as Release Access Bearer (RAB), Modify Bearer Request (MBR), and DDN Acknowledgment. For example, a drop in MBR from MME with an existing voice bearer results in a call drop. A drop of MBR from MME with a voice bearer transfer as part of EPS Fallback (EPSFB) results in an EPSFB call setup failure (Voice Access Failure Rate (AFR)).

## Configuring Dispatcher for S11 Interface

To configure a dispatcher for S11 interface, use the following sample configuration:

```
config
   interface interface_name
      sla response response_time_value
      dispatcher
         count count_value
         capacity capacity_value
         outbound [ true | false ]
         cache [ true | false ]
         threshold threshold_value
         expiry expiry_time_value
         nonresponsive nonresponsive_time_value
     exit
```

**NOTES:**

- **interface** *interface_name*: Specify the interface name as S11.

- **sla response** *response_time_value*: Specify the SLA response time. *response_time_value* must be an integer.
- **count** *count_value*: Specify the count of dispatcher queues.
- **capacity** *capacity_value*: Specify the capacity of each dispatcher queue.
- **outbound [ true | false ]**: Specify this value to enable or disable the queue support for the outbound messages.
- **cache [ true | false ]**: Specify this value to enable or disable the retransmission of cache.
- **threshold** *threshold_value*: Specify the threshold value for the outstanding requests per queue cache.
- **expiry** *expiry_time_value*: Specify the responded cache entry expiry duration in milliseconds.
- **nonresponsive** *nonresponsive_time_value*: Specify this value for the non-responsive cache entry expiry duration in milliseconds.

## Configuration Example

The following is an example configuration of dispatcher for S11 interface.

```
interface s11
   sla response 5500
   dispatcher
    count         10
    capacity      100000
    outbound      true
    threshold     160000
    expiry        7000
    nonresponsive 6000
   exit
```

# OAM Support

This section details the metrics supported for the dispatcher.

## Metrics

Following new metrics is introduced to capture the specific event when the dispatcher queue is configured:

**Metric Name:** outgoing_drop_queued_total

**Description:** The metric captures the specific event when the dispatcher queue configured at the UDP client is full and is loaded with too many messages beyond the configured queue capacity

Following labels are added in this metric:

- **local_address:** Local UDP address where the UDP endpoint is started.
- **interface:** Name of the UDP interface configured in the endpoint configuration.
- **protocol:** Name of the protocol configured in the endpoint configuration. For example, UDP
- **queue_id:** Queue index number based on the source IP address.
- **peer_address:** IP address of the peer endpoint.

- **cause:** The reason for the drop of message from the queue. For example, one of the reasons is dispatcher queue being full.
- **queue_type:** Type of queue configured in the dispatcher configuration.