# UDM Integration



# Feature Summary and Revision History

## Summary Data

***Table 1: Summary Data***

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

## Revision History

***Table 2: Revision History***

| Revision Details | Release |
|---|---|
| Added the N10 fail open support. | 2023.04.0 |

| Revision Details | Release |
|---|---|
| First introduced. | 2020.02.2 |

# Feature Description

The Unified Data Management (UDM) is responsible for primarily storing the subscriber data, which SMF accesses for managing the user sessions on the network. The SMF explicitly subscribes to receive the notifications about the events that occur in the subscriber data such session terminate.

The N10 interface is between Unified Data Management (UDM) and SMF (Session Management Function). The UDM provides the following services to SMF via the Nudm interface:

- Nudm_SubscriberDataManagement Service
- Nudm_UEContextManagement Service

# How it Works

This section describes how this feature works.

When the SMF skips UDM subscription, then it stops sending the following messages:

- Fetch-Subscription during session establishment
- Subscribe-for-Notification during session establishment
- Unsubscribe-to-Notification during session release and when the UDCM receives the UECM messages

The SMF allows any dynamic changes to the UDM subscription skip configuration. That is, new value is applicable for the new session being established. The existing sessions continue to use the old values.

# Configuring UDM

This section provides all the configurations related to UDM:

## Configuring Options for Controlling SDM Messages

This section describes how to configure controlling Subscription Data Management (SDM) messages over the N10 interface.

### Configuring RAT Type

To configure the RAT type with the local authorization under the DNN profile, use the following sample configuration:

```
config
   profile dnn dnn_profile
```

```
authorization local rat-type [ nr | eutra | wlan ]
end
```

**NOTES:**

- **authorization local**: This command skips the SDM messages for EPS sessions only. Upon configuring this command under the selected DNN profile, the SMF skips the UDM interaction for fetch subscription. The SMF uses the values received in the Create Session Request message. The SMF skips the UDM interaction to receive 'Subscribe-for-Notifications' from the UDM.
- **rat-type [ nr | eutra | wlan ]**: This keyword skips the following SDM messages based on the specified RAT type.
  - udm subscription-fetch
  - subscribe-to-notifications
  - unsubscribe-to-notifications

  Upon configuring the RAT type with **authorization local** command in the selected DNN profile, then for sessions on that RAT-type, the SMF skips the UDM interaction for the following messages:
  - udm subscription-fetch during session establishment
  - subscribe-for-notifications during session establishment
  - unsubscribe-for-notifications during session release
- **no authorization local rat-type [ nr | eutra | wlan ]**: Disables the local authorization under the DNN profile.

### Configuration Verification

To verify the configuration, use the **show running-config profile dnn** *dnn_profile_name* command.

The output of this show command displays all the configurations including the RAT type information that is configured within the specified DNN profile.

```
[smf] smf# show running-config profile dnn intershat
profile dnn intershat
network-element-profiles chf chf1
network-element-profiles amf amf1
network-element-profiles pcf pcf1
network-element-profiles udm udm1
charging-profile chgprf1
virtual-mac b6:6d:47:47:47:47
ssc-mode 2 allowed [ 3 ]
session type IPV6 allowed [ IPV6 ]
authorization local rat-type nr
upf apn intershat
dcnr true
exit
```

## Configuring Session Type

The SMF uses both subscription type data from UDM response and the session type configuration in DNN profile to allow or reject the call. The SMF selects the session type based on the initial look up of UE-requested PDN type in the UDM subscription data. Then, the SMF provisions session type for the session based on the selected session type and the session type configured in the DNN profile.

To configure the PDU session type in DNN profile, use the following sample configuration.

```
config
   profile dnn dnnprofile
      session type { IPV4 | IPV4V6 | IPV6 } allowed [ IPV4 | IPV4V6 |
IPV6 ]
      end
```

**NOTES**:

- **session type { IPV4 | IPV4V6 | IPV6 } allowed [ IPV4 | IPV4V6 | IPV6 ]**: Specify the IP type for the PDU session. The **allowed** keyword allows you to specify two IP types other than the default session type.
- The SMF uses this session type configuration to process the call. For example, if the UE requested type is IPv4 and the UDM subscription type is IPv4v6, the SMF selects IPv4 in the first pass and subsequently checks against the session type configuration. If the configured session type is IPv6, then the SMF rejects the call with a cause "#51 - PDU session type IPv6 only= IPV4 allowed".
- If the IPAM configuration includes the IP address pool that is different from the finally selected PDU session type, the SMF rejects the call with a cause "#31 - request rejected, unspecified". For example, this cause value will be generated under the following conditions:
  - UeReq-PdnType = V4
  - UdmSubscription-PdnType = V4V6
  - SessionType-Config = V4V6
  - IP-Pool = V6

### Configuration Verification

To verify the configuration, use the **show running-config profile dnn** *dnn_profile_name* command.

The output of this show command displays all the configurations including the session type information that is configured within the specified DNN profile.

```
[smf] smf# show running-config profile dnn intershat
profile dnn intershat
network-element-profiles chf chf1
network-element-profiles amf amf1
network-element-profiles pcf pcf1
network-element-profiles udm udm1
charging-profile chgprf1
virtual-mac b6:6d:47:47:47:47
ssc-mode 2 allowed [ 3 ]
session type IPV6 allowed [ IPV6 ]
upf apn intershat
dcnr true
exit
```

# Configuration to Disable Optional IEs

To disable optional IEs such as **epdgInd** and **registrationTime** attributes in the N10 Registration Request, use the following sample configuration:

```
config
  profile message-handling nf-type nf_ype
     mh-profile mh_profile_name
     service name type { nudm-ee | nudm-pp | nudm-sdm | nudm-ueau |
nudm-uecm }
     message type { UdmRegistrationReq | UdmSdmGetUESMSubscriptionData |
 UdmSdmSubscribeToNotification | UdmSdmUnsubscribeToNotification |
UdmSubscriptionReq | UdmUecmRegisterSMF skip optional-ies [ epdgInd |
registrationTime ] | UdmUecmUnregisterSMF }
    exit
  exit
exit
```

**NOTES**:

- **mh-profile** *mh_profile_name*: Specify the name of the message handling profile.
- **skip optional-ies [ epdgInd | registrationTime ]** : Skips ePDG indication and registration time in the Registration Request.

**Note** The **skip optional-ies [ epdgInd | registrationTime ]** command is available only with the UdmUecmRegisterSMF message.

# Configuration-based Control of Subscription Notification with Immediate Report

*Table 3: Feature History*

| Feature Name | Release Information | Description |
|---|---|---|
| Configuration-based Control of Subscription Notification with Immediate Report Indication on N10 Interface | 2024.03.0 | With the "immediateReport" attribute enabled on UDM, the SMF ignores the UDM subscription fetch message and continues with the Subscription Notification Request, in which Subscription Management Data is received in the Subscription Notification Response.<br><br>With this controlled UDM notification subscription with the immediate report indication, the interactions between SMF and UDM over the N10 interface are minimized during the 4G, 5G, and Wi-Fi attach procedures and hence prevents message overload and failures.<br><br>This feature introduces the new CLI command **subscription notify-immediate** in the DNN profile.<br><br>**Default Setting:** Disabled – Configuration Required |

## Feature Description

When the "immediateReport" attribute is enabled on UDM in the Nudm_SDM_Subscribe service operation request, the SMF ignores the UDM subscription fetch message and continues with the subscription notification. The UDM sends the subscription data in the response to the SMF. The "immediateReport" attribute indicates whether an immediate report is required or not. By default, this attribute is disabled.

By enabling this feature, the interactions between SMF and UDM over the N10 interface are minimized during the 4G, 5G, and Wi-Fi attach procedures. The minimized interactions prevent message overload and failures. You can configure this feature through the **subscription notify-immediate** CLI command.

**Note** The "immediateReport" attribute for UDM subscription is supported only for *3GPP TS 29.503 version 16.9.0, Release 16*.

# How it Works

When the subscription notification with immediate report feature is enabled, SMF performs the following tasks:

- When the **subscription notify-immediate** CLI is configured, it's assumed that all the UDMs support the immediate report feature.
- SMF skips the N10 subscription fetch and continues with the N10 subscription notification. SMF includes the "immediateReport" attribute in the Subscribe to Notify request.
- SMF triggers the subscription fetch procedure if the SMF doesn't receive the subscription data in the Subscribe to Notify response.
- SMF triggers subscription notification with the "immediateReport" attribute to UDM through Service Communication Proxy (SCP). If the response includes the subscription data, the SMF doesn't perform the subscription fetch again. Else, the SMF triggers subscription fetch.

**Note** If the subscription notification fails, SMF performs an action based on the failure handling configuration.

- SMF enables the **subscription local rat-type** CLI functionality only when both **subscription local rat-type** and **subscription notify-immediate** CLI commands are configured for the same RAT type.
- SMF ignores the **authorization local** CLI functionality when the **subscription [ local | notify-immediate ]** CLI is configured.

# Configuring Subscription Notification with Immediate Report

To control the UDM Subscription Notification with Immediate Report messages, use the following sample configuration:

```
config
   profile dnn dnn_profile
      subscription [ local | notify-immediate ] rat-type [ nr | eutra |
wlan ]
      end
```

**NOTES:**

- **subscription notify-immediate**: SMF skips the subscription fetch procedure and continues with Subscribe to Notify procedure with the ImmediateReport.

## Verifying Subscription Notification with Immediate Report

This section describes how to verify the subscription notification when the "immediateReport" attribute is enabled.

Use the **show running-config profile dnn** command to verify subscription notification when the "immediateReport" attribute is enabled for various RAT types.

The following is an example output of the **show running-config profile dnn** command for the configured **subscription local rat-type** and **subscription notify-immediate rat-type** values.

```
show running-config profile dnn dnnprof-ims.prod
 profile dnn dnnprof-ims.prod
   subscription local rat-type [ wlan ]
   subscription notify-immediate rat-type [ nr wlan ]
exit
```

# OAM Support

This section describes operations, administration, and maintenance information for the subscription notification with immediate report feature.

## Metrics

As part of this feature, the following label is added:

- **UDM_LABEL_IMMEDIATE_RPT**: This label is added to the RecordUdmMsgProcessingStats **udm_msg_processing_status** metrics to determine the number of PDU sessions that have the "immediateReport" attribute enabled.

The following is an example of metrics when the immediate report feature is disabled.

```
udm_msg_processing_status{app_name="smf",cluster="Local",data_center="DC",
gr_instance_id="1",immediate_report="false",instance_id="0",msg_status="accepted",
rat_type="nr",service_name="smf-service",snssai="",udm_end_point="",udm_msg="UdmRegistration"}
 1

udm_msg_processing_status{app_name="smf",cluster="Local",data_center="DC",
gr_instance_id="1",immediate_report="false",instance_id="0",msg_status="accepted",
rat_type="nr",service_name="smf-service",snssai="",udm_end_point="",udm_msg="UdmSmSubscription"}
 1

udm_msg_processing_status{app_name="smf",cluster="Local",data_center="DC",
gr_instance_id="1",immediate_report="false",instance_id="0",msg_status="accepted",
rat_type="nr",service_name="smf-service",snssai="",udm_end_point="10.105.35.112:8001",
udm_msg="UdmSubscribeToNotify"} 1

udm_msg_processing_status{app_name="smf",cluster="Local",data_center="DC",
gr_instance_id="1",immediate_report="false",instance_id="0",msg_status="attempted",
rat_type="nr",service_name="smf-service",snssai="",udm_end_point="",udm_msg="UdmRegistration"}
 1

udm_msg_processing_status{app_name="smf",cluster="Local",data_center="DC",
gr_instance_id="1",immediate_report="false",instance_id="0",msg_status="attempted",
rat_type="nr",service_name="smf-service",snssai="",udm_end_point="",udm_msg="UdmSmSubscription"}
 1

udm_msg_processing_status{app_name="smf",cluster="Local",data_center="DC",
gr_instance_id="1",immediate_report="false",instance_id="0",msg_status="attempted",
rat_type="nr",service_name="smf-service",snssai="",udm_end_point="10.105.35.112:8001",
udm_msg="UdmDeregistration"} 1

udm_msg_processing_status{app_name="smf",cluster="Local",data_center="DC",
gr_instance_id="1",immediate_report="false",instance_id="0",msg_status="attempted",
rat_type="nr",service_name="smf-service",snssai="",udm_end_point="10.105.35.112:8001",
udm_msg="UdmSubscribeToNotify"} 1
```

The following is an example of metrics when the immediate report feature is enabled.

```
udm_msg_processing_status{app_name="smf",cluster="Local",data_center="DC",
gr_instance_id="1",immediate_report="false",instance_id="0",msg_status="accepted",
rat_type="nr",service_name="smf-service",snssai="",udm_end_point="",udm_msg="UdmRegistration"}
 1

udm_msg_processing_status{app_name="smf",cluster="Local",data_center="DC",
gr_instance_id="1",immediate_report="true",instance_id="0",msg_status="accepted",
rat_type="nr",service_name="smf-service",snssai="",udm_end_point="",udm_msg="UdmSmSubscription"}
 1

udm_msg_processing_status{app_name="smf",cluster="Local",data_center="DC",
gr_instance_id="1",immediate_report="true",instance_id="0",msg_status="accepted",
rat_type="nr",service_name="smf-service",snssai="",udm_end_point="10.105.35.112:8001",
udm_msg="UdmSubscribeToNotify"} 1

udm_msg_processing_status{app_name="smf",cluster="Local",data_center="DC",
gr_instance_id="1",immediate_report="false",instance_id="0",msg_status="attempted",
rat_type="nr",service_name="smf-service",snssai="",udm_end_point="",udm_msg="UdmRegistration"}
 1

udm_msg_processing_status{app_name="smf",cluster="Local",data_center="DC",
gr_instance_id="1",immediate_report="true",instance_id="0",msg_status="attempted",
rat_type="nr",service_name="smf-service",snssai="",udm_end_point="",udm_msg="UdmSmSubscription"}
 1

udm_msg_processing_status{app_name="smf",cluster="Local",data_center="DC",
gr_instance_id="1",immediate_report="false",instance_id="0",msg_status="attempted",
rat_type="nr",service_name="smf-service",snssai="",udm_end_point="10.105.35.112:8001",
udm_msg="UdmDeregistration"} 1

udm_msg_processing_status{app_name="smf",cluster="Local",data_center="DC",
gr_instance_id="1",immediate_report="true",instance_id="0",msg_status="attempted",
rat_type="nr",service_name="smf-service",snssai="",udm_end_point="10.105.35.112:8001",
udm_msg="UdmSubscribeToNotify"} 1
```

# N10 Fail Open on Converged Core

***Table 4: Feature History***

| Feature Name | Release Information | Description |
|---|---|---|
| Minimizing SMF and UDM interactions on N10 interface through fail open support | 2023.04 | With the fail open feature, the SMF supports the ignore and continue failure handling actions as well for the call during the N10 message failures. These failures include the UDM registration, Subscriber Fetch, and Subscribe to Notify for all RAT types. |

## Feature Description

This feature provides the following support:

- Fail open—The SMF supports the ignore and continue failure handling actions as well for the call during the N10 message failures. These failures include the UDM registration, Subscriber Fetch, and Subscribe to Notify for all RAT types, which are LTE, NR and Wi-Fi.

  If the registration isn't performed or failed during the session creation, then based on the configured failure handling action, the SMF performs the N10 registration during handover. The failure handling action can either be ignore or continue. If you haven't configured the failure action to terminate the session, and the failure happens during handover, then SMF ignores the failure and continues with the handover procedure.

- Interaction with SCP Model-D—If you have configured the SCP failure handling as continue or ignore, then the preceding failure handling is applicable. If you have configured the SCP failure handling as a fallback and a failure happens after the fallback, then the action that you configured under UDM failure handling is applicable.

# Configuration-based Control of Subscription Messages

## Feature Description

The Unified Data Management (UDM) is responsible for primarily storing the subscriber data, which SMF accesses for managing the user sessions on the network. The SMF explicitly subscribes to receive the notifications about the events that occur in the subscriber data such session terminate. When the SMF wants to stop receiving the notifications, it initiates the Unsubscribe-to-Notification messages to UDM. Upon receiving these messages, the UDM cancels the subscription by removing the notification subscription for the subscribed session.

**Note** The SMF does not receive notification when the UDM-triggered subscription change is observed. However, for UDM-triggered session terminations, the SMF receives notifications from UDM.

## How it Works

This section provides a overview of how the SMF and UDM communicate over the Unsubscribe-to-Notifications message:

1. The NF, such as SMF, sends an Unsubscribe-to-Notifications request to the resource identified by the URI to the UDM. The SMF transacts the request to the UDM over the N10 interface. The Unsubscribe-to-Notifications request allows the SMF unsubscribe from notifications for a specific subscriber session. The SMF receives the URI details during the subscription creation process.

   The Unsubscribe-to-Notifications request contains the 'SUPI' and 'subscriptionId' in the URI.

2. The UDM processes the request, and based on the response; it sends a response code to the SMF. For example, if the unsubscription is successful, then UDM sends 204 code. If the request is not processed, then the appropriate HTTP status code indicating the error is returned in the response body along with the additional error information.

3. The SMF handles the timeout and failure that occurs when sending the Unsubscribe-to-Notifications messages to the UDM. In case the Unsubscribe-to-Notifications request fails, the SMF continues to purge the corresponding sessions.

   The Unsubscribe-to-Notification message is required for sessions that are hosted on the EUTRA network. Being on this network may not be a requirement for sessions that are released on the NR and WLAN network. For these access types, the SMF sends the UDM registration and deregistration messages that include subscription to notifications through implicit-unsubscribe during the deregistration.

## Standards Compliance

The Support for the Unsubscribe-To-Notifications Messages feature complies with the following standards:

- *3GPP TS 29.503 - 5G System; Unified Data Management Services*

## Call Flows

This section describes the call flow for the Unsubscribe-To-Notifications message support.

### Unsubscribe-to-Notifications Call Flow

This section describes the call flow on how the SMF sends a request to the UDM to unsubscribe from notifications of data changes.

***Figure 1: Unsubscribe-to-Notifications Communication with UDM***

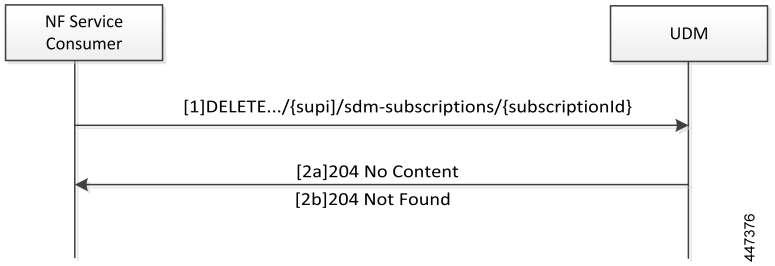


***Table 5: Unsubscribe-to-Notifications Communication Call Flow Description***

| Step | Description |
|---|---|
| 1 | The NF service consumer, such as SMF, sends a request to the UDM to unsubscribe from notifications. By unsubscribing, the UDM no longer sends notifications to SMF when the data modifications occur in the respective subscriber session.<br>The NF service consumer sends a DELETE request to the resource identified by the URI. The NF service consumer receives the URI when the subscription gets created. |
| 2a | If the deletion of request is successful, the UDM responds with "204 No Content". |
| 2b | If the subscription is invalid, which can be due to an unknown subscriptionId value, then the HTTP status code "404 Not Found" is returned along with the additional error information in the response body (as part of the "ProblemDetails" element).<br>If the request is not processed, then the appropriate HTTP status code indicating the error is returned in the DELETE response body along with the additional error information. |

# OAM Support for the Unsubscribe-To-Notifications Messages

This section describes operations, administration, and maintenance information for this feature.

## Statistics Support

The SMF maintains the following labels on the smf-rest-ep pod for monitoring the number of unsubscribe-to-notifications messages that are initiated towards UDM:

- nfType – “udm”
- messageDirection – “outbound”
- apiName – “sdm_unsubscription_req”
- nfUri – “nf_uri”
- respStatus – “response_status”
- rspCause – “response_cause”

# Configuration-based Control of UDM Registration Messages

***Table 6: Feature History***

| Feature Name | Release Information | Description |
|---|---|---|
| Configuration-based Control of UDM Registration Messages | 2024.01.2 | SMF allows the user to ignore the UDM registration messages during the PDU setup and Wi-Fi attach procedures.<br>With this controlled UDM registrations, the interactions between SMF and UDM over the N10 interface are minimized to handle the message overload and attach failures on the N10 interface.<br>This feature introduces the new CLI command **skip-n10-registration rat-type [ NR \| WIFI \| ALL ]** in the DNN profile.<br>**Default Setting:** Disabled – Configuration Required |

# Feature Description

When the UE attach failures are caused by message overload on N10 interface, SMF ignores the UE registration messages from reaching the UDM. This feature optimizes the N10 interactions in the 5G and Wi-Fi network. You can configure this feature through the **skip-n10-registration rat-type [ NR | WIFI | ALL ]** CLI command.

# How it Works

When you configure the **skip-n10-registration rat-type [ NR | WIFI | ALL ]** CLI command, SMF ignores the UDM registration during session creation in the 5G and Wi-Fi network. The handover between 5G and Wi-Fi fails because ePDG doesn't find the target SMF that was handling the session. If the UE registration fails during 5G or Wi-Fi attach and if the UDM failure handling template (FHT) is configured as continue or ignore, then SMF reattempts the registration during Wi-Fi-to 5G and 5G to Wi-Fi handovers.

# Configuring Control for UDM Registration Messages

To control the UDM registration messages, use the following sample configuration:

```
config
   profile dnn dnn_profile
      skip-n10-registration rat-type [ NR | WIFI | ALL ]
      exit
```

**NOTES:**

- **skip-n10-registration rat-type [ NR | WIFI | ALL ]**: Specify the RAT type for which you want to ignore the UDM registration.

# Session Management based on UDM Data Change Notification

## Session management based on UDM data change notification

*Table 7: Feature History*

| Feature Name | Release Information | Description |
|---|---|---|
| Session management based on UDM data change notification | 2025.01.0 | With this feature, SMF terminates or continues the session on receiving the data change notification with REPLACE operation from Unified Data Management (UDM). This feature enables the User Equipment (UE) to avail new subscription changes.<br>This feature introduces a new event **subscription-change** in the existing Event Management Policy configuration.<br>**Command Enhanced**:<br>**policy eventmgmt** *policy_eventmgmt_name* **priority** *priority_number* **event** *subscription-change*<br>**Default Setting**: Disabled–Configuration Required |

When there is a change in the subscriber data such as upgrading the data speed from 2 Mbps to 8 Mbps, UDM sends this data change notification with the changed attributes to SMF. SMF continues or terminates the session. When SMF terminates the session the new subscription value gets updated in the UE. When SMF continues the session, no action takes place in UE.

## How SMF manages the session using UDM data change notification

When SMF receives UDM data change notification, SMF takes action based on user-defined configuration in the event management policy or the system-defined configuration. When SMF finds the user-defined configuration for subscription change in the event management policy, then SMF manages the session based on that configuration. Otherwise, SMF manages the session based on the system-defined configuration.

UDM data change notification includes the ModificationNotification with a list of NotifyItems. The NotifyItems includes the ChangeItem field with the following changed attributes:

- Ambr
- 3gppChargingCharacteristics
- sscModes

- Snssai
- SubscribedDefaultQos
- staticIpAddress
- pduSessionTypes

The ChangeItem field includes the following parameters:

- op—Indicates the type of operation that happens to the resource.

**Note** SMF takes action only if the op attribute is set to REPLACE operation.

- path—Contains the JSON pointer value which indicates the target location within the resource where the change is applied.

**Note** SMF supports the UDM change notify path as empty (“”).

- origValue—Indicates the original value at the target location. This attribute is present if the op attribute is of value MOVE, REPLACE, or REMOVE.
- newValue—Indicates a new value at the target location. This attribute is present if the op attribute is of value ADD or REPLACE.

**Note** On hSMF (home SMF), SMF continues the call on receiving the UDM change notification and performs no other action.

## Manage session with user-defined configuration

These stages describe how SMF manages the session with user-defined configuration:

1. When SMF receives the data change notification with the op attribute value set to REPLACE. SMF terminates or continues the session based on the user-defined action of the *subscription-change* event in the policy eventmgmt CLI command.
2. If the action *terminate* is configured for the subscription-change event, SMF terminates the session and sends N1 cause as Reactivation-Require message to UE. The new subscription value gets updated in the UE. If the action *continue* is configured for the subscription-change event, SMF continues the session and changes are not updated in the UE.

For more information, refer to the Enable subscribtion change event in the event management policy section.

## Manage session with system-defined configuration

These stages describe how SMF manages the session based on the system-defined configuration:

1. When SMF receives the UDM data change notification that has origValue and newValue with op set to REPLACE operation in the ChangeItem field of NotifyItem, then SMF terminates or continues the session.
2. On receiving the NotifyItems in the notification, SMF compares the origValue of the NotifyItem with its subscribed data.
3. If the origValue matches the subscribed data, the SMF selects the relevant NotifyItem. Within the selected NotifyItem, the SMF compares the origValue with the newValue of the attributes. This comparison identifies which attribute is changed. Based on the changed attributes identified in the origValue and newValue comparison, the SMF decides whether to terminate or continue the session. When SMF terminates the session, the new subscription value is updated in the UE. When SMF continues the session, no action is taken.

**Note** When SMF receives the data change notification without the origValue in the ChangeItem field of NotifyItem, then SMF continues the session and sends an error response 503 to UDM.

The SMF manages the session as shown in the following table:

| Attribute | Condition | Action |
|---|---|---|
| Ambr | SMF does not compare the newValue and origValue of Ambr. | Continues the session and takes no action. |
| 3gppChargingCharacteristics | Condition 1: If the newValue of ChargingCharacteristics is different from the origValue of ChargingCharacteristics. | Terminates the session. |
| | Condition 2: If the newValue of ChargingCharacteristics is the same as the origValue of ChargingCharacteristics. | Continues the session and takes no action. |
| Snssai | Condition 1: If the newValue of Snssai has no sst and sd match or no sst match (in case when Snssai includes only sst) with the origValue. | Terminate the session. |
| | Condition 2: If the newValue of Snssai has sst and sd match or sst match (in case when Snssai includes only sst) with the origValue. | Continues the session and takes no action. |
| SubscribedDefaultQos | SMF does not compare the newValue and origValue of SubscribedDefaultQos. | Continues the session and takes no action. |

| Attribute | Condition | Action |
|---|---|---|
| sscModes | Only SSCmode1 is supported on SMF. If SSCmode1 is not preset on newValue, then the call is disconnected. | Terminates the session. |
| pduSessionTypes | Condition 1: If the newValue of pduSessionType is not the same or superset of origValue. | Terminate the session. |
| | Condition 2: If the newValue of pduSessionType is the same of origValue. | Continues the session and takes no action. |
| staticIpAddress | Condition 1:If the newValue of staticIpAddress is not the same as of origValue. | Terminate the session. |
| | Condition 2: If the newValue of staticIpAddress is the same as of origValue. | Continues the session and takes no action. |

# Enable subscription change event in the event management policy

Follow these steps to perform the user-defined configuration in the event management policy:

**Procedure**

**Step 1** Configure the subscription change event, on page 17

**Step 2** Define rules and condition for subscription change event, on page 18

**Step 3** Configure action for subscription change event, on page 19

## Configure the subscription change event

Follow these steps to configure the subscription change event:

**Procedure**

**Step 1** Enter the event management policy configuration mode.

**policy eventmgmt** *policy_eventmgmt_name*

**Example:**

```
[smf] smf# config
[smf] smf(config)# policy eventmgmt em1
```

**Step 2** Define the priority of the event management policy and then configure the subscription-change event, rule, and action name to be executed.

**priority** *priority_number* **event** *event_name* **ruledef** *ruledef_name* **actiondef** *actiondef_name*

**Example:**

```
[smf] smf(config-eventmgmt-em1)# priority  16 event subscription change ruledef
rd-udmDataChangeNotif  actiondef ad-udmDataChangeNotif
```

The valid priority range is from 1 to 65535. Both *ruledef_name* and *actiondef_name* are alphanumeric strings that can be between 1 and 63 characters long.

**Step 3** Save and commit the configuration.

**Example:**

```
[smf] smf(config-eventmgmt-em1))# end
```

**Step 4** [*Optional*] Use **show running-config policy eventmgmt em1** command to verify if the subscription change is enabled.

**Example:**

```
smf] smf# show running-config policy eventmgmt em1
Fri Dec  6  08:34:32.360 UTC+00:00
policy eventmgmt em1
priority 16 event subscription-change ruledef rd-udmDataChangeNotif actiondef ad-udmDataChangeNotif
exit
```

## Define rules and condition for subscription change event

Follow these steps to define the rules and condition for the subscription change event.

### Procedure

**Step 1** Enter the policy rule management configuration mode.

**policy rulemgmt** *policy_rulemgmt_name*

**Example:**

```
[smf] smf# config
[smf] smf(config)# policy rulemgmt rm1
```

**Step 2** Specify the name of the ruledef to add to the policy.

**ruledef** *ruledef_name*

**Example:**

```
[smf] smf(config-rulemgmt-rm1)# ruledef rd-udmDataChangeNotif
```

**Step 3** Define the condition as any.

**Example:**

```
[smf] smf(config-ruledef-rd-udmDataChangeNotif)# condition any
```

The ruledef gets updated with this condition **any**.

**Step 4** Save and commit the configuration.

**Example:**

```
[smf] smf(config-ruledef-rm1))# end
```

**Step 5** [*Optional*] Use **show running-config policy rulemgmt rm1** command to verify if the rules and conditions are defined.

**Example:**

```
smf] smf# show running-config policy rulemgmt rm1
Fri Dec  6  08:34:32.360 UTC+00:00
policy rulemgmt rm1
 ruledef rd-udmDataChangeNotif
 condition any
 exit
```

## Configure action for subscription change event

Follow these steps to configure the action.

### Procedure

**Step 1** Enter the action management policy configuration mode.

**policy actionmgmt** *policy_actionmgmt_name*

**Example:**

```
[smf] smf# config
[smf] smf(config)# policy actionmgmt act1
```

**Step 2** Specify the action definition policy and define the action attributes to be executed.

**actiondef** *actiondef_name*

**Example:**

```
[smf] smf(config-actionmgmt-act1)# actiondef  ad-udmDataChangeNotif
```

**Step 3** Specify the priority in which the actions are to be executed. Then, configure one of the following action:

- terminate—Configure this action to terminate the session. Terminating the session enables to update the new subscription changes in the UE.
- continue—Configure this action to continue the session.

**priority** *priority_number* **action** *action_name*

**Example:**

```
[smf] smf(config-actiondef-act1)# priority  12 action terminate-session
```

The action is configured. Configuring this action enables SMF to decide either to terminate or continue the session.

**Step 4** Save and commit the configuration.

**Example:**

```
[smf] smf(config-actiondef-act1)# end
```

**Step 5** [*Optional*] Use **show running-config policy actionmgmt act1** command to verify if the action is enabled.

**Example:**

```
[smf] smf# show running-config policy actionmgmt act1
Fri Dec  6  08:34:32.360 UTC+00:00
policy actionmgmt act1
 actiondef ad-udmDataChangeNotif
  priority 1 action terminate-session
 exit
```

## Sample configuration for subscription change event

This section provides the sample configuration to enable subscription change in the event management policy with appropriate rules and actions.

```
policy eventmgmt em1-SynAndSemError
      priority 16 event subscription-change ruledef rd-udmDataChangeNotif actiondef
ad-udmDataChangeNotif
exit
policy rulemgmt rm1
     ruledef rd-udmDataChangeNotif
     condition any
exit
policy actionmgmt act1
     actiondef ad-udmDataChangeNotif
     priority 1 action terminate-session
exit
```

# Utilizing bulk statistics to manage session

Use these bulk statistics for managing session using the UDM data change notification.

- **UdmDataNotification:** This label denotes the data change notification message.
- **disc_udm_subscription_change:** This label is added under the Disconnect Stats category to denote the reason associated with the session disconnect based on UDM data change notification.