



# Stale Session Handling and Clearing

- [Feature Summary and Revision History, on page 1](#)
- [Feature Description, on page 2](#)
- [How it Works, on page 2](#)
- [Feature Configuration, on page 4](#)
- [OAM Support, on page 5](#)

## Feature Summary and Revision History

### Summary Data

*Table 1: Summary Data*

Applicable Product or Functional Area	cnSGW-C
Applicable Platform	SMI
Feature Default Setting	Disabled - Configuration required to enable
Related Documentation	Not Applicable

### Revision History

*Table 2: Revision History*

Revision Details	Release
First introduced	2021.02.3

## Feature Description

The cnSGW-C supports identifying and clearing stale sessions using a *session-stale-timer* parameter in the SGW Profile configuration. An example of a stale session is one that is inactive and not read or modified for a specific period of time.

## How it Works

This section describes how this feature works.

## Call Flows

This section describes the key call flows for this feature.

### Timer Expiry Handling Call Flow

This section describes the Timer Expiry Handling call flow.

Figure 1: Timer Expiry Handling Call Flow

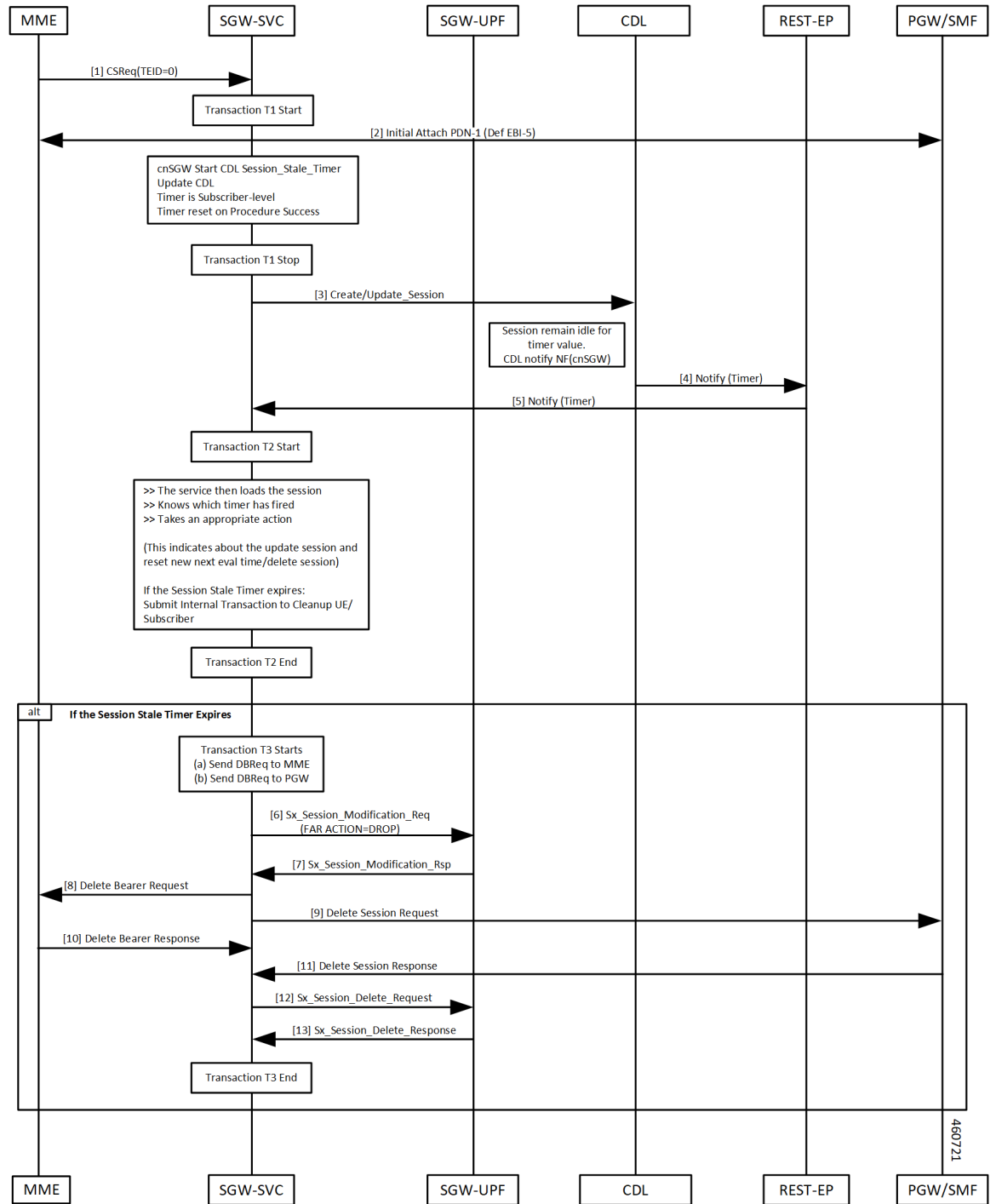


Table 3: Timer Expiry Handling Call Flow Description

Step	Description
1, 2, 3	<p>The initial attach and Session Stale Timer updates are sent to CDL and the timer starts. The sequence is as follows:</p> <ul style="list-style-type: none"> <li>• Initial Attach Success</li> <li>• Session Stale Timer created</li> <li>• CDL updates done.</li> <li>• CDL starts Time (eval time) and waits for update session.</li> </ul> <p><b>Note</b> Session Stale Timer is Reset or Restart, when any of the activity or transaction happens on the Control Plane (cnSGW-C).</p>
4, 5	<p>The Timer expires on CDL pod, and the timer sends notification to cnSGW-C. The sequence is as follows:</p> <ul style="list-style-type: none"> <li>• If no Session update received for eval timer duration</li> <li>• Timer Expiry on CDL pod</li> <li>• CDL sends Timer Notification to cnSGW-C</li> </ul>
6–13	<p>The session cleanup is activated, when the Session Stale Timer expires, and the timer reset isn't required. The sequence is as follows:</p> <ul style="list-style-type: none"> <li>• Receives Timer Notification on cnSGW-C</li> <li>• If the Timer Notification is for Session Stale Timer and if the timer reset isn't required, it starts UE session clean-up activities.</li> <li>• Sends Sx_Session_Modification_Req to UPF to set FAR Action=Drop, Sx_Session_Modification response received</li> <li>• Sends Delete Bearer Request towards MME</li> <li>• Sends Delete Session Request toward PGW</li> <li>• Sends Sx_Session_Delete_Request toward UPF to clean up User Plane data</li> </ul>

## Feature Configuration

To configure this feature, use the following configuration:

```

config
  profile sgw sgw_profile_name
  session-stale-timer session_stale_timer
end

```

### NOTES:

- **session-stale-timer** *session\_stale\_timer*—Specify the maximum number of seconds for which a session can remain idle without any signaling or event, after which the session will be terminated.

The *session\_stale\_timer* value must be in the range of 0–4294967295, and must be greater than the **setup-timeout** and **session-idle-timeout** timer values.

To disable the session-stale-timer configuration, set it to 0.

## Configuration Example

The following is an example configuration.

```
config
  profile sgw sgw1
  session-stale-timer 120
end
```

## Configuration Verification

To verify the configuration:

```
show running-config profile sgw sgw1
session-stale-timer 120
```

The output of the show command includes the following field:

**session-stale-timer**—Indicates the maximum number of seconds for which a session can remain idle without any signaling or event, after which the session is terminated.

## OAM Support

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

## Bulk Statistics

The following statistics are supported for this feature

```
sgw_service_stats{app_name="smf",cluster="Local",data_center="DC",
fail_reason="",gr_instance_id="1",instance_id="0",interface="interface_sgw_egress",reject_cause="",
service_name="sgw-service",sgw_procedure_type="stale_session_initiated_deletion",status="attempted",
sub_fail_reason=""} 1
```

```
sgw_service_stats{app_name="smf",cluster="Local",data_center="DC",fail_reason="",
gr_instance_id="1",instance_id="0",interface="interface_sgw_egress",reject_cause="",
service_name="sgw-service",sgw_procedure_type="stale_session_initiated_deletion",status="success",
sub_fail_reason=""} 1
```

```
sgw_service_stats{app_name="smf",cluster="Local",data_center="DC",fail_reason="",
gr_instance_id="1",instance_id="0",interface="interface_sgw_ingress",reject_cause="",
service_name="sgw-service",sgw_procedure_type="stale_session_initiated_deletion",status="attempted",
sub_fail_reason=""} 1
```

```
sgw_service_stats{app_name="smf",cluster="Local",data_center="DC",fail_reason="",  
gr_instance_id="1",instance_id="0",interface="interface_sgw_ingress",reject_cause="",  
service_name="sgw-service",sgw_procedure_type="stale_session_initiated_deletion",status="success",  
sub_fail_reason="" } 1
```

```
sgw_ue_disconnect_stats{app_name="smf",cluster="Local",data_center="DC",  
gr_instance_id="1",instance_id="0",reason="stale_session_init_disconnect",  
service_name="sgw-service"} 1
```

```
sgw_pdn_disconnect_stats{app_name="smf",cluster="Local",data_center="DC",  
gr_instance_id="1",instance_id="0",pdn_type="ipv4",rat_type="EUTRAN",  
reason="stale_session_init_disconnect",service_name="sgw-service"} 1
```