High Availability and CP Reconciliation

- Feature Summary and Revision History, on page 1
- Feature Description, on page 2
- How it Works, on page 3
- Configuring High Availability and CP Reconciliation, on page 6

Feature Summary and Revision History

Summary Data

Table 1: Summary Data

Applicable Product(s) or Functional Area	cnBNG
Applicable Platform(s)	SMI
Feature Default Setting	Disabled - Configuration Required
Related Documentation	Not Applicable

Revision History

Table 2: Revision History

Revision Details	Release
CP HA and Audit support for LAC and LNS sessions.	2022.03.0
First introduced.	2021.04.0

Feature Description

The high availability (HA) and Reconciliation feature for the control plane supports all cnBNG-specific service pods. This feature supports the IPoE, PTA, LAC, and LNS session types.

CP Audit

When subscriber sessions desynchronize across all pods in the control plane during HA events and session inconsistency, cnBNG runs CP reconciliation (that is, CP audit) to synchronize the sessions.

Note CP reconciliation is also referred to as CP audit in this document.

- Reconciliation between SM and DHCP for IPoE sessions
- Reconciliation between SM, DHCP, and PPP for PTA and LNS sessions
- Reconciliation between SM and PPP for LAC sessions
- Reconciliation between PPP and L2TP for LAC and LNS sessions
- Reconciliation between Node Manager (NM) and FSOLs for all session types

To recover L2TP service after HA events and to avoid service impact, critical information such as L2TP sequence numbers (Ns/Nr), Session Count, and SessionID bitmap must be recovered. The tunnel state for L2TP HA is recovered through recovery from another L2TP service.

Pod Restart

The cnBNG-specific service pods support the pod restart functionality.

This feature supports one BNG-specific service pod restart with a minimum gap of 10 minutes between pod restarts. It also supports one VM restart with a minimum gap of 30 minutes between VM restarts.

The HA support for L2TP is provided using a local peer service to synchronize the necessary L2TP tunnel information and recover it after restart. It also supports hitless operation during pod restart, and the restart of pods without impacting the existing sessions and tunnels. This feature helps to recover the L2TP control connection information such as Ns/Nr sequence numbers, Tunnel Context, and Session Bitmap, and also resume the control channel.

Note VM reboot is not supported for LAC and LNS sessions.

The restart of service pods has the following impact:

- CPU or memory spike can occur if there is a churn of sessions during pod restart. For example, if SM has two replicas such as instance 1 and instance 2, and if instance 1 restarts, there will be spike in the CPU or memory in instance 2.
- Service pod IPCs can timeout if the destination service pod restarts before responding to the ongoing IPCs.

- CDL updates of ongoing sessions can fail and result in desynchronization of sessions between pods.
- If subscriber sessions desynchronize between CP and UP, cnBNG runs CP to UP reconciliation.
- If IP address leaks occur in IPAM, run the IPAM reconciliation CLI command **reconcile ipam**.
- ID leaks (CP SubcriberID and PPPoE Session ID) can occur in the NM.
- Reset of Grafana metrics for the restarted pods.

How it Works

This section describes how the high availability and Reconciliation feature for the control plane works.

CP Reconciliation Process

This section describes the CP reconciliation scenarios and processes.

On issuing the manual CP Audit CLI command, FSOL services (DHCP and PPPoE) start reconciling their respective sessions with the SM service to check if the session exists and if the Audit-ID matches. When this check is passed, it proceeds to the next step, else, FSOL disconnects the session.

In the next step, FSOLs tries to audit the session with the Node Manager (NM) to check if the IP address and ID resources are matching. This is to ensure the consistency of the IP and ID resource across session database and NM.

After reconciliation from FSOLs, SM triggers the final reconcile to remove any extra sessions. At the end of this step, all services are expected to have a consistent session database.

CP reconciliation supports the following functionality:

- Supports a maximum of five CP reconciliations in parallel for different UPs.
- Configure the mandatory **cdl datastore session slot notification max-concurrent-bulk-notifications** CLI command to run CDL bulk notifications in parallel for multiple bulk notification requests. Without this configuration, the CP reconciliation process can be slow.

  For information, see the Configuring CDL Bulk Notifications, on page 7.

  New bulk notification requests are put in queue and these requests are dequeued one at a time when the ongoing request is complete.

  Each CP reconciliation request invokes three bulk notification requests to the CDL. Hence, five CP reconciliation requests invoke a maximum of 15 bulk notifications. With this configuration, the **clear subscriber** CLI command is executed in parallel.

  Each **clear subscriber** CLI command invokes one CDL bulk notification request to the CDL. Executing more than 5 **clear subscriber** and **show subscriber** CLI commands slows down the CP reconciliation process. Therefore, it is recommended to avoid these commands while CP reconciliation is in progress.

- CP reconciliation deletes the session in the following scenarios:
  - Extra sessions in DHCP or PPP compared to SM
  - Extra sessions in SM compared to DHCP or PPP
  - Mismatch in session data between DHCP, PPP, and SM

- Mismatch between IP and ID resources between FSOLs and NM

- When a session is deleted in the CP or UP because of a mismatch, the same deleted session could be present in the CPE. This causes traffic loss for the deleted subscriber until the subscriber is recreated after lease expiry for an IPoE session.

Note

- If any pod (SM, DHCP, or PPPoE) restarts while CP reconciliation is in progress, there may still be a session mismatch across pods even after completing the CP reconciliation.
- CP reconciliation without churn and HA events in CP or UP—if it is executed within the supported TPS limits and sessions across pods in the CP synchronize after completing CP reconciliation.
- CP reconciliation with churn (session bring-up or bring-down, CoA) and no HA events in CP and UP:
  - If CP audit is executed within TPS limits and sessions across pods in CP synchronize after completing CP audit.
  - CP audit reconciles sessions that are created before starting the audit. CP audit does not reconcile sessions that are created after starting audit.
  - CP audit does not reconcile sessions that are updated 60 seconds before audit start time. For example, session update time is T1 and audit start time is T2, if T2 minus T1 is less than or equal to 60 seconds, then that session is not audited.

L2TP Audit Process

Depending on the time of HA event and session or tunnel churn, session inconsistency can occur across CP pods mainly between PPPoE, SM, and L2TP. The L2TP tunnel and session ID syncing is required between NM and L2TP, PPPoE and L2TP respectively.

The L2TP Audit has the following two stages:

1. The PPPoE service is modified to check if the session has L2TP sessions or not. It will sync the session with L2TP.
2. The L2TP service checks the audited bitmap. It sweeps the remaining bitmap to clear any stale or non-matching sessions and synchronizes the PPPoE services. DHCP, SM, and PPPoE services will synchronise for LNS sessions.

Automatic Session Mismatch Detection

An existing Audit ID is incremented and sent to the SM for every new transaction initiated from DHCP or PPP to SM. If the transaction is successful, this Audit ID is stored in DHCP or PPP, and in the SM CDL records.

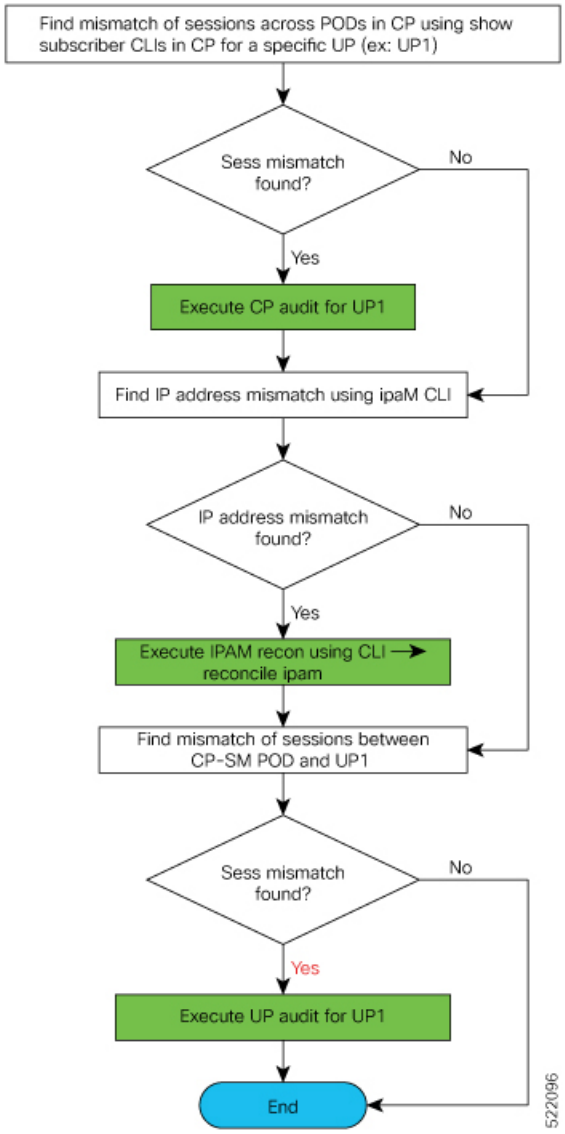
The SM validates the Audit ID received in every request from DHCP or PPP. When a received Audit ID is not incremental to the Audit ID present in the SM, the SM discards the transaction and responds to the DHCP or PPP with an Audit ID mismatch error. The SM then initiates a new transaction to disconnect the session in CP and UP.

Synchronizing Sessions Across CP Pods and UP

CP reconciliation or UP reconciliation (that is, reconciliation between CP-SM pod and UP) is executed for a specific UP.

The following figure depicts the procedure to synchronize sessions across CP pods and UP, for a specific UP.

Figure 1: Synchronizing Sessions Across CP Pods and UP



Limitations and Restrictions

The High Availability and CP Reconciliation feature has the following limitations and restrictions in this release:

- Supports only IPoE and PPPoE sessions for High Availability and CP reconciliation.

- Supports one BNG-specific service pod restart with a minimum gap of 10 minutes between pod restarts.
- Supports one VM restart with a minimum gap of 30 minutes between VM restarts.
- Does not support double fault for infrastructure pods (cache, CDL, and Node Manager). The system goes to "bad" state with double faults.

Configuring High Availability and CP Reconciliation

This section describes how to configure the High Availability and CP Reconciliation feature.

Configuring the High Availability and CP Reconciliation feature involves the following steps:

- Reconciling Sessions Across CP Pods and UP, on page 6
- Configuring CDL Bulk Notifications, on page 7

Reconciling Sessions Across CP Pods and UP

Use the following commands to reconcile subscriber sessions across PPP, DHCP, and SM pods in the CP with the specified UP.

```
subscriber session-synchronize-cp upf upf_name { abort |
     timeout timeout_value | tps tps_value }
```

NOTES:

- **upf** *upf_name* : Configures CP reconciliaton for this UPF.

Note The maximum number of CP reconciliations supported is 5.

- **{ abort | timeout** *timeout_value*| **tps** *tps_value* **}**: Specifies the following parameters for subscriber session reconciliation:

  abort : Aborts the ongoing CP reconciliation for only the specified UPF.

  timeout *timeout_in_seconds* : Specifies the number of seconds the reconciliation can run. If it runs longer than the specified timeout, the reconciliation process is aborted. The valid values range from 2 to 100 minutes. The default value is 60 minutes.

  tps *tps_value* : Specifies the number of notifications sent from the CDL to Node Manager per second. The valid values range from 40 to 1000. The default is 40.

Note Set this value based on the scale and churn of sessions during the CP reconciliation.

Verifying High Availability and CP Reconciliation

Use the following **show** command to verify ongoing or completed CP audit details for the specified UPF.

Note Only one CP audit detail is stored per UPF.

show subscriber synchronize-cp upf *upf_name*

Example

The following is a configuration example.

```
[cnbng-smi-40g-tb03/bng] bng# show subscriber synchronize-cp upf lps_asr9k-1
subscriber-details
{
"Audit ID": 1634722199,
"Session Audit Statistics": {
"DHCP": {
"Audit State": "Completed",
"Session Count": 430,
"Notifications Received": 430
},
"LNS": {
"Audit State": "N/A",
"Session Count": 0,
"Notifications Received": 0
},
"PTA & LAC": {
"Audit State": "N/A",
"Session Count": 0,
"Notifications Received": 0
},
"Session Manager": {
"Audit State": "Completed",
"Session Count": 404,
"Notifications Received": 404
}
},
"Audit State": "Completed",
"Notifications/Sec": 40,
"Timeout": 6000,
"Audit Started": "2021-10-20 09:29:59 +0000 UTC",
"Fsol Audit Started": "2021-10-20 09:29:59 +0000 UTC",
"SM Audit Started": "2021-10-20 09:30:10 +0000 UTC",
"Audit Ended": "2021-10-20 09:30:22 +0000 UTC",
"Total Time Taken": "23 Seconds"
}
```

Configuring CDL Bulk Notifications

Use the following commands to run CDL bulk notifications in parallel for multiple bulk notification requests.

Note This is a mandatory configuration for CP reconciliation.

```
config
  cdl datastore session slot notification max-concurrent-bulk-notifications
 value
  exit
```

NOTES:

- **max-concurrent-bulk-notifications** *value*: Specifies the maximum number of bulk task notifications that CDL can process concurrently. The valid values range from 1 to 20.

  Configure this value to 20 for CP reconciliation.

Sample Configuration

The following is a sample configuration of the CDL bulk notification where a maximum of 20 bulk notifications are executed in parallel for multiple bulk notification requests.

```
config
   cdl datastore session slot notification max-concurrent-bulk-notifications 20
   exit
```