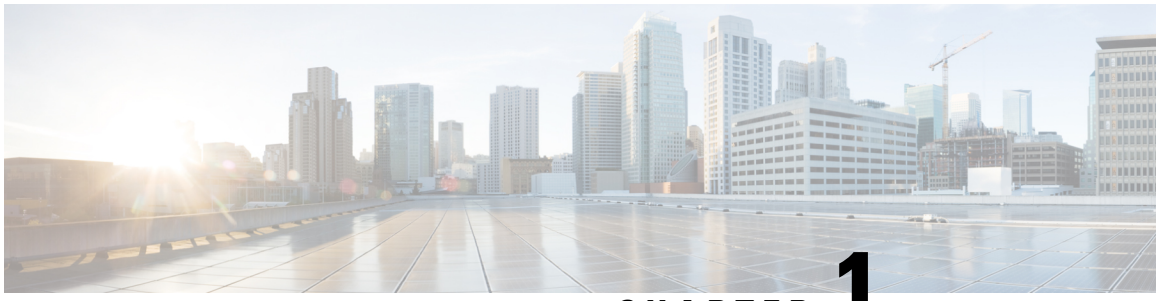
Cisco Ultra Cloud Core Serving Gateway Control Plane Function, Release 2023.01 - Release Change Reference

First Published: 2023-01-24

Last Modified: 2023-01-23

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS



CHAPTER 1

UCC 5G cnSGW-C - Release Change Reference

Feature Defaults Quick Reference

The following table indicates what features are enabled or disabled by default in this release.

Features/ Behavior Changes	Default	Release Introduced/ Modified
Cache Pod Optimization	Enabled – Always-on	2023.01.0
PFCP Session Report with DLDR Throttling Support, on page 4	Disabled – Configuration required to enable	2023.01.0
Resiliency Handling, on page 4	Disabled – Configuration required to enable	2023.01.0

Performance Optimization Features

Feature Summary and Revision History

Summary Data

Table 1: Summary Data

Applicable Products or Functional Area	cnSGW-C
Applicable Platforms	SMI

Feature Default Setting	Async BG IPC from GTPC-EP towards SGW-Service: Enabled – Always-on Batch ID Allocation, Release, Reconciliation Support: Disabled – Configuration required to enable Cache Pod Optimization: Enabled – Always-on CDL Flush Interval and Session Expiration Tuning Configuration: Enabled – Configuration required to disable DDN Call Flow Optimization: Disabled – Configuration required to enable DDN Timeout Configuration: Disabled – Configuration required to enable Edge Echo Implementation: Enabled – Always-on ETCD Peer Optimization Support: Enabled - Always-on Flag the DB Database Updates: Enabled – Always-on GTPC Interface Split: Disabled – Configuration required to enable GTPC IPC Cross-rack Support: Disabled – Configuration required to enable Interservice Pod Communication: Disabled – Configuration required to enable Maintenance Mode: Disabled – Configuration required to enable MBR Call Flow Optimization: Disabled – Configuration required to enable Optimized GTPv2 Encoder and Decoder: Enabled – Always-on Partial CDL Update for Idle Active Call Flow: Enabled – Always-on PFCP Session Report with DLDR Throttling Support: Disabled – Configuration required to enable Resiliency Handling: Disabled – Configuration required to enable Roaming Peer Path Management Optimization: Disabled – Configuration required to enable UDP Proxy functionality merge into Protocol microservices: Enabled – Configuration required to disable
Related Documentation	Not Applicable

Revision History

Table 2: Revision History

Revision Details	Release
Support for the following sub-features were introduced: • Added support for cache pod optimization. • PFCP Session Report with DLDR Throttling Support • Resiliency Handling	2023.01.0
Support for the following sub-features were introduced: • Batch ID Allocation, Release, and Reconciliation • CDL Flush Interval and Session Expiration Tuning Configuration • DDN Call Flow Optimization • DDN Timeout Configuration • Edge Echo Implementation • ETCD Peer Optimization Support • Flag the DB Database Updates • GR Split • GTPC Interface Split • GTPC IPC Cross-rack Support • Interservice Pod Communication • Introduced support for IPv6. • Maintenance Mode • Optimization of Modify Bearer Request and Response (MBR) call flows • Optimized GTPv2 Encoder and Decoder is provided for additional Request and Response messages. • UDP Proxy and GTPC-EP Merge • UDP Proxy and PFCP-EP Merge	2022.04.0
First introduced.	2021.02.3

Cache Pod Optimization

Feature Description

The cnSGW-C supports the cache pod optimization to reduce the cache pod query at the GTPC endpoint.

The get affinity query is used to receive the affinity information in an outgoing request or response message toward the GTPC endpoint. With this optimization, the GTPC endpoint pod doesn't send the query to the cache pod for the upcoming request messages.

To receive this affinity information, an affinity query is used in an outgoing request or response message toward the GTPC endpoint. With this optimization, the GTPC endpoint pod doesn't send the query to the cache pod for the upcoming request messages.

In the previous releases, after the cnSGW-C sent out the DDN and received the MBR from the MME, the GTPC endpoint had to send the query to the cache pod to get affinity information. Later, the cnSGW-C used the affinity information so that an MBR can be forwarded to the correct service pod.

With this optimization, you can prevent the extra cache pod query.

For more information, refer to the UCC 5G cnSGWc Configuration and Administration Guide > Performance Optimization Support chapter.

PFCP Session Report with DLDR Throttling Support

Feature Description

In a live network deployment, due to some external events, all or most of the idle sessions become active at the same time.

When idle sessions become active, the UPF sends the session report request with the report type DLDR to the cnSGW-C. When the cnSGW-C receives the session report with the report type as DLDR, the cnSGW-C sends the DDN message to page UE. To turn the UE active, the MME initiates the paging procedure for the UE. If paging is successful, the MME initiates the service request Modify Bearer Request. On delivering data to the UE, the UE initiates the Release Access Bearer Request and turns idle. This call flow increases an overall load on the system. When the entire call flow occurs for all subscribers in a short time, there’s a huge process overhead on the system.

The PFCP Session Report with the DLDR Throttling Support feature enables the cnSGW-C to limit the number of session report requests. These requests enter the system to prevent the process overload on the system.

For more information, see the *UCC 5G cnSGW-C Configuration and Administration Guide > Performance Optimization Support* chapter.

Resiliency Handling

Feature Description

The Resiliency Handling feature introduces a CLI-controlled framework to support the service pod recovery, when you observe a system fault or a reported crash. It helps in recovering one of the following service pods:

- sgw-service pod
- smf-service pod

- gtpc-ep pod
- protocol pod

These service pods are software modules containing the logic to handle several session messages. The service pods are fault-prone due to any one of the following or a combination of multiple scenarios:

- Complex call flow and collision handling
- Inconsistent session state
- Incorrect processing of inbound messages against the session state
- Unexpected and unhandled content in the inbound messages

Whenever you observe the system fault or a crash, the fault behavior results into a forced restart of the service pod. It impacts the ongoing transaction processing of other sessions. The crash reoccurs even after the pod restart.

To mitigate this risk, use the CLI-based framework with actions defined to clean up subscriber sessions or terminate the current processing.

For more information, see the *UCC 5G cnSGW-C Configuration and Administration Guide* > *Performance Optimization Support* chapter.