



# End-to-End Flow Control

- [Feature Summary and Revision History, on page 1](#)
- [Feature Description, on page 2](#)
- [How it Works, on page 2](#)
- [Configuring End-to-End Flow Control, on page 4](#)

## 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   |
|-------------------|-----------|
| First introduced. | 2021.04.0 |

# Feature Description



**Note** This feature is Network Services Orchestrator (NSO) integrated.

The Cloud Native Broadband Network Gateway (cnBNG) manages residential subscribers from different access planes in a centralized way. It accepts and identifies subscriber Control Plane (CP) traffic coming from multiple User Planes (UPs) associated with the CP. When the number of UPs scale, the amount of CP traffic coming from each UP multiplies.

The traffic flow between CP and UP must be regulated to ensure that the CP attends all the service requests without service interruption.

The following scenarios create burstiness or higher flow rates in the traffic flows:

- Power outage in a residential area
- Access network outage for a specific period
- UP catastrophic events such as process crash, route processor reboots, and chassis reload

These scenarios generate sudden spike in traffic going to the CP. To handle these traffic spikes, it is necessary to flow control and rate limit the CP ingress to ensure that service applications are not overwhelmed with these bursts. The End-to-End Flow Control feature optimizes flow control and rate limit of the traffic toward the CP ingress.

## How it Works

This section describes how End-to-End Flow Control works in cnBNG.

There are two types of traffic that enter or exit the CP:

- Control traffic that is responsible for subscriber session creation
- Control traffic that is already provisioned for a subscriber session

The following application infrastructure (App-Infra) features facilitates the cnBNG CP ingress packet flow control:

- [Dispatcher, on page 2](#)
- [Overload Control, on page 3](#)

## Dispatcher

In the dispatcher, if you configure the right dequeue rate, the packets do not pile up in the PFCP queue. The dequeue rate must be higher than the incoming rate from the UP.

All PFCP packets land into a single queue because there is no packet segregation. Any rate control that is applied on this queue is per UPF PFCP packet rate control. It is not possible to control a particular type of

packet per UPF. For example, DHCP release, PPPoE PADT, or keepalive failure notification packet cannot be controlled per UPF at the dispatcher queue.

The dispatcher queue size configuration handles the burst of packets. This functionality supports the following:

- Dedicated queue for each PFCP or N4 interface, and GTPu interface for each UPF connected to the control plane
- Configuration of queue size and flow control rate limits

## Overload Control

Overload control is applied to a packet after it is released from a dispatcher. This creates a queue based on the packet type at the aggregate level across all UPF data. Because overload control enables packet type based queues, rate control is applied for that type of packet at the aggregate level of all UPFs.

Special treatment of the packet is indirectly achieved by having different queues for a packet at overload control feature and aggregate of all UPF level.

The dispatcher supports the following categories of virtual message groups:

- PFCP keepalive messages between CP and UPF
- PFCP LCP keepalive failure notification messages
- PFCP Response messages
- Session Report messages
- Other message types which are not listed in different categories

The Overload Control feature provides aggregate queues for a message type coming from UPF functions. Group IDs are supported for each message group and the message type is configurable for each group. When configured, a virtual queue is created for each message type and treated based on the configured attributes for that group. For each queue, the size and rate limit can be configured.

For each message, the configured rate of packets are dequeued and sent to the CPF. For priority packets such as PFCP keepalives, dedicated queues are allocated so that they are not impacted with other queue sizes.

Based on the cluster capacity, specific values for each queue and message type must be configured. The values are adjusted based on the capacity.

## Limitations and Restrictions

The End-to-End Flow Control feature has the following limitations and restrictions:

- Session bring-down rate (DHCP release, PPPOE PADT, L2TP, CDN rate control) cannot be enforced using the CP flow control configuration. Also, UP does not have flow control of these packets. Therefore, solution level flow control for session disconnect triggers for all session types is not supported.
- Packet level flow control for DHCPv4 and DHCPv6 Renew, and DHCPv6 Relay forwarded messages is not supported.
- L2TP LAC and LNS FSOL rate control are not supported on the ASR 9000 UP in this release. The CP does not have rate control based on FSOL. Because PPPoE bring-up controls LAC, PPPoE FSOL rate control on ASR 9000 can be used to control LAC session bring-up.

- Dispatcher configuration changes require restarting of the CP.
- Flow control must be configured at the UP level for the following packets at the UPF. This ensures that the packet rate from UP to CP is controlled because CP cannot provide per packet rate control, per UPF.
  - FSOL
  - Session delete notifications
  - LCP keepalive failure notifications
  - Session statistics report

## Configuring End-to-End Flow Control

This section describes how to configure the End-to-End Flow Control feature on Control Plane (CP).

The configuration involves the following procedures:

- [Configuring Dispatcher for GTPu Interface, on page 4](#)
- [Configuring Dispatcher for N4 Interface, on page 5](#)
- [Configuring Overload Control for Message Types, on page 7](#)

## Configuring Dispatcher for GTPu Interface

To configure dispatcher for GTPu interface, use the following commands:

```
config
instance instance-id instance_id
  endpoint udp-proxy
    interface gtpu dispatcher { cache { true | false } |
      capacity queue_capacity | count queue_count |

      outbound { true | false } | rate-limit rate_limit |
      threshold threshold_value }
    exit
```

### NOTES:

- **instance** *instance\_id*: Configure multiple instances for the specified instance and enters the instance sub-mode.
- **endpoint udp-proxy**: Configure parameters for the UDP-proxy endpoint and enters the endpoint sub-mode.
- **interface gtpu dispatcher { cache { true | false } | capacity *queue\_capacity* | count *count* | outbound { true | false } | rate-limit *value* | threshold *threshold\_value* }**: Specify the dispatcher parameters for the GTPu interface.
  - **cache { true | false }**: Enable (false ) or disable (true) cache retransmission support. The default value **false** indicates that the cache retransmission support is enabled.
  - **capacity *queue\_capacity***: Specify the number of packets that this queue holds.



**Note** Ensure that there is sufficient memory when configuring higher capacity queues.

- **count** *queue\_count*: Specify the number of N4 queues to be created. Each queue is associated or dedicated to an UPF. For example, if the count is 2, two N4 queues are created and two UPs can be connected.
- **outbound { true | false }**: Enable (true) or disable (false) queue support for outbound messages. Default value: **false**.



**Note** Outbound flow control for BNG is not supported.

- **rate-limit** *rate\_limit*: Specify the rate limit for each queue, that is, when packets are dequeued. The rate limit is defined in seconds.
- **threshold** *threshold\_value*: Specify the queue size before packets are dropped.

### Example

The following is a configuration example.

```
interface gtpu
  sla response 150000
  dispatcher
    count 1
    capacity 1000000
    outbound true
    rate-limit 500
    cache true
    threshold 950000
    flowctrl-group group1
      capacity 2000
      rate-limit 200
    exit
  exit
exit
exit
exit
exit
```

## Configuring Dispatcher for N4 Interface

To configure dispatcher for N4 interface, use the following commands:

```
config
  instance instance_id
    endpoint udp-proxy
      interface n4 dispatcher { cache { true | false } |
        capacity queue_capacity | count queue_count |

        outbound { true | false } | rate-limit rate_limit |
        threshold threshold_value }
      exit
```

**NOTES:**

- **instance** *instance\_id*: Configure multiple instances for the specified instance and enters the instance sub-mode.
- **endpoint udp-proxy**: Configure parameters for the UDP-proxy endpoint and enters the endpoint sub-mode.
- **interface n4 dispatcher { cache { true | false } | capacity queue\_capacity | count count | outbound { true | false } | rate-limit value | threshold threshold\_value }**: Specify dispatcher parameters for the N4 interface.
  - **cache { true | false }**: Enable (false ) or disable (true) cache retransmission support. The default value **false** indicates that the cache retransmission support is enabled.
  - **capacity queue\_capacity**: Specify the number of packets that this queue holds.




---

**Note** Ensure that there is sufficient memory when configuring higher capacity queues.

---

- **count queue\_count**: Specify the number of N4 queues to be created. Each queue is associated or dedicated to an UPF. For example, if the count is 2, two N4 queues are created and two UPs can be connected.
- **outbound { true | false }**: Enable (true) or disable (false) queue support for outbound messages. Default value: **false**.




---

**Note** Outbound flow control for BNG is not supported.

---

- **rate-limit per\_second**: Specify the rate limit for each queue, that is, when packets are dequeued. The rate limit is defined in seconds.
- **threshold threshold**: Specify the queue size before packets are dropped.

**Example**

The following is an example configuration.

```

endpoint udp-proxy
  replicas 1
  nodes 2
  vip-ip 201.201.201.51
  vip-ipv6 2001::10:1:39.191
  interface n4
    sla response 150000
    dispatcher
      count 1
      capacity 500000
      outbound true
      rate-limit 300
      cache false
      threshold 950000
    flowctrl-group group1
      capacity 1000
      rate-limit 100
  
```

```
exit
exit
```

## Configuring Overload Control for Message Types

To configure overload control for all message types, use the following commands:

```
config
    overload-control msg-type { all | lcpkeepalive | pfckeepalive |
pfcprresponse | sessionreport }
        msg-priority msg_priority | rate-limit rate_value | queue-size queue_size
    | reject-threshold reject_threshold | pending-request pending_request |
discard-behavior { drop | true }
    commit
```

### NOTES:

- **overload-control msg-type { all | lcpkeepalive | pfckeepalive | pfcprresponse | sessionreport }:** Configure overload control for the specified message type.
- **msg-priority *msg\_priority*:** Specify the message priority. This keyword is not applicable in the BNG context.
- **rate-limit *rate\_value*:** Specify the rate limit for each queue, that is, when packets are dequeued. The rate limit is defined in seconds.
- **queue-size *queue\_size*:** Specify the size of the queue to be created.
- **reject-threshold *threshold\_limit*:** Specify the percentage of the pending-request value.
- **pending-request *pending\_request*:** Specify the number of packets present in the queue at any time.
- **discard-behavior { drop | true }:** Specify whether to drop or process the packets. Default value: **drop**.

### Example

The following is a configuration example.

```
overload-control msg-type all
    rate-limit 13000 queue-size 200000 reject-threshold 95 pending-request 200000
exit
overload-control msg-type lcpkeepalive
    rate-limit 1100 queue-size 25000 reject-threshold 95 pending-request 25000
exit
overload-control msg-type sessionreport
    rate-limit 1000 queue-size 25000 reject-threshold 95 pending-request 25000
exit
overload-control msg-type pfckeepalive
    rate-limit 100 queue-size 1000 reject-threshold 95 pending-request 1000
exit
overload-control msg-type pfcprresponse
    rate-limit 4000 queue-size 25000 reject-threshold 95 pending-request 25000
exit
exit
```

