



## Configuring Quota Server Support

---

The CSG2 uses quota servers to return billing quota values for subscribers. The quota server interfaces with the billing system balance manager to reserve credit. The quota server then translates the reserved credit for the subscriber into quota based on the business and rating rules for multiple subscriber services on the CSG2.

For each CSG2 content billing service, the CSG2 downloads a separate quota, and deducts from that quota. Quotas are specified in units called *quadrans*. A quadran is a generic unit whose “value” is defined by each quota server. A quadran can represent, for example, a click for a per-click service (for example, an HTTP request), or a byte for a per-volume service. The value of a quadran is transparent to the CSG2; the CSG2 simply requests and downloads quadrans as needed from quota servers.

The CSG2 provides the following features for the quota server:

- [Configuring the Quota Server Local Port, page 4-2](#)
- [Configuring a Quota Server, page 4-2](#)
- [Configuring the Quota Server Keepalive Time, page 4-3](#)
- [Configuring the Quota Server GTP' Message Buffer, page 4-3](#)
- [Configuring the Quota Server Retransmit Time, page 4-3](#)
- [Configuring the Quota Server Retry Number, page 4-4](#)
- [Configuring the Quota Server Window Size, page 4-4](#)
- [Configuring Quota Server Load Sharing, page 4-5](#)
- [Reassigning Subscribers to a New Quota Server, page 4-5](#)
- [Sending User Profile Requests to Quota Servers, page 4-6](#)
- [Quota Push, page 4-6](#)
- [Replacing Quota Balance, page 4-6](#)
- [Delaying Quota Reauthorization, page 4-7](#)
- [Asynchronous Quota Return, page 4-7](#)
- [Reporting the Billing Plan ID to the Quota Server, page 4-7](#)
- [Pricing by Quota Server Configuration Example, page 4-8](#)
- [Differentiating Prices Configuration Example, page 4-8](#)
- [Reducing the Number of Services Configuration Example, page 4-9](#)

## Configuring the Quota Server Local Port

The first step when configuring CSG2 support for the quota server is to configure the local port on which the CSG2 is to communicate with the quota server.

For prepaid billing, you must configure at least one quota server local port, the local port on which the CSG2 is to communicate with quota servers.

To configure a local port for quota servers, enter the following command in global configuration mode:

Command	Purpose
<code>csg2(config)# ip csg quota-server local-port port-number</code>	Configures the local port on which the CSG2 communicates with quota servers.  The quota server local port number must be different from the BMA local port number and from the PSD local port number (configured with the <code>ip csg bma local-port</code> command and the <code>ip csg psd local-port</code> command, respectively).

## Configuring a Quota Server

For prepaid billing, you must configure at least one quota server.

You can configure up to 32 quota servers. Each quota server must have a unique priority and a unique IP address (or a unique IP address-VRF name combination, if VRF is configured).

When you configure a quota server, make sure that its IP address and port number match on both the active CSG2 and the standby CSG2.

The CSG2 differentiates quota servers on the basis of their IP addresses and port numbers. You can configure multiple quota servers with the same IP address, but the CSG2 does not support nodealive or redirect for multiple quota servers with the same IP address.

You can also enable the quota server for eGGSN.

If you have enabled interface awareness, you can also associate a VLAN's Virtual Routing and Forwarding (VRF) table name with the quota server.

To configure a quota server, enter the following command in global configuration mode:

Command	Purpose
<code>csg2(config)# ip csg quota-server [vrf vrf-name] ipv4-address port-number {priority   eggnsn}</code>	Configures CSG2 quota servers.  To enable the quota server for eGGSN, specify the <code>eggnsn</code> keyword.  <b>Note</b> The VRF table name is defined using the <code>vrf definition</code> command in global configuration mode. VRF table names supported by the CSG2 can be from 1 to 32 characters long, and can include uppercase or lowercase letters, numbers, and any special characters.  The CSG2 does not support the use of the word <code>forwarding</code> as a valid VRF name.

## Configuring the Quota Server Keepalive Time

By default, the CSG2 sends keepalive messages to the quota servers once every 60 seconds. That setting is sufficient in most environments, but the CSG2 also allows you to change the time between keepalive messages, if necessary.



### Note

We recommend that you change the keepalive time interval only when directed to do so by Cisco Technical Assistance Center (TAC) engineers. In most environments, the default value is the most appropriate setting.

To change the keepalive timer for the quota servers, enter the following command in global configuration mode:

Command	Purpose
<code>csg2(config)# ip csg quota-server keepalive number-of-seconds</code>	Defines the quota server keepalive time interval for the CSG2.

## Configuring the Quota Server GTP' Message Buffer

By default, the CSG2 can buffer up to 10,000 general packet radio service (GPRS) tunneling protocol prime (GTP') messages for the quota servers. That setting is sufficient in most environments, but the CSG2 also allows you to change the quota server GTP' message buffer, if necessary.



### Note

We recommend that you change the number of GTP' messages that can be buffered only when directed to do so by Cisco Technical Assistance Center (TAC) engineers. In most environments, the default value is the most appropriate setting.

To change the maximum number of GTP' messages, enter the following command in global configuration mode:

Command	Purpose
<code>csg2(config)# ip csg quota-server messages number</code>	Specifies the maximum number of GTP' messages that the CSG2 can buffer for all quota servers.

## Configuring the Quota Server Retransmit Time

By default, the CSG2 retransmits packets to a quota server once every four seconds. That setting is sufficient in most environments, but the CSG2 also allows you to change the time between retransmits, if necessary.



### Note

We recommend that you change the retransmit time interval only when directed to do so by Cisco Technical Assistance Center (TAC) engineers. In most environments, the default value is the most appropriate setting.

To change the quota server retransmit time interval for the CSG2, enter the following command in global configuration mode:

Command	Purpose
<code>csg2(config)# ip csg quota-server retransmit number-of-seconds</code>	Defines the quota server retransmit time interval for the CSG2.

## Configuring the Quota Server Retry Number

By default, the CSG2 retries communication with a quota server three times before determining that the link has failed. That setting is sufficient in most environments, but the CSG2 also allows you to change the number of retries, if necessary.



### Note

We recommend that you change the number of retries allowed only when directed to do so by Cisco Technical Assistance Center (TAC) engineers. In most environments, the default value is the most appropriate setting.

To change the maximum number of quota server retries allowed before the CSG2 determines that the link has failed, enter the following command in global configuration mode:

Command	Purpose
<code>csg2(config)# ip csg quota-server retries number-of-retries</code>	Defines the maximum number of quota server retries allowed before the CSG2 determines that the link has failed.

## Configuring the Quota Server Window Size

By default, the CSG2 sets the maximum quota server transmit window size to 128 packets, and sets the minimum quota server transmit window size automatically. Those settings are sufficient in most environments, but the CSG2 also allows you to change the maximum and minimum quota server transmit window sizes, if necessary.



### Note

We recommend that you change the transmit window size only when directed to do so by Cisco Technical Assistance Center (TAC) engineers. In most environments, the default value is the most appropriate setting.

To define the quota server transmit window size for the CSG2, enter the following command in global configuration mode:

Command	Purpose
<code>csg2(config)# ip csg quota-server window {max window-size   min window-size   min auto}</code>	Defines the quota server transmit window size for the CSG2.

## Configuring Quota Server Load Sharing

The CSG2 allows load sharing among quota servers. This support is useful in environments in which the number of quota transactions sent by the CSG2 could overwhelm a single quota server. Multiple quota servers can be simultaneously active, and the CSG2 assigns a quota server to each subscriber. All quota transactions for the subscriber are handled by the same quota server.

If a quota server fails, all subscribers associated with that quota server are distributed among the other active other quota servers.



### Note

The CSG2 differentiates quota servers on the basis of their IPv4 addresses and port numbers. You can configure multiple quota servers with the same IPv4 address, but the CSG2 does not support nodealive or redirect for multiple quota servers with the same IPv4 address.

The CSG2 creates a sticky object to ensure that all the quota transactions for a subscriber are sent to the same quota server. If the user ID is not available (for example, if the internal table is too small to hold all user ID entries, or if the CSG2 cannot access the user ID database), the CSG2 creates a sticky object for the subscriber IP address. In addition to activating multiple quota servers, the CSG2 allows you to specify the time to wait before it deletes inactive sticky objects.

To configure quota server load sharing, enter the following command in global configuration mode:

Command	Purpose
<code>csg2(config)# ip csg quota-server activate <i>number</i></code>	<p>Activates one or more quota servers.</p> <p>You do not need to use this command to activate a quota server that is enabled for eGGSN (that is, a quota server that is configured with the <b>eggsn</b> option on the <b>ip csg quota-server</b> command). A quota server that is enabled for eGGSN activates as soon as it is configured.</p>

## Reassigning Subscribers to a New Quota Server

If a quota server fails, you can reassign the subscribers to a different quota server.

To reassign subscribers to a different quota server, enter the following command in global configuration mode:

Command	Purpose
<code>csg2(config)# ip csg quota-server reassign</code>	<p>Reassigns subscribers to a different CSG2 quota server after a failure.</p> <p><b>Note</b> This command is not supported for quota servers that are enabled for eGGSN (that is, quota servers that are configured with the <b>eggsn</b> option on the <b>ip csg quota-server</b> command). If a quota server that is enabled for eGGSN fails, it does not fail over to the next available quota server.</p>

## Sending User Profile Requests to Quota Servers

By default, the CSG2 can send user profile requests to quota servers as needed. However, you can configure the CSG2 to prevent it from sending those user profile requests.

To enable the CSG2 to send user profile requests (the default setting), enter the following command in global configuration mode:

Command	Purpose
<code>csg2(config)# ip csg quota-server user-profile</code>	Enables the CSG2 to send user profile requests to quota servers.

To prevent the CSG2 from sending user profile requests, enter the **no** form of the command in global configuration mode:

Command	Purpose
<code>csg2(config)# no ip csg quota-server user-profile</code>	Prevents the CSG2 from sending user profile requests to quota servers.

## Quota Push

This feature enables operators to “push” quota for a service instance associated with an individual subscriber. This enables quota servers to provide quota for a subscriber or service before traffic from that subscriber or service reaches the CSG2. This eliminates the delay that can occur when quota is obtained through a service authorization request and response. A sophisticated quota server could also use quota push for better control of quota levels during active sessions.

The CSG2 accepts a quota push for a subscriber at any point after the subscriber and billing plan are known to the CSG2 (that is, when a CSG2 User Table element exists for the subscriber). For example, the CSG2 accepts a quota push after receiving an accounting start but does not require an existing service for the subscriber (one is created). The CSG2 does not begin charging against quota until traffic begins to arrive and a session is created. Zero-quota might be granted so that cause code and authorization actions can be set (for example, for a free service). A quota download message is sent to the BMA in response to receiving a quota push.

The service idle timer starts whenever quota is pushed, in case the expected traffic never arrives.

The CSG2 rejects the Quota Push message if the “Replace current balance” flag is not set in the Granted Quadrans TLV.

There are no commands required to enable quota push.

## Replacing Quota Balance

By default, when the CSG2 receives a quota grant from the quota server, the CSG2 adds the granted quota to the current balance for a subscriber’s service. Quota balance replacement enables the quota server to instruct the CSG2 to replace the current quota balance with the amount of granted quota for a subscriber’s service. If the replacement grant is provided in a Service Authorization Response, the CSG2 subtracts the amount of quota used since the Service Reauthorization Requests from the granted quota before replacing the balance.

There are no commands required to enable quota balance replacement.

## Delaying Quota Reauthorization

The CSG2 accepts the Reauthorization Delay TLV, which specifies the number of seconds the CSG2 delays its next reauthorization request to the quota server for the service specified in the message. This TLV also specifies the action the CSG2 is to take for the service between the time the message is received and the next reauthorization:

- **Wait**—The CSG2 keeps transactions in a pending state during the delay period. In pending state, the CSG2 maintains the transaction state but drops packets.
- **Deny**—The CSG2 drops new transactions during the delay period. Existing transactions are dropped if quota expires during the delay period. The CSG2 does not maintain the session state; the subscriber must open a new connection after the delay period.

**Note**

For HTTP pipelining, dropping new transactions can also affect existing transactions if they share the same TCP connection.

Quota servers can use delayed quota reauthorization to deny subscribers access to CSG2 categories without having to continually deny authorization requests (that is, for blacklisting services). To do so, the quota server sends a grant of 0 quadran in a Service Authorization Response, Quota Push Request, or Service Verification Response message, with a long reauthorization delay timer (0xFFFFFFFF), a Deny action, and a cause code of 0x03.

There are no commands required to enable delayed quota reauthorization.

## Asynchronous Quota Return

The Asynchronous Quota Return feature allows the quota server to request the CSG2 to return quota for a defined subscriber and service, and to send a Quota Return. The quota reserved for ongoing transactions is recalled from the transactions and included in the quota returned in the Quota Return message.

Because all of the quota is returned, there is no longer any reserved quota to process ongoing pending transactions. Packets received for pending transactions are dropped. However, time-based billing holds back 5 seconds of quota, so transactions can proceed while returning quota for time-based billing.

There are no commands required to enable asynchronous quota return.

## Reporting the Billing Plan ID to the Quota Server

The CSG2 reports the billing plan identifier in messages to the quota server. The CSG2 also reports the billing plan ID in billing records.

There are no commands required to enable billing plan ID reporting.

## Pricing by Quota Server Configuration Example

The following example shows a CSG2 configuration in which the quota server performs all pricing. In this example:

- Assume that Subscriber X has \$10.00 in his account.
- There are two types of content:
  - C1—This content is billed per object (for example, URL GET); each object costs \$0.01.
  - C2—This content is billed per byte; each kilobyte costs \$0.01.
- The quota server controls each object transaction for content C1.
- The quota server controls all the pricing.

```
ip csg content C1
  policy P1
  inservice
!
ip csg content C2
  policy P2
  inservice
!
ip csg service PERCLICK
  basis fixed
  content C1 policy P1
!
ip csg service PERBYTE
  basis byte ip exclude mms
  content C2 policy P2
!
ip csg billing REGULAR
  service PERCLICK
  service PERBYTE
```

When Subscriber X, with a subscription to billing plan REGULAR, tries to access content that matches C1, the CSG2 tries to download quota for Subscriber X for service PERCLICK.

The quota server borrows money from Subscriber X's \$10.00, and returns some quadrans to the CSG2. Each quadran is good for one object download, or one click. If the quota server is configured for the CSG2 to query for each click, it can choose to send just one quadran at a time, so that the CSG2 queries the quota server each time. However, if the quota server is configured to grant \$2.00 worth of quadrans to the CSG2 in one shot, it can send 200 quadrans to the CSG2, which the CSG2 keeps using for Subscriber X's access to C1.

When Subscriber X tries to access content that matches C2, the CSG2 makes another request to the quota server to get Subscriber X's quota for C2. C2 is billed per IP byte. The quota server borrows another \$5.00 from Subscriber X's account, and sends 500000 quadrans to the CSG2. As Subscriber X continues to access C2, his traffic is metered for volume. For each byte, the CSG2 deducts one quadran.

## Differentiating Prices Configuration Example

The following example extends the previous example by adding a content type that is priced differently. In this example:

- Assume that Subscriber X has \$10.00 in his account.
- There are three types of content:

- C1—This content is billed per \*.jpg file, where each JPG file costs \$0.01.
- C2—This content is billed per byte, where each kilobyte costs \$0.01.
- C3—This content is billed per \*.mp3 file, where each MP3 file costs \$0.05.
- The quota server controls each object transaction for content C1.
- The quota server controls all the pricing.

This configuration requires an additional service type, MP3, which allows the quota server to price object downloads (clicks) differently for MP3 files.

```
ip csg content C1
  policy P1
  inservice
!
ip csg content C2
  policy P2
  inservice
!
ip csg content MP3
  policy P1
  inservice
!
ip csg service PERCLICK
  basis fixed
  content C1 policy P1
!
ip csg service PERBYTE
  basis byte ip
  content C2 policy P2
!
ip csg service MP3
  basis fixed
  content C1 policy P1
!
ip csg billing REGULAR
  service PERCLICK
  service PERBYTE
  service MP3
```

When Subscriber X tries to download an MP3 file (that is, a file that matches content type MP3), the CSG2 requests the MP3 quota for Subscriber X. Each download of an MP3 file costs \$0.05, so the quota server borrows \$1.00 from Subscriber X's account, and returns 20 quadrans to the CSG2 for service MP3. The CSG2 can use the quadrans for 20 downloads of MP3 files.

Alternatively, the quota server could send just one quadran, which is enough for only one transaction. This would force the CSG2 to ask for quota before each download of an MP3 file.

## Reducing the Number of Services Configuration Example

The [“Differentiating Prices Configuration Example”](#) section on page 4-8 shows that you can create a new service for one type of content and differentiate its billing from other types of content.

However, with each new service, the subscriber's quota fragments further, and traffic between the CSG2 and the quota server increases.

You can reduce traffic by specifying a symbolic weight on the CSG2. In the following example, each MP3 download (\$0.05) costs five times as much as each JPG download (\$0.01). By assigning a weight of 5 to MP3 downloads, you can keep both content C1 and content MP3 under service PERCLICK, thereby reducing the overall number of services and reducing the traffic between the CSG2 and the quota server.

```
ip csg content C1
  policy P1
  inservice
!
ip csg content C2
  policy P2
  inservice
!
ip csg content MP3
  policy P1
  inservice
!
ip csg service PERBYTE
  basis byte ip
  content C2 policy P2
!
ip csg service PERCLICK
  basis fixed
  content C1 policy P1
  content MP3 policy P1 weight 5
!
ip csg billing REGULAR
  service PERCLICK
  service PERBYTE
```

When the quota server borrows \$1.00 from Subscriber X's account and sends 100 quadrans for service PERCLICK, the CSG2 can use the quadrans for 100 JPG files, or for 20 MP3 files, or for a mix of the two content types.