CHAPTER 5

Configuring Service Support

A CSG2 content billing service is a component of a billing plan to which subscribers subscribe.

You can configure one or more content billing services for the CSG2. Each service represents a group of content that is billed the same way, such as billing per-click (or per-request) or billing per-IP byte, and that shares part of a subscriber's quota. Grouping content into one or more services enables you to separate, for example, a subscriber's prepaid quota for Internet browsing from his quota for e-mails.

For each 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 requests an additional quota grant when a subscriber's per-click quota falls below a specified percentage of the last quota grant, or when a subscriber's per-volume quota falls below a specified percentage of the last quota grant or 32 KB, whichever is greater.

For each service that a subscriber tries to access, the CSG2 maintains a separate logical accounting session. When a subscriber's quota is divided among multiple services, the CSG2 requests an additional quota grant for each service individually, based on its usage.

If a subscriber fails authorization for a service, but continues to send new requests for that service, the CSG2 waits a specified time before sending the quota server a reauthorization request for that subscriber. This ensures that the quota server is not inundated with reauthorization requests from unauthorized subscribers.

The CSG2 allows you to define a pool of up to 1024 services. You can authorize each subscriber for any number of services from that pool, but we recommend that the billing system not authorize each subscriber for more than 10 active services. Exceeding this guideline could lead to the following problems:

- The increase in the number of quota authorizations per subscriber can overload both the quota server and the CSG2.
- As the number of services for which a subscriber is actively authorized increases, the subscriber's quota becomes fragmented. Although the CSG2 allows the billing system to recall and redistribute the quota, so that the subscriber is not denied service because of quota fragmentation, the process increases overhead in both the quota server and the CSG2.

The CSG2 supports multiple protocols under a single service definition.

The CSG2 provides the following features for content billing services:

Configuring a Basic Content Billing Service

Each content billing service is associated with one or more contents and policies.

Multiple services can include the same content/policy pair, as long as the services are not associated with the same billing plan. They cannot be associated with the same billing plan because then the match of content/policy pair to service would not be unique.

To configure a content billing service, enter the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`csg2(config)# ip csg service service-name`	Configures a CSG2 content billing service, and enters CSG2 service configuration mode.
Step 2	`csg2(config-csg-service)# content content-name policy policy-name [weight weight-name]`	Configures a content and policy as a member of a CSG2 billing service, and optionally assigns a weight to the content.

Configuring the Billing Basis for a Service

The billing basis specifies how billing is to be charged:

- Per-click (fixed-cost) billing is charged at a fixed cost, which is deducted each time the first packet for a transaction hits a content-policy pair (that is, deducted for each request).
- Volume-based billing can be based on either the number of IP bytes or the number of TCP bytes.
- Duration-based billing can be based on either service duration time or connection duration time.

To configure the billing basis for a service, enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# basis {byte ip | byte tcp | fixed | second [connect | transaction]} [dual [byte ip | byte tcp | fixed | second transaction}]]`	Specifies the billing basis for a CSG2 content billing service. **Note** The **basis** command applies to prepaid and virtual prepaid billing only. It has no impact on postpaid billing. When changing the basis for a service, the content must be taken out of service.

To specify the activation mode for a CSG2 Connection Duration service, enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# activation [automatic | user-profile]`	Specifies the activation mode for a CSG2 Connection Duration service (that is, a service configured with **basis second connect**). • **automatic**—Activates the Connection Duration service, unless the billing profile indicates that no service is to be activated. • **user-profile**—Activates the Connection Duration service only if the billing profile specifies this service as the connect service. This is the default setting.

Specifying a Service Owner

The CSG2 enables you to specify an identifier or name for a CSG2 service owner to be used with fixed-record format. The owner is responsible for the content associated with the service.

The administrator who configures owner identification is responsible for its accuracy. Correct configuration requires that contents for this service, their policies, and any associated URL or header maps, identify all data transfers with this owner, and only data transfers with this owner.

To do so, enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# owner {id` *id* `| name` *name*`}`	Specifies an identifier or name for a CSG2 service owner.

Specifying a Service Class

The CSG2 enables you to specify a service class value to be used with fixed-record format. The class is opaque to the CSG2 and has meaning only for the administrator. It is reported as tariff-class in fixed-record format call detail records (CDRs).

To specify a service class, enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# class value`	Specifies a service class value.

Configuring a Service Idle Time

The CSG2 enables you to configure an idle timer for a service. The timer begins when there are no sessions. If the timer expires and the subscriber's quota for the service has not been used, the CSG2 assumes that the service is idle and sends a Service Stop to free up the resources.

For services configured with **basis second**, make sure the idle timeout value for the content configurations, set using the **idle** command in CSG2 content configuration mode, does not exceed the service idle timeout value, set using the **idle** command in CSG2 service configuration mode.

To configure a service idle timer, enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# idle duration`	(Optional) Specifies the minimum amount of time that the CSG2 maintains a service with no subscriber sessions.

Configuring Advice of Charge

Advice of Charge (AoC) is a function that enables a service provider to provide messaging and authorization prompts to its subscribers. The CSG2's support for AoC uses a quota server and a customer-provided notification server to host the actual messaging:

- The quota server is responsible for telling the CSG2 to block subscriber requests and redirect them to the notification server when the subscriber must make a decision to pay for the service. It is also responsible for telling the CSG2 to allow the subscriber request to flow when the subscriber has agreed to pay.
- The notification server is responsible for communicating fees to the subscriber and providing the option to pay. The subscriber's payment decision must be communicated from the notification server to the quota server.

The CSG2's role in the AoC process is to redirect subscriber data requests to the notification server. The CSG2 provides URL-redirect support for HTTP and wireless application protocol 1.x (WAP 1.x), for redirecting to the notification server. With URL-redirect, the notification server can be a standard web server, because the CSG2 does the redirection at the protocol level.

The CSG2 allows the redirection for AoC to be triggered once per service (when the first access to the service is made by the subscriber), or at the start of any new data transaction. The former is accomplished using the CSG2's service verification function, the latter using the CSG2's content authorization function. The URL can be pre-configured, or it can be provided dynamically by the quota server (the more flexible option). You can configure content authorization to request a pass/fail authorization for any transaction (for example, for individual SMTP e-mails), but the CSG2 does not honor redirect requests from the quota server in the middle of a TCP connection.

In general, the method by which the notification server communicates success or failure of the AoC to the quota server is outside the scope of the CSG2's role in the process. However, the CSG2 does provide some additional assists for URL-redirects that greatly ease the burden on the backend systems. For example, the CSG2 provides the ability to strip trailing tokens from a URL. Therefore, an HTTP-based notification server can be deployed such that it will append the results of the AoC to the subscriber's HTTP request when redirecting the subscriber to the final requested content. The CSG2 reports this URL, token and all, to the quota server on the next Content Authorization Request. If configured to do so, upon successfully receiving permission from the quota server to forward the flow, the CSG2 strips the token from the request so that the content server is not confused by the extra data.

You can instruct the CSG2 to obtain authorization from the quota server for each subscriber request for content.

The CSG2's support for AoC has the following restrictions:

- AoC is supported for all protocols except IMAP and POP3. However, AoC via content authorization and URL-redirect is supported for only HTTP, SIP, WAP 1.x, and WAP 2.0.
- AoC token-stripping is not supported for requests in which the URL is fragmented over more than one IP packet.
- AoC is not supported for Connection Duration services (that is, services configured with **basis second connect**).
- When performing AoC for a TCP connection carrying pipelined HTTP requests, the CSG2 responds with the redirect to the subscriber as soon as the quota server requests the redirect. This could result in the redirect arriving at the subscriber before responses for previous requests arrive, and the subscriber might associate the redirect with a different request in the pipeline.
- When a CSG2 prepaid service is configured for AoC, the weighting value for charging the content is not determined until the CSG2 processes the Content Authorization Response. For SMTP billing (**parse protocol smtp**), the CSG2 does not send the Content Authorization Request until it processes the SMTP DATA command. If the CSG2 does not process the SMTP DATA command for a session, then the CSG2 does not charge the session for volume and event billing.
- If a Content Authorization Request is queued to a quota server, and the quota server fails, the CSG2 reassigns the request to the standby quota server or to some other active quota server. If there is no standby quota server or other active quota server, the CSG2 completes the AoC request with action code 0 (Drop).

Note If the quota server queue is very long and very slow, the Interprocessor Communication (IPC) request for the AoC message might timeout on the Traffic Processor (TP). If this occurs, and if the timeout handler on the TP has already triggered, then the CSG2 might treat the response from the quota server as a failed message, dropping the packet.

This section contains the following information:

Enabling AoC URL-Rewriting

When AoC URL-rewriting is enabled, the CSG2 alerts the quota server of a new transaction, and allows it to direct the CSG2 to perform any of the following mutually exclusive actions:

- **DROP**: Drop all packets for this flow.
- **FORWARD**: Forward the flow without altering the destination (a weight might be specified).
- **REDIRECT-URL**: Redirect subscriber requests to the URL provided by the quota server. The CSG2 sends a Layer 7 redirect to the subscriber (for example, HTTP 302 response) that contains the redirect URL.

To enable AoC URL-rewriting, enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# aoc enable`	(Optional) Enables Advice of Charge (AoC) URL-rewriting for the CSG2.

Configuring an AoC Token

When direct communication is not possible between the quota server and the notification server, payment decision information can be shared indirectly by modifying the URL in the subscriber request. The notification server appends a string beginning with a token to the originally requested URL and sends it to the subscriber as part of a redirect reply after the subscriber agrees to pay. The CSG2 receives the subsequent GET request containing the rewritten URL and sends it to the quota server in a Content Authorization Request. The quota server recognizes the token string and understands that the subscriber has agreed to pay for the request. It responds to the CSG2 with a FORWARD action code in the Content Authorization Response. The CSG2 detects the token, creates a new GET request that contains the original URL (without the appended token and any characters following it), and sends the GET on behalf of the subscriber. The token must be known by the CSG2, the quota server, and the notification server. The token is administratively defined on the CSG2 by using the CLI. The token must be chosen carefully to ensure that it is present only in URLs that are rewritten by the notification server and not in other subscriber requests.

URL-rewriting allows a top-off server to append parameters to a URL in order to convey state information to the quota server during a Content Authorization Request. Whenever a Content Authorization Response contains the forward action code, and the URL contains the AoC confirmation token, the token and all trailing characters are removed from the URL before the request is forwarded to the server.

If the token uses the URL-escape format, the redirect URL to which the token is being matched must also use the URL-escape format.

The CSG2 supports AoC URL-rewriting for only HTTP and WAP 1.x.

If you have enabled AoC URL-rewriting, you can define a URL-rewriting token for AoC. To do so, enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# aoc confirm` *token*	(Optional) Configures a token for use in AoC URL-rewriting.

Configuring AoC URL-Appending

Whenever a Content Authorization Response contains a REDIRECT_URL action code for a WAP Content Authorization Request, the CSG2 can optionally append the originally requested URL to the one returned by the quota server.

For example, if the subscriber requests the following URL:

http://www.redirect_url.com/home.wml

and the quota server returns the following URL in a REDIRECT_URL Content Authorization Response:

http://www.redirect_url. .com/charges.wml

then the CSG2 sends the following URL as part of a redirect message to the subscriber:

http://www.redirect_url. .com/charges.wml?www.redirect_url. .com/home.wml

The default behavior is to pass the redirect URL to the subscriber as specified by the quota server without modification.

The CSG2 supports AoC URL-appending for WAP 1.x and WAP 2.0 only.

If you have enabled AoC URL-rewriting, you can enable AoC URL-appending for AoC. To do so, enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# aoc append url`	(Optional) Specifies that the CSG2 is to append the original URL to the redirect URL sent by the quota server on a Content Authorization REDIRECT_URL response for use in AoC URL-rewriting.

Redirect Flexibility

A quota server can request a redirect for multiple reasons (top-up, "sorry" indication, login request). The CSG2 allows the quota server to return the IP address and port number for each redirect. Thus, a different port number, or even a different network, can be used for every reason that the quota server might request the redirect. The CSG2 stores the most recent redirect address and port number for each service under each subscriber profile, and uses that address and port instead of the globally defined default.

Configuring Service Verification

Service verification is a capability similar to Advice of Charge (AoC), which is provided the first time a subscriber accesses a service using HTTP or WAP 1.x. A Service Verify Request quota management message supplies the quota server with content from the subscriber request (the URL, header information, subscriber agent, and so on). The quota server responds with a Service Verification Response that includes a decision to redirect the request to a notification server, to forward it, or to drop it.

Service verification provides the same URL-rewriting capabilities that are provided by AoC. An administrator uses the command-line interface (CLI) to define the service confirmation token that is used in URL-rewriting.

As long as service verification is enabled, sessions of any type for this subscriber do not trigger service reauthorization requests. Service reauthorization resumes for the subscriber when service verification is disabled.

Service verification supports forward, redirect-URL, and drop authorization action codes sent in a Service Verification Response. Service verification also supports optional downloading of quota for a subscriber in a Service Verification Response. The CSG2 sends service verification requests even when no quota is supplied in the Service Verification Response, if the Service Authorization Response contains the cause TLV with value 0x04 (subscriber low on quota, but service access is permitted). Quota Download call detail records (CDRs) are sent to the BMA, as appropriate, whenever the quota server supplies quota in a Service Verification Response.

Service verification can be used in conjunction with existing AoC functionality.

Service verification is supported only for HTTP and WAP 1.x.

This section contains the following information:

Enabling Service Verification URL-Rewriting

When service verification URL-rewriting is enabled, the CSG2 alerts the quota server of a new transaction, and allows it to direct the CSG2 to perform any of the following mutually exclusive actions:

- **DROP**: Drop all packets for this flow.
- **FORWARD**: Forward the flow without altering the destination (a weight might be specified).
- **REDIRECT-URL**: Redirect subscriber requests to the URL provided by the quota server. The CSG2 sends a Layer 7 redirect to the subscriber (for example, HTTP 302 response) that contains the redirect URL.

To enable service verification, enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# verify enable`	(Optional) Enables service verification for the CSG2. Service verification is disabled when you enter the **no** form of this command, or when the quota server sends a Service Verify Tag-Length-Value (TLV) in a Service Authorization Response or Service Verification Response.

Configuring a Service Verification Token

URL-rewriting allows a top-off server to append parameters to a URL in order to convey state information to the quota server during a Service Verification Request. Whenever a Service Verification Response contains the forward action code, and the URL contains the verify confirmation token, the token and all trailing characters are removed from the URL before the request is forwarded to the server.

If the token uses the URL-escape format, the redirect URL to which the token is being matched must also use the URL-escape format.

The CSG2 supports URL-rewriting for HTTP, WAP 1.x, and WAP 2.0.

If you have enabled service verification URL-rewriting, you can define a URL-rewriting token for service verification. To do so, enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# verify confirm` *token*	(Optional) Configures a token for use in CSG2 service verification URL-rewriting.

Enabling Service-Level CDR Summarization

By default, the CSG2 generates billing records for each transaction. This large number of records might overwhelm the charging gateway (CG) or the collector. To prevent this situation, the CSG2 can summarize CDRs at the service level, instead of at the transaction level.

For example, if a subscriber accesses the open Internet service, and the data is billed solely on the basis of volume, it is of little use to generate records for each HTTP transaction. With service-level CDR summarization enabled, the CSG2 generates only consolidated records on service-level usage. Information from individual events is not reported (for example, no URLs).

The CSG2 uses the Service Usage - variable format (0x0040) CDR for service-level CDR summarization.

Service-level CDRs differ from Service Stop CDRs as follows:

- The CSG2 sends a Service Stop message to the quota server when a prepaid service instance ends. At the same time, the CSG2 sends a companion CDR, the Service Stop Notification CDR (0x11), to the BMA.
- The CSG2 sends a service-level CDR to the BMA when a service instance ends, or, if configured, when a volume- or time-based threshold is met. The CSG2 sends service-level CDRs only to the BMA, and only if the service is configured for service-level CDRs.

The CSG2 can generate intermediate service-level CDRs for volume-based billing or for time-based billing. Cause codes 0x0A and 0x0B are used for service-level CDRs generated for volume- or time-based billing.

The CSG2 supports the following protocols in both fixed and variable format: FTP, IP, HTTP, IMAP, POP3, RTSP, SMTP, and WAP 1.x. (POP3 and IMAP are supported in postpaid mode only.)

To configure service-level CDR summarization, enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)#` **`records granularity service`** {**`bytes`** *`bytes`* | **`seconds`** *`seconds`* | **`bytes`** *`bytes`* **`seconds`** *`seconds`*}	(Optional) Specifies that the CSG2 is to generate summarized, service-level CDRs.

Note To enable service-level CDR summarization in postpaid mode, you must also specify that the associated billing plan is postpaid by using the **mode postpaid** command in CSG2 billing configuration mode.

Service-level CDRs are generated only for subscribers with entries in the CSG2 User Table entry. If a subscriber does not have an entry in the User Table, the CSG2 generates transaction-level CDRs.

If there are no quota servers configured on the CSG2, and you want to use service-level CDRs in a postpaid environment (that is, all users are postpaid), you can configure a single postpaid billing plan and assign all users to that billing plan. In the following example, all postpaid users are automatically assigned to billing plan EVERYBODY:

```
ip csg map SPORTS
  match url http://www.nhl.com/*
!
ip csg map MOVIES
  match url http://www.hollywood.com/*
!
```

```
ip csg policy SPORTS
  map SPORTS
!
ip csg policy MOVIES
  map MOVIES
!
ip csg content HTTP
  ip any tcp 80
  policy SPORTS
  policy MOVIES
  inservice
!
ip csg service SPORTS
  content HTTP policy SPORTS
  records granularity service byte 128000
!
ip csg service MOVIES
  content HTTP policy MOVIES
  records granularity service bytes 128000
!
ip csg billing EVERYBODY
  mode postpaid
  service SPORTS
  service MOVIES
```

Configuring Passthrough Mode and the Default Quota

For prepaid subscribers, sessions are blocked when a quota server is not available for authorization grant of quota. In passthrough mode, the CSG2 grants quota for services and their sessions when a quota server is not available. The CSG2 allows all traffic to pass, and CDRs are flagged for special consideration by the BMA.

For each service for which you want to use passthrough mode, you must enable it by entering the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# passthrough` *quota-grant*	(Optional) Enables passthrough mode for a CSG2 service.

You also use this command to specify the size of each quota grant (the default quota) to assign to a service. When passthrough mode is enabled for a service, and a session for a service needs quota, and no quota server is active, the CSG2 grants the service the amount of quadrans specified on the **passthrough** command. (There are three types of quadrans: **basis byte** for volume-based billing, **basis fixed** for event-based billing, and **basis second** for duration-based billing.) The CSG2 continues to grant quota as long as a quota server is inactive.

When the service becomes idle, the CSG2 generates and stores a Service Stop Request message, containing the total usage for this instance of the service. When a quota server becomes active, the CSG2 forwards all stored Service Stop Request messages to the quota server.

This section contains the following information about passthrough mode:

- Flagging of Messages, page 5-12
- User Profile Requests, page 5-12
- Quota Server Recovery, page 5-12

Flagging of Messages

To facilitate billing recovery, some messages to the quota server and the BMA include a QuotaServerFlags TLV. The CSG2 adds this TLV whenever it grants a passthrough mode quota to a service.

User Profile Requests

When the CSG2 learns of new subscribers, it typically sends a User Profile Request to an active quota server. This enables the CSG2 to learn the billing plan to use for each subscriber. If the quota server returns a NULL billing plan, this indicates that a subscriber is postpaid.

Quota Server Recovery

When a quota server becomes active, the CSG2 forwards stored Service Stop Requests to it. Additional actions taken by the CSG2 depend on subscriber traffic.

Prepaid subscribers might have some services that were granted quota in passthrough mode. For those services, when quota runs low, the CSG2 sends a Service Reauthorization Request to the quota server, flagging the request with the QuotaServerFlags TLV. The usage TLV and remaining TLV contain the sum total of quota granted to the service since it began. This total might be a combination of quota granted by the quota server before the failure and quota granted by the CSG2 in passthrough mode. The requested quadrans TLV contains a request for an additional quota amount.

When the quota server responds to a Service Stop or a Service Reauthorization Request, the CSG2 moves the service out of passthrough mode. If the quota server denies quota when it sends a Service Authorization Response message, the CSG2 blocks the traffic. The CSG2 also flags CDRs generated by traffic for these services, which received passthrough mode quota grants, with QuotaServerFlags TLVs, until a Service Stop Request is sent. That is, once a service is granted a passthrough mode quota, the CSG2 flags all CDRs for that serviced, up to and including the Service Stop. This applies only to prepaid subscribers. Postpaid subscribers CDRs are never flagged.

Configuring Metering

The CSG2 enables you to control some aspects of metering.

This section contains the following information:

Configuring an Initial Quota for Metering

TheCSG2 enables you to specify the initial quota, in quadrans, debited from the balance at the beginning of a service when the service is configured for Service Duration Billing. The debit occurs when the CSG2 grants the first network access for a session that has been mapped to the service. The initial value is not rounded up to the nearest increment value.

Specifying the initial quota allows you to apply "connection setup charges" to a service.

To specify the initial quota, enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# meter initial value`	(Optional) Specifies the initial quota debited by the CSG2 from the balance at the beginning of a service when the service is configured for Service Duration Billing.

Configuring a Minimum Quota for Metering

The CSG2enables you to specify the minimum number of quadrans debited for a service or session, excluding the value in **meter initial**.

For example, to force the CSG2 to debit 90 quadrans when less than 90 quadrans of network usage were used for the service, specify **meter minimum 90**. If the initial value is 20 quadrans and the minimum is 90 quadrans, then the minimum total charge is 110 quadrans. The minimum value is applied only if at least 1 session is granted network access for the service.

If **basis second** is configured for the service, the usage is rounded up to the minimum value when the Service Stop is sent. For a minimum value of 90, 150 seconds of network usage is not rounded up for the purpose of calculating usage in the Service Stop, but, for example, 63 seconds of network usage is rounded up to 90 quadrans.

Note The rounding-up of network usage is not reflected in calculations for the Usage Tag-Length-Value (TLV) in Service Reauthorization Requests.

To specify the minimum number of quadrans debited by the CSG2 for a service or session, enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# meter minimum value`	(Optional) Specifies the minimum number of quadrans debited for a service or session.

Configuring a Debit Increment for Metering

The CSG2enables you to specify the increment, in seconds, for debiting quota upon completion of a service configured for Service Duration Billing. For example, to enable the CSG2 to charge quota per minute instead of per second, specify **meter increment 60**.

If **basis second** is configured for the service, the network usage (usage excluding the initial charge) is rounded up to the nearest integer multiple of the increment value when the Service Stop is sent. For an increment value of 60, the CSG2 does not round up 120 seconds of network usage; however, the CSG2 does round up, say, 163 seconds of network usage to 180 quadrans before it calculates total usage for reporting in the Service Stop.

Note The rounding-up of network usage is not reflected in calculations for the Usage Tag-Length-Value (TLV) in Service Reauthorization Requests.

The increment value is considered when determining whether sufficient quota exists for granting network access for a session. For instance, if the increment is 60, the network usage is 50, and the balance is 10, network access is permitted. However, if the increment is 60, the network usage is 70, and the balance is 10, network access is not permitted because the balance is not sufficient to satisfy the entire increment (that is, a minimum of 1 minute of quota would be required to allow access for a portion of the minute).

To specify the debiting increment, enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# meter increment value`	(Optional) Specifies the increments for debiting quota by the CSG2 upon completion of a service configured for Service Duration Billing.

Excluding RTSP PAUSE from Metering

The CSG2 monitors the RTSP control session between the RTSP subscriber and network and scans for PAUSE and PLAY methods. When a PAUSE method is detected, the CSG2 initiates an event to inform the prepaid service to stop charging for duration-based billing. Then, when a PLAY method is detected, the CSG2 initiates an event to inform the prepaid service to resume the charging for duration-based billing. The event corresponding to the PLAY is only necessary when the CSG2 is in the PAUSE state.

When configuring RTSP PAUSE support, keep the following considerations in mind:

- RTSP pause is supported only for duration-based billing (**basis second**).
- Duration-based billing applies only to a billing service, and RTSP might not be the only application that is operating over a given billing service. Therefore, the suspension of billing for the PAUSE period applies only if there are no other applications operating over the same billing service.
- Both last billable and intermediate idles for RTSP are excluded from duration-based billing if RTSP PAUSE support is configured.
- RTSP PAUSE support applies only to classical RTSP transport, in which the stream data is transmitted over a separate User Datagram Protocol (UDP) connection. RTSP PAUSE support does not apply to the TCP or HTTP interleaved transport modes.
- RTSP pause support is independent of the UDP content idle timer value and of the RTSP service idle timer. RTSP pause support does not stop the UDP content idle timer.
- RTSP PAUSE support applies only to charges for prepaid quota usage (quadrans TLV).
- The RTSP PAUSE time is not reflected in any CDRs.

To exclude the RTSP PAUSE time from the duration-based billing calculation, enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# meter exclude pause rtsp`	(Optional) Excludes the RTSP PAUSE time from the CSG2 usage calculation duration billing.

Including IMAP Bytes in Metering

The CSG2 provides transaction support for IMAP. The CSG2 defines an IMAP transaction as a tagged response from an IMAP server that contains TEXT. TEXT is the part of the e-mail that follows the envelope; the presence of TEXT results in a classification of BODY. The CSG2 includes IMAP transaction counts in the Completed Transactions TLV. The CSG2 does not include any envelope information in the IMAP transaction CDRs.

For requests and responses that are not transactions (they do not contain TEXT), the CSG2 accumulates the bytes and includes them in the next transaction. When the IMAP session ends, the CSG2 reports any remaining bytes.

Consider the following simple example of an IMAP transaction with BODY:

Subscriber request: 1 FETCH 5 BODY[]
Network response: * 5 FETCH (BODY[]{55}cr-lf-55-bytes-of-e-mail-followed-by-cr-lf)cr-lf
1 OK FETCH COMPLETE

The CSG2 handles this request and response as follows:

Step 1 The subscriber request is tagged **1**. The CSG2 parses the request and increments the body up byte counts, because the request was for a **BODY[]**.

Step 2 The CSG2 parses the untagged response from the network and notes that it contains TEXT (**BODY[]**).

Step 3 The CSG2 parses the tagged response **1 OK FETCH COMPLETE**, which means this is an IMAP transaction (a tagged response that contains TEXT).

Here is a more complicated example:

Subscriber request: 8 FETCH 1:100 BODY[]<0.5>
Network response: * 1 FETCH (BODY[]<0> “. . . . .”)cr-lf
*** 2 FETCH (BODY[]<0> “. . . . .”)cr-lf**
*** 3 FETCH (BODY[]<0> “. . . . .”)cr-lf**
*** 4 FETCH (BODY[]<0> “. . . . .”)cr-lf**
...
*** 100 FETCH (BODY[]<0> “. . . . .”)cr-lf**
8 OK FETCH COMPLETE

The CSG2 handles this request and response as follows:

Step 1 The subscriber request is tagged **8**. The CSG2 parses the request and increments the body up byte counts, because the request was for a **BODY[]**.

Step 2 The network sends 100 untagged responses which the CSG2 parses, noting that the response contains TEXT (**BODY[]**).

Step 3 The CSG2 parses the tagged response **8 OK FETCH COMPLETE**, which means this is an IMAP transaction (a tagged response that contains TEXT). The CDR reports 100 **BODY** fetches, the request bytes are allocated to body up, and the response bytes are allocated to body down.

The CSG2 categorizes bytes as BODY, HEADER, and OTHER, determined as follows:

- BODY—The bytes are classified as BODY if a fetch request or response is encountered for one of the following specifications (including any appended "<>" subset variants):
  - BODY[]
  - BODY[#]
  - BODY[TEXT]
  - BODY[#.TEXT]
  - BODY.PEEK[]
  - BODY.PEEK[#]
  - BODY.PEEK[TEXT]
  - BODY.PEEK[#.TEXT]
  - RFC822
  - RFC822.TEXT
- HEADER—If the bytes cannot be classified as BODY, then they are classified as HEADER if a fetch request or response is encountered for one of the following specifications (including any appended "<>" subset variants):
  - BODY[HEADER]
  - BODY[#.HEADER]
  - BODY.PEEK[HEADER]
  - BODY.PEEK[#.HEADER]
  - RFC822.HEADER
- OTHER—If request or response cannot be classified as BODY or HEADER, then it is classified as OTHER. OTHER examples include:
  - SYN/FIN/ACK/RST packets that do not contain a payload
  - Non-HEADER or BODY IMAP commands such as **3 select inbox**
  - Retransmitted packets
  - Anything else that is not considered BODY or HEADER
  - If the session becomes encrypted or enters PASSTHRU mode, subsequent packets for the session cannot be parsed and are treated as OTHER.

Note Any IMAP transaction that is not an OK tagged response (such as **1 OK FETCH COMPLETE**) is subject to a prepaid refund.

To specify which IMAP bytes are billed for when doing prepaid debits (BODY only, BODY and HEADER only, or BODY and OTHER only), enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# meter include imap body {header | only | other}`	(Optional) Specifies which IMAP bytes are billed for by the CSG2 when doing prepaid debits. Because IMAP metering is byte-based, you cannot configure both **meter include imap** and **basis fixed** or **basis second** in the same service. Only **basis byte** is meaningful with **meter include imap**.

Excluding MMS from Metering

By default, the CSG2 treats Multimedia Messaging Service (MMS) traffic like any other WAP 1.x traffic and generates prepaid and postpaid WAP statistics reports for it. The content type distinguishes it as MMS traffic. You can disable MMS prepaid billing by performing the following task:

To disable MMS prepaid billing, excluding MMS bytes from the CSG2 usage calculation, enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# meter exclude mms wap`	(Optional) Excludes bytes for a WAP 1.x Multimedia Messaging Service (MMS) session from the CSG2 usage calculation.

Excluding the Final Service Idle from Metering

The CSG2 enables you to exclude the final service idle from the CSG2 usage calculation duration billing.

Excluding the final service idle might result in reduced charging because the next service access occurs after the service idles, rather than occurring before the service idles.

You cannot configure both **meter exclude svc-idle** and **basis byte** or **basis fixed** in the same service. Only **basis second** is meaningful with **meter exclude svc-idle**.

To exclude the final service idle, enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# meter exclude svc-idle`	(Optional) Excludes the final service idle from the CSG2 usage calculation duration billing.

Configuring the Quota Reauthorization Threshold

You can configure the threshold of available quota that triggers CSG2 service reauthorization.

To configure the reauthorization threshold, enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# reauthorization threshold` *threshold*	(Optional) Configures the CSG2 reauthorization threshold.

For services configured for fixed-cost billing (**basis fixed**), the reauthorization trigger is the smaller of the following values:

- The threshold configured using the **reauthorization threshold** command
- 25% of the last quota grant returned from the quota server

For services configured for volume-based billing (**basis byte**), the reauthorization trigger is the smaller of the following values:

- The threshold configured using the **reauthorization threshold** command
- 32 KB or 25% of the last quota grant returned from the quota server, whichever is larger

For services configured for duration-based billing (**basis second**), the reauthorization trigger is the threshold configured using the **reauthorization threshold** command.

Note The CSG2 can also accept a threshold specified by the quota server in a quota grant to the CSG2. The threshold must appear in each and every quota server response. The quota server threshold, if present, overrides the threshold specified using the **reauthorization threshold** command.

Configuring the Quota Reauthorization Timeout

After the CSG2 receives a grant of zero quadrans in a Service Authorization Response, the CSG2 waits before it requests quota in a Service Reauthorization Request. You can configure a timer to trigger the service reauthorization.

To specify the CSG2 reauthorization timeout, enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# reauthorization timeout` [**initial** *initial-timeout*] [**maximum** *maximum-timeout*]	(Optional) Configures the CSG2 reauthorization timeout.

For every quota grant of zero, the reauthorization time doubles, until the maximum timeout is reached. For example, if the initial timeout is set to 30 seconds, and the maximum timeout is set to 250 seconds, the reauthorization times (assuming quota grants of zero) would be:

- 30 seconds
- 60 seconds

- 120 seconds
- 240 seconds
- 250 seconds
- 250 seconds

And so on.

Note The CSG2 can also accept a timeout specified by the quota server in a quota grant to the CSG2. The timeout must appear in each and every quota server response. The quota server timeout, if present, overrides the timeout specified using the **reauthorization timeout** command.

Enabling a Refund Policy for a Service

The prepaid error reimbursement feature allows the CSG2 to automatically refund quota for failed transactions. If you have configured a refund policy for the CSG2, you can enable that refund policy for use by a prepaid service.

For more information about refunding in CSG2, see the "Configuring a Refund Policy on the CSG2" section on page 2-21.

To enable a refund policy, enter the following command in CSG2 service configuration mode:

Command	Purpose
`csg2(config-csg-service)# refund` *policy-name*	(Optional) Specifies the refund policy for a CSG2 prepaid service.