CHAPTER 1

Overview

The Cisco Content Services Gateway (CSG) is a high-speed processing module that brings content billing and user awareness to the Cisco Catalyst 6500 series switch or Cisco 7600 series router platforms. The CSG is typically located at the edge of a network in an Internet service provider (ISP) Point of Presence (POP), or Regional Data Center.

The CSG offers more than standard IP flow accounting; the CSG also examines various protocol requests (Wireless Application Protocol [WAP], Mail, FTP, RTSP, and HTTP) to gather URLs and other header information for accounting purposes. Additionally, the CSG gathers information on usernames and usage statistics, and enables differentiated billing for individual transactions based on hostname, on the directory accessed, or on individual files.

The CSG inspects IP traffic at levels deeper than typical routers. When doing so, the CSG behaves partly as a proxy server. As such, you should design your network security strategy to protect the CSG as you would any proxy or server.

This section includes the following information:

What's New

The CSG 3.1(3)C5(5) includes the following new features:

Additional features are described in the "Features from Previous Releases" section on page 1-13.

HTTP Pipelining and Chunked Transfer Encoding

The CSG now supports full HTTP pipelining and chunked transfer encoding. No new configuration is required to enable this function.

When performing AoC for a TCP connection carrying pipelined HTTP requests, the CSG responds with the redirect to the client as soon as the quota server requests the redirect. This could result in the redirect arriving at the client before responses for previous requests arrive, and the client might associate the redirect with a different request in the pipeline.

TCP Byte Counts for HTTP Billing

Support for full HTTP pipelining and chunked transfer encoding required extensive redesign of the TCP engine and of HTTP parsing, which in turn impacted the way the CSG counts bytes. For HTTP billing, the CSG now reports only TCP byte counts. To maintain backward compatibility, the CSG still reports IP byte counts, but the values reported are the same as the TCP byte counts. Packet counts for pipelined HTTP operations are a snapshot of the number of packets detected on the connection since the previous statistics were reported. The packet count might even be zero if two pipelined operations share the same packet.

WAP Support for URL Rewriting

URL-rewriting support is extended to HTTP, to WAP 1.x, and to WAP 2.x. The URL-rewriting token, defined using either the **aoc confirmation** or **verify confirmation** command, now applies to HTTP, to WAP 1.x, and to WAP 2.x.

Service Verification

Service verification is a capability like advice of charge (AoC), provided the first time a user accesses a service using HTTP or WAP. A Service Verify Request quota management message supplies the quota server with content from the user request (the URL, header information, user 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 CLI to define the service confirmation token that is used in URL-rewriting.

To enable or disable service verification, use the **verify** command in CSG service configuration mode. Service verification is also disabled when the quota server sends a Service Verify Tag-Length-Value (TLV) in a Service Authorization Response or Service Verification Response.

Service verification is supported only for HTTP and WAP.

As long as service verification is enabled, sessions of any type for this user do not trigger service reauthorization requests. Service reauthorization resumes for the user 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 user in a Service Verification Response. The CSG sends service verification requests even when no quota is supplied on the Service Verification Response, if the service authorization response contains the cause TLV with value 0x04 (user low on quota, but service access is permitted). Quota Download (0x0008) 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.

RADIUS Handoff Support

In networks that do not use Cisco Home Agents (HAs), the CSG's RADIUS handoff feature can manage handoffs for roaming users.

The User Table identifies all users known to the CSG. The table is populated based on the contents of RADIUS Accounting Start messages, or from the user database, if either feature is enabled in your configuration. When RADIUS handoff is configured, and a RADIUS Accounting Stop is received, the CSG starts a handoff timer instead of deleting the CSG User Table entry for the roaming user immediately.

- When a handoff occurs, the CSG detects an accounting start for the same user with a different NAS IP address. The CSG then uses the existing User Table entry for the user, to preserve the user information, and turns off the timer.
- If the handoff timer expires before the CSG detects an accounting start for the user, the CSG assumes a handoff did not occur and deletes the User Table entry for the user.

The handoff timer does not run on the backup CSG. Instead, when the backup CSG becomes the primary CSG, it examines all User Table entries. For all entries that have the HANDOFF flag set, the CSG starts a handoff timer. If the handoff timer is not configured on the backup CSG, all User Table entries that have the HANDOFF flag set are deleted.

To configure RADIUS handoff support, use the **radius handoff** command in CSG user group configuration mode.

Fixed CDR Support for HTTP

The CSG now provides fixed Call Detail Record (CDR) support for HTTP as well as for WAP. This support generates one fixed CDR for every HTTP transaction, instead of two CDRs that are typically generated at the beginning and end of the transaction.

The new single CDR (0x0038) contains all fields included in the HTTP header and HTTP statistics records, in a fixed format. In addition, the same fixed format service TLVs that were included for WAP are also included for HTTP.

The new single CDR also includes RADIUS TLVs, in ascending order, based on the RADIUS TLVs configured using the **report radius attribute** command in CSG accounting configuration mode. This is a change from the CSG 3.1(3)C5(1), in which you hard-coded up to 10 specific RADIUS attributes which were included in the CDR in a predefined order. The new scheme is more flexible, enabling you to add new RADIUS attributes as you go. This change in the handling of RADIUS TLVs applies to both WAP and HTTP fixed CDRs.

Fixed CDR support does not support RADIUS attribute 26 (the Vendor-Specific Attribute, or VSA), because the list of attributes defined within the VSA is in itself variable. Therefore, a predefined "fixed" list of attributes cannot be determined when RADIUS attribute 26 is configured.

To enable the fixed format feature for HTTP and for WAP, use the **records format fixed** command in CSG accounting configuration mode.

The CSG also supports fixed HTTP intermediate records. The fixed intermediate record format is identical to the format of the fixed record created at the end of the transaction, except for the new message type (0x003A), which is necessary to differentiate the two records. The intermediate statistics, such as TCP byte counts, are per intermediate period, and are not cumulative. This differs from the existing HTTP intermediate support for variable format CDRs, in which the TCP byte counts are cumulative.

The Content Delivered TLV (0x003D) contains a value of 0x00 (not delivered) if the HTTP response code is greater than or equal to 400, or if the TCP byte download count is less than 12 bytes.

Fixed CDR Support for RTSP

This feature enables the CSG to send the existing RTSP stream CDRs in a new fixed format (type 0xYY). The same fixed format service TLVs that were included for WAP are also included for RTSP.

To enable the fixed format feature for RTSP, use the **records format fixed** command in CSG accounting configuration mode.

Single CDR Support for WAP Connectionless and HTTP

The CSG already reduces the multiple CDRs generated for WAP connection-oriented traffic down to a single CDR, which is reported at the end of the transaction. This feature is now extended to WAP connectionless traffic and HTTP traffic.

The single CDR contains the standard variable format, and it also includes a comprehensive list of TLVs containing all pertinent information for the transaction. For WAP connectionless transactions, it includes everything that is included in a WAP GET and REPLY CDR. For HTTP transactions, it includes everything in the HTTP header and HTTP statistics records.

To enable single CDR support for WAP connection-oriented, WAP connectionless, and HTTP traffic, use the **variable-single-cdr** keyword on the **records format** command in CSG accounting configuration mode.

When you configure single CDR support, the CSG suppresses HTTP intermediate record generation.

Fixed CDR Support for IMAP

The CSG now supports postpaid billing for the Internet Message Access Protocol (IMAP), in addition to postpaid billing for Post Office Protocol, version 3 (POP3) and Simple Mail Transfer Protocol (SMTP). This feature enables the CSG to report service-level fixed format CDRs for IMAP. The service-level CDR includes the following IMAP-specific counts:

- Number of header retrievals. That is, the number of times the CSG retrieved the header attribute of an e-mail message (for example, **BODY[HEADER]**, **RFC822.HEADER**).
- Header IP bytes sent upstream (client to server)
- Header IP bytes sent downstream (server to client)
- Header TCP bytes sent upstream
- Header TCP bytes sent downstream
- Number of body retrievals. That is, the number of times the CSG retrieved any portion of the body text of an e-mail message (for example, **BODY[]**, **BODY[TEXT]**, **BODY[3]**, **BODY[]<0.4096>**, **RFC822**, **RFC822.TEXT**).
- Body IP bytes sent upstream
- Body IP bytes sent downstream
- Body TCP bytes sent upstream
- Body TCP bytes sent downstream

The CSG reports incremental byte counts for the IMAP service-level fixed format CDRs. For example, if 100 KB of traffic is generated for the first 15 minutes, 50 KB for the next 15 minutes, and the CSG generates intermediate CDRs every 15 minutes, then the CSG reports the delta of the total byte count from the point in which the last CDR was reported to the point at which the current CDR is reported. So, the first CDR would report 100 KB and the second would report 50 KB.

With fixed format CDRs, they may be reported at a given time interval or after a volume threshold has been reached (for example, every 15 minutes, or after every 100 KB.)

To enable the CSG to support IMAP, enter the **records format fixed** command in CSG accounting configuration mode, and the **accounting type imap** command in CSG policy configuration mode.

When configuring CSG support for IMAP, keep in mind the following considerations:

- The CSG supports only postpaid billing for IMAP. IMAP transactions for a prepaid user are treated as postpaid.
- The CSG does not support A0C for IMAP. If AoC is configured for an IMAP user, AoC is ignored for that user.
- The CSG cannot examine IMAP flows sent over an encrypted tunnel, such as Secure Socket Layer (SSL) or Transport Layer Security (TLS). Therefore, when an encrypted tunnel is used for IMAP traffic, the CSG records only IP and TCP upstream and downstream byte counts. No other counts are provided.

SMTP Prepaid/Envelope Support

The CSG now provides SMTP postpaid and prepaid support, including the addition of envelope information in the CDR.

SMTP prepaid support includes all existing billing options (including IP bytes, TCP bytes excluding retransmissions, duration, and fixed). SMTP CDRs include mail envelope information as well as IP byte counts, TCP byte counts, and mail data (X-CSG-SIZE) byte counts for each mail message. When multiple e-mails are sent over a single TCP connection, each mail message is assigned byte counts until the start of the next mail message. The last mail is assigned bytes from the start of that mail until the end of the TCP connection.

The return code reported in the CDR is the one returned for the DATA portion of the mail message. If the CSG does not receive that data return code, it reports the last error return code (other than **250**) received for individual recipients (because a bad recipient return code might be the cause of the mail not being sent). If the CSG receives a QUIT before receiving any return code, it reports a default return code of **554** (Transaction failed). This enables the CSG to apply refunding via the SMTP return code value.

If the user runs out of quota in the middle of a transaction, the session is terminated and all known information is reported in a CDR. The application return code indicates whether the mail was received, and the authentication failure bit is set in the TCP **flags** field.

The CSG no longer uses the TCP Stats CDR, which was generated at the end of the TCP connection, because the information in the TCP Stats CDR duplicates the information in the new SMTP CDR.

SMTP Advice of Charge Support

The CSG handles advice of charge (AoC) support for SMTP in a slightly different manner than for other protocols.

Typically, the CSG sends the content authorization request immediately after performing service authorization. The CSG can do this because all of the information in the content authorization request is contained in the initial flow received by the CSG.

However, for SMTP, the information needed in the content authorization request—number of recipients with a good return code, number of recipients with a bad return code, size of mail in bytes (if present) and the sender of the e-mail—are not known until after the CSG processes the SMTP envelope. Therefore, when AoC is configured, the CSG allows all envelope information to flow through, even if the user has no quota (however, access is not permitted if the user is not authorized). The CSG initiates the content authorization request when it receives the DATA command. The CSG queues the packet containing the DATA command until AoC processing is resolved.

If the CSG receives a DROP or REDIRECT in the content authorization response, it drops the DATA command packet, terminates the session, and generates a CDR containing the envelope information and an invalid application return code.

If the CSG receives a FORWARD, it uses the weight that is returned in the response for prepaid processing.

Multiple e-mails over the same TCP connection result in multiple content authorization requests. Each mail is treated as a separate transaction.

To enable AoC support for SMTP, use the **authorize content** command in CSG service configuration mode.

POP3 Support

The CSG now generates a single CDR for each POP3 e-mail. The CDR includes all necessary information, such as the IP byte count and the TCP byte count. The CSG no longer generates a final TCP Stats record. If a user downloads multiple e-mails during a single TCP session, the CSG generates a CDR for the previous e-mail each time it processes a new RETR or TOP command. The CSG generates a CDR for the last email when it processes the STATS command (for TCP termination).

The CSG supports POP3 in only postpaid mode.

If a user tries to download e-mail and no e-mail exists, the CSG generates a POP3 CDR that contains an application return code TLV with a value of 554. This is the only condition in which the CSG includes a non-zero return code in a POP3 CDR.

To define the POP3 accounting type for a billing policy, use the **accounting type pop3** command in CSG policy configuration mode.

RADIUS Packet of Disconnect

The quota server can instruct the CSG to disconnect a user. The CSG then sends a RADIUS Packet of Disconnect (PoD) message to the NAS identifying the user, and the NAS then sends a RADIUS Accounting Stop message, which also clears the User Table entry.

The quota server instructs the CSG to disconnect a user using one of the following methods:

- The quota server can send the new UserDisconnectRequest message (0x2D) to the CSG. This message uses the UserIndex TLV to identify the user to be disconnected.
- The quota server can use Action Code 4 in the Action TLV in one of the following requests and responses:
  - The ServiceAuthResponse (indicating that the CSG is to send the PoD message when the quota runs out)
  - The ServiceStopRequest (indicating that the CSG is to send the PoD message immediately)
  - The UserProfileResponse (indicating that the CSG is to send the PoD message immediately)

The CSG sends the PoD message to the NAS that is specified by the NAS-IP-Address attribute (4) in the Accounting Start.

To configure RADIUS PoD support, use the following commands in CSG user group configuration mode:

- Use the **radius pod attribute** command to specify the RADIUS attributes to be copied from the RADIUS Start message and sent to the NAS in the PoD message.
- Use the **radius pod nas** command to specify the NAS port to which the CSG should send the PoD message, and the key to use in calculating the Authenticator.
- Use the **radius pod timeout** command to specify the number of times to retry the RADIUS PoD message if it is not acknowledged, and the interval between retries.

The CSG can send PoD messages only if the CSG received the user information on a virtual interface configured using the **radius proxy** or **radius endpoint** command in module CSG configuration mode.

RADIUS Endpoint

To configure the CSG as a RADIUS Accounting endpoint, you can use the existing **radius key** and **radius acct-port** commands in CSG user group configuration mode, or you can use the new **radius endpoint** command in module CSG configuration mode. The CSG 3.1(3)C5(5) supports both endpoint configuration methods. However, if you plan to use RADIUS PoD with RADIUS endpoint, then you must use the **radius endpoint** command in module CSG configuration mode.

We do not recommend using both configuration methods in the same environment.

RADIUS Proxy Source IP Address

You can now configure a source IP address using the **radius proxy** command in module CSG configuration mode. The CSG uses the source IP address when it forwards a RADIUS message to the server. Prior to 3.1(3)C5(5), the CSG used the CSG IP address as the source IP address.

Service-Level CDR Summarization

By default, the CSG generates billing records for each transaction. This has the potential to overwhelm the Charging Gateway or the collector. To prevent this situation, the CSG can summarize CDRs at the service level, instead of at the transaction level.

For example, if a user is accessing the open Internet service, and the data is to be billed only based on volume, generating records for each HTTP transaction is of little use. With service-level CDR summarization enabled, the CSG generates only consolidated records containing service-level usage. Be aware that no information from individual events is reported (for example, no URLs).

The CSG supports the following protocols in both fixed and variable format: IP, HTTP, SMTP, POP3, and IMAP. (POP3 and IMAP are supported in postpaid mode only.)

To enable service-level CDR summarization, use the **records granularity** command in CSG service configuration mode.

Note If you specify both **type http** and any other type (**type other**, **type ftp**, **type imap**, and so on) for a service, and you enable service-level CDR summarization for the service, the CSG's incremental and cumulative byte counts are not valid, because they are a mix of TCP bytes (for the HTTP traffic) and IP bytes (for all other traffic).

Passthrough Mode and the Default Quota

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

For each service that you want to use passthrough mode, you must enable it using the new **passthrough** command in CSG service configuration mode.

You also use this command to specify the size of each quota grant (the default quota) to give to a service. When passthrough mode is enabled for a service, and a session for the service needs quota, and a quota server is not active, the CSG grants the service the amount of quadrans specified on the **passthrough**

command. (Quadrans come in three forms, "basis byte" for volume-based billing, "basis fixed" for event-based billing, and "basis second" for duration-based billing.) The CSG continues to grant quota for as long as a quota server is not active.

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

This section contains the following additional information about passthrough mode:

- Flagging of Messages, page 1-9
- User Profile Requests, page 1-9
- Quota Server Recovery, page 1-10

Flagging of Messages

To facilitate billing recovery, certain messages to the quota server and the BMA include a QuotaServerFlags TLV. The CSG adds this TLV whenever it grants a passthrough mode quota to a service. The CSG_**QSFLAGS_PASSTHRU_USED** bit is set in the **flags** field of the TLV. The QuotaServerFlags TLV is added to the following messages:

- FTP
- HTTP Statistics
- IP
- POP3
- QuotaDownload sent to BMA
- RTSP
- ServiceReauthorizationRequest
- ServiceStop sent to BMA
- ServiceStop sent to quota server
- Service Usage
- Single Variable HTTP Statistics
- SMTP
- TCP
- UDP
- WAP Connectionless PDU Types
- WAP Connection Oriented Statistics

User Profile Requests

When the CSG learns of new users, it typically sends a UserProfileRequest to an active quota server. This enables the CSG to learn the billing plan to use for each user. If the quota server returns a NULL billing plan, a user is postpaid.

When passthrough mode is in use for any service, the CSG changes the way it processes UserProfileRequests. When there is no active quota server, the CSG assigns all new users to postpaid processing. The CSG bills all sessions for these users as postpaid, and does not flag generated CDRs with a QuotaServerFlags TLV.

If a user is still on the network when the quota server becomes active, the CSG sends a normal UserProfileRequest to the quota server for the user. When the CSG receives a response, it updates the user's billing plan. If the updated billing plan is now a prepaid billing plan, the CSG blocks new IP sessions started by the user until the quota server grants a quota. IP sessions that were active before the billing plan was updated to prepaid are kept as postpaid, and generate postpaid CDRs until they end.

Quota Server Recovery

When a quota server becomes active, the CSG forwards stored ServiceStopRequests to it. Additional actions taken by the CSG depend on user traffic.

When a user who was forced to postpaid while the quota server was absent creates a new IP session, the CSG issues a UserProfileRequest followed by a ServiceAuthorizationRequest, and blocks new traffic until quota has been granted.

Prepaid users might have some services that were granted quota in passthrough mode. For those services, when quota runs low, the CSG sends a ServiceReauthorizationRequest 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 CSG in passthrough mode. The requested quadrans TLV contains a request for an additional quota amount.

When the quota server responds to a ServiceStopRequest or a ServiceReauthorizationRequest, the CSG moves the service out of passthrough mode. If the quota server denies quota when it sends a ServiceAuthorizationResponse message, the CSG blocks the traffic. The CSG also flags CDRs generated by traffic for these services, which received passthrough mode quota grants, with QuotaServerFlags TLVs, until a ServiceStop is sent. That is, once a service is granted a passthrough mode quota, the CSG flags all CDRs for that serviced, up to and including the ServiceStop. Again, this only applies to prepaid users. Postpaid users CDRs are never flagged.

Fragment Support

The CSG now supports IP fragments for both TCP and non-TCP flows, including fragments that arrive out of order.

Connection Duration Billing

Connection Duration Billing enables the CSG to deduct quota based on the time that a user is logged on to the IP network. That differs from Service Duration Billing, which charges based on the duration of a service. Because the service measures the duration of subscriber access, the service is never idle—it is destroyed only when the user logs out, or when a Service Stop Request is received from the quota server.

The CSG charges based on the following rules:

- The First Billable Time (FBT) is the timestamp, in seconds, of the first non-zero grant of quota in a Service Authorization Response for the Connection Duration service. A Service Authorization Request is generated when the following conditions are met:
  - A User Table entry is created (typically due to a RADIUS Accounting Start message),.
  - A Connection Duration service is part of the billing profile for the User Table entry (indicated in a RADIUS Access-Accept message, a RADIUS Accounting-Start message, or a Quota Server User Profile Response).

  If the user has quota, the FBT is typically the same time as the RADIUS Accounting-Start.

- The Last Billable Time (LBT) is the timestamp, in seconds, when the User Table entry is destroyed. During the service lifetime, the CSG updates the LBT when either of the following situations occurs:
  - An IP session starts or ends.
  - A Service Reauthorization Request is sent by the CSG. This results in an update to the service balance and usage before the Service Reauthorization Request is sent. The CSG uses the following algorithm to calculate the usage:

    Usage = LBT - FBT

    or

    Usage = OutOfQuota Timestamp - FBT

    whichever is smaller.

    Therefore, if the service does not run out of quota, the algorithm is simply:

    Usage = LBT - FBT

    If the user runs out of quota, but refreshes the quota before the service times out, the periods of zero quota are not included in the usage calculation. When the user runs out of quota, existing prepaid and postpaid IP sessions for the subscriber are terminated. If the user does not have quota to proceed, no IP sessions for the user are allowed to proceed. The CSG provides enforcement for only those policies that have accounting configured.

To configure Connection Duration Billing, use the **activation** and **basis second** commands in CSG service configuration mode

URL MAP Support for RTSP

The CSG uses maps to match URLs or headers against a pattern, to determine whether flows are to be processed by the CSG accounting services. The CSG now supports RTSP URL mapping and filtering, in addition to WAP and HTTP URL mapping and filtering.

For more information about URL mapping, including special considerations for RTSP URL mapping, see the description of the **match (URL map)** command.

Postpaid Service Tagging

This feature enables the CSG to map postpaid content to a CSG service, and to report the service name in a CSG Service ID TLV in transaction-level CDRs to the BMA. (The CSG Service Session ID TLV is not sent in variable-format records for postpaid service tagging.)

The service must be associated with a billing plan configured for postpaid mode. As in the case of prepaid billing plans, the user can be associated with a billing plan via a RADIUS message or a User Profile Response from the quota server. If no quota server is configured, and the billing plan cannot be determined from RADIUS messages, the user is automatically associated with any billing plan configured for postpaid mode. In such cases, we strongly recommend that you configure only one billing plan for postpaid mode.

Stateful Redundancy and Failover

The CSG supports stateful redundancy for FTP, HTTP, IMAP, POP3, SMTP, TCP, and WAP connections.

Stateful redundancy is the configuration of the CSG to share information related to billing with its backup CSG in the event of a failure. That is, the session continues to be billed even when the primary CSG fails and the backup CSG takes over.

As described in the "Configuring Fault Tolerance" section on page 4-4, the primary and backup CSGs use a private VLAN to exchange connection and billing status information. The configurations must be the same on each CSG. The quota server, BMA, and user ID database definitions should also be the same, although this is not required.

During normal operation, connection and billing state information is sent by the primary CSG to the backup CSG, and from the primary quota server to the backup quota server. Both the primary and the backup CSGs maintain state information for the configured BMAs, and the primary CSG keeps the backup CSG informed as to which BMAs and quota servers are being used. If the primary CSG fails, the backup CSG takes over operation and tries to use the same BMAs or quota servers, if it has connectivity. Otherwise, the backup CSG selects the BMAs or quota servers with the highest priority.

The primary CSG also informs the backup CSG when user IDs are added to or removed from the User Table, and sends the correlators to the backup CSG to ensure consistency when sending billing records for recovered connections to the BMAs. Quota use is also correlated.

The CSG provides full stateful failover for FTP and IMAP sessions.

The CSG provides limited stateful failover for HTTP, POP3, SMTP, and WAP sessions. User information and quota information is maintained on the backup CSG; however, in-flight transactions are not. If the primary CSG fails, the user transaction completes on the backup, but no quota is charged for the transaction. Normal billing resumes with the user's next transaction.

The CSG also supports stateful redundancy for TCP connections. That is, the session continues to be billed even when the primary CSG fails and the backup CSG takes over.

The CSG does not support stateful redundancy for IP, RTSP, or UDP connections.

"Default" Policy

The CSG matches content on a best-match basis, based on Layer 3 and Layer 4 information. When there is a successful content match, the CSG then matches against the policies configured within that content, linearly, on a first-match basis. If no policy within the content matches, the CSG matches against an implicit "default" policy, which matches all traffic. Matching this "default" policy does not generate a CDR, because no accounting policies can be configured for the "default" policy.

For example, given the following policy and content configuration:

```
ip csg policy PHTTP1
  accounting type http customer-string HTTP-POL1
ip csg policy PHTTP1
  accounting type http customer-string HTTP-POL2
ip csg content HTTP
  policy PHTTP1
  policy PHTTP2
```

The output from the **show module contentServicesGateway 5 content name HTTP detail** command is as follows:

```
HTTP, state = OPERATIONAL, index = 10
  destination = 198.133.219.0/24:80, TCP
  idle = 3600, replicate = none, vlan = ALL, pending = 30
  max parse len = 4000, persist rebalance = TRUE
  conns = 0, total conns = 0

  policy          total conn   client pkts  server pkts
  ------------------------------------------------------
  PHTTP1          0            0            0
  PHTTP2          0            0            0
  (default)    4760        30056        26534
```

In this example, any TCP traffic that does not match either the PHTTP1 policy or the HTTP2 policy matches the “default” policy, and is reflected in the **(default)** row.

Features from Previous Releases

In addition to new features introduced in this release, the CSG provides the following features and functionality that were introduced prior to the CSG 3.1(3)C5(5):

Support for WAP 2.0

The CSG supports WAP 2.0—a specific dialect of XML that can be transported over HTTP to convey content to the mobile devices. The incomplete WAP 2.0 standard specification defines five network flows in which mobile devices may participate. The first four flows described below are generally implemented over WAP 2/HTTP/TCP across a WAP 2.0 Proxy/Push Proxy Gateway (PPG). The last flow (Flow number 5, TO-TCP) is not supported.

General flows supported in the CSG

1. Retrieve a message from the network using HTTP.request-method: GET
2. Post a message into the network using HTTP.request-method: POST
3. Acknowledge a PUSH indication using HTTP.request-method: POST

Push flows generally implemented as WAP 2 over SMS OTA-Push

These flows are generally implemented as WAP 2/SMS rather than WAP 2/HTTP.

4. PO-TCP: The PPG establishes a direct connection to the mobile device using prior knowledge of the mobile device's IP address. The PPG negotiates an understanding of the mobile device's identity and capabilities using HTTP.request-method: OPTIONS, then uses HTTP.request-method: POST to deliver the push notification as a WAP 2.0 XML message. The WAP 2.0 XML may wrap other content such as MMS-encoded notifications or URLs.

5. TO-TCP: This form of PUSH is a hybrid between the conventional pure Short Message Service (SMS) notification and previous flow. It provides a bridge during a transition period where the PPG does not have prior knowledge of the mobile device's IP address. This push is implemented as a WAP 2 Session Initiation Request (SIR) over SMS. Upon receipt of the SIR, the mobile device connects to the PPG via TCP. The PPG then begins the HTTP exchange described in flow 4 above.

Note The TO-TCP flow is not supported, but is provided for informational purposes.

The CSG supports billing WAP 2.0 traffic (the first three flows above) using existing configuration commands. WAP 2.0 mobile devices may be configured to use or to ignore the WAP 2.0 Proxy; however, if a WAP 2.0 Proxy is not configured, the configuration resembles HTML over HTTP (in that you must choose the appropriate content rules so that HTTP policies can be applied to the WAP 2 traffic). The WAP 2.0 Proxy enables you to identify WAP 2.0 traffic by configuring a content that examines traffic to and from the WAP 2.0 Proxy. Using an account type of **http** enables billing of WAP 2.0, including support for policies based on the HTTP method, URL and HTTP header values. The current limitations of HTTP billing (with respect to Transport Layer Security [TLS]) apply to billing WAP 2.0/HTTP and MMS/WAP 2.0/HTTP.

Differentiated Billing of MMS Over WAP 2.0

WAP 2.0 mobile devices generally implement support for extensive Multimedia Messaging Service (MMS). This is generally implemented over WAP 2.0. Service providers use MMS to differentiate and promote their products, which necessitates differentiating the billing of MMS over WAP 2.0 from other WAP 2.0 billing.

The CSG supports the ability to bill MMS over the supported WAP 2.0 flows at a differentiated rate. When WAP 2.0 billing is configured, MMS may be differentiated by using the capabilities of the **http** accounting type to detect some or all of the following characteristics of MMS/WAP 2/HTTP traffic:

- The URL of a GET of MMS content points to the MMSC and encodes an MMS message ID.
- The URL of the POST of an MMS message or an MMS message notification acknowledgement points to the MMSC.
- The Content-Type HTTP header of the POST of an MMS message or an MMS message notification acknowledgement is "application/vnd.wap.mms-message".

MMS over WAP 2.0 allows the following three types of notification:

1. SMS-based notification carrying the URI for the MMS. The handset then initiates a GET request to that URI to retrieve the information.
2. TO-TCP (Terminal-Originated TCP) starts with SMS, but only provides the IP address of the PPG. The handset must then open a TCP connection and wait for an HTTP request from the PPG. This HTTP request is an OPTIONS method and must succeed before the handset can retrieve the notification.
3. PO-TCP (PPG Originated TCP) is similar to TO-TCP except the TCP connection is opened by the PPG, and is followed by the OPTIONS method.

The CSG Layer 7 billing for MMS relies entirely on options one and three. TO-TCP is not supported.

Note If a terminal reuses a persistent PO-TCP to initiate a new method request, the packets are dropped and the PO-TCP connection appears hung until TCP retry attempts expire.

Real Time Streaming Protocol Billing

The Real Time Streaming Protocol (RTSP) Billing feature adds the following functionality to the CSG:

- Correlates various streams associated with an RTSP session.
- Reports application-level information (for example, filename) to the billing system.

RTSP uses four different protocols for streaming to the client. The client presents the server with a choice of acceptable protocols and port numbers, the server responds with its choice of protocol that includes:

- RTSP requires a control TCP connection to server port 554.
- RTSP also requires a UDP server-to-client stream for RTP (audio/video stream delivery), and a bidirectional UDP flow pair for exchanging synchronization information. The ports for the UDP flows are negotiated on the TCP connection during the SETUP exchange.
- RTSP can use RealNetworks RDT for the stream transport. This establishes a UDP flow in each direction: one for stream delivery from the server, and the other for requesting resends of lost media packets.
- RTSP can operate completely over the single TCP connection.
- RTSP can be tunneled over HTTP.

RTSP transport modes are negotiated on the control connection using the following methods:

- Client sends SETUP request suggesting one or more modes it can support.
- Server responds with mode it has selected and ports that are to be used.

Per-Click Authorization

Per-click authorization implements functions like advice of charge (AoC) redirection and retrieval of price from an external server. For the control session, the CSG sends a contentAuthorizationRequest at the beginning of the TCP session. For each transaction involving a data stream, the CSG sends a contentAuthorizationRequest before allowing the data stream. This request allows the quota server to inspect the filename before granting authorization.

The CSG only allows Network Address Translation (NAT-based) redirection for RTSP traffic.

RTSP allows multiplexing multiple data streams over the same transport. For example, audio and video presentations can be multiplexed over the same UDP flows. In these cases, the quota server must ensure that it does not send contradictory responses to the various contentAuthorizationRequests. For example, if one request is allowed and the other one denied, the CSG's behavior is undefined.

Correlation

The CSG provides RTSP correlation at the RTSP session level. All TCP/UDP flows associated with an RTSP session share a correlator.

The CSG does not correlate RTSP streams that do not share the RTSP session ID.

Correlating Multiple Streams Controlled by a Single RTSP Session

An RTSP session can control multiple streams, such as audio and video stream for a movie. For instance, if M is the media server, a client (C) can perform the following operations over the same RTSP session:

Table 1-1 Multiple Streams Controlled by Single RSTP Session

Client	Server	Protocol	Method/URL
C	M	RTSP	DESCRIBE rtsp://a.ex.com/movie.sdp Client requests description of a movie. The server assigns a session ID to the client, and sends the.sdp file containing information about the movie.
C	M	RTSP	SETUP rtsp://a.ex.com/movie/audio Client requests setup of a stream.
C	M	RTSP	SETUP rtsp://a.ex.com/movie/video Client requests setup of a second stream. This results in setting up of four UDP flows.
C	M	RTSP	PLAY rtsp://a.ex.com/movie.sdp

In this example, all the streams share the RTSP session and the session ID. There is one RTSP control TCP session, and four UDP streams associated with it. The CSG is able to correlate all these four UDP streams together with the control session.

Correlating Multiple Streams Controlled by HTTP

HTTP sessions can be used to correlate multiple, related RTSP streams. Different RTSP streams could go to different servers. The CSG has no easy way to find out that these two streams are related. A typical situation is when a web server (W) hosts the media description file, movie.sdp, a video server (V) contains the video stream, and an audio server (A) contains the audio stream. The following interactions take place:

Table 1-2 Multiple Streams Controlled by HTTP

Client	Server	Protocol	Method/URL
C	M	HTTP	GET /movie.sdp
C	V	RSTP	SETUP rtsp://v.eg.com/video
C	A	RSTP	SETUP rtsp://a.eg.com/audio
C	V	RSTP	PLAY rtsp://v.eg.com/video
C	A	RSTP	PLAY rtsp://a.eg.com/audio

In the previous example, there are three concurrent sessions:

- HTTP 1.1 sessions: 1
- RTSP Video Session: 2, 4
- RTSP Audio Session: 3, 5

All of the sessions (TCP and UDP) associated with an RTSP session can be correlated. In this same example, the sessions associated with the video on server V are correlated. Similarly, the sessions associated with the audio on server A are correlated; however, there is no correlation between the audio and video flows, and no correlation with the HTTP session.

Implications of Container Files:

A container file is a storage entity in which multiple, continuous media types pertaining to the same end-user presentation are present. A container file represents an RTSP presentation, with each of its components being RTSP streams. While the components are transported as independent streams, it is desirable to maintain a common context for these streams at the server. Synchronized Multimedia Integration Language (SMIL) is an example of describing the contents of a container file.

The CSG does not correlate the streams within a container file.

Interleaved RTSP

Interleaved RTSP passes RTSP data in the TCP control session. Because the CSG parses the control session, it could cause a large performance bottleneck.

To avoid a bottleneck, the CSG does the following for interleaved RTSP sessions:

- Wait for a SETUP request/reply to determine whether this is an interleaved RTSP session.
- Remember the URL information.
- After determining interleaved RTSP, report RTSP information to BMA/quota server, and shortcut the connection to fastpath. Any subsequent transactions on the same RTSP control connection is not visible to the CSG's billing function.

This method provides some RTSP level information, but avoids making the RTSP path a target of DoS attacks. If most of the RTSP streaming billing applications are in the walled garden, customers have some control over the servers to ensure that the use of interleaved RTSP is not too much.

CDRs

The CSG generates the following the CDRs for RTSP:

- TCP control session: TCP, TCPInt, RTSP
- Data streams: RTSP stream
- UDP CDRs for each UDP session

Note If you are using fixed CDR support, the CSG does not generate any UDP CDRs.

RTSP billing in the CSG is based on inspection of the RTSP SETUP and TEARDOWN messages that are exchanged between the client and server. The CSG builds the RTSP CDR immediately after the RTSP TEARDOWN signal if the URL exactly matches that from the RTSP SETUP signal. Otherwise, the CSG builds the CDR after any condition that causes the flows to be terminated. Examples include:

- When the idle content timer expires. By default, this timer is set to 3600 seconds (1 hour). To receive the RTSP CDRs sooner, set the timer to a smaller value, such as 60 seconds, using the **idle** command in CSG content configuration mode.
- When a service_stop is triggered (for example, when the access server sends a RADIUS Accounting Stop for the user).

Session Processing

RTSP control session processing is similar to FTP control sessions. The RTSP control session is assigned an 8-byte correlator. The most significant 6 bytes of the correlator are assigned from the session ID and the session ID sequence. The least significant 2 bytes of the correlator are zeroed (for example, 0x0000).

The CSG keeps track of RTSP sessions and an RTSP session is used to correlate multiple streams associated with the session.

Note An RTSP session may be comprised of more than one TCP session; alternatively, the RTSP session can exist without a TCP session between client and server.

When the client sends a **setup** command, the CSG notes the client ports and extracts server ports from the SETUP response. Data connections to these ports are processed as if they hit the *content, policy* definition for the control server.

The following example (from RFC 2326) uses a single RTSP session to control multiple streams. The CSG actions are annotated after various steps.

In this example, client C requests a presentation from media server M. The movie is stored in a container file. The client has attached an RTSP URL to the container file.

```
C->M: SYN port=RTSP
M->C: SYN-ACK
Assign 8 byte correlator X. Lower two bytes of the correlator are 0.

C->M: DESCRIBE rtsp://foo/twister RTSP/1.0
            CSeq: 1

M->C: RTSP/1.0 200 OK
            CSeq: 1
            Content-Type: application/sdp
            Content-Length: 164

            v=0
            o=- 2890844256 2890842807 IN IP4 172.16.2.93
            s=RTSP Session
            i=An Example of RTSP Session Usage
            a=control:rtsp://foo/twister
            t=0 0
            m=audio 0 RTP/AVP 0
            a=control:rtsp://foo/twister/audio
            m=video 0 RTP/AVP 26
            a=control:rtsp://foo/twister/video

C->M: SETUP rtsp://foo/twister/audio RTSP/1.0
            CSeq: 2
            Transport: RTP/AVP;unicast;client_port=8000-8001

M->C: RTSP/1.0 200 OK
            CSeq: 2
            Transport: RTP/AVP;unicast;client_port=8000-8001;
                       server_port=9000-9001
            Session: 12345678
```

Build RTSP record. Correlator = X+i. The CSG makes sure that X+i is even. RTSP usage records for these two UDP flows carry X+i and X+i+1 as the correlators. The correlators share 63 bits to help bind the various flows for an RTSP transaction together, and also enable you to distinguish the various interim records for one UDP flow from another.

```
C->M: SETUP rtsp://foo/twister/video RTSP/1.0
            CSeq: 3
            Transport: RTP/AVP;unicast;client_port=8002-8003
            Session: 12345678

M->C: RTSP/1.0 200 OK
            CSeq: 3
            Transport: RTP/AVP;unicast;client_port=8002-8003;
                       server_port=9004-9005
            Session: 12345678
```

Build RTSP record. Correlator = X+3. RTSP usage records generated for these two UDP flows carry the same correlator.

```
C->M: PLAY rtsp://foo/twister RTSP/1.0
            CSeq: 4
            Range: npt=0-
            Session: 12345678

M->C: RTSP/1.0 200 OK
            CSeq: 4
            Session: 12345678
            RTP-Info: url=rtsp://foo/twister/video;
              seq=9810092;rtptime=3450012

C->M: TEARDOWN rtsp://foo/twister RTSP/1.0
            CSeq: 6
            Session: 12345678
V->C: RTSP/1.0 200 OK
            CSeq: 6
```

This TEARDOWN does not correspond to the SETUP URL, so the CSG lets the streams idle out and sends usage records when the streams idle out.

Prepaid Error Reimbursement

The Prepaid Error Reimbursement feature allows the CSG to automatically refund quota for failed transactions, as defined by the CLI. Refund conditions can be configured using session flag (IP, TCP or WAP) settings and application return codes.

The CSG also adds a refund TLV to the statistics records on the BMA interface. The refund TLV is added for transactions that meet one of the refund conditions. The refund amount contains the quota amount to be refunded for the transaction. The refund amount is the same number that is reported in the quadrans TLV. Thus, the full charge for the transaction is always refunded for these protocols.

Note For duration-based services, error reimbursement is not possible.

WAP Cutoff

When a user's quota is entirely depleted during the middle of a transaction, the corresponding action varies depending on the protocol. For WAP, the current transaction is allowed to complete, and the user is charged for all bytes used in the transaction. The result is that the user has a negative quota balance. On the next transaction request, the user is redirected to the top-off server. While this behavior provides the best user experience, it also allows some leakage. For small transactions, the leakage is minimal; however, for large transactions the leakage can be significant.

Because there is a trade-off between end-user experience and leakage, a CSG configuration option allows you to choose what behavior you want to enforce. To configure this feature, enter the **zero-quota abort type** command in global configuration mode. The configuration option is enabled on a per-service basis. This option is only supported for WAP, and the default is to not terminate a transaction midstream when the user runs out of quota. For all other protocols, the user is cut off midstream.

Note Configuring the cut-off option for WAP affects only connection-oriented sessions, and not connectionless traffic.

When configured, this condition causes the existing transaction to be aborted. The CSG sends aborts to both the client and server, terminating the transaction. A BMA record for the transaction is generated with a new flag setting (0x04) in the Wireless Transaction Protocol (WTP) information record (0x0020) that indicates the transaction was intentionally aborted. In the report, the user is charged for the number of bytes that were processed for the transaction, including the bytes that caused it to exceed the quota balance. Typically, the user should not be charged for this transaction because it was not allowed to complete. The user is reimbursed by the billing agent for transactions with the 0x04 flag set, or by the prepaid refund feature.

Service Duration Billing

The Service Duration Billing feature enables the CSG to deduct quota based on the time of network usage for prepaid (or "managed") users. With this feature, the user is charged for the time duration of the CSG service. The charging is performed according to the following rules:

- For TCP sessions, the Last Billable Session Time (LBST) is the timestamp of the end of the session. The end of the session is detected using TCP session-termination signaling (RST, FIN/ACK signals) or with content idleness. Because non-TCP sessions (such as UDP) do not have a Layer 4 session termination mechanism, the LBST for non-TCP sessions is the last packet forwarded for the IP session.
- The First Billable Time (FBT) is the timestamp (in seconds) of the first grant of network access to a session mapped to a duration-based charging prepaid service. Typically, this time is equal to the timestamp of the first Service Authorization Response with a non-zero quota.
- The Last Billable Time (LBT) is the greatest timestamp (in seconds) of the LBST for all IP sessions mapping to the service for this user. Optionally (and by default), the value for service idle is added to the maximum interval of the LBST when calculating the LBT. The reason for adding the service idle timeout to the duration is because the duration calculation already includes the intermediate (between IP sessions) idle intervals, so the last idle interval should also be included.
- If the service object is destroyed due to service idleness, the calculation for usage is:

  Usage = LBT - FBT

  If the service object is destroyed due to an asynchronous event such as user logoff, the calculation for usage is:

  Usage = LBT - FBT

  or

  Usage = Async Event Timestamp - FBT

  whichever is smaller.

  If the service object runs out of quota, the calculation for usage is:

  Usage = LBT - FBT

  or

  Usage = OutOfQuota Timestamp - FBT

  whichever is smaller.

Note If the user runs out of quota, but the user refreshes the quota before the service idles out, the periods (or gaps) of zero quota is not included in the usage calculation.

When a Service Duration Billing Service is a member of a billing plan, and an accounting definition is in service and downloaded to a CSG module, you cannot modify the basis or meter configuration. You are instructed at the console to configure **no inservice** on the downloaded Accounting definitions.

Reporting to the BMA

For service duration billing, the unit for quadrans reported to the quota server and BMA is seconds. In messages sent to the BMA on a per-IP-session basis (such as TCP statistics), the prepaid TLVs (Session ID, Service ID, Quadran) are present; the value for quadrans in the Quadran TLV is zero because the duration is based on service, not individual sessions or the sum of durations of individual sessions.

Out of Quota

When a subscriber runs out of quota, the CSG terminates the user sessions mapped to the service using the same asynchronous session kill mechanism that is used when a subscriber User Table entry is deleted. The CSG reauthorizes when the remaining time is low (instead of 0) in order to more quickly determine session processing when zero quota is reached.

Report Billing Plan ID to BMA and Quota Server

The CSG reports the billing plan identifier string in BMA records, and in messages to the quota server.

Asynchronous Quota Return

The Asynchronous Quota Return feature allows the quota server to request the CSG return quota for a defined user and service, and send a Quota Return.

Asynchronous Service Stop

The Asynchronous Service Stop feature allows the quota server to request the CSG to stop a prepaid service for a defined user and service, and send a Service Stop.

RADIUS Features

The following changes to RADIUS functionality exist in the CSG:

- RADIUS Monitor provides a way to insert the CSG into the RADIUS flow without changing the authentication, authorization, and accounting (AAA) or Network Access Server (NAS) addresses in the network. The CSG watches for RADIUS messages flowing through it that match the configured rule.
- The CSG can accept the RADIUS accounting on/off, and can clear the sessions established from the respective NAS.

- RADIUS Proxy provides a way to remove the possibility of routing errors (RADIUS is targeted at the CSG addresses directly), and must be used in place of the **radius monitor** command when the CSGs are being load-balanced.
- RADIUS Proxy supports both RADIUS Access and RADIUS Accounting.

Note A single CSG environment does not require a route in the Content Services Module (CSM) pointing to the CSG RADIUS virtual IP address. However, in a fault-tolerant setup, or a multiple-CSG setup, a route to the CSG RADIUS virtual IP address is required. This route must point to the Alias IP of the appropriate CSG.

See "Configuring RADIUS Support: Learning Who the Subscriber Is" section on page 5-1 or more information on RADIUS features.

User Profile Retrieval from RADIUS Access Accept or Accounting Request

The user profile (billing plan) can be specified in a RADIUS message using the Cisco subattribute 1 VSA. The format of the VSA is:

csg:billing_plan= *billing_plan_name*

The *billing_plan_name* can be null, to indicate a postpaid user. Otherwise, the billing plan name must be sent as an uppercase string to match a configured billing plan on the CSG.

The billing plan may be included in the RADIUS Access-Accept or Accounting-Request message.

If the RADIUS Access-Accept includes the billing plan, the user ID (RADIUS attribute 1 or 31, as configured) must also be included; otherwise, the CSG is not able to associate the billing plan with the user.

Use the **user-profile server radius** command to retrieve the billing plan from the RADIUS message.

Reporting RADIUS Attributes to Quota Server

Specified RADIUS attributes are included in messages to the quota server, in addition to being reported to the BMA.

User Cleanup on RADIUS Accounting Start

A subscriber's connectivity attributes may change over time without a RADIUS Accounting Stop arriving to close down the previous accounting. Instead, it is possible that a new RADIUS Accounting Start or Interim Accounting message may arrive with the updated information. Some customers may choose to close all of the user's services if a significant change has occurred in the user's status.

With the **radius start restart session-id** command configured, the CSG deletes the user entry as if it had received a stop, closes all of the subscriber's services, and creates a new entry.

To avoid deleting the user entry because of a retransmission of the RADIUS message, the **radius start restart session-id** command specifies an attribute to detect duplicate messages. If the contents of the attribute in the message match the contents of the previous message, the existing entry is not deleted.

Processing Multiple RADIUS Accounting Stops

For enhanced network connectivity options, such as secondary packet data protocol (PDP) contexts, the NAS sends multiple RADIUS Accounting Stop messages. In the case of secondary PDP contexts, for example, the NAS sends a RADIUS Accounting Stop as each context is terminated.

Only the final stop, which contains an attribute indicating it is final, should be used to remove the subscriber from the CSG User Table. The CSG support for this functionality allows the specific attribute to be configured. If this function is configured, the CSG processes only the RADIUS Accounting Stop that contains the configured attribute. The contents of the specified attribute are not examined.

Monitor

RADIUS Monitor provides a way to insert the CSG without changing the AAA or NAS addresses in the network. The CSG monitors the traffic between the RADIUS client and the RADIUS server, and watches for RADIUS messages flowing through it that match the configured rule. The address of the server must be configured.

Optionally, a RADIUS key is configured. If the key is configured, the CSG parses and acts on the message only if the RADIUS Authenticator is correct. If the key is not configured, the CSG always parses the message. The message is forwarded regardless of the key being configured or correct.

Here is a sample configuration:

```
ip csg user-group U1
  radius userid User-Name
  radius monitor 10.2.3.4 1234 key cisco --> Address, Port, and Key for RADIUS AAA Server.
  radius monitor 10.2.3.9 1234 key cisco2
  radius monitor 10.2.7.4 3901 key cisco --> Multiple AAA destinations can be monitored.
```

All RADIUS messages, including Access messages, are forwarded, except when the IP or UDP headers specify a length larger than the physical packet size.

Proxy

The CSG enhances the proxy function to allow operation with clients that use large numbers of port numbers. RADIUS Proxy provides a way to remove the chance of routing errors (RADIUS is targeted at the CSG addresses directly), and must be used in place of Monitor when the CSGs are being load-balanced.

The old proxy function is still supported, and operates as before if you configure it in the old way. But Cisco recommends that you use the new proxy support.

The RADIUS client sends messages to the configured CSG (virtual) address. The CSG accepts messages for all ports on the configured address. The address of the RADIUS server is also configured. Optionally, a RADIUS key is configured. If the key is configured, the CSG parses and acts on the message only if the RADIUS Authenticator is correct. If the key is not configured, the CSG parses the message with no conditions. The message is forwarded regardless of the key being configured or correct.

All RADIUS messages (including Access messages) are forwarded except when the IP or UDP headers specify a length larger than the physical packet size.

There is a limit of 64,511 clients, where a client is defined by its IP address and port.

The CSG address specified in the command must not be defined by other CSG commands (for example, **vlan, content**, **radius monitor**, or other **radius proxy** commands).

Reporting of RADIUS Attributes

You can specify a set of attributes to be extracted from RADIUS Accounting Start messages for each subscriber, and reported with each transaction record. The CSG reports these attributes to the BMA and to the quota server. The CSG extracts these attributes from the RADIUS Accounting Start, and refreshes (replaces) its stored attributes whenever a RADIUS Interim Accounting message is received, to ensure that the latest user information is stored.

You can use Arbitrary RADIUS attributes to understand where a user is connecting to the network, and for correlation purposes. Examples of these attributes and their uses include:

- NAS-IP-Address (4) identifies the gateway that provides accounting control for the subscriber. Examples of such devices include the gateway GPRS support node (GGSN), Packet Data Serving Node (PDSN), HomeAgent, Cisco AS5300, and so on.
- SGSN IP (26/10415/6) identifies the SGSN the subscriber is accessing.

Note Note that, in this case, the CSG must be configured to report all RADIUS attribute 26 (the Vendor-Specific Attribute, or VSA) instances.

- Acct-session-ID (44) uniquely identifies the session on this NAS and can be used for correlation to GGSN accounting records.

Note The CSG cannot separate the RADIUS attribute 26s—the CSG sends all of them.

Configuring RADIUS Inspection: Endpoint

The CSG RADIUS features require that you configure the NAS to direct RADIUS messages to the CSG IP address (or to the alias address if this is a redundant configuration). You must also configure your NAS to the specific CSG port number. The following example illustrates the configuration:

```
module csg 3
  radius endpoint 1.2.3.4 key secret
```

You can still use the existing **radius key** and **radius acct-port** commands in CSG user group configuration mode to configure the CSG as a RADIUS Accounting endpoint, but we recommend that you use the **radius endpoint** command in module CSG configuration mode. The CSG 3.1(3)C5(5) supports both endpoint configuration methods. However, if you plan to use RADIUS PoD with RADIUS endpoint, then you must use the **radius endpoint** command in module CSG configuration mode.

We do not recommend using both configuration methods in the same environment.

HTTP URL Redirect

Prior to this release, HTTP redirect was done exclusively at the IP layer using the CSG redirect NAT feature. Layer 7, URL redirect support is now available. In a redirect scenario, the CSG responds to the HTTP client with response code 302 and a URL to which the client should redirect. The redirect URL is administratively configured using the CLI, or is specified by the quota server during service authorization or content authorization processing.

In the case of service authorization or reauthorization and content authorization, the quota server reply contains the REDIRECT-URL action code and the redirect URL. This action code has previously been used for WAP 1.x redirection and is being reused for HTTP. It may be appropriate in some network configurations for the quota server to return a single redirect URL for both WAP and HTTP.

In support of configurations where this is not appropriate, information is supplied in the service authorization or reauthorization request and the content authorization request to identify the request as being for WAP or HTTP. The service (re)authorization request reports when sessions are of type WAP or HTTP in a new, SessionType TLV. This optional TLV is only sent for WAP and HTTP sessions. Content authorization requests include HTTP and WAP specific information that distinguish the type of content.

A redirect URL returned from the quota server in a service authorization response, or in a content authorization response with the REDIRECT_URL action code, takes precedence over a redirect URL that is configured with the CLI. The CLI-specified redirect URL is used when the quota server responds with the FORWARD action code.

Redirection for HTTP URLs operate as follows:

Two new CSG variables are used to control the amount of time and the number of redirects that the CSG offers. This is provided to allow the operator to tune the time for redirection. This is especially useful when redirect occurs in the middle of a large index page, but before all GET's have been transmitted.

- CSG_REDIRECTS_INTERVAL defines the length of time for which the CSG should redirect.
- CSG_REDIRECTS_MAX defines the number of requests that are redirected before the CSG stops redirecting, but within the interval time.

The CSG starts the interval timer when the first request is redirected after it has received no quota. This counter is reset, and the timer is stopped after another quota grant of zero is given.

As an example, assume that CSG_REDIRECTS_MAX is set to 15 and CSG_REDIRECTS_INTERVAL is set to 8 seconds. If you receive a Service Auth Response with zero quadrans, and you have redirect information, then redirection occurs when you run out of quota (assuming you have not received quota since). The CSG_REDIRECTS_INTERVAL 8-second timer starts after your first redirect. Therefore, request 1 is redirected, and up to 14 more requests can be redirected, if they occur within the 8 seconds after the first redirect.

Note This operation functions in the same manner for NAT redirects.

Support for URL-Rewriting for AoC

An advice of charge (AoC) solution uses a quota server and a notification/top-off server. The quota server is responsible for telling the CSG to block client requests and redirect them to the notification server when the client must make a decision to pay for the service. It is also responsible for telling the CSG to allow the client request to flow when the client has agreed to pay. The notification/top-off server is responsible for communicating fees to the client and providing the option to pay. The client's payment decision must be communicated from the notification server to the quota server.

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 client request. The notification server appends a string beginning with a token to the originally requested URL and sends it to the client as part of a redirect reply after the client has agreed to pay. The CSG 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 client has agreed to pay for the request. It responds to the CSG with a FORWARD action code in the content authorization

response. The CSG detects the token, creates a new GET request containing the original URL with the token and any characters following it removed, and sends the GET on behalf of the client. The token must be known by the CSG, the quota server, and the notification server. It is administratively defined on the CSG using the CLI. The token should be chosen carefully to ensure that it is only present in URLs rewritten by the notification server and not in other client requests.

WAP AoC URL Appending

Whenever a content authorization response contains a REDIRECT_URL action code for a WAP content authorization request, the CSG can optionally append the originally requested URL to the one returned by the quota server. For example, if the client originally requested the URL http://www.yahoo.com/home.wml and the quota server returns http://www.yahoo.com/charges.wml in a REDIRECT_URL content authorization response, then the CSG would send the URL http://www.yahoo.com/charges.wml?www.yahoo.com/home.wml as part of a redirect message to the client.

The default behavior is to pass the redirect URL to the client as specified by the quota server without modification. To enable this feature, set the CSG_WAP_APPEND_AOC_URL environment variable using the Cisco IOS CLI.

Fixed Attribute CDRs

In support of some legacy billing systems, the CSG provides a fixed attribute format for WAP CDRs. The same set of attributes are reported in each CDR regardless of Wireless Session Protocol (WSP) protocol data unit (PDU) type. CDRs contain zero-length attributes when there is no information to report, but the same set of attributes are always reported in the same sequence. Some new attributes are included that convey provisioned information.

Support for Port Number Ranges

When you define content on the CSG, you can define a single port number, or a range of port numbers. This eliminates the need to define a content for each port.

When defining a range of port numbers, choose a range that is applicable to the associated policies. For example, defining a range of port numbers from 80 to 8080 for **accounting type http** means that the CSG must perform intensive HTTP inspection on many intermediate ports, ports that might not be expected to carry HTTP flows. HTTP inspection of such a high volume of non-HTTP flows can result in excessive processing by the CSG, as well as generating many CDRs that the customer had not planned for.

Support for the Cisco Persistent Storage Device

The CSG supports the Cisco Persistent Storage Device (PSD). The PSD provides persistent storage capabilities to the CSG, and allows the CSG to store data on the PSD's internal hard drive.

Under normal conditions, the CSG sends content billing records to the mediation partners' servers. If, for any reason, those servers become unreachable, records are sent to the PSD for safekeeping until contact is reestablished with the Billing Mediation Agent (BMA). Once contact is reestablished, the CSG retrieves the records from the PSD, and forwards them to the BMA.

Storage

Under normal conditions, the PSD provides backup capabilities when necessary—for example, during network outages. The PSD stores the payload from the packet in a queue, and is unaware of the content or format of that data, so that the data can be retrieved exactly as it was sent.

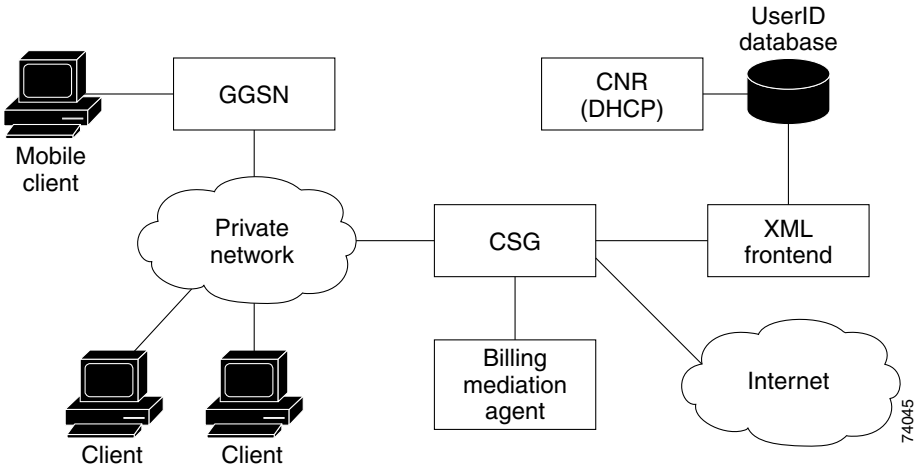
Retrieval

Once the CSG determines that the regular data server is again reachable (in this case, the BMA), it retrieves the stored data from the PSD. The data is returned to the CSG in the same order and form as it was deposited. The CSG is responsible for maintaining order, if necessary, or of mixing retrieved data with incoming "live" records. Once the CSG acknowledges to the PSD that it has successfully sent the data to the client server (the BMA), the PSD deletes that data. The PSD stores the data until it receives this acknowledgement.

Postpaid Billing

Figure 1-1 illustrates simple traffic flows between the various components in a simple postpaid CSG environment.

Figure 1-1 Traffic Flow Between Client and Server



Clients send requests that pass through a private network, or through a GGSN, before they reach the Internet.

The CSG monitors data flows and generates accounting records that can be used to bill customers at a content-level granularity. The CSG sends the accounting records to a Billing Mediation Agent (BMA), which formats the records as required by the customer's billing system.

User IDs are obtained from RADIUS accounting records, or by querying the user database.

BMA Load Sharing

The CSG can support multiple BMAs. This is useful in environments in which the number of billing records sent by the CSG could overwhelm a single BMA.

Note Multiple BMAs cannot have the same IP address.

The CSG maintains GTP' sequence numbers for each BMA.

All of the billing records for a given user are sent to the same BMA.

Quota Server Load Sharing

The CSG supports multiple quota servers. This is useful in environments in which the number of quota transactions sent by the CSG could overwhelm a single quota server. Multiple quota servers can be simultaneously active, and the CSG assigns a quota server to each user. All quota transactions for the user are done with the same quota server.

When a quota server fails, all users associated with that quota server are distributed among other quota servers.

Note Multiple quota servers cannot have the same IP address.

Support for the CSG MIB

The CSG supports the CISCO-CSG-MIB implemented in Cisco IOS.

HTTP 1.0 Content Billing

The CSG enables you to bill users for individual transactions by discriminating on a per-object basis, and on a per-user basis. Unlike traditional billing models, which bill for broad classes of traffic, this service enables differentiated billing based on the actual object being requested. You can even bill objects at different rates to different customers. For example, you can bill advertisements to the advertiser rather than to the end user.

HTTP 1.1 Content Billing

The CSG records each request over a persistent HTTP 1.1 session separately.

FTP Billing

The CSG supports both postpaid and prepaid FTP protocol-aware billing. The CSG can generate TCP billing records for FTP connections, and records that report FTP-specific information, such as the filename.

Users can define **basis fixed** and **basis byte** prepaid billing services for FTP.

Note There is no regular expression (map) support for differentiating FTP services.

FTP requires a control TCP connection to well-known server port 21.

Non-HTTP Traffic

For non-HTTP traffic, the CSG records information about data transfer based on flow information.

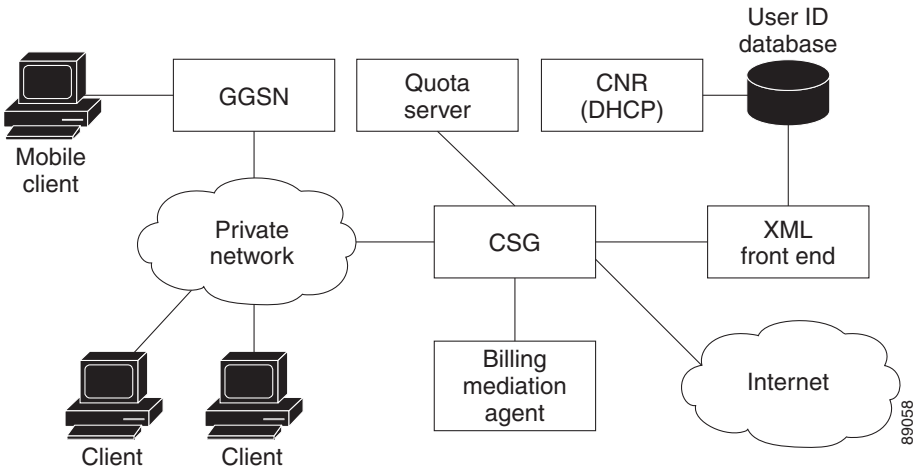
Prepaid Content Billing and Accounting

In addition to postpaid billing, the CSG provides prepaid content billing and accounting. You can configure multiple prepaid billing plans, and subscribers can choose the plan that best meets their needs. Each subscriber can use only one billing plan.

The CSG uses a BMA to interface with a billing server. At the end of each transaction, the CSG sends a billing record to the BMA, indicating the content accessed and the amount deducted. The BMA logs the information in the user's bill.

The CSG uses a quota server to keep track of the quota that is left in the user's account. Each CSG supports one quota server and multiple idle backup quota servers. The CSG allows multiple groups of users on each quota server, with one quota manager for each user. Figure 1-2 illustrates a typical CSG prepaid content billing network.

Figure 1-2 CSG Prepaid Content Billing Network



Quota is provided by the Quota Manager, on request for quota by the CSG. This quota is either for an initial service connection, or for subsequent re-authorization when the original/last quota grant is depleted. The Quota Manager is allowed to provide a value in the range of 0 to 2,147,483,647 (0x00000000 to 0x7FFFFFFF). This value— called "quadrans"— comes in three forms, "basis byte" for volume-based billing, "basis fixed" for event-based billing, and "basis second" for duration-based billing.

When quota is depleted to zero, the user can no longer access the service.

Quota is held on a per-service basis. Therefore, if a user is connected to more than one service, the CSG stores quota for each service that is open.

Once the user finishes a session by closing the bearer session (a RADIUS Accounting Stop is sent from a GGSN), the service is stopped, and any unused quota is returned to the Quota Manager.

While this prepaid system is in operation, the normal postpaid system runs by sending CDRs to the BMA.

The following example flow illustrates a basic prepaid flow between the CSG and the Quota Manager:

1. The NAS/GGSN sends an Access Request to the RADIUS Server.
2. On receipt of an Access Accept from the RADIUS Server, the NAS/GGSN sends an Accounting Start to the RADIUS Server.
3. The CSG creates a user entry, and links the user IP address to either the username or Calling Station ID (depending upon the configuration of the CSG).
4. The CSG sends a User Authorization Request to the Quota Manager.
5. The Quota Manager replies to the CSG with a valid billing plan for the user (User Authorization Response).
6. User traffic begins to flow from the NAS/GGSN toward the requested server.
7. The CSG sends a Service Authorization Request to the Quota Manager, requesting quota for this connection.
8. The Quota Manager returns a given quota in the Service Authorization Response (if it has quota to give).
9. The user traffic passes the CSG to the service, and prepaid billing begins.
10. A Service Stop occurs if either the NAS/GGSN sends a RADIUS Accounting Stop, or if the content and service idle out.
11. Service Stop provides the quota used and returns any remaining quota.

Support for WAP Traffic

The CSG can intercept WAP traffic and generate reports that include contextual WAP information and counts of the bytes transferred. WAP functionality provides protocol-level prepaid and postpaid billing, including the following functionality:

- Billing CDRs for WTP and WSP in support of WAP 1.2—The ability to generate billing records for each WAP GET, POST, PUSH/CONFIRMED PUSH, ABORT and REPLY PDUs, as well as a summary report at WAP Disconnect. Records include URL, User Agent, source and destination IP, separate IP byte and PDU counts from both the initiator and the responder. (The PDU count is not the same as the packet count. Multiple WAP PDUs can share a single packet.)
- Prepaid billing for WTP and WSP in support of WAP 1.2, including the ability to differentiate WAP browsing from MMS, and to exclude charging for MMS.
- Top-up capability using URL redirect.
- URL-map support for WAP.
- Support for multiple services.
- WAP 1.2.1 HTTP support: The CSG HTTP support is compatible with WAP 1.2.1 (HTTP over WP-TCP) traffic.

RADIUS Accounting Attribute Reporting

The CSG allows you to configure RADIUS accounting attributes that are reported to the BMA in every CDR. You can specify as many attributes as you desire. The attributes are copied from the RADIUS accounting message, and sent in each billing message to the BMA.

These attributes are configured by their standard number, as shown in the following example:

```
ip csg accounting USER-BMA1
  user-group GROUP1
  agent activate 2 sticky 30
  agent 210.0.0.102 3386 1
  report radius attribute 3
  report radius attribute 5
  report radius attribute 7
  report radius attribute 44
  inservice
```

It is possible for users to select so many attributes that the total message size is greater than a single UDP packet. The CSG supports continuation messages that allow users to select as many attributes as they desire. A continuation message includes a correlator, a continuation number (so messages received out of order can be reordered), and an indication of the final message.

Note The CSG only examines the standard RADIUS attribute number and is not aware of any special formatting or subclassing for Vendor-Specific Attributes (VSAs). If a VSA is desired, then the CSG reports all VSAs (that is, attribute 26s).

If the configured attributes change, only new RADIUS requests are subject to the new attributes. Attributes already saved for a user continues to be reported.

When a RADIUS Start request is received, any attributes received from a previous Start request are deleted. If there are multiple instances of an attribute, they are all reported. Attributes are reported in the order they exist in the RADIUS message.

Obtaining User IDs

The CSG uses two methods to obtain user IDs:

- The CSG can use an external user ID database to map IP addresses to user IDs. When the CSG receives a packet with an unknown IP address, and it needs to associate the IP address with a user ID, it queries the database. If the user ID is not available, the CSG generates an accounting record without it.
- The CSG can act as a RADIUS Accounting Server or a proxy for RADIUS accounting messages. The CSG can examine the accounting messages to determine user IDs. (The CSG does not support full RADIUS accounting.)

After identifying a user, the CSG associates the user's IP address with the user ID, and, if a quota server has been defined, tries to download the user's profile. The profile indicates whether the user is postpaid or prepaid, as well as the user's billing plan. If the user is a prepaid user, the CSG downloads also the user's quota, then forwards the user's flows.

Learning Client IP Addresses Using Inspection of X-Forwarded-For Headers

If your network is configured with a gateway or proxy placed between the client and the CSG, you can configure the CSG to determine the client's IP address by inspecting the X-Forwarded-For header (for HTTP connections only).

Filtering Accounting

Filtering lets you configure the following functionality:

- Specify sites to include or exclude for billing information. Specific sites are identified by URL, IP address, protocol, or port parameters.
- Specify a customer string to insert in billing records for the specified site.
- Specify that protocol-specific information is generated for billing records to a specified site.

WAP URL Mapping

The CSG maps are used to match URLs or headers against a pattern, to determine whether flows are to be processed by the CSG accounting services. The CSG provides URL mapping and filtering for WAP. A simple example configuration follows:

```
ip csg map WML url
  match url *.wml

ip csg policy WAP_POLICY
  accounting type wap
  url-map WML

ip csg content WAP_CONTENT
  up any udp 9201
  policy WAP_POLICY
  inservice
```

For WAP 1.x, URL maps take precedence over access lists.

Advice of Charge and Per-Event Filtering

The CSG supports the following per-event actions that require instruction from the billing system, and are supported as variations of the same design:

- Advice of Charge: Uses NAT to redirect the user data flow to a server that can ask the user to verify the charge before serving the content.
- Per-Event Filtering: Permits or denies a transaction as directed by the quota server.

To enable these functions, use the **authorize content** command in CSG service configuration mode.

SMTP and POP3 Data Mining

SMTP is the Internet mail transfer protocol that operates over TCP with port 25. End users send messages using SMTP, and it is also used to transfer messages between SMTP gateways (or relays). POP3 is a common protocol used to retrieve Internet mail from a mail server. POP3 also operates over TCP and typically uses port 110.

SMTP and POP3 messages consist of 3 parts:

- Envelope— The SMTP and POP3 commands and responses.
- Headers— RFC2822 headers that appear as contents to the SMTP and POP3 protocols. The RFC2822 headers are of the form “header field name: header field body”. Some common header field names are “To”, “From”, “Date”, “Subject”, “Cc”, and “Bcc”.
- Body—The part of the message that appears as contents to the SMTP and POP3 protocols, but does not include headers. The headers and body of the message are separated by a blank line (for example, <CR><LF><CR><LF> in RFC 2822).

The CSG inspects SMTP and POP3 messages and reports all RFC 2822 header field names and bodies that appear in the header section of the message (before the body of the message). SMTP and POP3 envelope information is not reported, with the exception of the SMTP return code from the DATA command. For SMTP, the sender and recipients in the SMTP MAIL and RCPT commands are not reported, but the values from the “To”, “From”, “Date”, “Cc”, and “Bcc” headers in the contents of the mail message are reported to identify senders and recipients.

Because the amount of information in the header section may be greater than an IP packet encapsulated in an Ethernet frame, the information may span multiple records by using the CSG Continue Data Record type. Because the amount of information in a single header field may also be greater than an IP packet over Ethernet, the CSG Report String Attribute reports also has a continuation option. This means that information for a single header may span multiple CSG Report String Attribute reports which may span multiple CSG Data Records.

Note If a TCP connection carries multiple mail messages, each mail message generates a separate SMTP or POP3 Data Record (plus Continuation Data Records if necessary).

Redirect Flexibility

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

RADIUS Proxy Support

The CSG can act as a RADIUS proxy, forwarding all RADIUS accounting messages it receives to a configured RADIUS server. When the RADIUS server acknowledges a message with an ACK, the CSG forwards the ACK back to the client.

The CSG allows you to specify only one RADIUS server, and the same RADIUS password is used throughout.

The CSG supports only a small number of clients (IP ports). Therefore, configure your GGSNs, for example, to reuse the same source port for all messages, rather than using a new port for new messages.

Cisco recommends that you use this support with a small number (approximately 20) of RADIUS senders (defined by an IP address and port number).

HTTP Records Reporting Flexibility

The client's IP address is included in the HTTP Header message. This enables the BMA to identify the client by user ID (as well as by IP address) immediately, without having to wait for the HTTP Statistics record.

You can configure the CSG to send the HTTP Header message as soon as it is generated, rather than batching it until an entire packet is filled. This reduces latency and notifies the BMA about the client's transaction as quickly as possible. This type of reporting is more efficient, but provides less information, and should be used only when the BMA needs to react to the client's activity very quickly.

You can configure the CSG to not send the HTTP Statistics message. This reduces the load on the BMA, and is useful when the billing policy depends only on the event and does not require detailed statistics. Note that the CSG still sends the HTTP Statistics message if the session fails (for example, if a Reset [RST] is received without a Finish [FIN], or if the session times out).

HTTP Error Code Reporting

The CSG reports HTTP-specific information about the request, such as the URL, as well as HTTP error codes (that is, response codes of 300 or higher).

Intermediate Billing Records

Typically, the CSG sends two billing records for each HTTP session. The CGS sends one record for all non-HTTP sessions, when the sessions end. However, for long-lived sessions, you might want to monitor the progress of the session. To monitor long-lived sessions, you can configure the CSG to send intermediate billing records after a specified number of seconds, or after a specified number of bytes, whichever occurs first.

Intermediate counts are also correlated between the active CSG and the standby CSG.

The CSG supports intermediate billing for FTP, HTTP, IP, TCP, and UDP. The CSG does not support intermediate billing for WAP or e-mail protocols (such as IMAP, POP3, and SMTP). The CSG does not support intermediate billing for RTSP control sessions unless the video/audio traffic is also transported over the control session.

Packet Forwarding

The CSG configurations allow users to specify multiple gateways; one for each of the VLANs. However, only one default gateway can be in effect at a time. So, the second default gateway configured does not take effect until the first default gateway is unconfigured. Generally, you should only specify one default gateway to avoid confusing which default gateway is being used by the CSG at any given moment.

All traffic that passes through the CSG (including the traffic to and from the CSG [GTP' traffic]) is routed and forwarded based on the VLAN interface, route, and gateway statements specified under the **module CSG** commands.

For example:

```
Module ContentServicesGateway 6

vlan 10 server
ip address 10.250.0.1 255.255.0.0
gateway 10.250.1.1
```

```
vlan 251 client
ip address 10.251.0.1 255.255.0.0
route 10.200.0.0 255.254.0.0 gateway 10.251.2.11
```

For packet forwarding, the following logic applies:

1. If the destination IP address of the packet is a subnet adjacent to one of the VLAN interfaces configured, the CSG tries to map the destination IP address to a MAC address, using the Address Resolution Protocol (ARP), and forwards the packet.
2. If the destination IP address of the packet is not subnet-adjacent, the CSG forwards it based on the routes specified.
3. If there are no matching routes, the CSG forwards the destination IP address of the packet based on the default gateway as defined. If no default gateway is specified, and if the CSG does not have an ARP entry in its ARP cache for the MAC address, the CSG drops the packet.

For packets that originated from the CSG (GTP'), the CSG uses the VLAN interface that resulted from the above forwarding logic to forward the packet. In the configuration example above, if a packet is forwarded using the default gateway, it uses the source interface of 10.250.0.1 to forward the packet.

In general, the default gateway is used to specify on the server VLAN to direct client traffic to the Internet. This prevents you from having to specify lots of subnets, as some of them are not easily identified. However, the default gateway can be placed in either the client or server VLAN.

In addition to using route/gateway to forward the traffic, client/server traffic can also be forwarded using next-hop as specified in the billing policy. For example:

```
ip csg policy FORWARD_PKT
      accounting customer-string CLIENT-TRAFFIC
      next-hop 1.1.1.1

   ip csg content FORWARD-INTERNET-TRAFFIC
      ip any
      policy FORWARD_PKT
      inservice
```

In the above example, if traffic hits this content and policy, the traffic is forwarded to next-hop router that has an IP address of 1.1.1.1.

The CSG supports next-hop packet forwarding for all protocols.

Packet Counts

The CSG reports the number of IP bytes uploaded and downloaded, the number of TCP bytes uploaded and downloaded by the application, and the packet counts (or PDU counts for WAP records). These counts exclude the IP and TCP headers, as well as retransmissions.

Additional Features

The CSG provides these additional features:

- More than one CSG can run in a Catalyst 6000 series switch or Cisco 7600 series router chassis.
- The CSG fault-tolerance support allows two CSG modules (in the same or in different chassis) to be configured in the active and standby modes.

Dependencies and Restrictions

- The CSG does not support IP packets larger than 1500 bytes.
- The CSG supports up to 256 total VLANs (client and server).
- The CSG supports up to 1024 content/policy pairs configured under services within a billing plan. Note that if two billing plans contain the same service, the content/policy pairs are counted multiple times.
- The CSG supports up to 4000 content definitions (or virtual server definitions in postpaid); up to 1000 unique IP addresses (virtual IP addresses plus VLAN IP addresses and alias IP addresses); and up to 127 contents for each unique IP address (each content counts as one and each VLAN/alias IP address counts as four).
- The CSG supports up to 255 services and up to 1024 services rules.
- The CSG supports up to 16,000 access control list (ACL) items.
- Up to six Cisco CSGs and/or CSMs can be installed in a Catalyst 6500 series switch or Cisco 7600 series router chassis.
- You cannot cascade two or more CSGs. For more information, see the TCP compliance exceptions in the "Layer 7 Inspection (accounting type=specific protocol)" section on page D-1.
- The CSG runs with Cisco IOS Release 12.1(12c)E4 or later.
- For IP Layer 4 and Layer 7 inspection, the CSG volume counters wrap at 0xFFFFFFFF (268435455 bytes). The volume counters are 32 bits unsigned.
- During chunked POST processing, the CSG can buffer up to 29,696 packets for all users. Therefore, the maximum theoretical POST size for a single user would be 44.5MB (29,696 packets at 1,500 bytes/packet). However, this theoretical maximum is reduced if other users have active chunked POSTs in process.
- For RTSP, keep the following considerations in mind:
  - RTSP requires a control TCP connection to server port 554.
  - RTSP offers minimal support for TCP-interleaved and HTTP-tunneled transport: only the first stream URL is reported. For authorized content, the first stream is sent to the quota server. The action must be identical to that sent for the control connection because the stream is interleaved on the control connection, and cannot be terminated/charged independent of the control connection.
  - RTSP supports multiple transport choices. RTSP clients and servers negotiate the transport choice dynamically before the stream is started.

    One such choice is to interleave the stream with the control channel. In this mode, the CSG cannot map the transport connection to a different policy, and URL mapping cannot be supported.

    The other shared mode for RTSP is to use a single HTTP connection. RTSP tunneled over HTTP has the same limitation as interleaved RTSP: The stream cannot be mapped to a policy different from the control connection, as both of them share the same transport.
  - For RTSP, all policies must use the same access control list (ACL) and the same next-hop IP address.
  - The CSG does not support multicast RTSP.
  - The CSG does not correlate streams described in a container file, for instance, SMIL.

  - The CSG parses only the RTSP control session. When multiple RTSP streams are multiplexed over the same transport, the CSG reports cumulative statistics for all such streams.
  - If RTSP URL mapping and filtering is used, and multiple RTSP streams share the same transport channel, the CSG generates a single content authorization request, and the request contains all URLs carried over that stream. Also, the RTSP stream CDR contains URLs for all streams that are multiplexed over the same transport channel.
  - If an RTSP proxy is used, the CSG should be placed on the client side of the proxy. If the CSG is placed on the network side of the proxy, the CSG sees packets originating from the proxy, and the CDRs reported contain the proxy's IP address, instead of the client's.
  - After a CSG failover, existing RTSP UDP streams continue to operate if a catch-all content rule was defined to pass unknown UDP flows. However, RTSP correlation for those streams ceases, and billing is limited only to the parameters defined for the catch-all content rule. New RTSP connections are processed normally.
  - For RTSP, all policies must use the same access control list (ACL) and the same next-hop IP address.
  - For RTSP, the policy used to determine the next hop address is chosen based solely on ACLs, not URL maps. As a result, you can choose the next hop from one policy for routing and from a different policy for billing.

- When you configure multiple fault-tolerant CSG pairs, *do not* configure multiple CSG pairs to use the same FT VLAN. Use a different FT VLAN for each fault-tolerant CSG pair.
- If you have a pair of CSG cards and a pair of CSM cards in your network, *do not* configure both the CSG pair and the CSM pair to use the same FT VLAN. Use a different FT VLAN for each pair. If you configure the CSG pair and the CSM pair to use the same FT VLAN, then either service, the CSG or the CSM, is down in the standby mode.
- Traffic coming from an unknown source MAC address on the client-side or server-side VLAN is dropped by the CSG.
- The IP address in the content definition cannot be in the same subnet as the IP address of the client VLANs.
- When you configure redundant CSGs, the backup CSG must use the same software release as the primary CSG, or a later software release. If your CSGs act as backups for each other, they must all use the same software release.
- Advice of Charge (AoC) is supported for only HTTP and WAP.
- When performing AoC for a TCP connection carrying pipelined HTTP requests, the CSG responds with the redirect to the client as soon as the quota server requests the redirect. This could result in the redirect arriving at the client before responses for previous requests arrive, and the client might associate the redirect with a different request in the pipeline.
- When a CSG prepaid service is configured for Advice of Charge (AoC), the weighting value for charging the content is not determined until the CSG processes the Content Authorization Response. For SMTP billing (**accounting type smtp**), the CSG does not send the Content Authorization Request until it processes the SMTP DATA command. If the CSG does not process the SMTP DATA command for a session, then the CSG does not charge the session for volume and event billing.
- The CSG does not support multiple protocols under a single service definition. Do not configure a CSG service with more than one accounting protocol type.
- Service verification is supported for only HTTP and WAP.
- FTP requires a control TCP connection to well-known server port 21.
- For WAP, keep the following considerations in mind:

  - All policies must use the same access control list (ACL) and the same next-hop IP address.
  - For WAP1.x, the policy used to determine the next hop address is chosen based solely on ACLs, not URL maps. As a result, you can choose the next hop from one policy for routing and from a different policy for billing.
  - The CSG supports only URL maps for WAP; header maps are not supported. You cannot use the CLI to configure header maps for WAP services. Policies defined as accounting type **wap** can accept only URL map definitions. For WAP 1.x, URL maps take precedence over access lists.

- For IMAP support, message tags cannot be longer than 100 bytes. If the CSG encounters a message with a tag length greater than 100 bytes, only IP and TCP upstream and downstream byte counts are reported.
- The RADIUS Accounting Start message which specifies the NAS IP to which to send the PoD message must be received on an IP address specified by the **radius proxy** or **radius endpoint** command configured in module CSG configuration mode.
- For CSG type=HTTP parsing, the CSG imposes the following restrictions:
  - The HTTP method must be initiated by the same endpoint that initiated the TCP connection (by the same side that sent the TCP SYN); the impact is that the client request transfers no data.
  - The maximum HTTP transaction volume is 268435455 bytes. If this length is exceeded, the CSG invokes Layer 4 billing for the remainder of the connection.
  - HTTP request parsing is limited to 64,000 bytes. Any headers beyond this limit are not recognized and are not used in matching URL or header maps.
  - Sharing of the same port for both HTTP and HTTPS is not supported. However, SSL can be tunneled over HTTP using the Connect method.
  - There are two types of maps for HTTP, URLs and headers.
  - If policy type=http, the only TCP option that is passed through the CSG is the Maximum Segment Size (MSS). The other options are dropped. This includes Wireless Profiled TCP Options (example: SACK) that are used with WAP2.0 implementations.
  - With RFC2818, an HTTP session can become encrypted via the UPGRADE method. If Layer 7 billing is defined for the HTTP port, then the session might time out when the UPGRADE occurs, because the CSG code cannot parse the encrypted data after TLS negotiation.
  - For an HTTP transaction, if any quota is granted, the CSG always sends the following packets, even if there is insufficient quota:

    - The first request (GET, POST, any other method) from the client—headers plus the part of the message that arrives before quota is granted.

    - The headers plus one packet of the response from the server.
  - For HTTP Layer 7 inspection, the CSG supports up to 65,535 concurrent HTTP TCP connections.
  - Some HTTP Layer 7 methods and content types cause the CSG to invoke Layer 4 processing for the remainder of the TCP connection. For details, see the HTTP compliance exceptions in the "Layer 7 Inspection (accounting type=specific protocol)" section on page D-1.
- For the CSG/Hybrid (that is, the CSG running in hybrid mode, with CatOS on the Supervisor Engine and Cisco IOS on the Multilayer Switch Feature Card (MSFC)):
  - Runs with Cisco IOS Release 12.1(13)E or later and CatOS 7.6.1 or later.
  - Supports only the CSG Release 2.2(3)C2(1) command-line interface (CLI), not the CSG Release 3.1(1)C3(1) CLI.

  - Does not support the **hw-module module** *slot* **reset** command. To reset the CSG/Hybrid, enter the **set module power** [**up** | **down**] *slot* command at the CatOS console.

- When replacing an adjacent device but retaining the same IP address (such that there is a different MAC address but the same IP address as before), you must either enter the **clear module csm** *slot* **connections** command in privileged EXEC mode, or you must recycle the CSG.

Note The **clear module csm** *slot* **connections** command clears all connections for the specified CSM; you cannot use this command to clear selected connections.

- Services configured for **basis second connect** (Connection Duration Billing) are subject to the following restrictions:
  - Service verification is not supported for Connection Duration services.
  - Advice of Charge (AoC) is not supported for Connection Duration services.
  - If redirect is to be performed when the Connection Duration Service runs out of quota, the URL location to which the CSG redirects must map to a policy that does not have accounting configured. This is due to the fact that all IP sessions mapped to policies with accounting configured (postpaid or prepaid) are dropped when the Connection Duration service has no quota.
- For Service Duration Billing:
  - The CSG does not support dual quota (that is, the ability to deduct quota based on multiple criteria at the same time for the same flow).
  - Content idle is not included for non-TCP connections. Therefore, the idle timeout for non-TCP content definitions is restricted to be less than the service idle timeout of any service that includes the non-TCP content definition, and that is configured for **basis second**.
  - The CSG does not allow you to specify weights for Service Duration Billing.
- If the CSG does not have an ARP entry in its ARP cache for the MAC address of a device or firewall, it drops packets received from that device or firewall.
- If you define content with a network mask of 255.255.255.255 or /32 (that is, all subnets), then, a virtual server is created and the CSG's MAC address is entered as the host's address in the CSG's ARP cache. Because of this, you cannot have hosts directly connected to the CSG, coupled with content with a network mask of 255.255.255.255 or /32 that matches those hosts.
- The CSG does not decrement the time to live (TTL) of an IP packet.