# Usage-Based Billing (SAMIS)

**First Published:** April 10, 2015

This document describes the Usage-based Billing feature for the Cisco Cable Modem Termination System (CMTS) routers, which provides subscriber account and billing information in the Subscriber Account Management Interface Specification (SAMIS) format. The SAMIS format is specified by the Data-over-Cable Service Interface Specifications (DOCSIS) Operations Support System Interface (OSSI) specification.

### Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://tools.cisco.com/ITDIT/CFN/. An account on http://www.cisco.com/ is not required.

### Contents



# Hardware Compatibility Matrix for Cisco cBR Series Routers

**Note** The hardware components introduced in a given Cisco IOS-XE Release are supported in all subsequent releases unless otherwise specified.

*Table 1: Hardware Compatibility Matrix for the Cisco cBR Series Routers*

| Cisco CMTS Platform | Processor Engine | Interface Cards |
|---|---|---|
| Cisco cBR-8 Converged Broadband Router | **Cisco IOS-XE Release 3.15.0S and Later Releases**<br>Cisco cBR-8 Supervisor :<br>• PID—CBR-CCAP-SUP-160G<br>• PID—CBR-CCAP-SUP-60G[1]<br>• PID—CBR-SUP-8X10G-PIC | **Cisco IOS-XE Release 3.15.0S and Later Releases**<br>Cisco cBR-8 CCAP Line Cards:<br>• PID—CBR-LC-8D30-16U30<br>• PID—CBR-LC-8D31-16U30<br>• PID—CBR-RF-PIC<br>• PID—CBR-RF-PROT-PIC<br>Cisco cBR-8 Downstream PHY Modules:<br>• PID—CBR-D30-DS-MOD<br>• PID—CBR-D31-DS-MOD<br>Cisco cBR-8 Upstream PHY Modules:<br>• PID—CBR-D30-US-MOD |

[1] Effective with Cisco IOS-XE Release 3.17.0S, CBR-CCAP-SUP-60G supports 8 cable line cards. The total traffic rate is limited to 60Gbps, the total number of downstream service flow is limited to 72268, and downstream unicast low-latency flow does not count against the limits.

# Prerequisites for Usage-Based Billing (SAMIS)

The Usage-based Billing feature has the following prerequisites:

- Cable modems must be compliant with DOCSIS 1.0 or DOCSIS 2.0, OSSI version 3.0 and DOCSIS 3.0.
- Cable modems that are being monitored should use a DOCSIS configuration file that defines upstream and downstream primary service flows using Service Class Naming (SCN [TLV 24/25, subTLV 4]). If dynamically-created service flows are to be monitored, they should also be created with SCN names.
- When the feature is operating in File mode, an external billing server must log into the Cisco CMTS to copy the billing records to the external server, using either Secure Copy (SCP) or Trivial File Transfer Protocol (TFTP). The Cisco CMTS cannot operate as a FTP or secure FTP (SFTP) server.
- When the feature is operating in Streaming mode in non-secure mode, an external billing server must be configured to receive the billing records at a configurable TCP port.
- When the feature is operating in Streaming mode in secure mode, the following are required:
  - The external billing server must be configured to receive the billing records at a configurable TCP port using a secure socket layer (SSL) connection.

**Tip** Several third-party solutions for SSL support on the billing application server are available http://www.openssl.org/index.html.

- • A Certificate Authority (CA) must be configured and available to provide the required digital certificates to the billing application and Cisco CMTS router. The CA can be a public CA, such as Verisign, or a server on your private management network that is running software such as the Cisco Provisioning Center (CPC).

- To use the **full-records** keyword, the Cisco CMTS router must be running the Cisco IOS-XE releases.
- To use the **flow-aggregate** keyword for ipdr/ipdr-d3 the Cisco CMTS router must be running the Cisco IOS-XE releases.
- To automatically generate sflogs for a recovered line card, after a line card switch-over or a line card process restart, activate background synchronization (bgsync) using the **cable bgsync active** command.

When **flow-aggregate** is enabled, the service flows are combined into one record per cable modem:

- • ServiceClassName element always returns a null value in IPDR records, even when service flows on the cable modem have a valid service class name.
  - ServiceIdentifier element always returns a zero value.

# Restrictions for Usage-based Billing

The Usage-based Billing feature has the following restrictions and limitations:

- SNMP commands can be used to display or modify the Usage-based Billing configuration, and SNMP traps can be used to notify the billing application system when a billing record is available. However, SNMP commands cannot be used to retrieve billing records.
- Enabling IPDR mode through SNMP is not supported.

During a line card switchover, the items in the line card side are lost. Similarly, during a PRE switchover, those items in the RP side of the sflog file are lost.

Items for failed service-flows are lost in the following scenarios:

- If you reload a line card without a redundancy protection mechanisms such as line card high availability (LCHA) or line card process restart (LCPR)
- If some service-flows are not successfully recovered

Moreover, for service-flows that are already destroyed, the sflogs that are not written to the hard disk and that are in the failed line card's memory, will be lost.

If the user uses the SAMIS file destination, a PRE switchover also reinitializes that output file

- Billing records do not include information about multicast service flows and traffic counters.
- The packet counters displayed by CLI commands are reset to zero whenever the Cisco CMTS router is rebooted. The packet counters displayed by SNMP commands are not retained across router reloads, and

SNMP MIB counters cannot be preserved during reloads. These counters are 64-bit values and could roll over to zero during periods of heavy usage.

- When configuring cable metering in the usage-based billing File Mode, the source-interface cannot be specified immediately after using the cable metering filesystem command. Once the cable metering filesystem command is used, the cable metering file will write to the bootflash. Until this operation is complete, no cable metering configuration will be allowed. After the file write operation is complete, the source-interface command (cable metering source-interface) can then be configured; and the metering file in the bootflash would need to be removed so that billing packets have the source-interface's IP address.

**Note** This cable metering restriction will not be a problem during reload.

- When configuring cable metering in the usage-based billing Streaming Mode, make sure that the loopback interface is accessible from the collector server. Telnetting to the IP address of the loopback interface from the collector server is a good method of testing whether the loopback interface is accessible from the collector server or not.

# Information About Usage-based Billing

## Feature Overview

The Usage-based Billing feature provides a standards-based, open application approach to recording and retrieving traffic billing information for DOCSIS networks. When enabled, this feature provides the following billing information about the cable modems and customer premises equipment (CPE) devices that are using the cable network:

- IP and MAC addresses of the cable modem.
- Service flows being used (both upstream and downstream service flows are tracked).
- IP addresses for the CPE devices that are using the cable modem.
- Total number of octets and packets received by the cable modem (downstream) or transmitted by the cable modem (upstream) during the collection period.
- Total number of downstream packets for the cable modem that the CMTS dropped or delayed because they would have exceeded the bandwidth levels allowed by the subscriber’s service level agreement (SLA).

Billing records are maintained in a standardized text format that the service provider can easily integrate into their existing billing applications. Service providers can use this information to determine which users might be potential customers for service upgrades, as well as those customers that might be trying to exceed their SLA limits on a regular basis.

## Usage-Based Billing and DOCSIS Support on the Cisco CMTS Routers

The usage-based billing feature supports these DOCSIS features on the Cisco CMTS routers:

- DOCSIS 1.0, DOCSIS 2.0, and DOCSIS 3.0 compliant cable modems are supported.
- Best Effort service flows are supported for DOCSIS-compliant cable modems.

- Secondary service flows are supported for DOCSIS-compliant cable modems.
- Dynamic service flows are supported for DOCSIS-compliant cable modems.
- Information about deleted service flows is available only for DOCSIS 1.1 service flows but not for DOCSIS 1.0 service flows.
- Support for terminated service flows must be enabled using the **cable sflog** command in global mode.

# Standards

The Usage-based Billing feature is based on several open standards, allowing it to be supported by a wide range of commercial and custom-written billing applications. The following standards provide the major guidelines for writing and using the billing records that the CMTS produces:

- Extensible Markup Language (XML)—A metalanguage that in turn can easily define other markup languages to contain any kind of structured information, such as billing records. An XML-based approach allows the collected billing information to be used by and distributed among many different billing applications from different vendors. It also allows the format to be easily updated and customized to meet the needs of different providers.
- IP Detail Record (IPDR)—An open, vendor-independent standard, defined in the *Network Data Management—Usage (NDM-U) For IP-Based Services* specification, to simplify billing and usage record-keeping for any type of services that can be delivered over an IP-based network. Service providers can use IPDR to create unified billing applications for all of their services, such as DOCSIS or Voice-over-IP, even though those services use different protocols and application servers.
- DOCSIS Operations Support System Interface (OSSI) specification—A DOCSIS specification that defines the requirements for the network management of a DOCSIS network, including a Subscriber Account Management Interface Specification (SAMIS) for a billing record interface. The DOCSIS 2.0 version of this specification states that a CMTS is not required to provide a billing interface, but if the CMTS does provide a billing interface, it must be based on the IPDR/XML standards.

**Tip** For further information about these standards, see the documents listed in the “Standards” section on page 38.

# IPDR Service Definition Schemas

To standardize the management of objects, service definition schemas are associated with IPDR just as MIBs are associated to SNMP.

For more information, see the OSSI specification document at http://www.cablelabs.com/wp-content/uploads/specdocs/CM-SP-OSSIv3.0-I02-070223.pdf.

The schemas are supported on Cisco IOS-XE releases.

*Table 2: IPDR Schema List for DOCSIS 3.0*

| Category | Service Definition | Schema Definition | Collection Method |
|---|---|---|---|
| SAMIS | SAMIS-TYPE-1 | DOCSIS-SAMIS-TYPE-1_3.5.1-A.1.xsd | time interval, ad-hoc |
| | SAMIS-TYPE-2 | DOCSIS-SAMIS-TYPE-2_3.5.1-A.1.xsd | time interval, ad-hoc |
| Diagnostic Log Service Definition Schemas | DIAG-LOG-TYPE | DOCSIS-DIAG-LOG-TYPE_3.5.1-A.1.xsd | ad-hoc |
| | DIAG-LOG-EVENT-TYPE | DOCSIS-DIAG-LOG-EVENT-TYPE_3.5.1-A.1.xsd | event |
| | DIAG-LOG-DETAIL-TYPE | DOCSIS-DIAG-LOG-DETAIL-TYPE_3.5.1-A.1.xsd | time interval, ad-hoc, event |
| Spectrum Management | SPECTRUM-MEASUREMENT-TYPE | DOCSIS-SPECTRUM-MEASUREMENT-TYPE_3.5.1-A.1.xsd | time interval, ad-hoc |
| CMTS CM Registration Status Information | CMTS-CM-REG-STATUS-TYPE | DOCSIS-CMTS-CM-REG-STATUS-TYPE_3.5.1-A.1.xsd | time interval, ad-hoc, event |
| CMTS CM Upstream Status Information | CMTS-CM-US-STATS-TYPE | DOCSIS-CMTS-CM-US-STATS-TYPE_3.5.1-A.1.xsd | time interval, ad-hoc |
| CMTS Topology | CMTS-TOPOLOGY-TYPE | DOCSIS-CMTS-TOPOLOGY-TYPE_3.5.1-A.1.xsd | ad-hoc, event |
| CPE Information | CPE-TYPE | DOCSIS-CPE-TYPE_3.5.1-A.1.xsd | ad-hoc, event |
| CMTS Utilization Statistics | CMTS-US-UTIL-STATS-TYPE | DOCSIS-CMTS-US-UTIL-STATS-TYPE_3.5.1-A.1.xsd | event |
| | CMTS-DS-UTIL-STATS-TYPE | DOCSIS-CMTS-DS-UTIL-STATS-TYPE_3.5.1-A.1.xsd | event |

The schemas listed in the table are supported by implementing the respective Collectors, which work as SNMP agents to generate these IPDR records according to management information of the system.

## IPDR CM-STATUS-2008

Effective from Cisco IOS-XE Release 3.17.0S, the IPDR CM-STATUS 2008 version is introduced for forward compatibility to support old IPDR collectors. In the IPDR CM-STATUS 2008 version, the CmtsRcsId and

CmtsTcsId objects are 16 bits in length whereas in the CM-STATUS version both these objects are 32 bits in length.

The CmtsRcsId object in the CM-STATUS-2008 version returns the lower 16 bits of value from the CM-STATUS version. But, the CmtsTcsId object returns the same value for both the CM-STATUS-2008 and CM-STATUS version since the value does not exceed 16 bits in both the schemas.

## DOCSIS SAMIS Service Definitions

SAMIS for DOCSIS 3.0 service definitions are well structured and has two versions—SAMIS-TYPE-1 and SAMIS-TYPE-2 and provide a different level of information details than SAMIS.

DOCSIS 2.0 SAMIS supports only event session (default type) and DOCSIS 3.0 SAMIS TYPE 1 and DOCSIS 3.0 SAMIS TYPE 2 support only interval and ad-hoc sessions.

SAMIS is collected based on configurable time intervals. Each interval is a different document and the Exporter stops and starts a new session for a new interval. The interval starts from the last metering that has either succeeded or failed, unlike the time-interval session that has a fixed starting point and an interval.

**Note** The SAMIS schema can be configured with the **cable metering ipdr session** command SAMIS-TYPE-1 and SAMIS-TYPE-2 schemas can be configured through the **cable metering ipdr-d3** command. These schemas are mutually exclusive of each other.

### Limitation To DOCSIS SAMIS

- Only a schema that is consistent with the **cable metering ipdr| ipdr-d3** command will work. If none of the schemas are consistent, none of them will work.
- Changing the SAMIS IPDR type will abort exporting IPDR data.

## DOCSIS Diagnostic Log Service Definitions

This service definition defines the IPDR Streaming using a two-step process:

- SNMP or other configuration management interface, such as the CLI is used to configure the Diagnostic Log.
- IPDR/SP is used to stream the Diagnostic Log instances.

These Diagnostic Log service definition schemas support the following collection methods:

- The Cisco CMTS supports streaming of the DIAG-LOG-TYPE record collections as an ad-hoc session.
- The Cisco CMTS supports streaming of DIAG-LOG-EVENT-TYPE record collections as an event session. For event-based Diagnostic Log records, the Cisco CMTS streams the record when the event is logged in the Diagnostic Log and an IPDR message is transmitted to the Collector.
- The DOCSIS-DIAG-LOG-DETAIL-TYPE supports the following collection methods:
  - Time interval—Follows a schedule based on session configuration to export data on a periodic time interval. When a given time interval end is reached, the Exporter collects the diagnostic log, then streams the record to the Collector associated with this session. For time interval based Diagnostic Log records, the Cisco CMTS streams a snapshot of the Diagnostic Log at the scheduled collection time.
  - Ad-hoc—When the Exporter receives a "FlowStart" message, it triggers the application to collect the diagnostic record and send the data to the Collector.

- Event—When a diagnostic log record is created, an ipdr message is transmitted to the Collector. For more information, see the Operations Support System Interface (OSSI) Specification.

## DOCSIS Spectrum Measurement Service Definition

This service definition schema defines the IPDR schema for the enhanced signal quality monitoring feature.

The DOCSIS-SPECTRUM-MEASUREMENT-TYPE schema supports the following collection methods:

- Time interval—Follows a schedule based on session configuration to export data on a periodic time interval. When a given time interval end is reached, the Exporter collects the spectrum information, then streams the records to the Collector.
- Ad-hoc—When the Exporter receives a "FlowStart" message, it triggers the application to collect the spectrum information and send the data to the Collector.

## DOCSIS CMTS CM Registration Status Service Definition

This service definition schema defines the IPDR service definition schema for the CMTS CM Registration Status information.

The DOCSIS-CMTS-CM-REG-STATUS-TYPE schema supports the following collection methods:

- Time interval—Follows a schedule based on session configuration to export data on a periodic time interval. When a given time interval end is reached, the Exporter collects the CM status information, then streams the records to the Collector.
- Ad-hoc—When the Exporter receives a "FlowStart" message, it triggers the application to collect all status information of the cable modems and send the data to the Collector.
- Event—When a cable modem goes from "offline" status to "online" or changes to "offline" from "online" (not including intermediate state changes), the Exporter invokes the application to collect the cable modem status information and sends the data to the Collector. For more information, see the Operations Support System Interface (OSSI) Specification.

## DOCSIS CMTS CM Upstream Status Service Definition

This service definition schema define the cable modem registration status objects and upstream status objects from the cable modem and the Cisco CMTS perspective. In the CmtsCmUsEqData IPDR schema field, configure the **cable upstream equalization-coefficient** command under the corresponding MAC domain to enable the feature to have data. For more information on this command, see the Cisco IOS CMTS Cable Command Reference Guide .

The DOCSIS-CMTS-CM-US-STATS-TYPE schema support the following collection methods:

- Time interval—Follows a schedule based on session configuration to export data on a periodic time interval. When a given time interval end is reached, the Exporter collects the cable modem upstream status information, then streams the records to the Collector.
- Ad-hoc—When the Exporter receives a "FlowStart" message, it triggers the application to collect all upstream status information of the cable modem and send the data to the Collector.

## DOCSIS CMTS Topology Service Definition

In the case of an event session, the event means a change of the topology.

This service definition schema defines the IPDR service definition schema for the CMTS Topology information.

The DOCSIS-CMTS-TOPOLOGY-TYPE schema supports the following collection methods:

- Ad-hoc—Sends the entire picture of all fiber-nodes.
- Event—Sends only the updated channels status of the fiber nodes.

## DOCSIS CPE Service Definition

The DOCSIS-CPE-TYPE schema supports the following collection methods:

- Ad-hoc—Follows a schedule based on session configuration to export data on a periodic time interval. When a given time interval end is reached, the Exporter collects the CPE status information, then transfers the records to the Collector.
- Event—When new CPE is added, the status of the CPE changes (including change in IP address), or a new CPE replaces an old one (in this case, two messages are displaced— removal of the old CPE and addition of the new CPE). For more information, see the Operations Support System Interface (OSSI) Specification.

## DOCSIS CMTS Utilization Statistics Service Definition

The CMTS Utilization Statistics mainly focuses on channel utilization. It covers CMTS MAC Domain, channel identifier, and the upstream or downstream utilization attributes and counters.

The DOCSIS-CMTS-US-UTIL-STATS-TYPE schemas defines upstream utilization statistics for a specified upstream logical channel interface for the specified Cisco CMTS. The interval can be configured through Channel Utilization Interval.

The DOCSIS-CMTS-DS-UTIL-STATS-TYPE schema defines downstream utilization statistics for a specified downstream interface for the specified Cisco CMTS. The interval can be configured through Channel Utilization Interval.

For more information, see the IPDR Streaming Protocol on the Cisco CMTS Routers guide at the following URL:

IPDR Streaming Protocol

These schemas support only interval-driven event session for the entire downstream and upstream. The interval is defined in the docsIfCmtsChannelUtilizationInterval MIB and it creates document for every exporting.

**Note** The UsUtilTotalCntnReqDataMslots, UsUtilUsedCntnReqDataMslots, and UsUtilCollCntnReqDataMslots MIBs are not supported on the Cisco CMTS implementation.

The DsUtilTotalBytes MIB for RF Gateway RF channels is the maximum counter of bytes this RF channel can pass during an interval.

# Modes of Operation

The Usage-based Billing feature can operate in three modes:

- File Mode—In file mode, the CMTS collects the billing record information and writes the billing records to a file on a local file system, using a file name that consists of the router's hostname followed by a timestamp of when the file was written. A remote application can then log into the CMTS and transfer the billing record file to an external server where the billing application can access it.

The remote application can use the Secure Copy Protocol (SCP) or the Trivial File Transfer Protocol (TFTP) to transfer the file. After a successful transfer, the remote application then deletes the billing record file, which

signals the CMTS that it can create a new file. The remote application can either periodically log into the CMTS to transfer the billing record file, or it can wait until the CMTS sends an SNMPv2 trap to notify the application that a billing record file is available.

- Streaming Mode—In streaming mode, the CMTS collects the billing record information and then regularly transmits the billing record file to an application on an external server, using either a non-secure TCP connection or a secure sockets layer (SSL) connection. The billing record data collected is streamed in real time; and if streaming is unsuccessful, then the SAMIS data is sent only at the next interval.

If the CMTS fails to establish a successful connection with the external server, it retries the connection between one to three times, depending on the configuration. If the CMTS continues to fail to connect with the external server, the Cisco CMTS sends an SNMPv2 trap to notify the SNMP manager that this failure occurred.

In streaming mode, you can configure the CMTS to transmit the billing record file at regular intervals. Typically, the interval chosen would depend on the number of cable modems and the size of the billing record files that the CMTS produces.

- IPDR Mode—In the IPDR mode, the IPDR export process communicates with IPDR Collectors. The architecture supports multiple Collectors distinguished by priority value for failover purposes. The smaller the number of Collectors, the higher is the priority value. Associating one session to two or more Collectors with the same priority value is regarded as random priority. At any given time, data is sent to only the available highest priority Collector. If the highest priority Collector connection fails due to any reason, the data is sent to the next available highest priority Collector. After a higher priority Collector comes back online, it will fail over again. Depending on the network configuration, you can have different primary Collectors for different IPDR sessions. For example, there may be a billing Collector or a diagnostic Collector.

# Billing Record Format

Each billing record is an ASCII text file using XML formatting to encode the billing record objects that are required by the DOCSIS specifications. This file can be read by any billing application that can be configured to parse XML data files.

The table lists the objects that are contained in each billing record that the CMTS generates. This table shows the object's name, as it appears in the billing record, and a description of that object.

***Table 3: Billing Record Objects***

| Object Name | Description |
|---|---|
| IPDRcreationTime | (Appears in header of billing record) Date and time that the CMTS created the billing created. |
| serviceClassName | Service Class Name (SCN) identifying the service flow (for example, BronzeDS). |
| CMmacAddress | MAC Address of the cable modem, expressed as six hexadecimal bytes separated by dashes (for example, 00-00-0C-01-02-03). |
| CMipAddress | IP address for the cable modem, expressed in dotted decimal notation (for example, 192.168.100.101). |
| CMdocsisMode | Version of DOCSIS QoS provision that the cable modem is currently using (DOCSIS 1.0 or 1.1). |

| Object Name | Description |
|---|---|
| CPEipAddress | IP address for each CPE device that is using this cable modem, expressed in dotted decimal notation. This object is optional and can be suppressed to improve performance by reducing the size of the billing record files. |
| CMTSipAddress | IP address for the CMTS, expressed in dotted decimal notation. |
| CMTShostName | Fully qualified hostname for the CMTS (for example, cmts01.cisco.com). |
| CMTSsysUpTime | Amount of time, in hundredths of a second, since the last initialization of the CMTS management interface, expressed as a 32-bit decimal number (0 to 4,294,967,296). |
| RecType (SFType renamed to RecType in Cisco IOS Release 12.3(17a)BC) | Type of service flow being described:<br>• **Interim**—the service flow was active throughout the collection period and should be reported as 1.<br>• **Stop**—the service flow was deleted at some point during the collection period and should be reported as 2. |
| serviceIdentifier | Service flow ID assigned to this service flow by the CMTS, expressed as a decimal number.<br>**Note**<br>For DOCSIS 1.0 cable modems, the SFID field always shows the primary service flow for the upstream or downstream. |
| serviceDirection | Direction for the service flow (**Downstream** or **Upstream**). |
| serviceOctetsPassed | Total number of octets received by the cable modem (downstream service flows) or transmitted by the cable modem (upstream service flows) during the collection period, expressed as a 64-bit decimal number. |
| servicePktsPassed | Total number of packets received by the cable modem (downstream service flows) or transmitted by the cable modem (upstream service flows) during the collection period, expressed as a 64-bit decimal number. |
| SLAdropPkts | (Downstream service flows only) Total number of downstream packets for the cable modem that the CMTS dropped because otherwise they would have exceeded the bandwidth levels allowed by the subscriber's service level agreement (SLA), expressed as a 64-bit decimal number. |
| SLAdelayPkts | (Downstream service flows only) Total number of packets that the CMTS delayed transmitting on the downstream to the cable modem because otherwise they would have exceeded bandwidth levels allowed by the subscriber's service level agreement (SLA), expressed as a 64-bit decimal number. |
| CMTScatvIfIndex | The ifIndex of the MAC interface. |
| CMTScatvIfName | The ifName of the CMTS CATV (MAC) interface associated with this cable modem. |

| Object Name | Description |
|---|---|
| CMTSupIfName | The ifName of the CMTS Upstream interface associated with this cable modem. |
| CMTSdownIfName | The ifName of the CMTS Downstream interface associated with this cable modem. |
| CMcpeFqdn | FQDNs for cable modem associated CPEs. |
| serviceTimeCreated | Timestamp for SF creation (consistent with QoS MIB model). |
| serviceTimeActive | The active time of the SF in seconds. |

**Note** Because the byte and packet counters are 64-bit values, it is possible for them to wrap around to zero during a billing period. The billing application should use the sysUpTime value along with the counters to determine whether the counters have wrapped since the last billing period. If a counter appears to regress, and if the current sysUpTime indicates this billing cycle is the next scheduled cycle for this particular cable modem, you can assume that the counter has wrapped during the billing cycle.

**Note** These billing record objects are defined in Appendix B, *IPDR Standards Submission for Cable Data Systems Subscriber Usage Billing Records* , in the *DOCSIS 2.0 OSSI Specification* (SP-OSSIv2.0-IO3-021218).

The following example shows a sample IPDR billing record for a downstream service flow:

```
<?xml version="1.0" encoding="UTF-8"?>
<IPDRDoc xmlns="http://www.ipdr.org/namespaces/ipdr"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="DOCSIS-3.1-B.0.xsd"
docId="C341A679-0000-0000-0000-000BBF54D000"
creationTime="2002-05-25T14:41:29Z"
IPDRRecorderInfo="CMTS01"
version="3.1">
</IPDR>
<IPDR xsi:type="DOCSIS-Type">
<IPDRcreationTime>2003-09-18T16:52:34Z</IPDRcreationTime>
<CMTShostname>R7519-UBR7246.cisco.com</CMTShostname>
<CMTSipAddress>1.8.8.21</CMTSipAddress>
<CMTSsysUpTime>287315    </CMTSsysUpTime>
<CMTScatvIfName>Cable8/0/0</CMTScatvIfName>
<CMTScatvIfIndex>13</CMTScatvIfIndex>
<CMTSupIfName>Ca8/0/0-upstream0</CMTSupIfName>
<CMTSupIfType>129</CMTSupIfType>
<CMTSdownIfName>Ca8/0/0-downstream</CMTSdownIfName>
<CMmacAddress>00-00-39-AB-D4-53</CMmacAddress>
<CMdocsisMode>1.0</CMdocsisMode>
<CMipAddress>3.8.21.3</CMipAddress>
<CPEipAddress></CPEipAddress>
<RecType>1</SFtype>
<serviceIdentifier>3</serviceIdentifier>
<serviceClassName></serviceClassName>
<serviceDirection>2</serviceDirection>
<serviceOctetsPassed>23457</ServiceOctetsPassed>
```

```
<servicePktsPassed>223</ServicePktsPassed>
<serviceSlaDropPkts>2</serviceSlaDropPkts>
<serviceSlaDelayPkts>0</serviceSlaDelayPkts>
<serviceTimeCreated>11000</serviceTimeCreated>
<serviceTimeActive>15890</serviceTimeActive>
</IPDR>
</IPDRDoc>
```

The following example shows a sample IPDR billing record for an upstream service flow:

<?xml version="1.0" encoding="UTF-8"?>

```
<IPDRDoc xmlns="http://www.ipdr.org/namespaces/ipdr"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="DOCSIS-3.1-B.0.xsd"
docId="docId="C3146152-0000-0000-0000-000BBF7D5800"
creationTime="2003-09-18T16:52:34Z"
IPDRRecorderInfo="CMTS01-UBR7246.cisco.com"
version="3.1">
<IPDR xsi:type=" DOCSIS-Type">
<IPDRcreationTime>2003-09-18T16:52:34Z</IPDRcreationTime>
<CMTShostname>R7519-UBR7246.cisco.com</CMTShostname>
<CMTSipAddress>1.8.8.21</CMTSipAddress>
<CMTSsysUpTime>287315    </CMTSsysUpTime>
<CMTScatvIfName>Cable8/0/0</CMTScatvIfName>
<CMTScatvIfIndex>13</CMTScatvIfIndex>
<CMTSupIfName>Ca8/0/0-upstream0</CMTSupIfName>
<CMTSupIfType>129</CMTSupIfType>
<CMTSdownIfName>Ca8/0/0-downstream</CMTSdownIfName>
<CMmacAddress>00-00-39-18-8A-4D</CMmacAddress>
<CMdocsisMode>1.0</CMdocsisMode>
<CMipAddress>3.8.21.14</CMipAddress>
<CPEipAddress></CPEipAddress>
<RecType>1</SFtype>
<serviceIdentifier>3</serviceIdentifier>
<serviceClassName></serviceClassName>
<serviceDirection>1</serviceDirection>
<serviceOctetsPassed>1404</ServiceOctetsPassed>
<servicePktsPassed>6</ServicePktsPassed>
<serviceSlaDropPkts>0</serviceSlaDropPkts>
<serviceSlaDelayPkts>0</serviceSlaDelayPkts>
<serviceTimeCreated>11000</serviceTimeCreated>
<serviceTimeActive>15890</serviceTimeActive>
</IPDR>
</IPDRDoc>
```

# SNMP Support

Cisco cBR Series Converged Broadband Router s support the following MIBs that provide SNMPv2 support for the Usage-based Billing feature:

CISCO-CABLE-METERING-MIB

- Supports configuration of the usage-based billing feature using SNMPv2 commands.
- Displays the current usage-based billing configuration using SNMPv2 commands.
- Sends SNMPv2 traps based on the following usage-based billing events:
  - The Cisco CMTS reports that a new billing record is available.

- The Cisco CMTS reports that a failure occurred in writing the most recent billing record (for example, the disk is full).
- The Cisco CMTS reports that it could not successfully open a secure SSL connection to stream a billing record to the billing server.

CISCO-CABLE-WIDEBAND-MIB

Sets the polling interval for calculating the utilization of an RF channel by using the **ccwbRFChanUtilInterval** object.

DOCS-QOS-MIB

- Sets the load and utilization of both upstream and downstream physical channels through the **docsIfCmtsChannelUtilizationInterval** object. This information may be used for capacity planning and incident analysis, and may be particularly helpful in provisioning high value QoS.
- Displays information about all service flows (DOCSIS 1.1 service flows only) including multicast service flow is maintained in the **docsQosServiceFlowLogTable** in DOCS-QOS-MIB, **docsIetfQosServiceFlowLogTable** in DOCS-IETF-QOS-MIB, and **docsQos3ServiceFlowLogTable** in DOCS-QOS3-MIB.

To view information about deleted service flows, enable logging of deleted service flows using the **cable sflog** global configuration command.

## Benefits

The usage-based billing feature provides the following benefits to cable service providers and their partners and customers:

- Allows service providers to integrate their billing applications for DOCSIS services with their other XML-capable billing applications.
- Standards-based approach that supports existing networks and services, such as DOCSIS and PacketCable, and is easily extensible to support future services as they are supported on the Cisco CMTS.

# How to Configure the Usage-based Billing Feature

This section describes the following tasks that are required to implement the Usage-based Billing feature:

## Enabling Usage-based Billing Feature File Mode Using CLI Commands

This section describes how to enable and configure the Usage-based Billing feature so that it operates in file mode, where it writes the billing record files to a local file system. The billing application must then log into the Cisco CMTS and retrieve the billing record files on a regular basis.

## Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable`<br>**Example:**<br>`Router#` | Enables privileged EXEC mode. Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal`<br>**Example:**<br>`Router(config)#` | Enters global configuration mode. |
| **Step 3** | **cable metering filesystem** *filesys* [**flow-aggregate**] [**cpe-list-suppress**] [**full-records**]<br>**Example:**<br>`Router(config)# cable metering filesystem harddisk:`<br>**Example:**<br>`Router(config)#` | Enables the Usage-based Billing feature for file mode and configures it.<br>The system will write the billing records on this file system using a file name that contains the hostname of the router followed by a timestamp when the record was written. |
| **Step 4** | **snmp-server enable traps cable metering**<br>**Example:**<br>`Router(config)# snmp-server enable traps cable metering`<br>**Example:**<br>`Router(config)#` | (Optional) Enables SNMP traps for usage-based billing events. Traps are sent when a new billing record is available, or when the system encountered a failure (such as insufficient disk space) in writing the new billing record. |
| **Step 5** | **cable sflog max-entry** *number* **entry-duration** *time*<br>**Example:**<br>`Router(config)# cable sflog max-entry 2000 entry-duration 7200`<br>**Example:** | (Optional) Enables logging for deleted SNMP service flows, which allows the billing feature to include information about deleted service flows. |

| | Command or Action | Purpose |
|---|---|---|
| | `Router(config)#` | |
| **Step 6** | **cable metering source-interface** *interface*<br>**Example:**<br>`Router(config)# cable metering source-interface loopback100`<br>**Example:**<br>`Router(config)#` | (Optional) Enables specification of the source-interface for the billing packets, usually a loopback interface. |
| **Step 7** | **end**<br>**Example:**<br>`Router(config)# end`<br>**Example:**<br>`Router#` | Exits global configuration mode and returns to privileged EXEC mode. |

# Enabling Usage-based Billing Feature File Mode Using SNMP Commands

This section describes how to enable and configure the Usage-based Billing feature so that it operates in file mode and writes the billing record files to a local file system. The billing application must then log into the Cisco CMTS and retrieve the billing record files on a regular basis.

To configure the Cisco CMTS for Usage-based Billing feature in file mode, you must set a number of objects in the CISCO-CABLE-METERING-MIB.

In addition, to include information about deleted service flows in the billing records (supported for DOCSIS 1.1 service flows), you must enable the logging of deleted service flows, using the **cable sflog** global configuration command.

***Table 4: SNMP Objects to be Configured for File Mode***

| Object | Type | Description |
|---|---|---|
| ccmtrCollectionType | Integer | Enables or disables the Usage-based Billing feature. The valid values are:<br>• 1—none. The Usage-based Billing feature is disabled (default).<br>• 2—local. The Usage-based Billing feature is enabled and configured for file mode.<br>• 3—stream. The Usage-based Billing feature is enabled and configured for streaming mode.<br>Set ccmtrCollectionType to 2 (local) to enable the feature for file mode. |

| Object | Type | Description |
|---|---|---|
| ccmtrCollectionFilesystem | DisplayString | Specifies the file system where the billing record file should be written. This object has a maximum length of 25 characters and must specify a valid file system on the router (such as slot0, disk1, or flash).<br>**Note**<br>The Cisco CMTS writes the billing records to this file system using a file name that consists of the router's hostname followed by a timestamp when the record was written. |
| ccmtrCollectionCpeList | TruthValue | (Optional) Indicates whether IP addresses for customer premises equipment (CPE) devices are omitted from the billing records, so as to reduce the size of the billing records and to improve performance. The valid values are the following:<br>• true—CPE information is present (default).<br>• false—CPE information is omitted.<br>**Note**<br>When set to true, a maximum of 5 CPE IP addresses for each cable modem. |
| ccmtrCollectionAggregate | TruthValue | (Optional) Indicates whether all information for an individual cable modem is combined into one record. Separate counters are maintained for upstream and downstream traffic, but those counters include all service flows in that direction. The valid values are as follows:<br>• true—All service flow information for each cable modem is aggregated into a single billing record. In this configuration, the service flow ID (SFID) for the billing record is set to 0 and the service class name (SCN) is blank.<br>• false—Information for each cable modem is not aggregated into a single billing record, but instead each service flow is recorded into its own record (default). |
| ccmtrCollectionSrcIfIndex | TruthValue | (Optional) Specifies the source-interface for the billing packets. |

**Note** The following steps use the standard SNMP commands that are available on many Unix and Linux systems. For each step, replace *ip-address* with the IP address of the Cisco CMTS, and replace *rw-community-string* with an SNMP community string that provides read-write access to the router.

## Procedure

**Step 1** Set the ccmtrCollectionType object to 2, to enable the Usage-based Billing feature and to configure it for file mode:

**Example:**

```
workstation# setany -v2c

ip-address rw-community-string

ccmtrCollectionType.0 -i 2
```

```
workstation#
```

**Step 2** Set the ccmtrCollectionFilesystem object to the local file system where the Cisco CMTS should write the billing records:

**Example:**

```
workstation# setany -v2c

ip-address rw-community-string
 ccmtrCollectionFilesystem.0 -D disk0:

workstation#
```

**Step 3** (Optional) To omit the IP addresses of CPE devices from the billing records, set the ccmtrCollectionCpeList object to 2 (false). The default is to include the CPE information.

**Example:**

```
workstation# setany -v2c

ip-address rw-community-string
 ccmtrCollectionCpeList.0 -i 2
workstation#
```

**Step 4** (Optional) To aggregate all service flow information for each cable modem in a single record, set the ccmtrCollectionAggregate object to 1 (true). The default is for each service flow to be written in a separate record:

**Example:**

```
workstation# setany -v2c

ip-address rw-community-string
 ccmtrCollectionAggregate.0 -i 1
workstation#
```

**Step 5** (Optional) To specify the source-interface for the billing packets, set the ccmtrtrCollectionSrcIfIndex object to 1 (true). The default is for the billing packets to automatically select a source-interface.

**Example:**

```
workstation# setany -v2c

ip-address rw-community-string
 ccmtrtrCollectionSrcIfIndex.0 -i 1
workstation#
```

## Examples for Enabling Usage Billing using SNMP Mode

The following example shows the Usage-based Billing feature being configured using SNMP commands. The following display shows that a Cisco CMTS router at IP address 10.8.8.21 is configured with the default configuration (the Usage-based Billing feature is disabled):

```
workstation# getmany -v2c 10.8.8.21 rw-string ciscoCableMeteringMIB
```

```
ccmtrCollectionType.0 = none(1)
ccmtrCollectionFilesystem.0 =
ccmtrCollectionCpeList.0 = true(1)
ccmtrCollectionAggregate.0 = false(2)
ccmtrCollectionStatus.0 = 0
ccmtrCollectionDestination.0 =
ccmtrCollectionTimestamp.0 = 00 00  00 00   00 00  00 00
ccmtrCollectionNotifEnable.0 = true(1)
workstation#
```

The following SNMP commands are then given to enable the Usage-based Billing feature and to configure it for file mode:

```
workstation# setany -v2c 10.8.8.21 rw-string ccmtrCollectionType.0 -i 2
workstation# setany -v2c 10.8.8.21 rw-string
ccmtrCollectionFilesystem
.0 -D disk1:
workstation#
```

These commands add the following line to the router's running configuration file:

```
Router# show running-config | include metering

cable metering filesystem disk1:
Router#
```

The following SNMP display shows the new configuration, after the Cisco CMTS has successfully written a billing record:

```
workstation# getmany -v2c 10.8.8.21 rw-string ciscoCableMeteringMIB

ccmtrCollectionType.0 = local(2)
ccmtrCollectionFilesystem.0 = disk1:
ccmtrCollectionCpeList.0 = true(1)
ccmtrCollectionAggregate.0 = false(2)
ccmtrCollectionStatus.0 = success(1)
ccmtrCollectionDestination.0 = disk1:UBR7246.cisco.com-20030925-185827
ccmtrCollectionTimestamp.0 = 07 d3  09 19   12 3a  1c 00
ccmtrCollectionNotifEnable.0 = true(1)
workstation#
```

# Enabling Usage-based Billing Feature Streaming Mode Using CLI Commands

This section describes how to enable and configure the Usage-based Billing feature so that it operates in streaming mode, where it regularly transmits the billing records to an external server for use by the billing application.

## Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable`<br>**Example:**<br>`Router#` | Enables privileged EXEC mode. Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal`<br>**Example:**<br>`Router(config)#` | Enters global configuration mode. |
| **Step 3** | **cable metering destination** *ip-address port* [*ip-address2 port2* ] *retries minutes* {**non-secure** \| **secure**} [**flow-aggregate**] [**cpe-list-suppress**] [**full-records**]<br>**Example:**<br>`Router(config)# cable metering destination 10.10.21.3 5300 10.10.21.4 5300 2 30 secure`<br>**Example:**<br>`Router(config)#` | Enables the Usage-based Billing feature for streaming mode and configures it with the following parameters: |
| **Step 4** | **snmp-server enable traps cable metering**<br>**Example:**<br>`Router(config)# snmp-server enable traps cable metering`<br>**Example:**<br>`Router(config)#` | (Optional) Enables SNMP traps for usage-based billing events. Traps are sent when a new billing record is available, or when the system encountered a failure (such as insufficient disk space) in writing the new billing record. |
| **Step 5** | **cable sflog max-entry** *number* **entry-duration** *time*<br>**Example:**<br>`Router(config)# cable sflog max-entry 2000 entry-duration 7200` | (Optional) Enables logging for deleted SNMP service flows, which allows the billing feature to include information about deleted service flows. |

| | Command or Action | Purpose |
|---|---|---|
| | **Example:**<br>`Router(config)#` | |
| **Step 6** | **cable metering source-interface** *interface*<br>**Example:**<br>`Router(config)# cable metering source-interface loopback100`<br>**Example:**<br>`Router(config)#` | (Optional) Enables specification of the source-interface for the billing packets, usually a loopback interface. |
| **Step 7** | **end**<br>**Example:**<br>`Router(config)# end`<br>**Example:**<br>`Router#` | Exits global configuration mode and returns to privileged EXEC mode. |

# Enabling Usage-based Billing Feature Streaming Mode Using SNMP Commands

This section describes how to use SNMP commands to enable and configure the Usage-based Billing feature so that it operates in streaming mode, where it regularly transmits the billing records to an external server for use by the billing application.

To configure the Cisco CMTS for Usage-based Billing feature in streaming mode, you must set a number of objects in the CISCO-CABLE-METERING-MIB.

**Note** In addition, to include information about deleted service flows (DOCSIS 1.1 service flows only) in the billing records, you must enable the logging of deleted service flows, using the **cable sflog** global configuration command. See the *Cisco IOS CMTS Cable Command Reference Guide* on Cisco.com:

Cisco CMTS Cable Command Reference

*Table 5: SNMP Objects to be Configured for Streaming Mode*

| Object | Type | Description |
|---|---|---|
| ccmCollectionType | Integer | Enables or disables the Usage-based Billing feature. The valid values are:<br>• 1—none. The Usage-based Billing feature is disabled (default).<br>• 2—local. The Usage-based Billing feature is enabled and configured for file mode.<br>• 3—stream. The Usage-based Billing feature is enabled and configured for streaming mode.<br>Set ccmCollectionType to 3 (stream) to enable the feature for streaming mode. |
| ccmCollectionIpAddress | InetAddress | IP address for the external collection server. This value must be specified. |
| ccmCollectionPort | Unsigned32 | TCP port number at the external collection server to which the billing records should be sent. The valid range is 0 to 65535, but you should not specify a port in the well-known range of 0 to 1024. This value must be specified. |
| **Note**<br>You can configure the ccmCollectionIpAddress and ccmCollectionPort objects twice, to specify a primary collection server and a secondary collection server. | | |
| ccmCollectionIpAddrType | InetAddressType | (Optional) Type of IP address being used for the collection server. The only valid value is ipv4, which is the default value. |
| ccmCollectionInterval | Unsigned32 | (Optional) Specifies how often, in minutes, the billing records are streamed to the external server. The valid range is 2 to 1440 minutes (24 hours), with a default of 30 minutes. (We recommend a minimum interval of 30 minutes.) |
| ccmCollectionRetries | Unsigned32 | (Optional) Specifies the number of retry attempts that the CMTS will make to establish a secure connection with the external server before using the secondary server (if configured) and sending an SNMP trap about the failure. The valid range for *n* is 0 to 5, with a default of 0. |
| **Note**<br>The ccmCollectionInterval and ccmCollectionRetries parameters are optional when configuring usage-based billing for streaming mode with SNMP commands, but these parameters are required when configuring the feature with CLI commands. | | |
| ccmCollectionSecure | TruthValue | (Optional) Specifies whether the Cisco CMTS should use a secure socket layer (SSL) connection when connecting with the billing application on the external server. The valid values are:<br>• true(1)—The Cisco CMTS uses a SSL connection. This option is available only on CMTS software images that support Baseline Privacy Interface (BPI) encryption.<br>• false(2)—The Cisco CMTS uses an unencrypted TCP connection. This is the default value. |

| Object | Type | Description |
|---|---|---|
| ccmCollectionCpeList | TruthValue | (Optional) Indicates whether IP addresses for customer premises equipment (CPE) devices are omitted from the billing records, so as to reduce the size of the billing records and to improve performance. The valid values are the following:<br>• true—CPE information is present (default).<br>• false—CPE information is omitted.<br>**Note**<br>When set to true, a maximum of 5 CPE IP addresses for each cable modem. |
| ccmCollectionAggregate | TruthValue | (Optional) Indicates whether all information for an individual cable modem is combined into one record. Separate counters are maintained for upstream and downstream traffic, but those counters include all service flows in that direction. The valid values are as follows:<br>• true—All service flow information for each cable modem is aggregated into a single billing record. In this configuration, the service flow ID (SFID) for the billing record is set to 0 and the service class name (SCN) is blank.<br>• false—Information for each cable modem is not aggregated into a single billing record, but instead each service flow is recorded into its own record (default). |
| ccmtrCollectionSrcIfIndex | TruthValue | (Optional) Specifies the source-interface for the billing packets. |

**Note** The following steps use the standard SNMP commands that are available on many Unix and Linux systems. For each step, replace *ip-address* with the IP address of the Cisco CMTS, and replace *rw-community-string* with an SNMP community string that provides read-write access to the router.

## Procedure

**Step 1** Set the ccmCollectionType object to 3, to enable the Usage-based Billing feature and to configure it for streaming mode:

**Example:**

```
workstation# setany -v2c
ip-address rw-community-string
 ccmCollectionType.0 -i 3
workstation#
```

**Step 2** Set the ccmCollectionIpAddress and ccmCollectionPort objects to the IP address of the external collection server and the TCP port number to which billing records should be sent:

**Example:**

```
workstation# setany -v2c
```

```
ip-address rw-community-string
 ccmCollectionIpAddress.1 -o '0a 08 06 0b'

workstation# setany -v2c
ip-address rw-community-string
 ccmCollectionPort.1 -g 6789

workstation#
```

**Step 3** (Optional) Set the ccmCollectionIpAddress and ccmCollectionPort objects a second time to specify the IP address and TCP port number of a second external collection server to which billing records should be sent, in the case that the Cisco CMTS cannot connect to the primary collection server:

**Example:**

```
workstation# setany -v2c
ip-address rw-community-string
 ccmCollectionIpAddress.1 -o '0a 08 06 0c'

workstation# setany -v2c
ip-address rw-community-string
 ccmCollectionPort.1 -g 7000

workstation#
```

**Step 4** (Optional) To change any of the other default parameters, set the appropriate objects to the desired values. For example, the following lines configure the Usage-based Billing feature for a non-secure connection, with a collection interval of 45 minutes, and a maximum number of 3 retries.

**Example:**

```
workstation# setany -v2c
ip-address rw-community-string
 ccmCollectionSecure.1 -i 2
workstation# setany -v2c
ip-address rw-community-string
 ccmCollectionInterval.1 -i 45
workstation# setany -v2c
ip-address rw-community-string
 ccmCollectionRetries.1 -i 3
workstation#
```

**Step 5** (Optional) To omit the IP addresses of CPE devices from the billing records, set the ccmCollectionCpeList object to 2 (false). The default is to include the CPE information.

**Example:**

```
workstation# setany -v2c

ip-address rw-community-string
 ccmCollectionCpeList.0 -i 2
workstation#
```

**Step 6** (Optional) To aggregate all service flow information for each cable modem in a single record, set the ccmCollectionAggregate object to 1 (true). The default is for each service flow to be written in a separate record:

**Example:**

```
workstation# setany -v2c
```

```
ip-address rw-community-string
 ccmCollectionAggregate.0 -i 1
workstation#
```

**Step 7** (Optional) To specify the source-interface for the billing packets, set the ccmtrCollectionSrcIfIndex object to 1 (true). The default is for the billing packets to automatically select a source-interface.

**Example:**

```
workstation# setany -v2c

ip-address rw-community-string
 ccmtrCollectionSrcIfIndex.0 -i 1
workstation#
```

## Examples for SNMP Commands

The following example shows the Usage-based Billing feature being configured using SNMP commands. The following display shows that a Cisco CMTS router at IP address 10.8.8.21 is configured with the default configuration (the Usage-based Billing feature is disabled):

```
workstation# getmany -v2c 10.8.8.21 rw-string ciscoCableMeteringMIB

ccmCollectionType.0 = none(1)
ccmCollectionFilesystem.0 =
ccmCollectionCpeList.0 = true(1)
ccmCollectionAggregate.0 = false(2)
ccmCollectionStatus.0 = 0
ccmCollectionDestination.0 =
ccmCollectionTimestamp.0 = 00 00  00 00   00 00  00 00
ccmCollectionNotifEnable.0 = true(1)
workstation#
```

The following SNMP commands are then given to enable the Usage-based Billing feature and to configure it for streaming mode:

```
workstation# setany -v2c 10.8.8.21 rw-string ccmCollectionType.0 -i 3
workstation# setany -v2c 10.8.8.21 rw-string ccmCollectionIpAddress.1 -o '0a 08 06 0b'

workstation# setany -v2c 10.8.8.21 rw-string ccmCollectionPort.1 -g 6789

workstation# setany -v2c 10.8.8.21 rw-string ccmCollectionSecure.1 -i 2
workstation# setany -v2c 10.8.8.21 rw-string ccmCollectionRetries.1 -i 3
workstation# setany -v2c 10.8.8.21 rw-string ccmCollectionInterval.1 -i 45
workstation#
```

These commands add the following line to the router's running configuration file:

```
Router# show running-config | include metering

cable metering destination 10.8.6.11 6789 3 45 non-secure
```

```
Router#
```

The following SNMP display shows the new configuration:

```
workstation# getmany -v2c 10.8.8.21 rw-string ciscoCableMeteringMIB

ccmCollectionType.0 = stream(3)
ccmCollectionFilesystem.0 =
ccmCollectionIpAddrType.1 = ipv4(1)
ccmCollectionIpAddress.1 = 0a 08  06 0b
ccmCollectionPort.1 = 6789
ccmCollectionInterval.1 = 45
ccmCollectionRetries.1 = 3
ccmCollectionSecure.1 = false(2)
ccmCollectionRowStatus.1 = active(1)
ccmCollectionCpeList.0 = true(1)
ccmCollectionAggregate.0 = false(2)
ccmCollectionStatus.0 = 0
ccmCollectionDestination.0 =
ccmCollectionTimestamp.0 = 00 00  00 00   00 00  00 00
ccmCollectionNotifEnable.0 = true(1)
workstation#
```

# Enabling and Configuring the Secure Copy Protocol (optional)

This section describes how to configure the Cisco CMTS for the Secure Copy Protocol (SCP), which allow an external server to log in to the Cisco CMTS and copy the billing records from the Cisco CMTS to the external server.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable`<br>**Example:**<br>`Router#` | Enables privileged EXEC mode. Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal`<br>**Example:**<br>`Router(config)#` | Enters global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | **aaa new-model**<br>**Example:**<br>`Router(config)# aaa new-model`<br>**Example:**<br>`Router(config)#` | Enables the Authentication, Authorization, and Accounting (AAA) access control model. |
| **Step 4** | **aaa authentication login** {**default** \| *list-name* } *method1* [*method2* ...]<br>**Example:**<br>`Router(config)# aaa authentication login default enable`<br>**Example:**<br>`Router(config)#` | Enables AAA access control authentication at login, using the following parameters:<br>Valid methods include **enable**, **line**, and **local**. |
| **Step 5** | **aaa authorization exec** {**default** \| *list-name* } *method1* [*method2* ...]<br>**Example:**<br>`Router(config)# aaa authorization exec default local`<br>**Example:**<br>`Router(config)#` | Configures the CMTS to allow users to run an EXEC shell and access the CLI to run the Secure Copy commands.<br>Valid methods include **local**. |
| **Step 6** | **username** *name* **privilege** *level* **password** *encryption-type password*<br>**Example:**<br>`Router(config)# username billingapp privilege 15 password 7 billing-password`<br>**Example:**<br>`Router(config)#` | (Optional) Creates a user account for login access and specifies the privilege level and password for that account:<br>**Note** This step is optional but for the purposes of security and management, Cisco recommends creating a unique account for the billing application to use when logging into the CMTS. |
| **Step 7** | **ip ssh time-out** *seconds*<br>**Example:**<br>`Router(config)# ip ssh time-out 120`<br>**Example:**<br>`Router(config)#` | Enables Secure Shell (SSH) access on the Cisco CMTS, which is required for SCP use. The *seconds* parameter specifies the maximum time allowed for SSH authentication, in seconds, with a valid range of 0 to 120 seconds, with a default of 120 seconds. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 8** | **ip ssh authentication-retries** *n*<br>**Example:**<br>`Router(config)# ip ssh authentication-retries 3`<br>**Example:**<br>`Router(config)#` | Specifies the maximum number of login attempts a user is allowed before the router disconnects the SSH session. The valid range is 1 to 5, with a default of 3 attempts. |
| **Step 9** | **ip scp server enable**<br>**Example:**<br>`Router(config)# ip scp server enable`<br>**Example:**<br>`Router(config)#` | Enables SCP access on the Cisco CMTS. |
| **Step 10** | **end**<br>**Example:**<br>`Router(config)# end`<br>`Router#` | Exits global configuration mode and returns to privileged EXEC mode. |

# Configuring the Cisco CMTS for SSL Operation

This section describes the procedures to configure the Cisco CMTS for secure socket layer (SSL) operation, so that the Usage-based Billing feature can use an SSL connection to transfer the billing record files in streaming mode.

**Note** This procedure is required only when using the **secure** option with the **cable metering destination** command.

## Prerequisites for CA

- The billing application server must be configured for SSL operations.
- A Certificate Authority (CA) must be configured to provide the required digital certificates to the billing application and Cisco CMTS router. The CA can be a public CA, such as Verisign, or a server on your private management network that is running software such as the Cisco Provisioning Center (CPC).

SUMMARY STEPS

To prepare the Cisco CMTS router for SSL operation, you must perform the following configuration steps:

- Configuring the router's host name and IP domain name, if not already done.
- Generating an RSA key pair.
- Declaring a Certification Authority.
- Configuring a Root CA (Trusted Root).

- Authenticating the CA.
- Requesting the Certificates.

For the detailed steps in performing these procedures, see the Configuring Certification Authority Interoperability

# Retrieving Records from a Cisco CMTS in File Mode

When the Usage-based Billing feature is enabled and configured for File mode, the billing application server must regularly retrieve the billing records from the Cisco CMTS. This is typically done by a script that either logs in to the Cisco CMTS and uses CLI commands to transfer the file, or by a script that uses SNMP commands to transfer the file.

When using CLI commands, the procedure is typically as follows:

1. The billing application server receives an SNMP trap from the Cisco CMTS when a billing record is written. This notification contains the file name of the billing record that should be retrieved.
2. The billing application server starts a custom-written script to retrieve the billing record. This script would do one of the following:
   a. If using CLI commands, the script logs in to the Cisco CMTS using a telnet connection, and then transfers the billing record to the billing application server, using the **copy** CLI command. The transfer can be done using either the Secure Copy Protocol (SCP) or the Trivial File Transfer Protocol (TFTP).

**Note** You could also use the File Transfer Protocol (FTP) to transfer files from the Cisco CMTS to an external FTP server, but this is not recommended, because the FTP protocol transmits the login username and password in cleartext.

   b. If using SNMP commands, the script sets the ciscoFlashCopyEntry objects in the CISCO-FLASH-MIB to transfer the billing record to the application server, using TFTP.
3. After transferring the billing record, the script deletes it on the Cisco CMTS file system, so that the Cisco CMTS can begin writing a new billing record.

The following sections show examples of how this can be done, using each method.

**Tip** The following examples are given for illustration only. Typically, these commands would be incorporated in automated scripts that would retrieve the billing records.

## Using SCP

To transfer billing records using SCP, you must first enable and configure the router for SCP operation, using the procedure given in the “Enabling and Configuring Secure Copy (optional)” section on page 21 . Then, the application server must log in to the Cisco CMTS and use the **copy** command at the privileged EXEC prompt. The **copy** command needs to specify the location of the billing record on the local filesystem and the destination server for the SCP transfer.

The following example shows a typical session where a billing record on slot0 is transferred to an FTP server with the hostname of billserver.mso-example.com:

```
CMTS01# copy slot0:CMTS01_20030211-155025 scp://billingapp-server.mso-example.com/

Address or name of remote host [billingapp-server.mso-example.com]?
Destination username [billing-app]?
Destination filename [CMTS01_20030211-155025]?
Writing CMTS01_20030211-155025
Password: billing-password

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
[OK - 1403352/1024 bytes]
1403352 bytes copied in 17.204 secs (85631 bytes/sec)
CMTS01# delete slot0:CMTS01_20030211-155025

CMTS01# squeeze slot0:

CMTS01#
```

**Note** The billing application must delete the billing record after it has been successfully transferred, so that the Cisco CMTS can write the next record. The **squeeze** command frees up the deleted disk space on Flash Memory and old-style PCMCIA cards (bootflash, flash, slot0, slot1). It is not needed on the newer ATA-style PCMCIA cards (disk0, disk1, disk2). However, because the **squeeze** command takes several seconds to complete, it should be given only when insufficient disk space exists for a new billing record. To avoid this problem, Cisco recommends using a 64 MB (or larger) ATA-style PCMCIA memory card, which automatically reclaims disk space for deleted files.

## Using TFTP

To transfer billing records using TFTP, you must first configure an external workstation to be a TFTP server. For security, the TFTP server should be isolated from the Internet or any external networks, so that only authorized TFTP clients, such as the Cisco CMTS router, can access the server.

To transfer the billing records, the application server must log in to the Cisco CMTS and use the **copy** command at the privileged EXEC prompt. The **copy** command needs to specify the location of the billing record on the local filesystem and the destination server for the TFTP transfer.

The following example shows a typical session where a billing record on slot0 is transferred to an TFTP server with the hostname of billserver.mso-example.com.

```
Router# copy slot0:CMTS01_20030211-155025 tftp://billingapp-server.mso-example.com/incoming

Address or name of remote host [billingapp-server.mso-example.com]?
Destination username [billing-app]?
Destination filename [CMTS01_20030211-155025]?
Writing CMTS01_20030211-155025
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
[OK - 1102348/1024 bytes]
1102348 bytes copied in 14.716 secs (63631 bytes/sec)
Router# delete slot0:CMTS01_20030211-155025

Router# squeeze slot0:

Router#
```

**Note** The billing application must delete the billing record after it has been successfully transferred, so that the Cisco CMTS can write the next record. The **squeeze** command frees up the deleted disk space on Flash Memory and old-style PCMCIA cards (bootflash, flash, slot0, slot1). It is not needed on the newer ATA-style PCMCIA cards (disk0, disk1, disk2). However, because the **squeeze** command takes several seconds to complete, it should be given only when insufficient disk space exists for a new billing record. To avoid this problem, Cisco recommends using a 64 MB (or larger) ATA-style PCMCIA memory card, which automatically reclaims disk space for deleted files.

## Using SNMP

To transfer billing record file using SNMP commands, you must set a number of objects in the CISCO-FLASH-MIB to transfer the file to a TFTP server. After the file has been successfully transferred, you can then use SNMp commands to delete the billing record file.

**Note** Before proceeding with these steps, ensure that the TFTP server is properly configured to receive to receive the billing records. At the very least, this means creating a directory that is readable and writable by all users. On some servers, the TFTP server software also requires that you create a file with the same name as the file that is to be received, and this file should also be readable and writable by all users.

To transfer a billing record file to a TFTP server, using SNMP commands, you must set a number of objects in the CISCO-FLASH-MIB.

***Table 6: Transferring a File to a TFTP Server Using SNMP Commands***

| Object | Type | Description |
|---|---|---|
| ciscoFlashCopyEntryStatus | RowStatus | Status of this table entry. Typically, this object is first set to 5 (create-and-wait). Then after all other parameters are specified, it is set to Active (1) to execute the command. |
| ciscoFlashCopyCommand | INTEGER | Type of copy command to be performed. To copy a billing record file to a TFTP server, set this object to 3 (copyFromFlash). |
| ciscoFlashCopyServerAddress | IpAddress | IP address of the TFTP server.<br>**Note**<br>This parameter defaults to the broadcast address of 255.255.255.255, which means it will transfer the billing record file to the first TFTP server that responds. For security, this object should always be set to the IP address of the authorized TFTP server. |
| ciscoFlashCopySourceName | DisplayString | Name of the billing record file to be transferred, including the Flash device on which it is stored. |

| Object | Type | Description |
|---|---|---|
| ciscoFlashCopyDestinationName | DisplayString | (Optional) Name for the billing record, including path, on the TFTP server. If not specified, the copy operation defaults to saving the billing record at the top-most directory on the TFTP server, using the original file name.<br>**Note**<br>A file with the destination file name should already exist on the TFTP server. This file should be readable and writable by all users, so that it can be replaced with the billing record file. |
| ciscoFlashCopyProtocol | INTEGER | (Optional) Specifies the protocol to be used when copying the file. For a TFTP transfer, set this object to 1 (tftp), which is the default. |
| ciscoFlashCopyNotifyOnCompletion | TruthValue | (Optional) Specifies whether the Cisco CMTS should generate a trap upon the completion of the copy operation. The default is false (no trap is generated). |

After transferring the billing records file, you must then set a number of objects in the CISCO-FLASH-MIB to delete the file, so that the Cisco CMTS can begin writing a new file. If the Flash memory is not ATA-compatible, you must also set a number of objects to squeeze the Flash memory to make the deleted space available for new files. Table 7: Deleting a File Using SNMP Commands , on page 32 describes each of these objects, and whether they are required or optional.

***Table 7: Deleting a File Using SNMP Commands***

| Object | Type | Description |
|---|---|---|
| ciscoFlashMiscOpCommand | INTEGER | Specifies the operation to be performed:<br>• 3—Delete the file.<br>• 5—Squeeze the Flash memory, so as to recover the deleted space and make it available for new files. |
| ciscoFlashMiscOpDestinationName | DisplayString | When deleting a file, the name of the file to be deleted, including the name of the file system, up to a maximum of 255 characters.<br>When squeezing a file system, the name of the file system to be squeezed (slot0:, slot1:, flash:, or bootflash:). |
| ciscoFlashMiscOpEntryStatus | RowStatus | Status of this table entry. Typically, this object is first set to 5 (create-and-wait). Then after all other parameters are specified, it is set to Active (1) to execute the command. |
| ciscoFlashMiscOpNotifyOnCompletion | TruthValue | (Optional) Specifies whether the Cisco CMTS should generate a trap upon the completion of the operation. The default is false (no trap is generated). |

DETAILED STEPS

**Note** The following steps use the standard SNMP commands that are available on many Unix and Linux systems. For each step, replace *ip-address* with the IP address of the Cisco CMTS, and replace *rw-community-string* with an SNMP community string that provides read-write access to the router.

### Copying the Billing Record File to the TFTP Server

## Procedure

**Step 1** The script performing the copy should generate a 32-bit number to be used as the index entry for this copy command. The script can generate this number in any convenient way, so long as the index number is not currently being used for another operation.

**Step 2** Create the table entry for the copy command, by using the number that was generated in Step 1 and setting the ciscoFlashCopyEntryStatus object to the create-and-wait state (5):

**Example:**

```
workstation# setany -v2c ip-address rw-community-string ciscoFlashCopyEntryStatus.582 -i 5

workstation#
```

**Step 3** Set the ciscoFlashCopyCommand to 3 (copyFromFlash) to specify that the billing record file should be copied from the router's Flash file system:

**Example:**

```
workstation# setany -v2c ip-address rw-community-string ciscoFlashCopyCommand
.582

-i 3
workstatioN#
```

**Step 4** Set the ciscoFlashCopyServerAddress object to the IP address of the TFTP server:

**Example:**

```
workstation# setany -v2c ip-address rw-community-string ciscoFlashCopyServerAddress
.582

-a "172.20.12.193"

workstatioN#
```

**Step 5** Set the ciscoFlashCopySourceName object to the file name, including the device name, of the billing record file to be transferred:

**Example:**

```
workstation# setany -v2c ip-address rw-community-string ciscoFlashCopySourceName
.582
-D
"slot0:CMTS01_20030211-155025
"
workstation#
```

**Step 6** (Optional) To specify a specific destination on the TFTP server, set the ciscoFlashCopyDestinationName object to the path name and file name for the billing record file on the TFTP server. (Typically, the path name and file name should already exist on the TFTP server.)

**Example:**

```
workstation# setany -v2c ip-address rw-community-string ciscoFlashCopyDestinationName
.582
-D
"/cmts01-billing/billing-file
"
workstation#
```

**Step 7** To execute the command, set the ciscoFlashCopyEntryStatus object to the active state (1):

**Example:**

```
workstation# setany -v2c ip-address rw-community-string ciscoFlashCopyEntryStatus.582 -i 1

workstation#
```

**Step 8** Periodically poll the ciscoFlashCopyStatus object until the file transfer completes:

**Example:**

```
workstation# getmany -v2c ip-address rw-community-string ciscoFlashCopyStatus
.582
  ciscoFlashCopyStatus.582 = copyInProgress(1)
workstation# getmany -v2c ip-address rw-community-string ciscoFlashCopyStatus
.582
  ciscoFlashCopyStatus.582 = copyInProgress(1)
workstation# getmany -v2c ip-address rw-community-string ciscoFlashCopyStatus
.582
  ciscoFlashCopyStatus.582 = copyOperationSuccess(2)
workstation#
```

If the file transfer fails, the most common status values that are reported by the ciscoFlashCopyStatus object are:

- 3—copyInvalidOperation. This indicates that the operation failed on the TFTP server, typically because the destination file name and path name do not exist on the TFTP server, or they exist but are not writable by all users.
- 5—copyInvalidSourceName. The file name for the billing record, as specified in ciscoFlashCopySourceName does not exist. Verify that you specified the correct device name and that no spaces exist in the file name.
- 6—copyInvalidDestName. The destination path name and file name specified in ciscoFlashCopyDestinationName is not accessible on the TFTP server. This could be because the path name does not exist or is not configured to allow write-access. This error could also occur if a file with the same path name and file name already exists on the TFTP server.
- 7—copyInvalidServerAddress. The IP address of the TFTP server specified in ciscoFlashCopyServerAddress is invalid, or the TFTP server is not responding.
- 14—copyFileTransferError. A network error occurred that prevented the file transfer from completing.

**Step 9** After the file transfer has completed successfully, set the ciscoFlashCopyEntryStatus object to 6 (delete) to delete the row entry for this copy command:

**Example:**

```
workstation# setany -v2c ip-address rw-community-string ciscoFlashCopyEntryStatus.582 -i 6

workstation#
```

### What to do next

**Deleting the Billing Record File**

## Using SNMP

After the billing record file has been successfully transferred, use the following procedure to delete the billing record on the Cisco CMTS flash file system, so that the Cisco CMTS can write the new billing record.

### Procedure

**Step 1** Generate another random number to be used as an index entry and configure the following objects in the ciscoFlashMiscOpTable:

**Example:**

```
workstation# setany -v2c ip-address rw-community-string ciscoFlashMiscOpEntryStatus
.31 -i 5

workstation# setany -v2c ip-address rw-community-string ciscoFlashMiscOpCommand
.31 -i 3
workstation# setany -v2c ip-address rw-community-string ciscoFlashMiscOpDestinationName
.31 -D
"/cmts01-billing/CMTS01_20030211-155025
"
workstation# setany -v2c ip-address rw-community-string ciscoFlashMiscOpEntryStatus
.31 -i 1

workstation#
```

**Step 2** Periodically poll the ciscoFlashMiscOpStatus object until the file transfer completes:

**Example:**

```
workstation# getmany -v2c ip-address rw-community-string ciscoFlashMiscOpStatus
.31
  ciscoFlashCopyStatus.31 = miscOpInProgress(1)
workstation# getmany -v2c ip-address rw-community-string ciscoFlashMiscOpStatus
.31
  ciscoFlashCopyStatus.582 = miscOpOperationSuccess(2)
workstation#
```

**Step 3** If the Flash memory system is not ATA-compatible (slot0:, slot1:, flash:, or bootflash:), configure the following objects in the ciscoFlashMiscOpTable to squeeze the Flash file system to recover the deleted file space:

**Example:**

```
workstation# setany -v2c ip-address rw-community-string ciscoFlashMiscOpEntryStatus
.32
-i 5

workstation# setany -v2c ip-address rw-community-string ciscoFlashMiscOpCommand
.32 -i 5
workstation# setany -v2c ip-address rw-community-string ciscoFlashMiscOpDestinationName
.32 -D slot0:
workstation# setany -v2c ip-address rw-community-string ciscoFlashMiscOpEntryStatus
.32
```

```
-i 1

workstation#
```

## Examples To Transfer Using SNMP

The following SNMP commands transfer a file named CMTS01_20030211-155025 to a TFTP server at the IP address 10.10.31.3. After the file is successfully transferred, the row entry for this copy command is deleted.

```
workstation# setany -v2c 10.8.8.21 rw-string
 ciscoFlashCopyEntryStatus.582 -i 5

workstation# setany -v2c 10.8.8.21 rw-string
 ciscoFlashCopyCommand
.582
 -i 3
workstation# setany -v2c 10.8.8.21 rw-string
 ciscoFlashCopyServerAddress
.582
 -a "10.10.31.3"

workstation# setany -v2c 10.8.8.21 rw-string
 ciscoFlashCopySourceName
.582 -D
"slot0:CMTS01_20030211-155025
"
workstation# setany -v2c 10.8.8.21 rw-string
 ciscoFlashCopyDestinationName
.582 -D
"/cmts01-billing/CMTS01_20030211-155025
"
workstation# setany -v2c 10.8.8.21 rw-string
 ciscoFlashCopyEntryStatus.582 -i 1

workstation# getmany -v2c 10.8.8.21 rw-string
 ciscoFlashCopyStatus
.582
  ciscoFlashCopyStatus.582 = copyInProgress(1)
workstation# getmany -v2c 10.8.8.21 rw-string
 ciscoFlashCopyStatus
.582
  ciscoFlashCopyStatus.582 = copyOperationSuccess(2)
workstation# setany -v2c 10.8.8.21 rw-string
 ciscoFlashCopyEntryStatus.582 -i 6

workstation#
```

The following commands show a billing record file being deleted on the Cisco CMTS file system, and the deleted file space being recovered by a squeeze operation:

```
workstation# setany -v2c 10.8.8.21 rw-string
 ciscoFlashMiscOpEntryStatus
.31 -i 5

workstation# setany -v2c 10.8.8.21 rw-string
 ciscoFlashMiscOpCommand
.31 -i 3
workstation# setany -v2c 10.8.8.21 rw-string
```

```
 ciscoFlashMiscOpDestinationName
.31 -D
"/cmts01-billing/CMTS01_20030211-155025
"
workstation# setany -v2c 10.8.8.21 rw-string
 ciscoFlashMiscOpEntryStatus
.31 -i 1

workstation# getmany -v2c 10.8.8.21 rw-string
 ciscoFlashMiscOpStatus
.31
  ciscoFlashCopyStatus.31 = miscOpInProgress(1)
workstation# getmany -v2c 10.8.8.21 rw-string
 ciscoFlashMiscOpStatus
.31
  ciscoFlashCopyStatus.582 = miscOpOperationSuccess(2)
workstation# setany -v2c 10.8.8.21 rw-string
 ciscoFlashMiscOpEntryStatus
.32 -i 5

workstation# setany -v2c 10.8.8.21 rw-string
 ciscoFlashMiscOpCommand
.32 -i 5
workstation# setany -v2c 10.8.8.21 rw-string
 ciscoFlashMiscOpDestinationName
.32 -D slot0:
workstation# setany -v2c 10.8.8.21 rw-string
 ciscoFlashMiscOpEntryStatus
.32 -i 1

workstation#
```

# Disabling the Usage-based Billing Feature

This section describes how to disable the Usage-based Billing. Giving this command immediately stops the collection of billing information. If a billing record is currently written or being streamed to an external server, the CMTS completes the operation before disabling the usage-based billing feature.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable`<br>**Example:**<br>`Router#` | Enables privileged EXEC mode. Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| | **Example:**<br>`Router(config)#` | |
| **Step 3** | **no cable metering**<br>**Example:**<br>`Router(config)# no cable metering`<br>**Example:**<br>`Router(config)#` | Immediately disables the Usage-based Billing feature and stops the collection of billing information. |
| **Step 4** | **no snmp-server enable traps cable metering**<br>**Example:**<br>`Router(config)# no snmp-server enable traps cable metering`<br>**Example:**<br>`Router(config)#` | (Optional) Disables SNMP traps for usage-based billing events. |
| **Step 5** | **no cable sflog**<br>**Example:**<br>`Router(config)# no cable sflog`<br>**Example:**<br>`Router(config)#` | (Optional) Disables the logging of deleted service flows. |
| **Step 6** | **no cable metering source-interface**<br>**Example:**<br>`Router(config)# no cable metering source-interface`<br>**Example:**<br>`Router(config)#` | (Optional) Disables a specified source-interface for the billing packets. |
| **Step 7** | **exit**<br>**Example:**<br>`Router(config)# exit`<br>**Example:**<br>`Router#` | Exits global configuration mode. |

# Configuring Certified SSL Servers for Usage-Based Billing

Cisco introduces adds support for the Secure Socket Layer (SSL) Server, used with the usage-based billing feature of the Cisco CMTS. Usage-based billing implements the DOCSIS Subscriber Account Management Interface Specification (SAMIS) format.

This new capability enables the configuration of the SSL server between the Cisco CMTS and a collection server. Certificate creation steps and **debug** commands are added or enhanced to support the SSL Server and certificates. This section describes general steps.

Refer also to the Configuring the Cisco CMTS for SSL Operation, on page 28 section.

## Generating SSL Server Certification

These general steps describe the creation and implementation of certification for the Secure Socket Layer (SSL) Server.

1. Generate the CA key.
2. Set up the open SSL environment, to include directory and sub-directory.
3. Copy files to the appropriate directories.
4. Generate the SSL Server certification request.
5. Grant the SSL Server certification request.
6. Convert the SSL Server certification to DER format.
7. Copy the SSL certification to Bootflash memory (write mem).
8. Start the SSL server.

## Configuring and Testing the Cisco CMTS for Certified SSL Server Support

Perform the following steps to configure the Cisco router to support the SSL Server and certification.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **ip domain name** *domain*<br>**Example:**<br>`Router(config)# ip domain name Cisco.com` | Defines a default domain name that the Cisco IOS software uses to complete unqualified host names (names without a dotted-decimal domain name). Do not include the initial period that separates an unqualified name from the domain name. |

| | Command or Action | Purpose |
|---|---|---|
| | | **Note** See the Domain Name System (DNS) document on Cisco.com for additional DNS information. |
| **Step 4** | **crypto key generate rsa**<br>**Example:**<br>`Router(config)# crypto key generate rsa` | Generates RSA key pairs. |
| **Step 5** | Ctrl-Z<br>**Example:**<br>`Router(config)# Ctrl-Z`<br>**Example:**<br>`Router#` | Returns to privileged EXEC mode. |
| **Step 6** | **test cable read certificate**<br>**Example:**<br>`Router# test cable read certificate` | Verifies the certificate is valid and operational on the Cisco CMTS. |
| **Step 7** | **show crypto ca certificate**<br>**Example:**<br>`Router# show crypto ca certificate` | Displays the available certificates on the Cisco CMTS. |
| **Step 8** | **configure terminal**<br>**Example:**<br>`Router# configure terminal`<br>**Example:**<br>`Router(config)#` | Enters global configuration mode. |
| **Step 9** | **cable metering destination** *ip-addr num-1 num-2 num-3* **secure**<br>**Example:**<br>`Router(config)# cable metering destination 1.7.7.7 6789 0 15 secure` | Defines the destination IP address for cable metering, to be used with the certificate. |
| **Step 10** | test cable metering<br>**Example:**<br>`Router# test cable metering` | Tests cable metering in light of the supported SSL server and metering configuration. |

# Monitoring the Usage-based Billing Feature

To display the most current billing record, use the **show cable metering-status** command. The following example shows typical output when usage-based billing is configured to write the billing records to a local file system:

```
CMTS01# show cable metering-status

destination                          complete-time   flow   cpe     status
                                                    aggr suppress
disk0:R7519-UBR7246-20000308-004428 Jun 12 09:33:05  No     No     success
CMTS01#
```

The following example shows a typical output for the **show cable metering-status** command when usage-based billing is configured to stream the billing records to an external server:

```
Router# show cable metering-status

destination                      complete-time   flow cpe  full status
                                                 aggr supp rec
10.11.37.2 :1234                 Jun 12 09:33:05  No   No  No success
Router#
```

The following example shows a typical output for the **show cable metering-status** command using verbose option:

```
Router# show cable metering-status verbose
Last export status
Destination : disk0:sunethra10k-20070129-190423
Complete Time : Jan29 19:04:38
Flow Aggregate : No
Full records : No
Cpe list suppression : No
Source interface : FastEthernet0/0/0
Status of last export : success
Current export status : In progress
```

The following example shows a typical output for the **show cable metering-status** command when usage-based billing is configured to use the IPDR Exporter to stream the billing records to an external server:

```
Router# show cable metering-status
destination                 complete-time   flow  cpe   full    status
              aggr  supp  rec
IPDR_Session2               Apr12 16:51:15  No    No    No    success
```

The following example shows a typical output for the verbose form of the **show cable metering-status** command when usage-based billing is configured to use the IPDR Exporter to stream the billing records to an external server:

```
Router# show cable metering-status
 verbose

Last export status
Destination               : IPDR_Session2
```

```
Complete Time          : Apr12 16:51:15
Flow Aggregate         : No
Full records                :No
Cpe list suppression   : No
Source interface       : Not defined
Status of last export  : success
```

**Note** If the **show cable metering-status** command displays the status of a streaming operation as "success" but the records were not received on the billing application server, verify that the Cisco CMTS and server are configured for the same type of communications (non-secure TCP or secure SSL). If the Cisco CMTS is configured for non-secure TCP and the server is configured for secure SSL, the Cisco CMTS transmits the billing record successfully, but the server discards all of the data, because it did not arrive in a secure SSL stream.

**Tip** The **show cable metering-status** command continues to show the status of the last billing record operation, until that billing record is deleted. If the record is not deleted, no new records are created.

To display information about the state of the IPDR Exporter, use the show ipdr Exporter command. The following example shows typical output:

```
Router#configure terminal
Router#show ipdr exporter
```

IPDR exporter is started.

# Configuration Examples for Usage-based Billing

This section lists the following sample configurations for the Usage-based Billing feature:

## File Mode Configuration (with Secure Copy)

The following excerpt from a configuration file shows a typical configuration for the Usage-based Billing feature when operating in file mode and enabling Secure Copy (SCP) for file transfers.

```
!
cable metering filesystem disk1:
snmp-server enable traps cable metering
...
aaa new-model
aaa authentication login default group tacacs+
aaa authorization exec default group tacacs+
username billingapp level 15 password 7 billing-password
ip ssh time-out 120
ip ssh authentication-retries 3
ip scp server enable
```

# Non-Secure Streaming Mode Configuration

The following excerpt from a configuration file shows a typical configuration for the Usage-based Billing feature when operating in streaming mode and specifying both a primary and a secondary external server. The data is sent using standard TCP packets, without any security.

```
cable metering destination 10.10.10.171 5321 10.10.10.173 5321 2 30 non-secure
snmp-server enable traps cable metering
```

The following excerpt from a configuration file shows a typical configuration for the Usage-based Billing feature when operating in streaming mode and specifying only a primary external server:

```
cable metering destination 10.10.11.181 6789 2 30 non-secure
snmp-server enable traps cable metering
```

**Note** You must ensure that the billing application server is configured for standard TCP communications. If the billing application server is configured for SSL communications when the Cisco CMTS is configured for standard TCP, the Cisco CMTS is able to send the billing records to the server, but the server discards all of that information because it is not arriving in a secure stream.

# Secure Streaming Mode Configuration

The following excerpt from a configuration file shows a typical configuration for the Usage-based Billing feature when operating in streaming mode and specifying only a primary external server. Secure socket layer (SSL) TCP connections are used to transmit the data, which requires the configuration of a digital certificate.

```
cable metering destination 10.10.11.181 6789 2 30 secure cpe-list-suppress
snmp-server enable traps cable metering
...
crypto ca trustpoint SSL-CERT
!
crypto ca certificate chain SSL-CERT
 certificate ca 00
  308204A6 3082038E A0030201 02020100 300D0609 2A864886 F70D0101 04050030
  8198310B 30090603 55040613 02555331 13301106 03550408 130A4361 6C69666F
  726E6961 3111300F 06035504 07130853 616E204A 6F736531 1C301A06 0355040A
  13134369 73636F20 53797374 656D732C 20496E63 2E311130 0F060355 040B1308
  4361626C 65204255 310E300C 06035504 03130553 65656D61 3120301E 06092A86
...
  3E65DBBA 337627E8 589980D6 C8836C7E 3D3C3BC1 F21973BF 7B287D7A 13B16DA2
  02B2B180 C2A125C7 368BDA4C 0B8C81B7 7D5BEFF9 A6618140 1E95D19E BD0A84F5
  B43702AB 39B5E632 87BA36AC A3A8A827 C5BAC0F1 B24B8F4D 55615C49 5B6E4B61
  B15CC48A 8EF566C8 6E449B49 BF8E9165 317C1734 9A48A240 78A356B5 403E9E9B
  88A51F5B 0FE38CC2 F431
  quit
!
```

**Note** You must ensure that the billing applications server is also configured for SSL communications.