

Configuring File Accounting

This chapter describes the method of capturing accounting records in comma separated value (.csv) format and storing the records to a file in internal flash or to an external FTP server.

Contents

- Prerequisites for File Accounting, page 3-39
- Restrictions for File Accounting, page 3-39
- Information About File Accounting, page 3-39
- How to Configure File Accounting, page 3-47
- Configuration Examples for File Accounting, page 3-52
- Feature Information for File Accounting, page 3-55

Prerequisites for File Accounting

- Cisco IOS XE Release 16.3.1 or a later release.
- Cisco IOS XE Cupertino 17.9.1a or a later release for transfer of call detail records (CDRs) using SFTP.

Restrictions for File Accounting

- This feature does not support Media Gateway Control Protocol (MGCP).

Information About File Accounting

To configure file accounting, you should understand the following concepts:

- File Accounting Method, page 3-40
- File Accounting Filtering, page 3-40

File Accounting Method

The file accounting feature provides a method for capturing accounting records in comma separated value (.csv) format and storing the records to a file in internal flash or to an external FTP server. It expands gateway accounting support which also includes the AAA and syslog mechanisms of logging accounting information.

The accounting process collects accounting data for each call leg created on a Cisco voice gateway. You can use this information for postprocessing activities such as generating billing records and network analysis. Cisco voice gateways capture accounting data in the form of call detail records (CDRs) containing attributes defined by Cisco. The gateway can send CDRs to a RADIUS server, syslog server, and with the new file method, to flash or an FTP server in .csv format.

Note For redundant solutions that use HSRP, CDRs are only generated by the active router.

CDRs in .csv format use the following conventions to capture accounting attributes:

- Each CDR has a fixed number and order of predefined attribute fields. Fields with no data are included as empty fields.
- Twelve fields are generic and are used to capture feature-related information. For a basic call, the call record is generated with basic call information in the feature part of the fields. The fields are static in terms of their position, however, the definitions of the feature_vsa fields are determined by the type of feature.
- A CDR is generated for each feature that is invoked. For example, if a call leg has a basic call and then a call transfer, two CDRs are generated for the following:
  - CDR with feature fields representing the basic feature
  - CDR with feature fields representing the supplementary service, for example, call transfer

The following output is an example of a CDR for a call generated using file accounting to capture records in .csv format:

```
1,48,964484051,"12345",”TWC”,1234,2345, "09/01/2006 15:39:44.747”
1,49,964484062,"12345",”CXFER”,1234,2345,3456, "09/01/2006 15:39:44.747”
```

Configuring file accounting includes defining the primary and secondary file location for storing call records. If the file transfer to the primary device fails, the gateway retries the primary device up to the configured number of times before automatically switching over to the secondary device. You can initiate a manual switchback to the primary device when it is restored. If the secondary device also fails, the accounting process ends and the system logs an error. New CDRs are dropped until one device comes back online and you manually reset.

The gateway holds call records in memory temporarily before writing the records to the specified accounting file. It appends call records to the accounting file after a configured flush-timer limit or whenever the memory buffer becomes full. The gateway closes the accounting file and opens a new file after a configured file-close time limit or you can initiate an immediate close. Other options allow you to select the specific attributes captured in the accounting record.

For configuration information, see the “Configuring File Accounting” section on page 3-47.

File Accounting Filtering

CDRs generated by the file accounting process can be filtered using one of the following three methods, depending on your data collection needs.

- Detailed File Accounting Format, page 3-41
- Compact File Accounting Format, page 3-47
- Customized Accounting Templates, page 3-47

Detailed File Accounting Format

Table 3-1 lists the name and order of the complete set of voice attribute fields generated in the detailed version of file accounting CDRs using the **cdr-format detailed** command.

Note Fields 0 to 22 are included in the compact version of the CDR.

Table 3-1 Detailed File Accounting Attributes

Field No.	Field Name	Type	Description
0	unix_time	Long	System time stamp when CDR is captured.
1	call-id	Long	Value of the Call-ID header.
2	cdr-type	Long	Template used: 0=None 1=Call history detail 2=Custom template
3	leg-type	Long	Call leg type: 1= Telephony 2=VoIP 3=MMOIP 4=Frame Relay 5=ATM
4	h323-conf-id	String	Unique call identifier generated by the gateway. Used to identify the separate billable events (calls) within a single calling session.
5	peer-address	String	Number that this call was connected to in E.164 format.
6	peer-sub-address	String	Subaddress configured under a dial peer.
7	h323-setup-time	String	Setup time in Network Time Protocol (NTP) format: hour, minutes, seconds, microseconds, time_zone, day, month, day_of_month, year.
8	alert-time	String	Time at which call is alerting.
9	h323-connect-time	String	Connect time in NTP format: hour, minutes, seconds, microseconds, time_zone, day, month, day_of_month, year.
10	h323-disconnect-time	String	Disconnect time in NTP format: hour, minutes, seconds, microseconds, time_zone, day, month, day_of_month, year.
11	h323-disconnect-cause	String	Q.931 disconnect cause code retrieved from Cisco IOS call-control application programming interface (Cisco IOS CCAPI).
12	disconnect-text	String	ASCII text describing the reason for call termination.
13	h323-call-origin	String	Gateway's behavior in relation to the connection that is active for this leg. `answer`= Legs 1 and 3 `originate`= Legs 2 and 4 `callback` = Legs 1 and 3

Table 3-1 Detailed File Accounting Attributes (continued)

Field No.	Field Name	Type	Description
14	charged-units	Long	Number of charged units for this connection. For incoming calls or if charging information is not supplied by the switch, the value is zero.
15	info-type	String	Type of information carried by media. 1=Other 9 not described 2=Speech 3=UnrestrictedDigital 4=UnrestrictedDigital56 5=RestrictedDigital 6- audio31 7=Audio7 8=Video 9=PacketSwitched
16	paks-out	Long	Total number of transmitted packets.
17	bytes-out	Long	Total number of transmitted bytes.
18	paks-in	Long	Total number of packets received.
19	bytes-in	Long	Total number of bytes received.
20	username	String	Username for authentication. Usually this is the same as the calling number.
21	clid	String	Calling number.
22	dnis	String	Called number.
23	gtd-orig-cic	String	Originating carrier identification code, used in routing to identify the network.
24	gtd-term-cic	String	Terminating carrier identification code.
25	tx-duration	String	Duration, in ms, of transmit path open from this peer to the voice gateway for the call.
26	peer-id	Long	ID value of the peer table entry to which this call was made. If a peer table entry for this call does not exist, the value of this object is zero.
27	peer-if-index	Long	ifIndex value of the peer table entry to which this call was made. If a peer table entry for this call does not exist, the value of this object is zero.
28	logical-if-index	Long	ifIndex value of the logical interface through which this call was made. For ISDN media, this is the ifIndex of the B channel that was used for this call.
29	acom-level	Long	Average ACOM level, in dB, for the call (ACOM is the combined loss achieved by the echo canceler). 1 indicates that the level cannot be determined or level detection is disabled.
30	noise-level	Long	Average noise level for the call, in dBm.
31	voice-tx-duration	String	Duration, in ms, for this call.
32	account-code	String	Account code entered using the **Acct** soft key during call setup or when connected to an active call.
33	codec-bytes	Long	Payload size of the voice packet.
34	codec-type-rate	String	Negotiated coder rate. Transmit rate of voice/fax compression to its associated call leg for the call.
35	ontime-rv-playout	Long	Duration, in ms, of voice playout from data received on time for this call.

Table 3-1 Detailed File Accounting Attributes (continued)

Field No.	Field Name	Type	Description
36	remote-udp-port	Long	Remote system UDP listener port to which voice packets are transmitted.
37	remote-media-udp-port	Long	Remote-media gateway UDP port.
38	vad-enable	String	Whether or not voice-activity detection (VAD) is enabled for the voice call.
39	receive-delay	String	Average playout FIFO delay plus the decoder delay during the voice call.
40	round-trip-delay	String	Voice-packet round-trip delay, in ms, between local and remote devices on the IP backbone during a call.
41	hiwater-playout-delay	String	High-water mark voice playout FIFO delay during the voice call.
42	lowater-playout-delay	String	Low-water mark voice playout FIFO delay during the voice call.
43	gapfill-with-interpolation	String	Duration, in ms, of the voice signal played out with the signal synthesized from parameters or samples of data preceding and following in time because of voice data not received on time (or lost) from the voice gateway for this call.
44	gapfill-with-redundancy	String	Duration, in ms, of the voice signal played out with signal synthesized from redundancy parameters available because of voice data not received on time (or lost) from the voice gateway for this call.
45	gapfill-with-silence	String	Duration, in ms, of the voice signal replaced with the signal played out during silence because of voice data not received on time (or lost) from the voice gateway for this call
46	gapfill-with-prediction	String	Duration, in ms, of voice signal played out with signal synthesized from parameters or samples of data preceding in time because of voice data not received on time (or lost) from voice gateway for this call.
47	early-packets	Long	Number of received voice packets that arrived too early to store in the jitter buffer during the call.
48	late-packets	Long	Number of received voice packets that arrived too late to play out with the codec during the call.
49	lost-packets	Long	Number of lost voice packets during the call.
50	max-bitrate	Long	Maximum bandwidth used by the video call.
51	faxrelay-start-time	String	Fax start time in a call. Multiple fax start/stop time stamps can exist in one call. Recorded for both VoIP and telephony call legs.
52	faxrelay-stop-time	String	Fax stop time in a call. Multiple fax start/stop time stamps can exist in one call. Recorded for both VoIP and telephony call legs.
53	faxrelay-max-jit-buf-depth	String	Depth of the jitter buffer, in ms.
54	faxrelay-jit-buf-ovflow	String	Number of jitter buffer overflow events during the call.
55	faxrelay-init-hs-mod	String	Initial high-speed modulation and baud rate negotiated at the time the call is connected.
56	faxrelay-mr-hs-mod	String	Most recent high-speed modulation and baud rate.
57	faxrelay-num-pages	String	Total number of transmitted and received fax pages.
58	faxrelay-tx-packets	String	Number of packets transmitted.
59	faxrelay-rx-packets	String	Number of packets received.

Table 3-1 Detailed File Accounting Attributes (continued)

Field No.	Field Name	Type	Description
60	faxrelay-direction	String	Whether a fax was originated (transmitted) or terminated (received) by this gateway.
61	faxrelay-pkt-conceal	String	Packet loss concealment; number of white scan lines inserted (nonzero for outbound gateway only).
62	faxrelay-ecm-status	String	Whether error correction mode is enabled.
63	faxrelay-encap-protocol	String	Encapsulation protocol used for fax transfer.
64	faxrelay-nsf-country-code	String	NSF country code of the local fax device; country name per T.35, Annex A.
65	faxrelay-nsf-manuf-code	String	NSF manufacturer code of the local fax device.
66	faxrelay-fax-success	String	Whether fax transfer was successful, the transfer failed, or indeterminate.
67	override-session-time	Long	Override session time.
68	h323-ivr-out	String	AV pairs sent from the voice gateway to the RADIUS server that you can define. You can set (write) the value with a customized Tcl IVR script.
69	internal-error-code	String	Cause of failed calls. For more information, see the "Internal Error Codes" section on page 4-92.
70	h323-voice-quality	String	Value representing impairment/calculated planning impairment factor (ICPIF) of the voice quality on the connection provided by lower-layer drivers (such as the digital-signal-processor). Low numbers represent better quality.
71	remote-media-address	String	Remote-media gateway IP address.
72	remote-media-id	Long	Remote-media gateway DNS name.
73	carrier-id	Long	ISUP carrier ID.
74	calling-party-category	String	Best-fit calling party category value extracted from the Generic Transparency Descriptor (GTD). Sent in start and stop accounting messages for call legs 1 and 4. Optionally, this field also contains: • 3-character country code representing the country variant extracted from the GTD Protocol Name (PRN) country field. • National value extracted from the GTD Field Compatibility Information (FDC) data field.
75	originating-line-info	Long	Sent in start and stop accounting messages for call legs 1 and 4.
76	charge-number	String	Charge number used for call.
77	transmission-medium-req	Long	Sent in start and stop accounting records for call legs 1 and 4.
78	service-descriptor	String	Gatekeeper-related.
79	outgoing-area	String	Gatekeeper identifier, or the destination zone or area, of the outgoing VoIP call.
80	incoming-area	String	Gatekeeper identifier, or the source zone or area, of the incoming VoIP call.
81	out-trunkgroup-label	String	Trunk-group label associated with the group of voice ports from which the outgoing TDM call leaves the gateway.
82	out-carrier-id	String	Carrier ID of the trunk group through which the call leaves the gateway or the partnering voice services provider identifier of the outgoing VoIP call.

Table 3-1 Detailed File Accounting Attributes (continued)

Field No.	Field Name	Type	Description
83	dsp-id	String	DSP ID used for the current call.
84	in-trunkgroup-label	String	Trunk group label associated with the group of voice ports from which the incoming TDM call arrived on the gateway.
85	in-carrier-id	String	Carrier ID of the trunk group through which the call arrived or the partnering voice service provider identifier of the incoming VoIP call.
86	cust-biz-grp-id	String	SIP business group ID.
87	supp-svc-xfer-by	String	Transferor information in the REFER/BYE/ALSO of SIP call. Used only in SIP call transfer.
88	voice-feature	String	Type of feature: BXFER = Blind transfer CFA = Call forward all CFBY = Call forward busy CFNA = Call forward no answer CXFER = Consultative transfer TWC = Two-way call
89	feature-operation	String	Feature operation.
90	feature-op-status	String	Success (0) or failure (1).
91	feature-op-time	String	Feature operation time. Time stamp of the operation start and stop time, if applicable for a given feature.
92	feature-id	String	Feature ID of the invocation. Identifies a unique instance of a feature attribute within a gateway. This number is incremented for each added feature attribute.
93	gw-rxd-cdn	String	Called number received in the incoming signaling message before any translation rules are applied.
94	gw-rxd-cgn	String	Calling number received in the incoming signaling message before any translation rules are applied.
95	gtd-gw-rxd-ocn	String	Original calling number received by the gateway.
96	gtd-gw-rxd-cnn	String	GTD connected number.
97	gw-rxd-rdn	String	Redirection number received by the gateway.
98	gw-final-xlated-cdn	String	Called number to be sent out of the gateway.
99	gw-final-xlated-cgn	String	Calling number to be sent out of the gateway.
100	gw-final-xlated-rdn	String	Final translated received number.
101	gk-xlated-cdn	String	Called number presented by the gatekeeper in the ACF RAS message. GK/GKTMP could modify the called number by appending a prefix or leave it unchanged.
102	gk-xlated-cgn	String	Calling number presented by the gatekeeper in the ACF RAS message. The GK/GKTMP could modify the calling number which is carried in the ACF nonstandard parameter.
103	gw-collected-cdn	String	Destination number collected by the gateway (application) that is used to route the call. Only applicable for 2-stage calls.

Table 3-1 Detailed File Accounting Attributes (continued)

Field No.	Field Name	Type	Description			
104	ip-hop	String	Maximum number of hops in the SIP invite message.			
105	redirected-station	String	Redirecting number extracted from the redirect number parameter. Sent in start accounting messages for all call legs. noa=Nature of address npi=Numbering plan indicator pi=Presentation indicator #=Address of the redirecting number			
106	subscriber	String	T1/channel associated signaling (CAS) or E1/R2 signal information about a subscriber.			
107	in-intrfc-desc	String	Description assigned to the voice port of the incoming call.			
108	out-intrfc-desc	String	Description assigned to the voice port of the outgoing call.			
109	session-protocol	String	Session protocol used for calls between the local and remote router through the IP backbone. Always equal to “sip” for SIP or “Cisco” for H.323.			
110	local-hostname	String	Local hostname that would be accessed or used by the SNMP MIBs. **Note** Only if hostname is configured under localhost in voice service voip/sip, local hostname field in CDR is generated.			
111	backward-call-id	String	Sent in stop accounting messages for call legs 1 and 4. Also included in interim-update packets.			
112	feature-id_field1	String	Feature name. Two-Way Call (TWC), Call Forward All (CFA), Call Forward Busy (CFBY), Call Forward No Answer (CFNA), Blind Transfer (BXFER), Consultive Transfer (CXFER), Hold (HOLD), Resume (RESUME).			
113	feature-id_field2	String	Feature invocation time.			
			TWC	**CFA, CFNA, CFBY**	**BXFER, CXFER**	**HOLD/RESUME**
114	feature-id_field3	String	calling number	feature status (frs)	frs	frs
115	feature-id_field4	String	called number	feature ID (fid)	fid	fid
116	feature-id_field5	String	frs	fcid	fcid	fcid
117	feature-id_field6	String	fid	legID	XconsID	legID
118	feature-id_field7	String	fcid	frson	legID	hrson
119	feature-id_field8	String	legID	fdcnt	frson	holding
120	feature-id_field9	String	Not used	fwder	xsts	held
121	feature-id_field10	String	Not used	fwdee	Xor	sl
122	feature-id_field11	String	Not used	fwdto	Xee	usr
123	feature-id_field12	String	Not used	frm	Xto	tag
124	ip-phone-info	String	Information about the ip-phone that is initiating the call			
125	ip-pbx-mode	String	Indication is the CDR was generated by CME or SRST			

Table 3-1 Detailed File Accounting Attributes (continued)

Field No.	Field Name	Type	Description
126	in-lpcor-group	String	Incoming LPCOR group number
127	out-lpcor-group	String	Outgoing LPCOR group number
128	fac-digit	String	Forced Authorization Code
129	fac-status	String	Forced Authorization Status
130	MOSQe (conversational quality MOS)	String	The conversational quality MOS. Conversational quality indicates the impact of the quality of the transmission on the dynamics of conversational exchanges between two parties; such metrics take into account delay, echo, and recency.
131	Packet-Loss-Rate	String	The total number of packets lost by the jitter buffer in the RTP stream.
132	Out-of-Order-Rate	String	The total number of packets that arrive at the jitter buffer out of sequence.

Note For description of fields 114 to 123, see the "Feature VSA Attributes" section on page 4-84.

Compact File Accounting Format

If you do not need the complete set of voice attributes supported by the file accounting process, a smaller, compact set is configurable using the **cdr-format compact** command. The compact version of the CDR captures the first 23 attributes (0 to 22) listed in Table 3-1, in the order listed.

Customized Accounting Templates

You can create accounting templates to customize your CDRs based on your billing needs. You create a template by using a text file that lists the names of the desired attributes. Only those attribute values defined in the template are sent to the accounting server.

Note For file accounting, you cannot delete attribute fields or change the order of the attributes using an accounting template. Any attribute not included in the template appears as a blank field in the CDR.

To use a customized template for filtering the specific voice attributes included in CDRs, see the "Customized Accounting Records" section on page 2-8.

How to Configure File Accounting

This section contains the following tasks:

- Configuring File Accounting, page 3-47 (required)
- Manually Initiating File Processes, page 3-51 (optional)
- Troubleshooting File Accounting, page 3-51 (optional)

Configuring File Accounting

To generate CDRs in file format (.csv), perform the following steps.

Note From Cisco IOS XE Cupertino 17.9.1a onwards, both FTP and SFTP passwords are encrypted.

Prerequisites

- Cisco IOS XE Release 16.3.1 or a later release.

Restrictions

FTP or SFTP servers in Cisco IOS software are not supported because they cannot append CDRs to a file, so every flush would create a new file.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **gw-accounting file**
4. **primary** {**{ftp | sftp }** *path/filename* **username** *username* **password** *password* | **ifs** *device***:***filename*}
5. **secondary** {{**ftp** | **sftp**} *path/filename* **username** *username* **password** *password* | **ifs** *device***:***filename*}
6. **maximum retry-count** *number*
7. **maximum buffer-size** *kbytes*
8. **maximum fileclose-timer** *minutes*
9. **maximum cdrflush-timer** *minutes*
10. **cdr-format** {**compact** | **detailed**}
11. **acct-template** {*template-name* | **callhistory-detail**}
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**gw-accounting file** **Example:** `Router(config)# gw-accounting file`	Enables the file method of accounting.

Step 4	**primary** {{**ftp** | *sftp} path/filename* **username** *username* **password** *password* | **ifs** *device*:*filename*}
	
	Example:
	`Router(config-gw-accounting-file)# primary ftp server1/cdrtest1 username bob password mypass`
	
	Example:
	`Router(config-gw-accounting-file)# primary sftp server1/cdrtest1 username bob password mypass`
		(Optional) Sets the primary location for storing the CDRs generated for file accounting.
		• **ftp** *path/filename*—Name and location of the file on an FTP server.
		• **sftp** *path/filename*—Name and location of the file on an SFTP server.
		• **ifs** *device*:*filename*—Name and location of the file in flash memory or other internal file system on this router. Values depend on the storage devices available on the router, for example flash or slot0.
		• **username** *username*—User ID for authentication.
		• **password** *password*—Password user enters for authentication.
		• Default: **flash:cdr**.

Step 5	**secondary** {{**ftp** | sftp} *path/filename* **username** *username* **password** *password* | **ifs** *device*:*filename*} **Example:** `Router(config-gw-accounting-file)# secondary ifs flash:cdrtest2`	(Optional) Sets the backup location for storing CDRs if the primary location becomes unavailable. • **ftp** *path/filename*—Name and location of the backup file on an FTP server. • **sftp** *path/filename*—Name and location of the file on an SFTP server. • **ifs** *device*:*filename*—Name and location of the backup file in flash memory or other internal file system on this router. Values depend on the storage devices available on the router, for example flash or slot0. • **username** *username*—User ID for authentication. • **password** *password*—Password user enters for authentication. • Default: **flash:cdr**.

	Command or Action	Purpose
Step 6	**maximum retry-count** *number* **Example:** `Router(config-gw-accounting-file)# maximum retry-count 3`	(Optional) Sets the maximum number of times the router attempts to connect to the primary file device before switching to the secondary device. • *number*—Number of connection attempts. Range:1 to 5. Default: 2.
Step 7	**maximum buffer-size** *kbytes* **Example:** `Router(config-gw-accounting-file)# maximum buffer-size 25`	(Optional) Sets the maximum size of the file accounting buffer. • *kbytes*—Maximum buffer size, in kilobytes. Range: 6 to 40. Default: 20.
Step 8	**maximum fileclose-timer** *minutes* **Example:** `Router(config-gw-accounting-file)# maximum fileclose-timer 300`	(Optional) Sets the maximum time for writing records to an accounting file before closing it and creating a new file. • *minutes*—Maximum time, in minutes, to write records to an accounting file. Range: 60 to 1,440. Default: 1,440 (24 hours). • Set this file close timer to at least five minutes longer than the flush timer set with the **maximum cdrflush-timer** command.
Step 9	**maximum cdrflush-timer** *minutes* **Example:** `Router(config-gw-accounting-file)# maximum cdrflush-timer 245`	(Optional) Sets the maximum time to hold call records in the buffer before appending the records to the accounting file. • *minutes*—Maximum time, in minutes, to hold call records in the accounting buffer. Range: 1 to 1,435. Default: 60 (1 hour). • Set this flush timer to at least five minutes less than the file close timer set with the **maximum fileclose-timer** command.
Step 10	**cdr-format** {**compact** | **detailed**} **Example:** `Router(config-gw-accounting-file)# cdr-format compact`	(Optional) Selects the format of the CDRs generated for file accounting. • **compact**—Compact set of voice attributes is generated in CDRs. • **detailed**—Full set of voice attributes is generated in CDRs. Default value.
Step 11	**acct-template** {*template-name* | **callhistory-detail**} **Example:** `Router(config-gw-accounting-file)# acct-template custom1`	(Optional) Selects the voice attributes to collect. • *template-name*—Name of custom accounting template that defines the attribute values to collect. • **callhistory-detail**—Collects all voice VSAs for accounting.
Step 12	**end** **Example:** `Router(config-gw-accounting-file)# end`	Exits to privileged EXEC mode.

Manually Initiating File Processes

To manually flush the buffer or to force a switch back to the primary file device from the secondary device, perform the following steps.

Prerequisites

- Cisco IOS XE Release 16.3.1 or a later release.

SUMMARY STEPS

1. **enable**
2. **file-acct flush** {**with-close** | **without-close**}
3. **file-acct reset**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**file-acct flush** {**with-close** | **without-close**} **Example:** `Router# file-acct flush with-close`	(Optional) Flushes pending accounting records to the file and closes the file.
Step 3	**file-acct reset** **Example:** `Router# file-acct reset`	(Optional) Switches back to the primary file location after flushing records to the active location.

Troubleshooting File Accounting

To troubleshoot the file accounting configuration, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **debug voip fileacct**
3. **debug voip dump-file-acct**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**debug voip fileacct** **Example:** `Router# debug voip fileacct`	Displays debugging messages related to generating attributes for file accounting.
Step 3	**debug voip dump-file-acct** **Example:** `Router# debug voip dump-file-acct`	Displays debugging messages related to file accounting flushing processes.

Configuration Examples for File Accounting

This section contains the following examples:

- File Accounting Configuration: Example, page 3-53
- File Accounting Filename: Example, page 3-53
- File Accounting Detailed CDR: Example, page 3-53
- File Accounting Compact CDR: Example, page 3-54
- Hold and Resume CDR: Example, page 3-54

File Accounting Configuration: Example

```
Router# show running-config | section gw-accounting

gw-accounting file
 primary ftp [server]/cdrtest1 username bob password 6 TI[^VcViOKEXJbU_I^UWNYBfHQbKfOAAB
 secondary ifs flash:cdrtest2
 maximum buffer-size  15
 maximum retry-count 3
 maximum fileclose-timer 300
 maximum cdrflush-timer 245
 cdr-format compact

gw-accounting file
 primary sftp 203.0.113.13/cdrtest username bob password 6 P^AV^_3
 secondary ifs flash:cdrtest2
 maximum buffer-size  15
 maximum retry-count 3
 maximum fileclose-timer 300
 maximum cdrflush-timer 245
 cdr-format compact

gw-accounting file
 primary sftp [2001:420:54ff:13::312:175]//cdrtest username bob password 6 P^AV^_3
 secondary ifs flash:cdrtest2
 maximum buffer-size  15
 maximum retry-count 3
 maximum fileclose-timer 300
 maximum cdrflush-timer 245
 cdr-format compact
```

File Accounting Filename: Example

The following examples show how the accounting file is given a unique name when it is created. The router hostname and time stamp are appended to the filename that you assign with the **primary** command at the time the accounting file is created.

```
cme-2821(config)# primary ftp server1/cdrtest1 username bob password temp
cme-2821(config)# primary sftp server1/cdrtest1 username bob password temp
```

The name of the accounting file that is created uses the *filename.hostname.timestamp* format:

```
cdrtest1.cme-2821.06_04_2007_18_44_51.785
```

File Accounting Detailed CDR: Example

The following example shows a CDR captured by file accounting using the detailed format. Because file accounting records are in .csv format, fields with no data are included as empty fields.

```
1746114646, 26529506, 1, 2, 7138BE00 10000 60D2A 4211A845, 6158626400, 15:46:29.800 UTC Thu May 1 2025,
15:46:35.470 UTC Thu May 1 2025, 15:50:46.640 UTC Thu May 1 2025, 10, normal call clearing (16), originate,
0, speech, 12703, 2540600, 12731, 2546200, 6153918077, 6153918077, 6158626400, "", "", 25117ms, 1102, 11, 0,
"", "", "", "", 160, g711ulaw, 0, 5060, 13088, disable, 0 ms, 11870 ms, 0 ms, 0 ms, 0 ms, 0 ms, 0 ms,
0, 0, 0, "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", 251, "", "", 0, 192.168.224.196,
"", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "",
ton:0,npi:0,#:+16158626400, ton:0,npi:0,pi:0,si:1,#:6153918077, "", "", "", ton:0,npi:0,#:6158626400,
ton:0,npi:0,pi:0,si:1,#:6153918077, "", "", "", "", "", "", Unknown, "", "", sipv2, "", "", TWC, 46:29.8,
6153918077, 6158626400, 0, 20768387, 7138BE00 10000 60D2A 4211A845, 194CEE2, "", "", "", "", "", "", "", "",
"", "", 1, 0.00%, 0.00%
```

```
11780434730,8,1,1,"9D4B0CA F74711DB 800D96DB A749148A","0163","","11:17:23.413 pdt Tue May 1
2007","11:17:23.413 pdt Tue May 1 2007","11:17:26.023 pdt Tue May 1 2007","11:17:53.243 pdt Tue May 1
2007","10  ","normal call clearing (16)","",0,"",0,0,0,0,"5105550160","5105550160","0163","","","0
ms",20005,29,28,0,0,"0
ms","","","g711ulaw","","","","","","","","","","","","","","","","","","","","","","","","","","","
","","",27,"Tariff:Unknown","","0","","","","","","","","","","","","","","","","","","","","","","","ton:
0,npi:0,pi:0,si:0,#:5105550160","","","","","ton:0,npi:0,pi:0,si:0,#:5105550160","","","","0163","","","Regul
arLine","","","","","","CXFER","05/01/2007 11:17:53.239",0,17,9D4B0CA F74711DB 800D96DB
A749148A,"1BD61",8,0,5,"5105550163","5105550160","3002"

11780434730,8,1,1,"9D4B0CA F74711DB 800D96DB A749148A","0163","","11:17:23.413 pdt Tue May 1
2007","11:17:23.413 pdt Tue May 1 2007","11:17:26.023 pdt Tue May 1 2007","11:17:53.243 pdt Tue May 1
2007","10  ","normal call clearing (16)","",0,"",0,0,0,0,"5105550160","5105550160","0163","","","0
ms",20005,29,28,0,0,"0
ms","","","g711ulaw","","","","","","","","","","","","","","","","","","","","","","","","","","","
","","",27,"Tariff:Unknown","","0","","","","","","","","","","","","","","","","","","","","","","","ton:
0,npi:0,pi:0,si:0,#:5105550160","","","","","ton:0,npi:0,pi:0,si:0,#:5105550160","","","","0163","","","Regul
arLine","","","","","","TWC","05/01/2007 11:17:23.407","5105550160","0163",0,14,9D4B0CA F74711DB 800D96DB
A749148A,8,"","","",""

11780434730,9,1,1,"1B795560 F74711DB 801296DB A749148A","5105550163","","11:17:51.323 pdt Tue May 1
2007","","11:17:53.263 pdt Tue May 1 2007","11:17:53.263 pdt Tue May 1 2007","10  ","normal call clearing
(16)","",0,"",0,0,0,0,"5105550163","5105550163","","","","0 ms",20006,30,28,0,0,"0
ms","","","None","","","","","","","","","","","","","","","","","","","","","","","","","","","","
,"",0,"Tariff:Unknown","","0","","","","","","","","","","","","","","","","","","","","","","","","ton:0,npi
:0,pi:0,si:0,#:5105550163","","","","","","","","","3002","","","RegularLine","","","","","","CXFER","05/01/2
007 11:17:53.251",0,22,9D4B0CA F74711DB 800D96DB A749148A,"1BD61",9,0,2,"5105550163","5105550160","3002"

11780434730,9,1,1,"1B795560 F74711DB 801296DB A749148A","5105550163","","11:17:51.323 pdt Tue May 1
2007","","11:17:53.263 pdt Tue May 1 2007","11:17:53.263 pdt Tue May 1 2007","10  ","normal call clearing
(16)","",0,"",0,0,0,0,"5105550163","5105550163","","","","0 ms",20006,30,28,0,0,"0
ms","","","None","","","","","","","","","","","","","","","","","","","","","","","","","","","","
,"",0,"Tariff:Unknown","","0","","","","","","","","","","","","","","","","","","","","","","","","ton:0,npi
:0,pi:0,si:0,#:5105550163","","","","","","","","","3002","","","RegularLine","","","","","","TWC","05/01/200
7 11:17:51.323","5105550163","",0,15,1B795560 F74711DB 801296DB A749148A,9,"","","",""
```

File Accounting Compact CDR: Example

The following example shows a CDR captured by file accounting using the compact format.

```
11783007890,16,1,1,"36CDEBEC F99E11DB 8025D2A3 19FAB826","6002","","10:46:26.329 pdt Fri May 4
2007","10:46:26.329 pdt Fri May 4 2007","10:46:27.149 pdt Fri May 4 2007","10:46:29.899 pdt Fri May 4
2007","10  ","normal call clearing (16)","",0,"",0,0,0,0,"5105550160","5105550160","6002","TWC","05/04/2007
10:46:26.333","5105550160","6002",0,16,36CDEBEC F99E11DB 8025D2A3 19FAB826,10,"","","",""
11783007890,15,1,1,"36CDEBEC F99E11DB 8025D2A3 19FAB826","5105550160","","10:46:25.709 pdt Fri May 4
2007","","10:46:27.159 pdt Fri May 4 2007","10:46:29.909 pdt Fri May 4 2007","10  ","normal call clearing
(16)","",0,"",0,0,0,0,"5105550160","5105550160","","TWC","05/04/2007
10:46:25.717","5105550160","",0,15,36CDEBEC F99E11DB 8025D2A3 19FAB826,F,"","","",""
```

Hold and Resume CDR: Example

The following example shows CDR stop records captured by file accounting for Hold and Resume. Because file accounting records are in .csv format, fields with no data are included as empty fields.

In this example, extension 3000 calls extension 5000, which is a shared line. Extension 5000 is shared by phone 5 (mbrown) and phone 7 (jsmith). The Hold record shows that Phone 7 answered the call and put the call on hold. Phone 5 then resumed the call as shown in the Resume record.

```
1194293905,6,0,1,"9C7B4D61 8B1311DC 8010A883 AC60BEF4","","","12:18:00.365 pst Mon Nov 5 2007","12:18:00.375
pst Mon Nov 5 2007","12:18:02.015 pst Mon Nov 5 2007","12:18:25.205 pst Mon Nov 5
2007","","","",0,"",0,0,1145,183200,"1.5.10.7","3000","5000","","","","","","","","","","","","","","","","
","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","",
23,"Tariff:Unknown","","0","","","","","","","","","","","","","","","","","","","","","","","","",
"ton:0,npi:0,#:5000","ton:0,npi:0,#:3000","","","","","ton:0,npi:0,#:3000","","","","","","","",
"Unknown","","","","","","HOLD","11/05/2007 12:18:13.179",0,31,11B2208B 8B1311DC 8010F5E9
5DD6F9BF,6,1,"5000","3000","sl:1,usr:jsmith,tag:7","dn:shared,usr:jsmith,tag:7","cme"

1194293905,6,0,1,"9C7B4D61 8B1311DC 8010A883 AC60BEF4","","","12:18:00.365 pst Mon Nov 5 2007","12:18:00.375
pst Mon Nov 5 2007","12:18:02.015 pst Mon Nov 5 2007","12:18:25.205 pst Mon Nov 5
2007","","","",0,"",0,0,1145,183200,"1.5.10.7","3000","5000","","","","","","","","","","","","","","","","
","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","",
","",23,"Tariff:Unknown","","0","","","","","","","","","","","","","","","","","","","","","","","","",
"","ton:0,npi:0,#:5000","ton:0,npi:0,#:3000","","","","","ton:0,npi:0,#:3000","","","","","","","Unknown","",
"","","","","RESUME","11/05/2007 12:18:11.043",0,28,106C3341 8B1311DC 800FF5E9
5DD6F9BF,6,0,"5000","3000","sl:1,usr:mbrown,tag:5","dn:shared,usr:jsmith,tag:7","cme"
```

Feature Information for File Accounting

Table 3-2 lists the release history for this feature.

Not all commands may be available in your Cisco IOS software release. For release information about a specific command, see the command reference documentation.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 3-2 lists only the Cisco IOS software release that introduced support for a given feature in a given Cisco IOS software release train. Unless noted otherwise, subsequent releases of that Cisco IOS software release train also support that feature.

Table 3-2 Feature Information for File Accounting with FTP and Flash Storage

Feature Name	Releases	Feature Information
Call Detail Records Comma Separated Value Format with FTP and Flash Storage	12.4(20)T 12.4(15)XY	Adds file accounting method that stores call records in .csv format. The following commands were introduced or modified by this feature: **acct-template**, **debug voip dump-file-acct**, **debug voip fileacct**, **file-acct flush**, **file-acct reset**, **gw-accounting**, **maximum buffer-size**, **maximum cdr-format**, **maximum cdrflush-timer**, **maximum fileclose-timer**, **maximum retry-count**, **primary**, **secondary**.