Call Duration Monitoring

Cisco Unified Border Element (SP Edition) supports the Call Duration Monitoring feature that is used to gracefully terminate calls whose duration has exceeded a configured maximum amount of time. You can configure the maximum call duration to be applied to a call.

Using this feature, the SBC can terminate SIP, H.323, and SIP to H.323 interworked calls, regardless of the signaling and media activity within those calls.

Cisco Unified Border Element (SP Edition) was formerly known as Integrated Session Border Controller and may be commonly referred to in this document as the session border controller (SBC).

For a complete description of the commands used in this chapter, refer to the *Cisco Unified Border Element (SP Edition) Command Reference: Unified Model* at:

http://www.cisco.com/en/US/docs/ios/sbc/command/reference/sbcu_book.html.

For information about all Cisco IOS commands, use the Command Lookup Tool at http://tools.cisco.com/Support/CLILookup or a Cisco IOS master commands list.

Note This feature is supported in the unified model for Cisco IOS XE Release 2.5 and later.

Feature History for Call Duration Monitoring Feature

Release	Modification
Cisco IOS XE Release 2.5	The Call Duration Monitoring feature was introduced on the Cisco IOS XR.

Contents

This chapter contains the following sections:

Prerequisites

The following prerequisite is required to implement this feature:

Before implementing these features, Cisco Unified Border Element (SP Edition) must already be configured.

Information About Call Duration Monitoring

Cisco Unified Border Element (SP Edition) supports the Call Duration Monitoring feature that is used to gracefully terminate calls whose duration has exceeded a maximum amount of time configured with this feature.

If the duration of a call exceeds the configured maximum set by the **max-call-duration** command, Cisco Unified Border Element (SP Edition) gracefully tears down the call.

- For SIP call branches, the SBC sends a SIP BYE to the endpoints.
- For H.323 call branches, the SBC sends a RELEASE COMPLETE message to the endpoints.

If there is a renegotiation in progress when the maximum duration is reached, the SBC attempts to terminate the call as gracefully as possible.

The SBC can terminate SIP, H.323, and SIP to H.323 interworked calls, regardless of the signaling and media activity within those calls.

The SBC will terminate calls under the following conditions:

- Duration has exceeded a maximum amount of time configured by the **max-call-duration** command for this feature.

  Note If the **max-call-duration** command is set to the default of zero (0), this results in disabling the Call Duration Monitoring function; and call duration can be determined by other factors, such as no media flow or calls not answered within a specified period of time.

- If no media has flowed on that call for a specified period of time, as configured by the **media-timeout** command.
- Calls that are not answered within a specified amount of time.

Calls are terminated according to whichever timer expires first.

After this feature is configured, the SBC starts a timer to individually monitor each call passing through it. The timer is started:

- For SIP calls, when the call is connected (*not* on receipt of the message establishing a new call).
- For H.323 calls, when the first SETUP message establishing a new call is received.

Once the timer has been started it cannot be reset.

However, if SBC re-routs a call during call setup, the maximum call duration timer is restarted because the configured maximum duration may have changed (based on the new routing information).

Configuring Call Duration Monitoring

This task configures Call Duration Monitoring.

SUMMARY STEPS

1. **configure terminal**
2. **sbc** *sbc-name*
3. **sbe**
4. **cac-policy-set** *policy-set-id*
5. **cac-table** *table-name*
6. **table-type** {**policy-set** | **limit** {*list of limit tables*}}
7. **entry** *entry-id*
8. **cac-scope** {*list of scope options*}
9. **max-call-duration** {*num*}
10. **action** [**next-table** *goto-table-name* | **cac-complete**]
11. **exit**
12. **exit**
13. **complete**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enables global configuration mode.
Step 2	**sbc** *service-name* **Example:** `Router(config)# sbc mysbc`	Enters the mode of an SBC service. Use the *service-name* argument to define the name of the service.
Step 3	**sbe** **Example:** `Router(config-sbc)# sbe`	Enters the mode of an SBE entity within an SBC service.
Step 4	**cac-policy-set** *policy-set-id* **Example:** `Router(config-sbc-sbe)# cac-policy-set 1`	Enters the mode of CAC policy set configuration within an SBE entity, creating a new policy set if necessary.

	Command or Action	Purpose
Step 5	**cac-table** *table-name* **Example:** `Router(config-sbc-sbe-cacpolicy)# cac-table StandardListByAccount`	Enters the mode for configuration of an admission control table (creating one if necessary) within the context of an SBE policy set.

Command or Action	Purpose
Step 6 **table-type** {**policy-set** | **limit** {*list of limit tables*}} **Example:** `Router(config-sbc-sbe-cacpolicy-cactable)# table-type policy-set`	Configures the table type of a CAC table within the context of an SBE policy set. The *list of limit tables* argument controls the syntax of the match-value fields of the entries in the table. Possible available Limit tables are: • *account*—Compare the name of the account. • *adj-group*—Compare the name of the adjacency group. • *adjacency*—Compare the name of the adjacency. • *all*—No comparison type. All events match this type. • *call-priority*—Compare with call priority. • *category*—Compare the number analysis assigned category. • *dst-account*—Compare the name of the destination account. • *dst-adj-group*—Compare the name of the destination adjacency group. • *dst-adjacency*—Compare the name of the destination adjacency. • *dst-prefix*—Compare the beginning of the dialed digit string. • *event-type*—Compare with CAC policy event types. • *src-account*—Compare the name of the source account. • *src-adj-group*—Compare the name of the source adjacency group. • *src-adjacency*—Compare the name of the source adjacency. • *src-prefix*—Compare the beginning of the calling number string. **Note** For Limit tables, the event or message or call matches only a single entry. Features can be enabled or disabled per adjacency group through CAC configuration the same way this is done per individual adjacencies. The adj-group table type matches on either source or destination adjacency group. When the policy-set keyword is specified, use the **cac-scope** command to configure the scope within each entry at which limits are applied in a CAC Policy Set table. **Note** For Policy Set tables, the event or call or message is applied to all entries in this table.

	Command or Action	Purpose
Step 7	**entry** *entry-id* **Example:** `Router(config-sbc-sbe-cacpolicy-cactable)# entry 1`	Enters the CAC table entry configuration mode to create or modify an entry in an admission control table.
Step 8	**cac-scope** {*list of scope options*} **Example:** `Router(config-sbc-sbe-cacpolicy-cactable-entry) # cac-scope call`	Configures the scope within each of the entries at which limits are applied in a policy set table. • *list of scope options*—Specifies one of the following strings used to match events: – **account**—Events that are from the same account. – **adjacency**—Events that are from the same adjacency. – **adj-group**—Events that are from members of the same adjacency group. – **call**—Scope limits are per single call. – **category**—Events that have same category. – **dst-account**—Events that are sent to the same account. – **dst-adj-group**—Events that are sent to the same adjacency group. – **dst-adjacency**—Events that are sent to the same adjacency. – **dst-number**—Events that have same destination. – **global**—Scope limits are global – **src-account**—Events that are from the same account. – **src-adj-group**—Events that are from the same adjacency group. – **src-adjacency**—Events that are from the same adjacency. – **src-number**—Events that have the same source number. – **sub-category**—The limits specified in this scope apply to all events sent to or received from members of the same subscriber category. – **sub-category-pfx**—The limits specified in this scope apply to all events sent to or received from members of the same subscriber category prefix. – **subscriber**—The limits specified in this scope apply to all events sent to or received from individual subscribers (a device that is registered with a Registrar server)

	Command or Action	Purpose
Step 9	**max-call-duration** {*num*} **Example:** `Router(config-sbc-sbe-cacpolicy-cactable-entry)# max-call-duration 6000`	Configures the maximum duration (in seconds) for which a call may exist. *num* range is from 0 to 2147483 seconds By default, the max-call-duration is 0, which results in disabling the Call Duration Monitoring feature.
Step 10	**action** [**next-table** *goto-table-name* | **cac-complete**] **Example:** `Router(config-sbc-sbe-cacpolicy-cactable-entry)# action cac-complete`	Configures the action to perform after this entry in an admission control table. Possible actions are: • Identify the next CAC table to process using the **next-table** keyword and the *goto-table-name* argument. • Stop processing for this scope using the **cac-complete** keyword.
Step 11	**exit** **Example:** `Router(config-sbc-sbe-cacpolicy-cactable-entry)# exit`	Exits from **entry** to **cactable** mode.
Step 12	**exit** **Example:** `Router(config-sbc-sbe-cacpolicy-cactable)# exit`	Exits from **cactable** to **cacpolicy** mode.
Step 13	**complete** **Example:** `Router(config-sbc-sbe-cacpolicy-cactable-entry)# complete`	Completes the CAC policy set when you have committed the full set.

Configuration Example—Call Duration Monitoring

The following is a configuration example for the Call Duration Monitoring feature:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# sbc test
Router(config-sbc)# sbe
Router(config-sbc-sbe)# cac-policy-set 1
Router(config-sbc-sbe-cacpolicy)# cac-table 1
Router(config-sbc-sbe-cacpolicy-cactable)# table-type policy-set
Router(config-sbc-sbe-cacpolicy-cactable)# entry 1
Router(config-sbc-sbe-cacpolicy-cactable-entry)# max-call-duration 6000
Router(config-sbc-sbe-cacpolicy-cactable-entry)#
```