



Codec Handling

A compressor-decompressor (codec) is a device or program that performs a transformation on a data stream or signal. Cisco Unified Border Element (SP Edition) is hard-coded with a set of recognized codecs (see [Table 17-1](#) to [Table 17-5](#)), including all commonly used voice and video codecs. The default behavior is to allow all recognized codecs on all calls. Any other codec present in call signaling is removed by Cisco Unified Border Element (SP Edition).

This enhancement allows you to restrict which codec(s) a particular call can use and to configure a minimum permissible packetization period for each permitted codec.

Cisco Unified Border Element (SP Edition) supports passthrough codecs that are passed through the SBC without transcoding. See [Table 17-4 on page 17-7](#) for a list of passthrough codecs by type.

H.323 only supports PCMA, PCMU, G.722, G.723, G.728, G.729, GSM, telephone-event, H.261, H.263, H.264, and T.38 codecs. Therefore in a SIP to H.323 call, if SIP codecs are sent to H.323 that H.323 cannot support, then these codecs are not passed through and the call may fail accordingly.

Cisco Unified Border Element (SP Edition) supports codec transcoding for SIP to SIP calls using an external DSP resource or transcoding resource. Transcoding is the process of translating a media stream encoded using one codec into a media stream encoded using another codec, for example, translating a media stream encoded as PCMU into one encoded as G.726-32. For more information on transcoding, see the [Implementing Transcoding](#) chapter.

Cisco Unified Border Element (SP Edition) enables H.323 TCS Codecs support in Cisco IOS XE Release 2.5.1. This support provides the ability to announce media capabilities on behalf of a SIP endpoint to an H.323 endpoint by adding extra offered codecs in the H.245 Terminal Capability Set (TCS) message. See the [?\\$paranum>H.323 TCS Codecs Support? section on page 17-16](#).



Note

For Cisco IOS XE Release 2.4 and later, this feature is supported in the unified model.

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.

Feature History for Codec Handling

Release	Modification
Cisco IOS XE Release 2.4	This feature was introduced on the Cisco IOS XR along with support for the unified model.
Cisco IOS XE Release 2.4.2	Support for packetization time, p-time, attribute in the SDP offer or answer configured in a codec list was added.
Cisco IOS XE Release 2.5	Support for H.323 calls, and H.323 video codecs H.261, H.263, and H.264 were added. Support for passthrough codecs without transcoding was added.
Cisco IOS XE Release 2.5.1	H.323 TCS Codecs support was added.
Cisco IOS XE Release 2.6	Support for Dynamic CODEC configuration, multiple audio codec, and multiple video codec were added. H.323 support for clear channel data and modem calls was added.
Cisco IOS XE Release 3.2S	Codec preference and reordering support was added.

Contents

This module contains the following sections:

- [Prerequisites for Codec Handling, page 17-3](#)
- [Restrictions for Codec Handling, page 17-3](#)
- [Restriction for H.323 TCS Codecs Support, page 17-3](#)
- [Restrictions for Dynamic Codec Configuration, page 17-4](#)
- [Restrictions for Codec Reordering, page 17-4](#)
- [Codec Handling and Restriction, page 17-4](#)
- [Dynamic Codec Configuration, page 17-8](#)
- [Configuring Codec Restriction, page 17-9](#)
- [Packetization Time, page 17-15](#)
- [H.323 TCS Codecs Support, page 17-16](#)
- [Codecs Preference and Reordering Support, page 17-17](#)
- [Configuration Examples—Configuring Codec Restriction, page 17-22](#)
- [Configuration Examples—H.323 TCS Codecs Support, page 17-24](#)
- [Configuration Example—Defining a Codec using Dynamic Codec Configuration, page 17-25](#)

Prerequisites for Codec Handling

The following prerequisite is required before you can restrict codecs:

- Before implementing Codec Handling, Cisco Unified Border Element (SP Edition) must already be configured.
- All signaling border element (SBE) and data border element (DBE) configurations required to make simple calls must already be configured.

Restrictions for Codec Handling

Review the following restrictions for codecs:

- For H.323 calls, SIP to H.323 and H.323 to SIP calls, both the callee and caller must use the same codec because any calls requiring transcoding will fail the call setup.
- The media packet forwarder on the DBE polices the bandwidth consumed by each media stream, but it cannot police the type of codecs or the packetization periods.
- Unrecognized codecs cannot be configured as members of the codec whitelist.
- Active calls are not released if there is a change in the codec whitelist during the call.
- If a codec whitelist is configured, Cisco Unified Border Element (SP Edition) removes any unlisted codecs from the call setup flow and media gate allocation.
- Multiple codec whitelists can be configured on a Call Admission Control (CAC) policy basis. For example, the list of codecs allowed for calls from SipAdj1 can be different than the list of codecs allowed for calls from SipAdj2.
- If a codec whitelist has not been configured, all recognized codecs (see [Table 17-1](#) to [Table 17-5](#)) are allowed for all calls.
- You must use the textual value of the codec description that appears on the Session Description Protocol (SDP) to configure the codec whitelist, for example “PCMU” or “telephone-event”.
- Disallowing all codecs is not supported. However, you could set a bandwidth limit of 0 to achieve the same result.
- Codec lists are not applied to media-bypass calls (those in which Cisco Unified Border Element (SP Edition) does not reserve media resources).
- The format of the codec name is the same as the string used to represent it in SDP, for example PCMU or VDVI. All recognized codec names are listed in [Table 17-1](#) – [Table 17-5](#).
- A single codec can only be added to each list once, with a single packetization period.
- For each codec on a list, CAC restricts the signaled packetization period for any stream using that codec to be greater than or equal to the packetization period configured along with the codec in the list. If a stream uses more than one codec in the list, then the greater of all the packetization periods configured for each codec in the list is applied to the stream.

Restriction for H.323 TCS Codecs Support

H.323 TCS Codecs support only applies to H.323-to-SIP or SIP-to-H.323 interworking calls with the caller or callee using SIP.

Restrictions for Dynamic Codec Configuration

The following restrictions apply for Dynamic codec configuration feature:

- Codecs provided with the SBC (system codec) may be modified but not deleted.
- Codec names are case insensitive.
- A maximum of 100 user defined codecs is supported.
- Custom codec payload ID should always be 96.
- When defining new codec, codec type must be specified before all other fields.
- A custom codec is not defined until a codec type is specified.
- The ID of a system codec should not be changed.
- The ID should be unique for a Codec.
- A custom codec should not be deleted if used in a codec list.
- Sample size is only valid for sampling codecs.
- Bandwidth cannot be specified for redundancy or sampling codecs.
- Channels may only be specified for sampling based codecs.

Restrictions for Codec Reordering

- SBC supports codec reordering only for SIP-SDP codecs. Reordering of codecs in H.323 is not supported.
- If media bypass is configured, codec reordering cannot be applied.

Codec Handling and Restriction



Note

The bandwidths listed in the tables below are the bandwidths without the transport layer overheads. Therefore, the actual bandwidths reserved by Cisco Unified Border Element (SP Edition) are higher than the listed values.

[Table 17-1](#) lists sample-based audio codecs.

Table 17-1 Sample-Based Audio Codecs with Packetization Time 10 ms

Payload Type	Codec Name	Clock Rate (Hz)	Sample Size (bits)	Channels
0	PCMU (also known as G.711)	8000	8	1
5	DVI4	8000	4	1
6	DVI4	16000	4	1
8	PCMA	8000	8	1
10	L16	44100	16	2
11	L16	44100	16	1

Table 17-1 Sample-Based Audio Codecs with Packetization Time 10 ms (continued)

Payload Type	Codec Name	Clock Rate (Hz)	Sample Size (bits)	Channels
15	G728	8000	2	1
16	DVI4	11025	4	1
17	DVI4	22050	4	1
—	G726-40	8000	5	1
2	G726-32	8000	4	1
—	G726-24	8000	3	1
—	G726-16	8000	2	1
—	L8	8000	8	1
—	DAT12	8000	12	2
—	L20	44100	10	2
—	L24	44100	24	2

Table 17-2 lists non-sample-based audio codecs.

Table 17-2 Non-Sample-Based Audio Codecs

Payload Type	Codec Name	Packetization Time (ms)	Allocated Bandwidth (bps)
3	GSM	10	13200
4	G723	30	6400
7	LPC	10	5600
9	G722	10	64000
12	QCELP	—	13300
13	CN	10	400
14	MPA	N/A	131072
18	G729	10	8000
18	G.729A	10	8000
—	G729B	20	8000
—	G729AB	10	8000
—	G729D	10	6400
—	G729E	10	11800
—	GSM-EFR	10	12400
—	iSAC	30	32000
—	VDVI	10	25000
—	AMR	10	12500
—	AMR-WB	10	24420
—	dsr-es201108	10	4800
—	EVRC	10	8550

Table 17-2 Non-Sample-Based Audio Codecs (continued)

Payload Type	Codec Name	Packetization Time (ms)	Allocated Bandwidth (bps)
—	EVRC0	10	8550
—	mpa-robust	10	327680
—	G7221	10	32000
—	MP4A-LATM	10	131072
—	SMV	10	8550
—	SMV0	10	8550

Table 17-3 lists video codecs.

Table 17-3 Video Codecs

Payload Type	Codec Name	Packetization Time (ms)	Allocated Bandwidth (bps)
25	CelB	—	524228
26	JPEG	—	524228
28	nv	—	524228
31	H261	—	524228
32	MPV	—	524228
33	MP2T	—	524228
34	H263	—	524228
NA	H264	—	21000000
—	BMPEG	—	524228
—	BT656	—	170000000
—	DV	—	1500000000
—	H263-1998	—	524228
—	H263-2000	—	524228
—	MP1S	—	1600000
—	MP2P	—	524228
—	MP4V-ES	—	524228
—	raw	N/A	1500000000
—	SMPTE292M	N/A	1500000000

Table 17-4 lists the supported passthrough codecs without transcoding.

Table 17-4 Passthrough Codecs without Transcoding

Codec Name	Codec Type	Packetization Time (ms)	Allocated Bandwidth (bps)
PCMA (also known as G.711)	sample-based audio	10	64000
PCMU (also known as G.711)	sample-based audio	10	64000
G726-16	audio	10	NA
G726-24	audio	10	NA
G726-32	audio	10	NA
G728	audio	10	NA
G729 A	non-sample-based audio	10	8000
G729 B	non-sample-based audio	10	8000
G723-53	non-sample-based audio	30	6400
G723-63	non-sample-based audio	30	6400
GSM/GSM-FR	non-sample-based audio	10	13200
GSM-EFR	non-sample-based audio	20	12400
GSM-HR	non-sample-based audio	20	5600
AMR	non-sample-based audio	10	12500
EVRC	non-sample-based audio	10	8550
G722	non-sample-based audio	10	64000
iLBC	non-sample-based audio	20	15200
H.261	video	NA	524228
H.263	video	NA	524228

Table 17-5 lists other audio codecs.

Table 17-5 Other Codecs

Codec Name	Packetization Time (ms)	Allocated Bandwidth (bps)
telephone-event	20	1600
tone	20	1600
RED	20	1
parityfec	20	1
T140	100	80
pointer	20	1600
H224	20	6560
T38	N/A	15500
X-NSE	20	1600

Dynamic Codec Configuration

The Dynamic Codec configuration feature allows you to:

- **Define new codecs**—You can create variants of codecs that are included in the SBC. For example, G.729 A is a codec that is included in SBC. You may define a variant called G.729.1 using Dynamic Codec feature.
- **Modify existing codecs**—You can modify certain attributes of the codecs included in the SBC. For example, you can change the bandwidth characteristics of H.264 video codecs, included in the SBC.
- **Display codecs supported on SBC**—You can view all codecs supported on an SBC.

**Note**

Aliasing and Codec conversion are not supported by Dynamic Codec Configuration feature in this release.

Audio and Video codecs that are not included in SBC can be defined using the Dynamic Codec Configuration feature.

Multiple Audio Codec Support

SBC transparently passes the following codecs:

- G.711 with silence suppression and RFC 2833 (and RFC 4733)
- G.723
- G.726
- G.726 with silence suppression (Silence suppression is used to save bandwidth by not sending the data when there is no voice during a call.)
- G.729
- G.729a
- G.729a/b
- AMR
- AAC-LD
- AMR-WB
- AMR-WB+
- G.718

Audio codecs not included in SBC can be defined using the Dynamic Codec Configuration feature.

Multiple Video Codec Support

SBC transparently passes media encoded using:

- H.263
- H.264
- H.264 AVC

- H.264 SVC
- Microsoft RT Video
- Helix Real Video

Video codecs not included in SBC can be defined using the Dynamic Codec Configuration feature.

H.323 Support for Clear Channel Data/Modem Calls

The clearmode codec is supported in existing VoIP equipments, and is used in the H.323, SIP, and MGCP signaling protocols. The Cisco SBC supports the clearmode codec in all places where it supports common audio codecs, such as G.711.

Configuring Codec Restriction

You first configure the codecs and then apply them as explained in the following sections:

- [Configuring Codecs, page 17-9](#)
- [Configuring a CAC Policy to Use a Codec List, page 17-11](#)

Configuring Codecs

To restrict which codec(s) a particular call can use and to configure a minimum permissible packetization period for each permitted codec, you must configure CAC with a list of codecs, provide a description for the list, and then add any codec(s) to the list.

SUMMARY STEPS

1. **configure terminal**
2. **sbc *service-name***
3. **sbe**
4. **codec list *name***
5. **description *text***
6. **codec *codec-name* [*packetization-period packetization-period*]**
7. **end**
8. **show sbc *service-name* sbe codec-list *list-name***

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 submode of an SBC service.
Step 3	sbe Example: Router(config-sbc)# sbe	Configures the submode of the SBE entity within a SBC service.
Step 4	codec list name Example: Router(config-sbc-sbe)# sbc mysbc sbe codec list my_codecs	Creates a codec list.
Step 5	description text Example: Router(config-sbc-sbe-codec-list)# description Legitimate codes	Adds a description for the specified codec list using a readable text string format. The no form of this command removes the description. This description is displayed when the show command is used for this codec list. It is also included for each codec list when a summary of all codec lists is displayed.
Step 6	codec codec-name [packetization-period packetization-period] Example: Router(config-sbc-sbe-codec-list)# codec PCMU packetization-period 20	Adds a codec to a codec list, and sets a minimum packetization period (optional) for the codec. The no form of this command (without the packetization period) removes the named codec from the codec list. Note If the no form of this command includes the packetization period, only the packetization period for the codec is removed.
Step 7	end Example: Router(config-sbc-sbe-codec-list)# end	Exits codec-list mode and enters Privileged EXEC mode.
Step 8	show sbc service-name sbe codec-list list-name Example: Router# show sbc mysbc sbe codec-list my_codecs	Displays detailed information about the codec lists configured on the SBE. If the list name is omitted, for example, my_codecs, then details are displayed for all codec lists on the SBE.

Configuring a CAC Policy to Use a Codec List

SUMMARY STEPS

1. **configure terminal**
2. **sbc** *service-name*
3. **sbe**
4. **cac-policy-set** *policy-set-id*
5. **first-cac-scope** *scope-name*
6. **first-cac-table** *table-name*
7. **cac-table** *table-name*
8. **table-type** {**policy-set** | **limit** {*list of limit tables*}}
9. **entry** *entry-id*
10. **cac-scope** {*list of scope options*}
11. **action** [next-table *goto-table-name* | cac-complete]
12. **codec-restrict-to-list** *list-name*
13. **complete**
14. **end**
15. **show sbc** *service-name* **sbe** **cac-policy-set** *id* **table** *name* **entry** *entry*

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Router# configure terminal	Enables global configuration mode.
Step 2	sbc <i>service-name</i> Example: Router(config)# sbc mysbc	Enters the submode of an SBC service.
Step 3	sbe Example: Router(config-sbc)# sbe	Configures the submode of the SBE entity within a SBC service.
Step 4	cac-policy-set <i>policy-set-id</i> Example: Router(config-sbc-sbe)# cac-policy-set 1	Enters the submode of CAC policy.

	Command or Action	Purpose
Step 5	first-cac-scope <i>scope-name</i> Example: Router(config-sbc-sbe-cacpolicy)# first-cac-scope global	Configures the scope at which to begin defining limits when performing the admission control stage of policy. Note The first-cac-scope definition is only relevant if the table type configured by the first-cac-table command is a Limit table. In that case, the scope of the first-cac-table is determined by first-cac-scope. If the first-cac-table is a Policy Set table, the first-cac-scope is ignored and defaults to global. The <i>scope-name</i> argument configures the scope at which limits should be initially defined. Possible values are: <ul style="list-style-type: none"> • adj-group • call • category • dst-account • dst-adj-group • dst-adjacency • dst-number • global • src-account • src-adj-group • src-adjacency • src-number Features can be enabled or disabled per adjacency group through CAC configuration the same way this is done per individual adjacencies.
Step 6	first-cac-table <i>table-name</i> Example: Router(config-sbc-sbe-cacpolicy)# first-cac-table first_policy_table	Configures the name of the first policy table to process when performing the admission control stage of policy.
Step 7	cac-table <i>table-name</i> Example: Router(config-sbc-sbe-cacpolicy)# cac-table first_policy_table	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 8	<p>table-type {<i>policy-set</i> <i>limit</i> {<i>list of limit tables</i>}}</p> <p>Example:</p> <pre>Router(config-sbc-sbe-cacpolicy-cactable)# table-type policy-set</pre>	<p>Configures the table type of a CAC table within the context of an SBE policy set.</p> <p>The <i>list of limit tables</i> argument controls the syntax of the match-value fields of the entries in the table. Possible available Limit tables are:</p> <ul style="list-style-type: none"> • <i>account</i>—Compare the name of the account. • <i>adj-group</i>—Compare the name of the adjacency group. • <i>adjacency</i>—Compare the name of the adjacency. • <i>all</i>—No comparison type. All events match this type. • <i>call-priority</i>—Compare with call priority. • <i>category</i>—Compare the number analysis assigned category. • <i>dst-account</i>—Compare the name of the destination account. • <i>dst-adj-group</i>—Compare the name of the destination adjacency group. • <i>dst-adjacency</i>—Compare the name of the destination adjacency. • <i>dst-prefix</i>—Compare the beginning of the dialed digit string. • <i>event-type</i>—Compare with CAC policy event types. • <i>src-account</i>—Compare the name of the source account. • <i>src-adj-group</i>—Compare the name of the source adjacency group. • <i>src-adjacency</i>—Compare the name of the source adjacency. • <i>src-prefix</i>—Compare the beginning of the calling number string. <p>Note For Limit tables, the event or message or call matches only a single entry.</p> <p>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.</p> <p>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.</p> <p>Note For Policy Set tables, the event or call or message is applied to all entries in this table.</p>

	Command or Action	Purpose
Step 9	entry <i>entry-id</i> Example: Router(config-sbc-sbe-cacpolicy-cactable)# entry 1	Enters the mode to create or modify an entry in an admission control table.
Step 10	cac-scope { <i>list of scope options</i> } 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. <ul style="list-style-type: none"> • <i>list of scope options</i>—Specifies one of the following strings used to match events: <ul style="list-style-type: none"> – 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 11	action [next-table <i>goto-table-name</i> 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: <ul style="list-style-type: none"> Identify the next CAC table to process using the next-table keyword and the <i>goto-table-name</i> argument. Stop processing for this scope using the cac-complete keyword.
Step 12	codec-restrict-to-list <i>list-name</i> Example: Router(config-sbc-sbe-cacpolicy-cactable-entry) # codec-restrict-to-list my_codecs	Configures CAC to restrict the codecs used in signaling a call to the set of codecs provided in the named list. If a codec list is empty, all codecs recognized by the SBE are allowed. The no form of this command, or not setting this command, allows any recognized codecs to be used without restrictions. Note This command replaces any codec list that was set up by an earlier CAC entry. To clear all restrictions from an earlier CAC entry, you must configure a codec-restrict-to-list <i>list-name</i> , where <i>list-name</i> is the name of a list containing no codecs.
Step 13	complete Example: Router(config-sbc-sbe-cacpolicy)# complete	Completes the cac-policy.
Step 14	end Example: Router(config-sbc-sbe-cacpolicy)# end	Exits the cac-policy-set configuration mode and enters Privileged EXEC mode.
Step 15	show sbc service-name sbe cac-policy-set id table name entry entry Example: Router# show sbc mysbc sbe cac-policy-set 1 table standard_policy_list entry 1	Displays detailed information for a specific entry in a CAC policy table, including any restricted codecs.

Packetization Time

The packetization time (p-time) is the time period a codec is applied to a media stream to build a single digital packet. When a call travels between two end points, both the codec and the p-time for that codec is negotiated. The SBC acts on the p-time attribute in the SDP based on the configuration in the codec list.

SBC adds the ptime attribute to an SDP offer or answer, when one is configured explicitly in a codec whitelist. The exception is if a minimum packetization time is configured as part of SBC policy. This configuration causes SBC to insert a=ptime lines into forwarded offers or answers.

SBC ensures that media clipping does not occur as a result of overestimating the ptime (and hence underestimating the bandwidth requirement).

Ifptime is signaled explicitly on the offer and answer, it uses the lower of the two values to calculate the bandwidth allowance for both directions of media.

H.323 TCS Codecs Support

Cisco Unified Border Element (SP Edition) enables H.323 TCS Codecs support in Cisco IOS XE Release 2.5.1. This support provides the ability to announce media capabilities on behalf of a SIP endpoint to an H.323 endpoint by adding extra offered codecs in the H.245 Terminal Capability Set (TCS) message. You configure the additional codecs in the TCS message with Call Admission Control (CAC) policy commands.

H.323 TCS Codecs support applies to SIP to H.323 or H.323 to SIP interworking calls when the caller or callee uses SIP.

Information About H.323 TCS Codecs Support

H.323 TCS Codecs support adds extra capability in the H.245 Terminal Capability Set (TCS) message by providing the ability to configure offered codecs in the TCS message, on behalf of a SIP endpoint for SIP to H.323 or H.323 to SIP interworking calls when the caller or callee uses SIP.

A TCS message is an H.245 message transmitted by an H.323 endpoint sent by the SBC. The TCS message transmitted by the H.323 endpoint is used to indicate the H.323 endpoint's media capabilities, as well as the version of H.245, to the other party.

However, a SIP endpoint's SDP offer and answer may not announce all its capabilities because the SDP offer may be missing T.38 and other codecs, and the SDP answer may be missing many codecs. For example, a call originated by SIP endpoint A may not indicate the support of T.38, which would cause the H.323 endpoint B to be unaware of the capability. Therefore, endpoint B would not be able to send a RequestMode message to the SBC asking endpoint A to switch to the T38 codec.

The H.323 TCS Codecs support feature overcomes this gap by allowing you to add offered codecs in the TCS message that will announce the SIP caller or callee media capabilities to the H.323 endpoint. You do this with the **caller-media-caps** or **callee-media-caps** command in CAC table entry configuration mode. These two commands configure the SIP caller or callee media capability by assigning a codec list that is used to announce media capabilities on behalf of either a SIP caller or callee in a SIP to H.323 or H.323 to SIP interworking call.

**Note**

In some scenarios, the **branch** command can be used as an alternative to the **caller** and **callee** commands. The **branch** command has been introduced in Release 3.5.0. See the [“Configuring Directed Nonlimiting CAC Policies” section on page 37](#) for information about this command.

You can also use the similar **tcs-extra-codecs** command in a CAC table entry to configure additional codecs in a codec list that sends an extra TCS message to the H.323 side. This command announces extra codecs capability to the H.323 endpoint on behalf of the SIP side, whether it is the SIP caller or SIP callee in a SIP to H.323 or H.323 to SIP interworking call.

You can use any one of the three commands independently.

**Note**

After a codec list has been assigned, it may not be deleted until it is removed from the CAC entry. A codec list must exist before it can be assigned to an entry in a CAC table.

Codecs Preference and Reordering Support

SBC supports reordering of the codecs in a list, and assigning a priority to each codec. You can also apply a codec preference to a list in the CAC policy entry.

For information about restrictions for the Codecs Preference and reordering feature, see the [Restrictions for Codec Reordering? section on page 17-4](#).

When a SIP endpoint makes an SDP call, a list of codecs is provided for each media stream in an endpoint. The codecs are listed by payload types in the m= attribute in the order of highest to lowest preference. The SIP endpoint chooses the highest-priority codec that is acceptable to the SIP.

For example, a call uses G.711 and G.723 codec, but uses the G.711 where possible. To set the preference to G.711, the SIP places the PCMU and PCMA before G.723 in its m-line:

```
m=audio 1234 RTP/AVP 8 0 4.
```

Configuring Reordering

To reorder the codecs in a list, and to prioritize a codec within the list, configure the list of codecs, provide a description for the list, and then add the priority to the codecs in the list.

SUMMARY STEPS

1. **configure terminal**
2. **sbc service-name**
3. **sbe**
4. **codec list name**
5. **description text**
6. **codec codec-name [packetization-period packet-period [priority priority-value] | priority priority-value [packetization-period packet-period]]**
7. **end**
8. **show sbc service-name sbe codec-list list-name**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Router# configure terminal	Enables the global configuration mode.
Step 2	sbc service-name Example: Router(config)# sbc mysbc	Enters the submode of an SBC service.

	Command or Action	Purpose
Step 3	sbe Example: Router(config-sbc)# sbe	Configures the submode of the SBE entity within an SBC service.
Step 4	codec list name Example: Router(config-sbc-sbe)# codec list my_codecs	Creates a codec list.
Step 5	description text Example: Router(config-sbc-sbe-codec-list)# description Legitimate codes	Adds a description for the specified codec list using a readable text string format. The no form of this command removes the description. This description is displayed when the show command is used for this codec list. It is also included for each codec list when a summary of all codec lists is displayed.
Step 6	codec codec-name [packetization-period <i>packet-period</i> [priority <i>priority-value</i>] priority <i>priority-value</i> [packetization-period <i>packet-period</i>]] Example: Router(config-sbc-sbe-codec-list)#codec G723 priority 1	Adds a codec to a codec list, sets a minimum packetization period (optional) for the codec, and sets a priority for the codec that is used for reordering. The default codec preference priority is 100 A smaller priority value indicates a higher priority, for example if the codec preference value is 1, which is the highest priority and places the codec at the top of the list. Using the no form of this command (without the packetization period) removes the named codec from the codec list. Note If the no form of this command includes the packetization period, only the packetization period for the codec is removed.
Step 7	end Example: Router(config-sbc-sbe-codec-list)# end	Exits the codec-list mode and enters the Privileged EXEC mode.
Step 8	show sbc service-name sbe codec-list list-name Example: Router# show sbc mysbc sbe codec-list my_codecs	Displays detailed information about the codec lists configured on the SBE. If the list name for example, my_codecs, is omitted, details are displayed for all the codec lists on the SBE.

Configuring Codec Preferences

To add codec preferences within a list, configure the CAC with a list of codecs, provide a description for the list, and then add the preference to the codecs.

SUMMARY STEPS

1. **configure terminal**
2. **sbc** *service-name*
3. **sbe**
4. **cac-policy-set** *policy-set-id*
5. **first-cac-scope** *scope-name*
6. **cac-table** *table-name*
7. **table-type** {**policy-set** | **limit** {*list of limit tables*}}
8. **entry** *entry-id*
9. **codec-preference-list** *list-name*
10. **end**
11. **show sbc** *service-name* **sbe** **cac-policy-set** *id* **table** *name* **entry** *entry*

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Router# configure terminal	Enables the global configuration mode.
Step 2	sbc <i>service-name</i> Example: Router(config)# sbc mysbc	Enters the submode of an SBC service.
Step 3	sbe Example: Router(config-sbc)# sbe	Configures the submode of the SBE entity within a SBC service.
Step 4	cac-policy-set <i>policy-set-id</i> Example: Router(config-sbc-sbe)# cac-policy-set 1	Enters the submode of CAC policy.
Step 5	first-cac-table <i>table-name</i> Example: Router(config-sbc-sbe-cacpolicy)# first-cac-table first_policy_table	Configures the name of the first policy table to processed when performing the admission control stage of the policy.

Command or Action	Purpose
Step 6	Enters the cactable mode for configuring an admission control table (creating one, if necessary) within the context of an SBE policy set.
Example: Router(config-sbc-sbe-cacpolicy)# cac-table first_policy_table	

	Command or Action	Purpose
Step 7	<p>table-type {policy-set limit {<i>list of limit tables</i>}}</p> <p>Example:</p> <pre>Router(config-sbc-sbe-cacpolicy-cactable)# table-type policy-set</pre>	<p>Configures the table type of a CAC table within the context of an SBE policy set.</p> <p>The <i>list of limit tables</i> argument controls the syntax of the match-value fields of the entries in the table. Possible available Limit tables are:</p> <ul style="list-style-type: none"> • <i>account</i>—Compare the name of the account. • <i>adj-group</i>—Compare the name of the adjacency group. • <i>adjacency</i>—Compare the name of the adjacency. • <i>all</i>—No comparison type. All events match this type. • <i>call-priority</i>—Compare with call priority. • <i>category</i>—Compare the number analysis assigned category. • <i>dst-account</i>—Compare the name of the destination account. • <i>dst-adj-group</i>—Compare the name of the destination adjacency group. • <i>dst-adjacency</i>—Compare the name of the destination adjacency. • <i>dst-prefix</i>—Compare the beginning of the dialed digit string. • <i>event-type</i>—Compare with CAC policy event types. • <i>src-account</i>—Compare the name of the source account. • <i>src-adj-group</i>—Compare the name of the source adjacency group. • <i>src-adjacency</i>—Compare the name of the source adjacency. • <i>src-prefix</i>—Compare the beginning of the calling number string. <p>Note For Limit tables, the event or message or call matches only a single entry.</p> <p>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.</p> <p>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.</p> <p>Note For Policy Set tables, the event or call or message is applied to all entries in this table.</p>

	Command or Action	Purpose
Step 8	entry <i>entry-id</i> Example: Router(config-sbc-sbe-cacpolicy-cactable)# entry 1	Enters the entry mode to create or modify an entry in an admission control table.
Step 9	codec-preference-list <i>list-name</i> Example: Router(config-sbc-sbe-cacpolicy-cactable-entry) # codec-preference-list my_codecs	Configures the CAC to set preferences for the codecs in the codecs list. Using the no form of this command, or not setting this command, allows the recognized codecs, if any, to be used without any usage preference. Note This command replaces the codec-preference-list, if any, set up by an earlier CAC entry. To clear all the preferences from an earlier CAC entry, configure a codec-preference-list <i>list-name</i> , where <i>list-name</i> is the name of a list containing no codecs.
Step 10	end Example: Router(config-sbc-sbe-cacpolicy)# end	Exits the cac-policy-set configuration mode and enters the Privileged EXEC mode.
Step 11	show sbc service-name sbe cac-policy-set id table name entry entry Example: Router# show sbc mysbc sbe cac-policy-set 1 table standard_policy_list entry 1	Displays detailed information pertaining to a specific entry in a CAC policy table, including codecs with preferences, if any.

For more information about the commands, refer to the [Cisco Unified Border Element \(SP Edition\) Command Reference: Unified Model](#).

Configuration Examples—Configuring Codec Restriction

This section provides a sample configuration and output for configuring restrictions on codecs and configuring a CAC policy to use a codec list.

Example of Configuring Codecs

The following example shows the commands required to configure codec restriction.

Figure 17-1 contains three adjacencies (A, B, and C). Any calls involving A need to be configured to use only the G729 and PCMU (G.711) codecs with a minimal preferred packetization period of 10 milliseconds. However, calls between B and C can use any available codecs.

To create a codec list containing the specified codecs configured with a minimal preferred packetization period, use the following commands:

```
Router# configure terminal
Router(config)# sbc mysbc
Router(config-sbc)# sbe
```

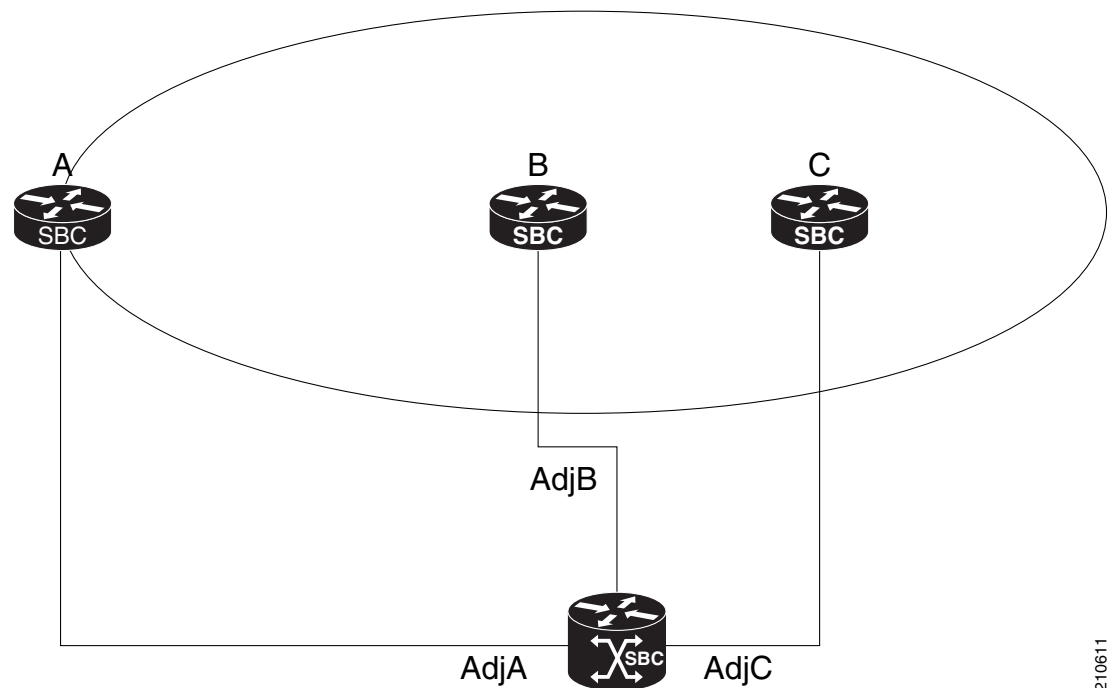
```

Router(config-sbc-sbe) # codec list allowable_codecs
Router(config-sbc-sbe-codec-list) # description The set of codecs allowed on adjacency AdjA
Router(config-sbc-sbe-codec-list) # codec g729 packetization-period 20
Router(config-sbc-sbe-codec-list) # codec pcmu packetization-period 10
Router(config-sbc-sbe-codec-list) # exit

```

After configuring codec restriction, you must configure a CAC policy to use the codec list. See [?\\$paranum>Example of Configuring a CAC Policy to Use a Codec List? section on page 17-23](#).

Figure 17-1 Example Scenario for Configuring Codec Restriction



210611

Example of Configuring a CAC Policy to Use a Codec List

The following example shows the commands required to configure a CAC policy to use a codec list. To configure a codec list, see [?\\$paranum>Example of Configuring Codecs? section on page 17-22](#).

```

Router# configure terminal
Router(config) # sbc mysbc
Router(config-sbc) # sbe
Router(config-sbc-sbe) # cac-policy-set 1
Router(config-sbc-sbe-cacpolicy) # first-cac-table table1
Router(config-sbc-sbe-cacpolicy) # first-cac-scope call
Router(config-sbc-sbe-cacpolicy) # cac-table table1
Router(config-sbc-sbe-cacpolicy-cactable) # match-type adjacency
Router(config-sbc-sbe-cacpolicy-cactable) # entry 1
Router(config-sbc-sbe-cacpolicy-cactable) # match AdjA
Router(config-sbc-sbe-cacpolicy-cactable) # codec-restrict-to-list allowable_codecs
Router(config-sbc-sbe-cacpolicy-cactable-entry) # action cac-complete
Router(config-sbc-sbe-cacpolicy-cactable-entry) # exit
Router(config-sbc-sbe-cacpolicy) # complete

```

**Note**

The codec list command line interface (CLI) commands can only be entered at the per-call level in the CAC policy tables. If you configure a codec list at any other level the CAC policy set will not activate. However, a log is displayed after you complete the configuration and the policy set is marked as “complete”.

Configuration Examples—H.323 TCS Codecs Support

The following example configures a codec list called “caller-media-caps-list” and assigns that list to the CAC table “cac-tbl-1” in entry 1 to announce media capabilities on behalf of the SIP caller:

```
Router(config)# sbc mySBC
Router(config-sbc)# sbe
Router(config-sbc-sbe)# codec list caller-media-caps-list
Router(config-sbc-sbe-codec-list)# codec t38
Router(config-sbc-sbe)# cac-policy-set 1
Router(config-sbc-sbe-cacpolicy)# cac-table cac-tbl-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)# caller-media-caps caller-media-caps-list
```

The following example configures a codec list called “callee-media-caps-list” and assigns that list to the CAC table “cac-tbl-1” in entry 1 to announce media capabilities on behalf of the SIP callee.

```
Router(config)# sbc mySBC
Router(config-sbc)# sbe
Router(config-sbc-sbe)# codec list callee-media-caps-list
Router(config-sbc-sbe-codec-list)# codec t38
Router(config-sbc-sbe)# cac-policy-set 1
Router(config-sbc-sbe-cacpolicy)# cac-table cac-tbl-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)# callee-media-caps callee-media-caps-list
```

The following example configures a codec list called “tcs-extra-caps-list” and assigns that list to the CAC table “cac-tbl-1” in entry 1 to announce extra codecs capability on behalf of the SIP side, whether it is the SIP caller or callee:

```
Router(config)# sbc mySBC
Router(config-sbc)# sbe
Router(config-sbc-sbe)# codec list tcs-extra-caps-list
Router(config-sbc-sbe-codec-list)# exit
Router(config-sbc-sbe)# cac-policy-set 1
Router(config-sbc-sbe-cacpolicy)# cac-table cac-tbl-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)# tcs-extra-codecs tcs-extra-caps-list
```

The following example show command lists the codec list names for caller and callee media capabilities and extra TCS capabilities for entry 1 in a CAC policy table:

```
Router# show sbc mySBC sbe cac-policy-set 1 table cac-tbl-1 entry 1
SBC Service mySBC

CAC Policy Set 1
  Active policy set: No
  Description:
  Averaging period: 60 sec
  First CAC table:
```



```

First CAC scope: global

Table name: cac-tbl-1
Description:
Table type: policy-set
Total call setup failures (due to non-media limits): 0

Entry 1
CAC scope:
CAC scope prefix length: 0
Action: Not set
Number of call setup failures (due to non-media limits): 0
Max calls per scope:      Unlimited      Max call rate per scope: Unlimited
Max in-call rate:         Unlimited      Max out-call rate:       Unlimited
Max reg. per scope:       Unlimited      Max reg. rate per scope: Unlimited
Max channels per scope:   Unlimited      Max updates per scope:   Unlimited
Early media:              Allowed        Early media direction:   Both
Early media timeout:      None           Transcoder per scope:    Allowed
Callee Bandwidth-Field:  None           Caller Bandwidth-Field:  None
Media bypass:              Allowed
Renegotiate Strategy:      Delta
Max bandwidth per scope:   Unlimited
...

Caller media capabilities:  caller-media-caps-list
Callee media capabilities: callee-media-caps-list
Extra TCS capabilities:     tcs-extra-caps-list

```

Configuration Example—Defining a Codec using Dynamic Codec Configuration

The following example shows the commands required to define a custom codec from a codec included in SBC (system codec):

```

Router# configure terminal
Router(config)# sbc mySBC
Router(config-sbc)# sbe
Router(config-sbc-sbe)# codec custom G726-40-4 id 4
Router(config-sbc-sbe-codec-def)# rate 64000
Router(config-sbc-sbe-codec-def)# packet time 100
Router(config-sbc-sbe-codec-def)# bandwidth 128000
Router(config-sbc-sbe-codec-def)# sample size 4
Router(config-sbc-sbe-codec-def)# channels 16
Router(config-sbc-sbe-codec-def)# max-frames-per-packet 12
Router(config-sbc-sbe-codec-def)# media video
Router(config-sbc-sbe-codec-def)# options transcode
Router(config-sbc-sbe-codec-def)# type sampling

```

