



Implementing SBC Transcoding

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 Pulse Code Modulation u-law (PCMU) into one encoded as ITU-T G.726-32.

The primary reason for transcoding configurations is to configure the capabilities of external media transcoding devices when these devices cannot be discovered automatically. In-band auto-discovery of transcoder capabilities is currently not supported. Therefore, this step must be done when configuring all connections to all current remote transcoding devices.

**Note**

Transcoding configurations can be skipped altogether if the described reason does not apply.

Media gateways are allowed to connect whether or not configuration has been supplied for them. To help avoid configuration errors, the signaling border element (SBE) logs a warning if an incoming connection is received from a media gateway that is not a data border element (DBE) and does not have transcoding configured.

**Note**

For ACE SBC Release 3.0.0 and later, this feature is supported in the unified model only.

For a complete description of commands used in this chapter, refer to [Chapter 39, “Cisco Session Border Controller Commands.”](#) To locate documentation for other commands that appear in this chapter, use the command reference master index, or search online.

Feature History for Implementing SBC Transcoding

Release	Modification
ACE SBC Release 3.0.00	This feature was introduced on the Cisco 7600 series router along with support for the SBC unified model.

Contents

This chapter contains the following sections:

- [Prerequisites for Implementing Transcoding, page 14-2](#)
- [Information About Transcoding, page 14-2](#)
- [Configuration Examples for Implementing Transcoding, page 14-9](#)

Prerequisites for Implementing Transcoding

The following prerequisites are required to implement SBC transcoding:

- On the Application Control Engine Module (ACE), you must be an Admin user to enter SBC commands. For more information, see the *Application Control Engine Module Administration Guide* at http://www.cisco.com/en/US/docs/interfaces_modules/services_modules/ace/vA4_2_0/configuration/administration/guide/AdminGd.html.
- Before implementing transcoding, the SBC must already be created. See the procedures described in Chapter 2, “ACE Configuration Prerequisites for the SBC”.
- All SBE and DBE configurations required to make simple calls must already be configured. Transcoding configurations follow these configurations.

Information About Transcoding

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.

Transcoding requires specialized digital signal processor (DSP) hardware, which is not available within SBC itself. A Cisco MGX 8880 device can be used to provide transcoding function for one or more SBCs.

The SBC supports two types of transcoding:

- [Transcoding After Rejection, page 14-2](#)
- [Codec Filtering, page 14-4](#)

Transcoding After Rejection

The SBC automatically brings the transcoding device into use for any call requiring transcoding between these codecs, as long as the Call Admission Control (CAC) policy configuration does not preclude the transcoder service from being supplied for the call. When a call that requires transcoding is set up, the SBE goes through the following steps:

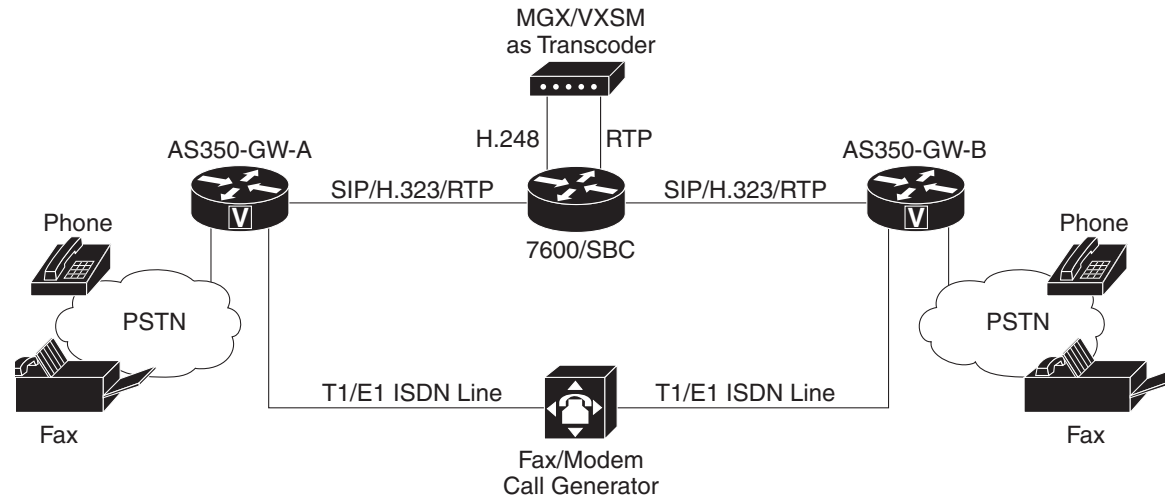
- Receives an initial signaling request from the calling endpoint. This triggers the SBE to perform initial call setup on the incoming and outgoing DBEs. The SBE then forwards the request to the called endpoint.
- Receives a response from the called endpoint that indicates that none of the codecs in the initial request are acceptable. These responses include:
 - 415—Unsupported media type (SIP)
 - 488—Not acceptable here (SIP)
 - Failure to identify common codec during Terminal Capability Exchange procedure of H.245 protocol.

This triggers the SBE to bring a transcoder into the call that is inserted in the media path between the incoming and outgoing DBEs. A new request is sent to the called endpoint, indicating the new codec type generated by the transcoder.

- SBE may then have to iterate through the list of codecs the transcoder supports until it finds one that is acceptable to the called endpoint. When this is done, the call is connected and media transmission begins.

Figure 14-1 shows where the transcoder sits in the network, and the path taken by the media in a transcoded call.

Figure 14-1 Transcoding Configuration



Note

Although Figure 14-1 shows two DBEs, transcoding is possible with a single DBE. With a single DBE, the media flows through the DBE twice, once on its way from the sending endpoint to the transcoder and a second time as it flows from the transcoder to the receiving endpoint.

For the Session Border Controller (SBC) to program the transcoder, it must be registered. The transcoding device acts as an H.248 media gateway, so it needs to be configured with the IP address and port of the SBE or SBC to connect to. The SBE or SBC acts as an H.248 Media Gateway Controller. (See the documentation for the transcoder device for notes on how to do this. The documentation for the Cisco MGX 8880 can be found at

http://www.cisco.com/en/US/docs/switches/wan/mgx/software/mgx_r5.6/voice/vxsm/configuration/guides/config5.html

In addition, the SBE must have the following specific configuration:

- An H.248 control address and port must be configured (using the **sbe control address ipv4** and **sbe control address h248 port** commands). By default, this is on port 2944, and it is the address and port to which the transcoder must connect.
- An explicit media gateway needs to be configured (using the **sbe media-gateway ipv4** command). The explicit media gateway must have its list of supported codecs defined so that the SBC knows which codecs the transcoder can translate between, and it must be identified as a transcoder (using the **sbe media-gateway ipv4 codecs** and **sbe media-gateway ipv4 transcoder** commands).
- The **show services sbc sbe media-gateway-associations** command can be used to check that the transcoder has correctly registered with the SBE. If this has happened, the transcoder should appear in the list of known media gateways with an active association.

Codec Filtering

The SBC allows you to restrict which codecs a particular call, caller and callee are allowed to use by whitelisting certain codecs. Initially all recognized codecs are on the whitelist. If a codec is requested which is absent from the call, caller, or callee codec whitelist, then the call still proceeds, but the forbidden codecs are removed from the offer and media gate configuration.

By supporting caller and callee codec lists, the SBC is able to make more intelligent transcoding decisions. If the codec support of either the calling or the called endpoint is known, then setting the caller and/or callee lists in a CAC policy is appropriate. However it may be that other considerations, such as the source adjacency, will affect the codec decision, in which case the per-call codec list can still be used.

For example, if the caller and callee codec lists are set to 'A and B', then all calls would use codec A. However, if a call had come across a transit network X (as indicated by the source adjacency) that only supported codec B, then the user could have an extra policy matching on source adjacency X, setting the per-call codec list to B. Calls crossing network X would then be forced to use codec B.

You can also limit the minimum packetization period of each codec, by configuring a value for the lowest acceptable minimum packetization period for each permitted codec. If a session is requested with a packetization period below this limit, the call still proceeds, but DC SBC increases the packetization period to the configured minimum value.

Restrictions for Transcoding

Review the following restrictions for transcoding:

- The H.323 fast-start calls will be dropped to slow-start procedure if transcoding is required. This can be achieved by either the callee side rejecting fast-start request or by configuring the **h245-tunnel disable** command in the adjacency table.
- No transcoding support for H.323 to SIP Interworked calls.

Configuring Transcoding After Rejection

In this configuration area, the user supplies a configuration for a list of remote media gateways that may need to be managed by the SBE. This is not required when transcoding is not needed.

The primary reason for transcoding configurations is to configure the capabilities of external media transcoding devices when these devices cannot be discovered automatically. In-band auto-discovery of transcoder capabilities is currently not supported. Therefore, this step must be done when configuring all connections to all current remote transcoding devices.

**Note**

Transcoding configurations can be skipped if the described reason does not apply.

By default, media gateways are allowed to connect whether or not configuration has been supplied for them. To help avoid configuration errors, the SBE logs a warning if an incoming connection is received from a media gateway that is not a DBE and does not have transcoding configured.

The basic steps for implementing transcoding are as follows:

1. Configure the IP address, port, and transport protocol for H.248 media gateway controller on SBC. This step may not be required if the Media Gateway Controller has already been configured.

2. Configure the media gateway IP address.
3. Configure the codecs to be transcoded (for example, between ITU-T G.711ulaw and ITU-T G.729A).
4. Specify the media gateway as a transcoder.
5. Activate SBE.

This task implements transcoding for SBC.

Once configured, the SBC automatically brings the transcoding device into use for any call requiring transcoding between the codecs as long as the call admission control (CAC) policy configuration does not preclude the transcoder service from being supplied for the call using the **transcode-deny** command (See the “[Configuring Call Admission Control Policy Sets and CAC Tables](#)” section in the [Implementing SBC Policies](#) module).

**Note**

In an H.323 adjacency configuration, you must use the **h245-tunnel disable** command for H.323 FastStart transcoded calls.

SUMMARY STEPS

1. **configure**
2. **sbc** *service-name*
3. **sbe**
4. **control address h248 index** *index-number*
5. **ipv4** *ipv4_IP_address*
6. **port** *port-number*
7. **transport** [*transport-type*]
8. **exit**
9. **media-gateway ipv4** *IPv4_IP_address*
10. **codecs** *codec-list*
11. **transcoder**
12. **exit**
13. **activate**
14. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: host1/Admin# configure	Enables global configuration mode.
Step 2	sbc service-name Example: host1/Admin(config)# sbc mysbc	Enters the mode of an SBC service. Use the <i>service-name</i> argument to define the name of the service.
Step 3	sbe Example: host1/Admin(config-sbc)# sbe	Enters the mode of an SBE entity within a SBC service.
Step 4	control address h248 index index-number Example: host1/Admin(config-sbc-sbe)# control address h248 index 0	Configures an SBE to use a given IPv4 H.248 control address
Step 5	ipv4 ipv4_IP_address Example: host1/Admin(config-sbc-sbe-ctrl-h248)# ipv4 1.1.1.1	Configures an SBE to use a given IPv4 H.248 control address.
Step 6	port port-number Example: host1/Admin(config-sbc-sbe-ctrl-h248)# port 2000	Configures an SBE to use a given IPv4 H.248 port for H.248 communications.
Step 7	transport [transport-type] Example: host1/Admin(config-sbc-sbe-ctrl-h248)# transport udp	Configures transport type for H.248 communications.
Step 8	exit Example: host1/Admin(config-sbc-sbe)# exit	Exits the current configuration mode.
Step 9	media-gateway ipv4 IPv4_IP_address Example: host1/Admin(config-sbc-sbe)# media-gateway ipv4 10.0.0.1	Configures a media gateway.

	Command or Action	Purpose
Step 10	codecs <i>codec-list</i> Example: host1/Admin(config-sbc-sbe-mg)# codecs m=audio 1234 RTP/AVP 0 18,a=rtpmap:0 PCMU/8000,a=rtpmap:18 G729A/8000	Configures the codecs supported by the media gateway.
Step 11	transcoder Example: host1/Admin(config-sbc-sbe-mg)# transcoder	Configures the media gateway with transcoder support.
Step 12	exit Example: host1/Admin(config-sbc-sbe-mg)# exit	Exits to the sbe command mode level.
Step 13	activate Example: host1/Admin(config-sbc-sbe)# activate	Initiates the SBC service after all SBE address configuration has been successfully committed.
Step 14	end Example: host1/Admin(config-sbc-sbe)# end	Ends the configuration session.

Configuring Codec Filtering Transcoding

Configure codec filtering transcoding as shown below.

SUMMARY STEPS

1. **configure**
2. **sbc** *service-name*
3. **sbe**
4. **media-gateway ipv4** *IPv4_IP_address*
5. **codecs** *codec-list*
6. **transcoder**
7. **exit**
8. **cac-policy-set**
9. **cac-table**
10. **entry** *entry_num*
11. **caller-codec-list** *list-name*
12. **exit**
13. **exit**

14. `codec-list list-name`
15. `codec codec-name`
16. `exit`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<code>configure</code> Example: host1/Admin# <code>configure</code>	Enables global configuration mode.
Step 2	<code>sbc service-name</code> Example: host1/Admin(config)# <code>sbc mysbc</code>	Enters the mode of an SBC service. Use the <i>service-name</i> argument to define the name of the service.
Step 3	<code>sbe</code> Example: host1/Admin(config-sbc)# <code>sbe</code>	Enters the mode of an SBE entity within a SBC service.
Step 4	<code>media-gateway ipv4 IPv4_IP_address</code> Example: host1/Admin(config-sbc-sbe)# <code>media-gateway ipv4 10.0.0.1</code>	Configures a media gateway.
Step 5	<code>codecs codec-list</code> Example: host1/Admin(config-sbc-sbe-mg)# <code>codecs m=audio 1234 RTP/AVP 0 18,a=rtpmap:0 PCMU/8000,a=rtpmap:18 G729A/8000</code>	Configures the codecs supported by the media gateway.
Step 6	<code>transcoder</code> Example: host1/Admin(config-sbc-sbe-mg)# <code>transcoder</code>	Configures the media gateway with transcoder support.
Step 7	<code>exit</code> Example: host1/Admin(config-sbc-sbe-mg)# <code>transcoder</code>	Exits the media gateway configuration mode.
Step 8	<code>cac-policy-set</code> Example: host1/Admin(config-sbc-sbe)# <code>cac-policy-set 1</code>	Enters the CAC policy submode.

	Command or Action	Purpose
Step 9	cac-table Example: host1/Admin(config-sbc-sbe) # cac-table 1	Creates or configures an admission control table.
Step 10	entry Example: host1/Admin(config-sbc-sbe-cacpolicy-cactable) # entry 1	Creates or modifies an entry in a table.
Step 11	caller-codec-list list-name Example: host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # caller-codec-list test	Lists the codecs which the caller leg of a call is allowed to use.
Step 12	exit Example: host1/Admin(config-sbc-sbe-cacpolicy-cactable) # exit	Exits the CAC table entry configuration mode.
Step 13	exit Example: host1/Admin(config-sbc-sbe-cacpolicy) # exit	Exits the CAC-policy-set configuration mode.
Step 14	codec-list list-name Example: host1/Admin(config-sbc-sbe) # codec-list my_codecs	Creates a codec list.
Step 15	codec codec-name Example: host1/Admin(config-sbc-sbe-codec-list) # codec PCMU	Adds a codec to a codec list.
Step 16	end Example: host1/Admin(config-sbc-sbe) # end	Ends the configuration session.

Configuration Examples for Implementing Transcoding

The example below is a configuration of transcoding after rejection.

```
host1/Admin(config) # sbc mySbc
host1/Admin(config-sbc) # sbe
host1/Admin(config-sbc-sbe) # control address h248 index 1
host1/Admin(config-sbc-sbe-ctrl-h248) # ipv4 88.88.133.2
host1/Admin(config-sbc-sbe-ctrl-h248) # port 2000
```

```

host1/Admin(config-sbc-sbe-ctrl-h248)# transport udp
host1/Admin(config-sbc-sbe-ctrl-h248)# exit
host1/Admin(config-sbc-sbe)# media-gateway ipv4 10.0.0.1
host1/Admin(config-sbc-sbe-mg)# codecs m=audio 1234 RTP/AVP 0 18,a=rtpmap:0
PCMU/8000,a=rtpmap:18 G729A/8000
host1/Admin(config-sbc-sbe-mg-codecs)# transcoder
host1/Admin(config-sbc-sbe-mg-codecs)# exit
host1/Admin(config-sbc-sbe)# activate
host1/Admin(config-sbc-sbe)# end

```

Below is an example of codec filtering transcoding.

```

host1/Admin(config)# sbc mySbc
host1/Admin(config-sbc)# sbe
Router(config)# interface vlan 130
Router(config)# ip address 10.130.10.2 255.255.255.240
host1/Admin(config-if)# alias 10.130.10.4 255.255.255.240
host1/Admin(config-if)# peer ip address 10.130.10.13 255.255.255.240
host1/Admin(config-if)# no shutdown
Router(config)# interface vlan 200
Router(config)# ip address 10.74.50.113 255.255.255.192

host1/Admin(config)# ft interface vlan 20
host1/Admin(config)# ip address 10.10.101.21 255.255.255.0
host1/Admin(config-if)# peer ip address 10.10.101.22 255.255.255.0
host1/Admin(config-if)# no shutdown

host1/Admin(config)# ft peer 1
host1/Admin(config-ft-peer)# heartbeat interval 300
host1/Admin(config-ft-peer)# heartbeat count 10
host1/Admin(config)# ft-interface vlan 20
host1/Admin(config)# ft group 1
host1/Admin(config-ft-group)# peer 1
host1/Admin(config-ft-group)# priority 90
host1/Admin(config-ft-group)# peer priority 110
host1/Admin(config-ft-group)# associate-context Admin
host1/Admin(config-ft-group)# inservice

host1/Admin(config)# ip route 10.0.20.33 255.255.255.255 10.130.10.33
host1/Admin(config)# ip route 0.0.0.0 0.0.0.0 10.74.50.114
host1/Admin(config)# ip route 0.0.0.0 0.0.0.0 10.130.10.1

host1/Admin(config)# snmp-server community cisco group Network-Monitor
host1/Admin(config)# snmp-server community public group Network-Monitor
host1/Admin(config)# snmp-server community private group Network-Monitor

host1/Admin# configure
host1/Admin(config)# sbc sbc-11
host1/Admin(config-sbc)# sbe
host1/Admin(config-sbc-sbe)# media-gateway ipv4 10.100.181.2
host1/Admin(config-sbc-sbe-mg)# codecs m=audio 20000 RTP/AVP 0 8 18,a=rtpmap:0
PCMU/8000,a=rtpmap:8 PCMA/8000,a=rtpmap:18 G729/8000
host1/Admin(config-sbc-sbe-mg)# transcoder

host1/Admin(config-sbc-sbe)# control address h248 index 1
host1/Admin(config-sbc-sbe-ctrl-h248)# ipv4 10.130.10.4
host1/Admin(config-sbc-sbe-ctrl-h248)# transport udp

host1/Admin(config-sbc-sbe)# adjacency sip SIPP81
host1/Admin(config-sbe-adj-sip)# nat force-off
host1/Admin(config-sbc-sbe-adj-sip)# preferred-transport udp
host1/Admin(config-sbc-sbe-adj-sip)# redirect-mode pass-through
host1/Admin(config-sbe-adj-sip)# authentication nonce timeout 300
host1/Admin(config-sbe-adj-sip)# signaling-address ipv4 10.130.10.4

```

```

host1/Admin(config-sbc-sbe-adj-sip) # signaling-port 5060
host1/Admin(config-sbc-sbe-adj-sip) # remote-address ipv4 10.0.244.81 255.255.255.255
host1/Admin(config-sbc-sbe-adj-sip) # signaling-peer 10.0.244.81
host1/Admin(config-sbc-sbe-adj-sip) # signaling-peer-port 5060
host1/Admin(config-sbc-sbe-adj-sip) # db-location-id 0
host1/Admin(config-sbc-sbe-adj-sip) # reg-min-expiry 3000
host1/Admin(config-sbc-sbe-adj-sip) # attach

host1/Admin(config-sbc-sbe) # adjacency sip SIPP91
host1/Admin(config-sbc-sbe-adj-sip) # nat force-off
host1/Admin(config-sbc-sbe-adj-sip) # preferred-transport udp
host1/Admin(config-sbc-sbe-adj-sip) # redirect-mode pass-through
host1/Admin(config-sbc-sbe-adj-sip) # authentication nonce timeout 300
host1/Admin(config-sbc-sbe-adj-sip) # signaling-address ipv4 10.130.10.4
host1/Admin(config-sbc-sbe-adj-sip) # signaling-port 5060
host1/Admin(config-sbc-sbe-adj-sip) # remote-address ipv4 10.0.244.91 255.255.255.255
host1/Admin(config-sbc-sbe-adj-sip) # signaling-peer 10.0.244.91
host1/Admin(config-sbc-sbe-adj-sip) # signaling-peer-port 5060
host1/Admin(config-sbc-sbe-adj-sip) # db-location-id 0
host1/Admin(config-sbc-sbe-adj-sip) # reg-min-expiry 3000
host1/Admin(config-sbc-sbe-adj-sip) # attach

host1/Admin(config-sbc-sbe) # sip inherit profile preset-core

host1/Admin(config-sbc-sbe) # cac-policy-set 1
host1/Admin(config-sbc-sbe-cacpolicy) # first-cac-table table
host1/Admin(config-sbc-sbe-cacpolicy) # first-cac-scope call
host1/Admin(config-sbc-sbe-cacpolicy) # averaging-period 60
host1/Admin(config-sbc-sbe) # cac-table table
host1/Admin(config-sbc-sbe-cacpolicy-cactable) # match-type adjacency
host1/Admin(config-sbc-sbe-cacpolicy-cactable) # entry 1
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # match-value SIPP81
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # action cac-complete
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # max-bandwidth 64009 Gbps
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # max-updates 4294967295
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # max-channels 4294967295
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # early-media-type full-duplex
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # early-media-timeout 0
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # caller-codec-list allow711u
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # callee-privacy never
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # caller-privacy never
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # callee-hold-setting standard
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # caller-hold-setting standard
host1/Admin(config-sbc-sbe-cacpolicy-cactable) # entry 2
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # match-value SIPP91
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # action cac-complete
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # max-bandwidth 64009 Gbps
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # max-updates 4294967295
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # max-channels 4294967295
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # early-media-type full-duplex
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # early-media-timeout 0
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # callee-codec-list allowg729
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # callee-privacy never
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # caller-privacy never
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # callee-hold-setting standard
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # caller-hold-setting standard
host1/Admin(config-sbc-sbe-cacpolicy-cactable-entry) # complete

host1/Admin (config-sbc-sbe) # active-cac-policy-set 1

host1/Admin (config-sbc-sbe) # retry-limit 3

host1/Admin (config-sbc-sbe) # call-policy-set 1
host1/Admin(config-sbc-sbe-rtgpolicy) # first-call-routing-table table

```

```

host1/Admin(config-sbc-sbe-rtgpolicy-rtgtable)# entry 1
host1/Admin(config-sbc-sbe-rtgpolicy-rtgtable-entry)# action complete
host1/Admin(config-sbc-sbe-rtgpolicy-rtgtable-entry)# dst-adjacency SIP91
host1/Admin(config-sbc-sbe-rtgpolicy-rtgtable-entry)# match-address 318X
host1/Admin(config-sbc-sbe-rtgpolicy-rtgtable-entry)# prefix
host1/Admin(config-sbc-sbe-rtgpolicy-rtgtable-entry)# complete

host1/Admin (config-sbc-sbe)# active-call-policy-set 1

host1/Admin(config-sbc-sbe)# sip max-connections 2
host1/Admin(config-sbc-sbe)# sip timer
host1/Admin(config-sbc-sbe-sip-tmr)# tcp-idle-timeout 120000
host1/Admin(config-sbc-sbe-sip-tmr)# tls-idle-timeout 3600000
host1/Admin(config-sbc-sbe-sip-tmr)# udp-response-linger-period 32000
host1/Admin(config-sbc-sbe-sip-tmr)# udp-first-retransmit-interval 500
host1/Admin(config-sbc-sbe-sip-tmr)# udp-max-retransmit-interval 4000
host1/Admin(config-sbc-sbe-sip-tmr)# invite-timeout 180

host1/Admin (config-sbc-sbe)# codec-list allow711u
host1/Admin(config-sbc-sbe-codec-list)# codec PCMU

host1/Admin (config-sbc-sbe)# codec-list allowg729
host1/Admin(config-sbc-sbe-codec-list)# codec G729

host1/Admin(config-sbc-sbe)# h323
host1/Admin(config-sbc-sbe-h323)# ras timeout arq 5000
host1/Admin(config-sbc-sbe-h323)# ras retry arq 2
host1/Admin(config-sbc-sbe-h323)# ras timeout brq 3000
host1/Admin(config-sbc-sbe-h323)# ras retry brq 2
host1/Admin(config-sbc-sbe-h323)# ras timeout drq 3000
host1/Admin(config-sbc-sbe-h323)# ras retry drq 2
host1/Admin(config-sbc-sbe-h323)# ras timeout grq 5000
host1/Admin(config-sbc-sbe-h323)# ras retry grq 2
host1/Admin(config-sbc-sbe-h323)# ras timeout rrq 3000
host1/Admin(config-sbc-sbe-h323)# ras retry rrq 2
host1/Admin(config-sbc-sbe-h323)# ras rrq ttl 60
host1/Admin(config-sbc-sbe-h323)# ras timeout urq 3000
host1/Admin(config-sbc-sbe-h323)# ras retry urq 1
host1/Admin(config-sbc-sbe-h323)# h225 timeout proceeding 10000
host1/Admin(config-sbc-sbe-h323)# h225 timeout establishment 180000
host1/Admin(config-sbc-sbe-h323)# h225 timeout setup 4000
host1/Admin(config-sbc-sbe-h323)# ras rrq keepalive 45000

host1/Admin(config-sbc-sbe)# h323
host1/Admin(config-sbc-sbe-adj-h323)# adjacency timeout 30000

host1/Admin(config-sbc-sbe)# blacklist
host1/Admin(config-sbc-sbe-blacklist)# global

host1/Admin(config-sbc-sbe)# blacklist
host1/Admin(config-sbc-sbe-blacklist)# address-default

host1/Admin(config-sbc-sbe)# redirect-limit 2
host1/Admin(config-sbc-sbe)# deact-mode normal
host1/Admin (config-sbc-sbe)# activate

host1/Admin(config-sbc)# dbe
host1/Admin(config-sbc-dbe)# location-id 0
host1/Admin(config-sbc-dbe)# media-timeout 360
host1/Admin(config-sbc-dbe)# deact-mode normal
host1/Admin (config-sbc-dbe)# activate

```

Verification

Use the following commands to verify operation:

Command	Purpose
<code>show services sbc sbe media-gateway-associations</code>	Displays a list of known media gateways with an active association.

The following example shows the SBC and media communications.

```

host1/Admin# show services sbc slt-n2 sbe media-gateway-associations
  SBC Service "slt-n2"
  Media gateway 192.169.125.1:2944
    Gateway Protocol = megaco
    Transport Protocol = UDP
    Local Address    = 22.46.0.11:2944

          Sent           Received         Failed         Retried
Requests  117              2              1              7
Replies   2              116            -              0

```

