CHAPTER **51**

SIP Instant Messaging

Cisco Unified Border Element (SP Edition) supports SIP instant messaging (IM). Two options for SIP instant messaging are configurable—record-route passthrough and privacy for SIP Instant Messaging.

The typical SIP instant messaging implementation uses end-to-end Record-Route passthrough, but privacy is not applied.

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.

Feature History for SIP Instant Messaging

Release	Modification
Cisco IOS XE Release 2.5	The SIP Instant Messaging feature was introduced on the Cisco IOS XR.

Contents

This chapter contains the following sections:

SIP Instant Messaging

Cisco Unified Border Element (SP Edition) supports SIP instant messaging (IM). For SIP instant messaging, SBC handles the calls as follows:

- Calls with the Media Announcement m=message or m=x-ms-message are allowed.
- Messages containing multiple m=message lines are permitted.

- SBC does not allocate any bandwidth to support IM dialogs because there is no media stream for such dialogs.

- The IM message is rejected if the SDP also contains any other media types.

- SBC forwards the SDP body unchanged unless privacy is configured.

SBC can be configured to pass through Record-Route headers from one side of the IM dialog to the other so that the end-to-end Route header set is available to caller and callee. In this case, SBC remains in the flow of messages by appending Record-Route headers and not rewriting the Contact.

Configurable Options for SIP Instant Messaging

Two options for SIP instant messaging are configurable and explained in the following sections:

The typical SIP instant messaging implementation will use end-to-end Record-Route passthrough but will not apply privacy.

Record-Route Passthrough for SIP Instant Messaging

This section explains Record-Route passthrough and contains the following subsections:

Record-Route Passthrough Overview

A Record-Route set represents the hop-by-hop route SIP messages must traverse between two endpoints as part of a SIP dialog. The final hop to the endpoint is represented by the Contact header. SBC's normal behavior is to rewrite the Contact and maintain two independent Record-Route sets, one for each side of the call. Each of these Record-Route sets represents the route between the endpoint and the adjacency on that side of the call.

Passing through the end-to-end Record-Route set means that the adjacencies on a call do not represent the last hop and therefore must not be identified by the Contact header. When passing through the end-to-end Record Route set, SBC also passes through the end-to-end Contacts without rewriting them. In this case, SBC remains in the flow by appending Record-Route headers representing the inbound and outbound adjacencies.

If inbound and outbound adjacencies have conflicting Record-Route passthrough configurations, the setting of the inbound adjacency is used. For example, if the inbound adjacency enables Record-Route passthrough but the outbound adjacency does not, the outbound adjacency will forward the supplied Record-Route set, append a Record-Route header corresponding to the outbound adjacency, and leave the Contact header unaltered.

Registered Subscribers and Record-Route Passthrough

Record-Route passthrough behavior does not apply to SIP messages received from or going to Registered endpoints regardless of the inbound adjacency's Record-Route passthrough configuration.

The SIP specification (RFC 326) mandates that registrars ignore Record-Route headers present on a REGISTER message. Therefore to ensure that SBC remains in subsequent dialogs created to or from registered subscribers, SBC must rewrite the Contact in REGISTER messages. SBC updates the Contacts in later dialogs created by a registered endpoint so that they match the Contact previously published to network. At the point that SBC rewrites the Contact, SBC terminates any existing Record-Route set and creates a new one.

Record-Route Passthrough Configuration

To pass through Record-Route sets, SBC provides configuration on a per adjacency basis using the **passthrough header record-route** command.

- If turned off (**no passthrough header record-route**), the Record-Route set is cached and returned to the endpoint. Thereafter, the Record-Route set is used to build Route headers on subsequent outgoing messages. Neither end will see the entire end-to-end Record-Route set. This is the default behavior.

- If turned on, the request is not from or to a registered subscriber, the following occurs:

  – For dialog-creating requests, any Record-Route set present on the request is passed through SBC and forwarded to the receiving endpoint.

  – The Contact present on the request is passed through SBC and forwarded to the receiving endpoint.

  – Record-Route headers representing the inbound and outbound adjacency are appended to the request.

  – The end-to-end Record-Route set is passed back through SBC on the response and forwarded to the calling endpoint. Both endpoints see the entire end-to-end Record-Route set.

  – The preceding will be the behavior regardless of outbound adjacency configuration.

  – Subsequent in-dialog requests do not have their Request URI updated to match the Contact received on the dialog-creating request.

If a request is from or to a registered subscriber, it is processed as though **record-route passthrough** was turned off.

If topology hiding or privacy is applied to a call, the Record-Route set is stripped from the request regardless of the **record-route passthrough** configuration.

Configuring Record-Route passthrough does not result in the Route headers being passed through on subsequent messages.

With Record-Route passthrough enabled, messages in an IM dialog are adjusted as described earlier in this section. As an example of these Record-Route passthrough adjustments, the following requests show how enabling Record-Route passthrough affects an INVITE request that is used for an IM dialog. In the first example, Record-Route passthrough is not used. In the second example, Record-Route passthrough is enabled, and SBC adds the Record-Route information in bold font to the INVITE request. In both examples, privacy is not used.

In these examples, the INVITE requests are outbound from the SBC to a SIP proxy server.

Outbound INVITE (without Record-Route passthrough):

```
INVITE sip:callee@callee.com SIP/2.0
```

```
Via: SIP/2.0/UDP sbc.com;branch=z9hG4bK-sbc-1
From: Caller <sip:caller@dcsbc.com>;tag=dcsbc
To: Callee <sip:callee@callee.com>
Call-ID: sbc-call-1
Contact: Caller <sip:abcd@sbc.com>
Content-Type: application/sdp
Content-Length: 135

v=0
o=- 0 0 IN IP4 192.168.1.121
s=session
c=IN IP4 192.168.1.121
t=0 0
m=message 5060 sip sip:caller@caller.com
c=IN IP4 192.168.1.122
```

Outbound INVITE (with Record-Route passthrough):

```
INVITE sip:callee@callee.com SIP/2.0
From: Caller <sip:caller@caller.com>;tag=caller-tag-1
To: Callee <sip:callee@callee.com>
Call-ID: call-1
Contact: Caller <sip:caller@caller.com>
Record-Route: <sip:proxy1@example.com;lr>
Record-Route: <sip:caller_adj@sbc.com;lr>
Record-Route: <sip:callee_adj@sbc.com;lr>
```

Privacy for SIP Instant Messaging

For IM privacy, the CAC configuration that is used for normal calls is also used for IM dialogs. The CAC table entry command option **caller-privacy** configures privacy. For information on using this command, see 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.

If privacy is enabled, SBC adjusts three fields in the SIP IM SDP to make them anonymous.

- The Connection lines (c=) may contain sensitive IP address information in the connection address field. This information is hidden by inserting a Session Description level Connection line with the local address of the corresponding adjacency. Any other Connection lines are removed.

- The o= line is replaced with SBC's own information. The address field in this line is set to match the c= line.

- For SDP indicating a SIP IM stream, the Media Descriptions lines (m=) will be of the form:

    m=message *port*[/*number_of_ports*] sip *format_list*

    In the preceding, *format_list* is a SIP URL and may contain user sensitive information. The user-sensitive information is hidden by stripping all messages indicating Media Description lines and forwarding an Offer containing the following Media Description line:

    m=message 5060 sip sip:anonymous@192.168.10.10

The following examples show how enabling **caller-privacy** affects the SDP messages that are used for an IM dialog. In the examples, 192.168.10.10 is the local address of the outbound adjacency, and 192.168.2.20 is the address of the inbound adjacency. In the examples, the information in bold font has been adjusted to make the caller anonymous. The following example is outbound from the SBC to a SIP proxy server.

Outbound INVITE (without privacy):

```
v=0
```

```
o=- 0 0 IN IP4 192.168.1.121
s=session
c=IN IP4 192.168.1.121
t=0 0
m=message 5060 sip sip:caller@caller.com
c=IN IP4 192.168.1.122
```

Outbound INVITE (with privacy):

```
v=0
o=- 0 0 IN IP4 192.168.10.10
s=session
c=IN IP4 192.168.10.10
t=0 0
m=message 5060 sip sip:anonymous@192.168.10.10
c=IN IP4 192.168.10.10
```

The following example is outbound from the SBC to a SIP gateway.

Outbound 200 OK (without privacy):

```
v=0
o=- 0 0 IN IP4 192.168.2.200
s=session
c=IN IP4 192.168.2.200
t=0 0
m=x-ms-message 5060 sip null
```

Outbound 200 OK (with privacy)

```
v=0
o=- 0 0 IN IP4 192.168.2.20
s=session
c=IN IP4 192.168.2.20
t=0 0
m=x-ms-message 5060 sip sip:anonymous@192.168.2.20
```

SDP Handling for SIP Instant Messaging

In the SDP, SBC identifies IM dialogs by the presence of m=message or m=x-ms-message. For an offer that includes both IM and audio/video media lines, SBC rejects the offer with error code 488, Not Acceptable Here. After the initial media negotiation, any subsequent reoffer that attempts to change a call from an IM to a non-IM or vice versa is also rejected with error code 488.

SBC passes through SDP for IM dialogs unchanged with the exception of modifications due to privacy (see ?$paranum>Privacy for SIP Instant Messaging? section on page 51-4). SBC does not allocate any gates or media pinholes.

The following sections provide more information on how SBC handles SDP for SIP instant messaging:

- ?$paranum>SIP URLs in the SDP? section on page 51-6
- ?$paranum>Miscellaneous SDP Handling? section on page 51-6

SIP URLs in the SDP

SDP attached to SIP IM dialogs can contain SIP URLs in the Media Description lines. For example:

m=message 5060 sip sip:example@home.net

If an endpoint sends an in-dialog message to the URL in the SDP (by placing the URL in the Request URI), the behavior is as follows:

- If SBC has not been configured to pass through the Record-Route set, the request may fail to route.
- If SBC has been configured to pass through the Record-Route set, the request will be forwarded by SBC without altering the Request URI.

For information on Record-Route passthrough, see the ?$paranum>Record-Route Passthrough for SIP Instant Messaging? section on page 51-2.

Miscellaneous SDP Handling

This section describes some miscellaneous SDP handling for SIP instant messaging.

- Transcoding—Since there is no media stream, SBC never attempts to bring in a transcoder. If the callee endpoint does not support the media type, the call fails.
- Special Media Descriptions—In some cases, SDP may include a proprietary media type of m=x-ms-message. SBC treats m=x-ms-message exactly the same as m=message. No support is added for any other proprietary media types.
- Interworking—SIP instant messaging does not interwork for SIP/H.323 calls, or for SIP-SIP late to early media interworking calls. If an IM dialogs is used in these scenarios, call setup fails with the response code 488, Not Acceptable Here.
- Invalid Connection Address—SBC does not edit the SDP of SIP IM dialogs except when privacy is configured. Therefore, the connection address passed through SBC may not be valid as far as the receiving endpoint is concerned. This is acceptable because no media is flowing between the endpoints.