



Understanding Session Initiation Protocol (SIP)

Understanding Session Initiation Protocol (SIP) describes SIP and the interaction between SIP and Cisco CallManager.

This section covers the following topics:

- [SIP Networks, page 37-1](#)
- [SIP and Cisco CallManager, page 37-2](#)
- [SIP Functions Supported in Cisco CallManager, page 37-4](#)
- [SIP Signaling/Trunk Interface Configuration Checklist, page 37-16](#)
- [Where to Find More Information, page 37-18](#)

SIP Networks

A SIP network uses the following components:

- **SIP Proxy Server**—The proxy server works as an intermediate device that receives SIP requests from a client and then forwards the requests on the client's behalf. Proxy servers can provide functions such as authentication, authorization, network access control, routing, reliable request retransmission, and security.
- **Redirect Server**—The redirect server provides the client with information about the next hop or hops that a message should take, and the client then contacts the next hop server or user agent server directly.

- Registrar Server—The registrar server processes requests from user agent clients for registration of their current location. Redirect or proxy servers often contain registrar servers.
- User Agent (UA)—A combination of user agent client (UAC) and user agent server (UAS) that initiates and receives calls. A UAC initiates a SIP request. A UAS is a server application that contacts the user when it receives a SIP request. The UAS then returns a response on behalf of the user. Cisco CallManager can act as both a server or client (a back-to-back user agent).

SIP uses a request/response method to establish communications between various components in the network and to ultimately establish a call or session between two or more endpoints. A single session may involve several clients and servers.

Identification of users in a SIP network works through

- A unique phone or extension number.
- A unique SIP address that appears similar to an e-mail address and uses the format `sip:<userID>@<domain>`. The user ID can be either a user name or an E.164 address. Cisco CallManager only supports E.164 addresses; it does not support email addresses.

Related Topics

- [SIP and Cisco CallManager, page 37-2](#)
- [SIP Functions Supported in Cisco CallManager, page 37-4](#)
- [SIP Signaling/Trunk Interface Configuration Checklist, page 37-16](#)

SIP and Cisco CallManager

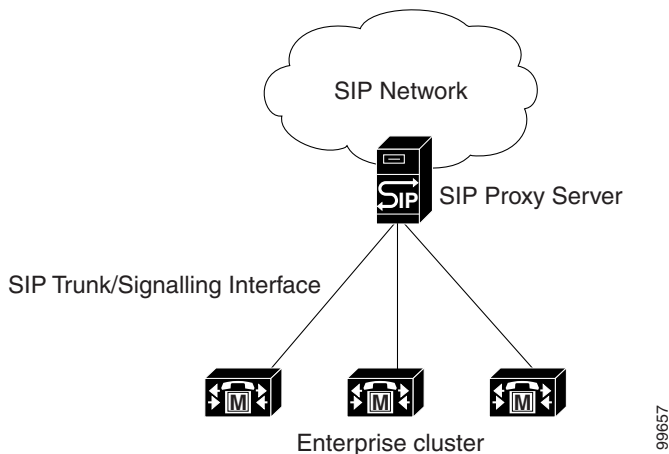
All protocols require that either a signaling interface (trunk) or a gateway be created to accept and originate calls. For SIP, the user must create a signaling interface. For more information, refer to the Trunk Configuration section in the *Cisco CallManager Administration Guide*.

SIP signaling interfaces connect Cisco CallManager networks and SIP networks that are served by a SIP proxy server. As with other protocols, SIP components fit under the device layer of Cisco CallManager architecture. As is true for the H.323 protocol, multiple logical SIP signaling interfaces can be configured in the Cisco CallManager database and associated with route groups, route lists, and

route patterns. To provide redundancy, in the event of failure of one logical SIP interface, other logical SIP interfaces provide services in the same route group list. Redundancy can also be achieved by assigning multiple Cisco CallManager modes under SIP signaling interface device pools.

SIP signaling interfaces use port-based routing, with one SIP signaling interface connecting to a SIP network. Cisco CallManager accepts calls from any SIP device as long as the SIP messages arrive on the configured incoming port. When configuring multiple signaling interfaces, configure a unique incoming port for each SIP interface. Use of the same port as an incoming port for multiple signaling interfaces causes an alarm.

Figure 37-1 SIP and Cisco CallManager Interaction



Related Topics

- [SIP Networks, page 37-1](#)
- [SIP Functions Supported in Cisco CallManager, page 37-4](#)

Media Termination Point (MTP) Devices

Cisco CallManager requires an RFC 2833 dual tone multifrequency (DTMF) compliant MTP device to make SIP calls. The current standard for SIP uses in-band Real-Time Transport Protocol (RTP) payload types to indicate DTMF

tones. AVVID components such as SCCP IP phones, support only out-of-band DTMF payload types. Thus, an RFC 2833 compliant MTP device acts as a translator between inband and out-of-band DTMF.

Related Topics

- [SIP and Cisco CallManager, page 37-2](#)
- [SIP Functions Supported in Cisco CallManager, page 37-4](#)

SIP Functions Supported in Cisco CallManager

Cisco CallManager supports the following functions and features for SIP calls:

- [Basic Calls Between SIP Endpoints and Cisco CallManager, page 37-4](#)
- [DTMF Relay Calls Between SIP Endpoints and Cisco CallManager, page 37-6](#)
- [Supplementary Services Initiated by SCCP Endpoint, page 37-8](#)
- [Supplementary Services Initiated by SIP Endpoint, page 37-9](#)
- [Enhanced Call Identification Services, page 37-10](#)
- [Redirecting Dial Number Identification Service \(RDNIS\), page 37-13](#)
- [SIP Service Parameters, page 37-14](#)

Basic Calls Between SIP Endpoints and Cisco CallManager

This section includes three basic calling scenarios. Two describe incoming and outgoing calls, while the other one describes the use of early media – a media connection prior to the connection or answer of a call.

- [Basic Outgoing Call, page 37-5](#)
- [Basic Incoming Call, page 37-5](#)
- [Use of Early Media, page 37-5](#)

Basic Outgoing Call

You can initiate outgoing calls to a SIP device from any Cisco CallManager device. A Cisco CallManager device includes SCCP IP Phones or fax devices that are connected to Foreign Exchange Station (FXS) gateways. For example, an SCCP IP Phone can place a call to a SIP endpoint. The SIP device answering the call triggers media establishment.

Basic Incoming Call

Any device on the SIP network, including SIP IP Phones or fax devices that are connected to FXS gateways can initiate incoming calls. For example, a SIP endpoint can initiate a call to an SCCP IP Phone. The SCCP IP Phone answering the call triggers media establishment.

Use of Early Media

While the PSTN provides inband progress information to signal early media (such as a ring tone or a busy signal), the same does not hold true for SIP. The originating party includes Session Description Protocol (SDP) information, such as codec usage, IP address, and port number, in the outgoing INVITE message. In response, the terminating party sends its codec, IP address, and port number in a 183 Session Progress message to indicate possible early media.

The 183 Session Progress response indicates that the message body contains information about the media session. Both 180 Alerting and 183 Session Progress messages may contain SDP, which allows an early media session to be established prior to the call being answered.

When early media needs to be delivered to SIP endpoints prior to connection, Cisco CallManager always sends a 183 Session Progress message with SDP. While Cisco CallManager does not generate a 180 Alerting message with SDP, it does support the 180 Alerting message with SDP when it receives one.

Related Topics

- [SIP Functions Supported in Cisco CallManager, page 37-4](#)
- [SIP and Cisco CallManager, page 37-2](#)

DTMF Relay Calls Between SIP Endpoints and Cisco CallManager

Based on RFC 2833, the current standard for SIP uses in-band payload types to indicate DTMF tones. AVVID components such as SCCP IP phones do not support in-band payload types. An RFC 2833 compliant MTP device monitors for payload type and translates between inband and out-of-band payload types.

The following call flows show how Cisco CallManager processes DTMF digits:

- [DTMF Relay Calls Between SIP Endpoints and Cisco CallManager, page 37-6](#)
- [Generating DTMF Digits, page 37-7](#)

Forwarding DTMF Digits from SIP Devices to Gateways or Interactive Voice Response (IVR) Systems

The following example shows the MTP software device processing inband DTMF digits from the SIP Phone to communicate with the Primary Rate Interface (PRI) gateway. The RTP stream carries RFC 2833 DTMF, as indicated by a dynamic payload type.

Figure 37-2 Forwarding DTMF Digits

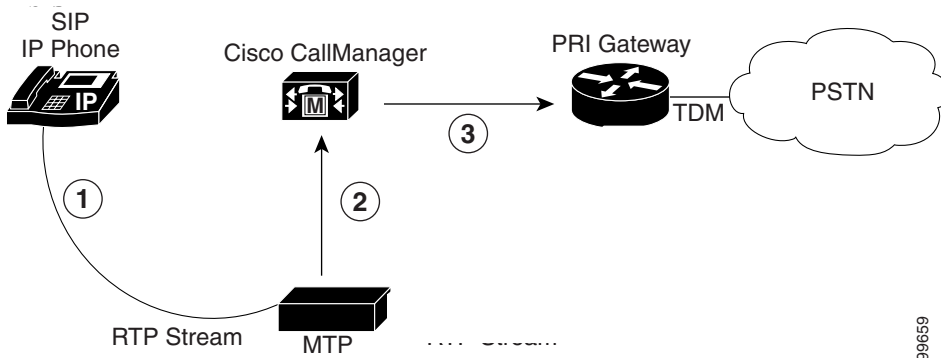


Figure 37-2 begins with media streaming, and the MTP device has been informed of the DTMF dynamic payload type.

1. The SIP Phone initiates a payload type response when the user enters a number on the keypad. The SIP Phone transfers the DTMF in-band digit (per RFC 2833) to the MTP device.
2. The MTP device extracts the in-band DTMF digit and passes the digit out of band to Cisco CallManager.
3. Cisco CallManager then relays the DTMF digit out of band to the gateway or IVR system.

Generating DTMF Digits

As discussed in [DTMF Relay Calls Between SIP Endpoints and Cisco CallManager](#), page 37-6, SIP sends DTMF in-band digits, while Cisco CallManager only supports out-of-band digits. The software MTP device receives the DTMF out-of-band tones and generates DTMF inband tones to the SIP client.

Figure 37-3 Generating DTMF Digits

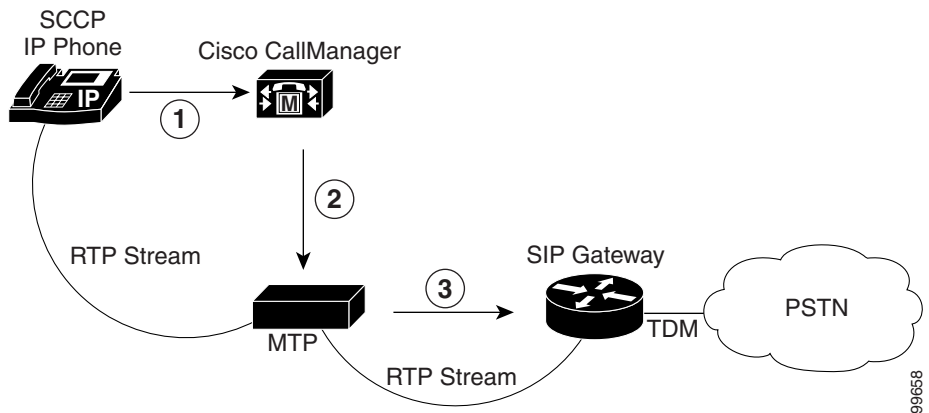


Figure 37-3 begins with media streaming, and the MTP device has been informed of the dynamic DTMF payload type.

1. The SCCP IP Phone user presses buttons on the keypad. Cisco CallManager collects the out-of-band digits from the SCCP IP phone.

2. Cisco CallManager passes the out-of-band digits to the MTP device.
3. The MTP device converts the digits to RFC 2833 RTP compliant inband digits and forwards them to the SIP client.

Related Topics

- [SIP Functions Supported in Cisco CallManager, page 37-4](#)
- [SIP and Cisco CallManager, page 37-2](#)

Supplementary Services Initiated by SCCP Endpoint

All supplementary services initiated by an SCCP endpoint during a SIP call are supported. SCCP endpoints are internally managed by Cisco CallManager without affecting the connecting SIP device. Any changes to the original connecting information are updated with re-INVITE messages that use the Remote-Party-ID header. Refer to *SIP Extensions for Caller Identity and Privacy* for more information on the Remote-Party-ID header.

The following section, [Ringback Tone During Blind Transfer, page 37-8](#), describes a blind transfer, which is unique as a supplementary service because it requires Cisco CallManager to provide a media announcement.

Ringback Tone During Blind Transfer

For SCCP initiated blind transfers, Cisco CallManager needs to generate tones or ringback after a call has already connected. In other words, Cisco CallManager provides a media announcement for blind transfers.

A blind transfer works when the transferring phone connects the caller to a destination line before the target of the transfer enters the call. A blind transfer differs from a consultative, or attended transfer, in which one of the transferring parties either connects the caller to a ringing phone (ringback received) or speaks with the third party before connecting the caller to the third party.

Blind transfers that are initiated from an SCCP IP Phone allow ringback to the original connected SIP device user. To accomplish ringback, Cisco CallManager uses an annunciator software device often located with an MTP device.

With an annunciator, Cisco CallManager can play predefined tones and announcements to SCCP IP Phones, gateways, and other IP telephony devices. These predefined tones and announcements provide users with specific information on the status of the call.

Related Topics

- [SIP Functions Supported in Cisco CallManager, page 37-4](#)
- [SIP and Cisco CallManager, page 37-2](#)

Supplementary Services Initiated by SIP Endpoint

The following sections describe supplementary services initiated by a SIP endpoint.

- [SIP–Initiated Call Transfer, page 37-9](#)
- [Call Hold, page 37-9](#)
- [Call Forward, page 37-10](#)

SIP–Initiated Call Transfer

Cisco CallManager does not support SIP-initiated call transfer and does not accept receiving REFER requests or INVITE messages that include a Replaces header. When Cisco CallManager receives a REFER request, it returns a 501 Not Implemented message. When Cisco CallManager receives an INVITE message with a Replaces header, it processes the call and ignores the Replaces header.

Call Hold

Cisco CallManager supports call hold and retrieve that a SIP device initiates or that a Cisco CallManager device initiates. For example, when a SCCP IP Phone user retrieves a call that was placed on hold by another user, Cisco CallManager sends a re-INVITE message to the SIP proxy. The re-INVITE message contains updated Remote-Party-ID information to reflect the current connected party. If Cisco CallManager originally initiated the call, the Party field in the Remote-Party-ID header gets set to calling; otherwise, it gets set to called. For more information on the Party field parameter, refer to [Enhanced Call Identification Services, page 37-10](#).

Call Forward

Cisco CallManager supports call forward that a SIP device initiates or that a Cisco CallManager device initiates. With call forwarding redirection requests from SIP devices, Cisco CallManager processes the requests. For call forwarding initiated by Cisco CallManager, no SIP redirection messages are used. Cisco CallManager handles redirection internally then conveys the connected party information to the originating SIP endpoint through the Remote-Party-Id header.

Related Topics

- [SIP Functions Supported in Cisco CallManager, page 37-4](#)
- [SIP and Cisco CallManager, page 37-2](#)

Enhanced Call Identification Services

This section describes the following SIP identification services in Cisco CallManager and how Cisco CallManager conveys these identification services in the SIP:

- Line Identification Services
 - Calling Line Presentation (CLIP) and Restriction (CLIR)
 - Connected Line Presentation (COLP) and Restriction (COLR)
- Name Identification Services
 - Calling Name Presentation (CNIP) and Restriction (CNIR)
 - Connected Name Presentation (CONP) and Restriction (CONR)

Cisco CallManager provides flexible configuration options to provide these identification services either on a call-by-call basis or statically preconfigured for each SIP signaling interface.

Calling Line and Name Identification Presentation

Cisco CallManager includes the calling line (or number) and name presentation information in the From and Remote-Party-ID headers of the initial INVITE message from Cisco CallManager. The From header field indicates the initiator of the request. Cisco CallManager uses Remote-Party-ID headers in 18x, 200 and

re-INVITE messages to convey connected name and identification information. The Remote-Party-ID header also gives detailed descriptions of caller identity and privacy. Cisco CallManager sets the Party field of the Remote-Party-ID header to calling for calling ID services.

**Note**

Refer to *SIP Extensions for Caller Identity and Privacy* for more general information on Remote-Party-ID header.

Example:

Bob Jones (with external phone number=8005550100) dials out to a SIP signaling interface; the From and Remote-Party-ID headers contain

```
From: "Bob Jones" <sip:8005550100@localhost>
Remote-Party-ID: "Bob Jones"<8005550100@localhost; user=phone>;
party=calling;screen=no;privacy=off
```

Calling Line and Name Identification Restriction

Calling line (or number) and name restrictions configuration occurs on the SIP signaling interface level or on a call-by-call basis. The SIP trunk level configuration takes precedence over the call-by-call configuration. To configure on a call-by-call basis, refer to the Route Group Configuration in the *Cisco CallManager Administration Guide*.

Calling line and name restrictions configuration also occurs independently of each other. For example, you may choose to restrict only numbers and allow names to be presented.

Example 1

With a restricted calling name, Cisco CallManager sets the calling name in the From header to a configurable string. Cisco CallManager sets the display field in the Remote-Party-ID header to include the actual name but sets the Privacy field to name:

```
From: "Anonymous" <sip:8005550100@localhost>
Remote-Party-ID: "Bob Jones"<9728135001@localhost; user=phone>;
party=calling;screen=no;privacy=name
```

Example 2

With a restricted calling number, Cisco CallManager leaves out the calling line in the From header; however, Cisco CallManager still includes the calling line in the Remote-Party-ID header but sets the Privacy field to `privacy=uri`:

```
From: "Bob Jones" <sip:localhost>
Remote-Party-ID: "Bob Jones"<8005550100@localhost; user=phone>;
party=calling;screen=no;privacy=uri
```

Example 3

With a restricted calling name and number, Cisco CallManager sets the Privacy field to `privacy=full` in the Remote-Party-ID header:

```
From: "Anonymous" <sip:localhost>
Remote-Party-ID: "Bob Jones"<8005550100@localhost; user=phone>;
party=calling;screen=no;privacy=full
```

Connected Line and Name Identification Presentation

Cisco CallManager uses connected line and name identification as a supplementary service to provide the calling party with the connected party's number and name. The From header field indicates the initiator of the request. Cisco CallManager uses Remote-Party-ID headers in 18x, 200 and re-INVITE messages to convey connected information. Cisco CallManager sets the Party field of Remote-Party-ID header to `called`.

Example 1

Cisco CallManager receives an INVITE message with a destination address of 800555. Cisco CallManager includes the connected party name in 18x and 200 messages as follows:

```
Remote-Party-ID: "Bob Jones"<98005550100@localhost; user=phone>;
party=called;screen=no;privacy=off
```

Connected Line and Name Identification Restriction

You can configure connected line (or number) and name restrictions on the SIP trunk level or on a call-by-call basis. The SIP trunk level configuration takes precedence over the call-by-call configuration. To configure on a call-by-call basis, refer to the Route Group Configuration in the *Cisco CallManager Administration Guide*.

Similar to Calling ID services, users can restrict connected number and name independently of each other.

Example 1

Cisco CallManager sets the display field in the Remote-Party-ID header to include the actual name, but sets the Privacy field to `privacy=name`:

```
Remote-Party-ID: "Bob Jones"<8005550100@localhost; user=phone>;  
party=called;screen=no;privacy=name
```

Example 2

With a restricted connected number, Cisco CallManager still includes the connected number in the Remote-Party-ID header but sets the Privacy field to `privacy=uri`:

```
Remote-Party-ID: "Bob Jones"<8005550100@localhost; user=phone>;  
party=called;screen=no;privacy=uri
```

Example 3

With a restricted connected name and number, Cisco CallManager sets the Privacy field to `privacy=full` in the Remote-Party-ID header:

```
Remote-Party-ID: "Bob Jones"<8005550100@localhost; user=phone>;  
party=called;screen=no;privacy=full
```

Related Topics

- [SIP Functions Supported in Cisco CallManager, page 37-4](#)
- [SIP and Cisco CallManager, page 37-2](#)

Redirecting Dial Number Identification Service (RDNIS)

Cisco CallManager uses the SIP Diversion header in the initial INVITE message to carry available RDNIS information.

Related Topics

- [SIP Functions Supported in Cisco CallManager, page 37-4](#)
- [SIP and Cisco CallManager, page 37-2](#)

SIP Service Parameters

SIP timers and counters can be individually configured for functionality on different servers. Refer to the Service Parameter Configuration chapter in the *Cisco CallManager Administration Guide* for full information on how to configure service parameters.

SIP Timers and Counters

SIP timers and counters are configurable service parameters. The following tables describe the various SIP timers and counters and give their default values and range values:

Table 37-1 SIP Timers that are Supported in Cisco CallManager

Timer	Default Value	Default Range	Definition
Trying	500 milliseconds	100 to 1000	Time that Cisco CallManager should wait for a 100 response before retransmitting the INVITE.
Connect	500 milliseconds	100 to 1000	Time that Cisco CallManager should wait for an ACK before retransmitting the 2xx response to the INVITE.
Disconnect	500 milliseconds	100 to 1000	Time that Cisco CallManager should wait for a 2xx response before retransmitting the BYE request.
Expires	180000 milliseconds	60000 to 300000	Valid time that is allowed for an INVITE request.

Table 37-1 SIP Timers that are Supported in Cisco CallManager (continued)

Timer	Default Value	Default Range	Definition
reliablexx	500 milliseconds	100 to 1000	Time that Cisco CallManager should wait before retransmitting the reliable1xx responses.
PRACK	500 milliseconds	100 to 1000	Time that Cisco CallManager should wait before retransmitting the PRACK request.

**Note**

When using TCP transport and a timer times out, the SIP device does not retransmit. The device relies on TCP to retry.

Table 37-2 SIP Retry Counters that are Supported in Cisco CallManager

Retry Counter	Default Value	Default Range	Definition
INVITE	5	1 to 10	Number of INVITE retries
Response	6	1 to 10	Number of RESPONSE retries
BYE	10	1 to 10	Number of BYE retries
Cancel	10	1 to 10	Number of Cancel retries
PRACK	6	1 to 10	Number of PRACK retries
Rel1xx	10	1 to 10	Number of Reliable 1xx response retries

Related Topics

- [SIP Functions Supported in Cisco CallManager, page 37-4](#)
- [SIP and Cisco CallManager, page 37-2](#)

SIP Signaling/Trunk Interface Configuration Checklist

[Table 37-3](#) provides an overview of the steps that are required to configure SIP signaling/trunk interfaces in Cisco CallManager, along with references to related procedures and topics.

Table 37-3 Trunk Configuration Checklist

Configuration Steps	Procedures and Related Topics
<p>Step 1 Create a SIP trunk.</p> <p>For outgoing calls, configure the destination address (the address of the SIP Proxy Server).</p> <p>Configure the destination port.</p> <p>Configure a unique incoming port for each SIP interface.</p>	<p>Adding a Trunk, <i>Cisco CallManager Administration Guide</i></p> <p>Trunk Configuration Settings, <i>Cisco CallManager Administration Guide</i></p>
<p>Step 2 Verify the RFC 2833 compliant MTP device is configured.</p>	<p>Trunk Configuration Settings, <i>Cisco CallManager Administration Guide</i></p> <p>In the trunk configuration, the MTP field is always checked. SIP requires an RFC 2833 compliant MTP device. For more information on MTP, see Media Termination Point Configuration, <i>Cisco CallManager Administration Guide</i>.</p>
<p>Step 3 Assign to a Route Pattern, Route Group, or Route List, if needed.</p>	<p>Route Pattern/Hunt Pilot Configuration, <i>Cisco CallManager Administration Guide</i></p> <p>Route Group Configuration, <i>Cisco CallManager Administration Guide</i></p> <p>Route/Hunt List Configuration, <i>Cisco CallManager Administration Guide</i></p>

Table 37-3 Trunk Configuration Checklist (continued)

Configuration Steps	Procedures and Related Topics
Step 4 Configure SIP timers, counters, and service parameters, if necessary.	Service Parameters Configuration , <i>Cisco CallManager Administration Guide</i> . For specific configurable values, see SIP Service Parameters , page 37-14.
Step 5 Verify the Annunciator is active, if necessary.	Annunciator Configuration , <i>Cisco CallManager Administration Guide</i> .
Step 6 If a SIP Proxy server is used as the destination address, configure static routes to point to all IP addresses or domain names of the SIP interface Call Manager Group.	Trunk Configuration Settings , <i>Cisco CallManager Administration Guide</i> The device pool field can be an IP address, fully-qualified domain name (FQDN), or DNS SRV name.

Related Topics

- [SIP Networks](#), page 37-1
- [SIP and Cisco CallManager](#), page 37-2
- [SIP Functions Supported in Cisco CallManager](#), page 37-4
- [SIP Signaling/Trunk Interface Configuration Checklist](#), page 37-16

Where to Find More Information

Related Topics

- [Understanding Cisco CallManager Trunk Types](#), page 38-1
- [Trunk Configuration](#), *Cisco CallManager Administration Guide*
- [Understanding IP Telephony Protocols](#), page 36-1
- [Caller Identification and Restriction](#), page 14-36

Additional Cisco Documentation

- *Cisco IP Telephony Solution Reference Network Design*