Session Initiation Protocol for Voice over IP Feature for Cisco uBR925 Cable Access Router and Cisco CVA122 Cable Voice Adapter

Feature History

Release	Modification
12.2(11)T	Support for the Session Initiation Protocol for Voice over IP feature was introduced for Cisco uBR925 cable access routers and Cisco CVA122 Cable Voice Adapters.

This document describes the Session Initiation Protocol for Voice over IP feature for Cisco uBR925 cable access routers and Cisco CVA122 Cable Voice Adapters in Cisco IOS Release 12.2(11)T. This document provides information on configuring the Session Initiation Protocol for Voice over IP feature to enable the setup of voice and multimedia calls across Internet Protocol (IP) networks.

This document includes the following sections:

Feature Overview

The Cisco Session Initiation Protocol (SIP) functionality enables Cisco access platforms to signal the setup of voice and multimedia calls over IP networks.

SIP is an ASCII-based, application-layer control protocol that can be used to establish, maintain, and terminate calls between two or more endpoints. SIP is an alternative protocol developed by the Internet Engineering Task Force (IETF) for multimedia conferencing over IP. SIP features are compliant with IETF RFC 2543, *SIP: Session Initiation Protocol*, published in March 1999. You can view RFC 2543 at http://www.ietf.org/rfc/rfc2543.txt.

Like other Voice-over-IP protocols, SIP is designed to address the functions of signaling and session management within a packet telephony network. Signaling allows call information to be carried across network boundaries. Session management provides the ability to control the attributes of an end-to-end call.

SIP provides the following capabilities:

- Determines the location of the target endpoint—SIP supports address resolution, name mapping, and call redirection.
- Determines the media capabilities of the target endpoint—SIP determines the lowest level of common services between the endpoints through Session Description Protocol (SDP). Conferences are established using only the media capabilities that can be supported by all endpoints.
- Determines the availability of the target endpoint—If a call cannot be completed because the target endpoint is unavailable, SIP determines whether the called party is connected to a call already or did not answer in the allotted number of rings. SIP then returns a message indicating why the target endpoint was unavailable.
- Establishes a session between the originating and target endpoints—If the call can be completed, SIP establishes a session between the endpoints. SIP also supports midcall changes, such as the addition of another endpoint to the conference or the changing of a media characteristic or codec.
- Handles the transfer and termination of calls—SIP supports the transfer of calls from one endpoint to another. During a call transfer, SIP simply establishes a session between the transferee and a new endpoint (specified by the transferring party) and terminates the session between the transferee and the transferring party. At the end of a call, SIP terminates the sessions among all parties.

Note The term conference means an established session (or call) between two or more endpoints. Conferences consist of two or more users and can be established using multicast or multiple unicast sessions.

Components of SIP

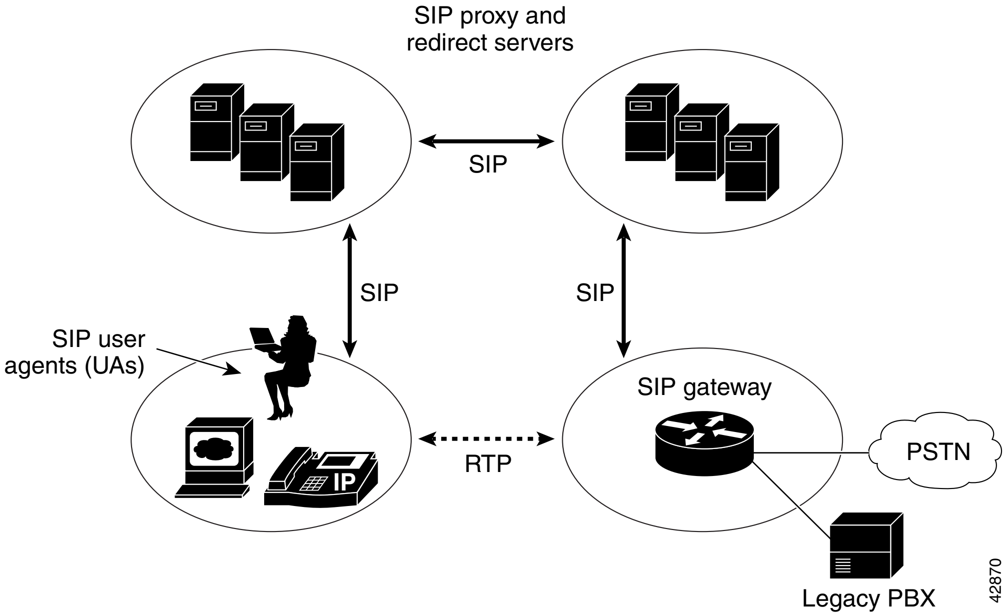
SIP is a peer-to-peer protocol. The peers in a session are called user agents (UAs). A user agent can function in one of the following roles:

- User agent client (UAC)—A client application that initiates the SIP request.
- User agent server (UAS)—A server application that contacts the user when a SIP request is received and that returns a response on behalf of the user.

Typically, a SIP endpoint is capable of functioning as both a UAC and a UAS, but functions only as one or the other per transaction. Whether the endpoint functions as a UAC or a UAS depends on the UA that initiated the request.

From an architectural standpoint, the physical components of a SIP network can be grouped into two categories: clients and servers. Figure 1 illustrates the architecture of a SIP network.

Figure 1 SIP Architecture



Note The SIP servers can interact with other application services, such as Lightweight Directory Access Protocol (LDAP) servers, location servers, a database application, or an extensible markup language (XML) application. These application services provide back-end services such as directory, authentication, and billing services.

SIP Clients

SIP clients include the following:

- Phones—Can act as either a UAS or UAC. SoftPhones (PCs that have phone capabilities installed) and Cisco SIP IP phones can initiate SIP requests and respond to requests.
- Gateways—Provide call control. Gateways provide many services, the most common being a translation function between SIP conferencing endpoints and other terminal types. This function includes translation between transmission formats and between communications procedures. In addition, the gateway translates between audio and video codecs and performs call setup and clearing on both the LAN side and the switched-circuit network side.

SIP Servers

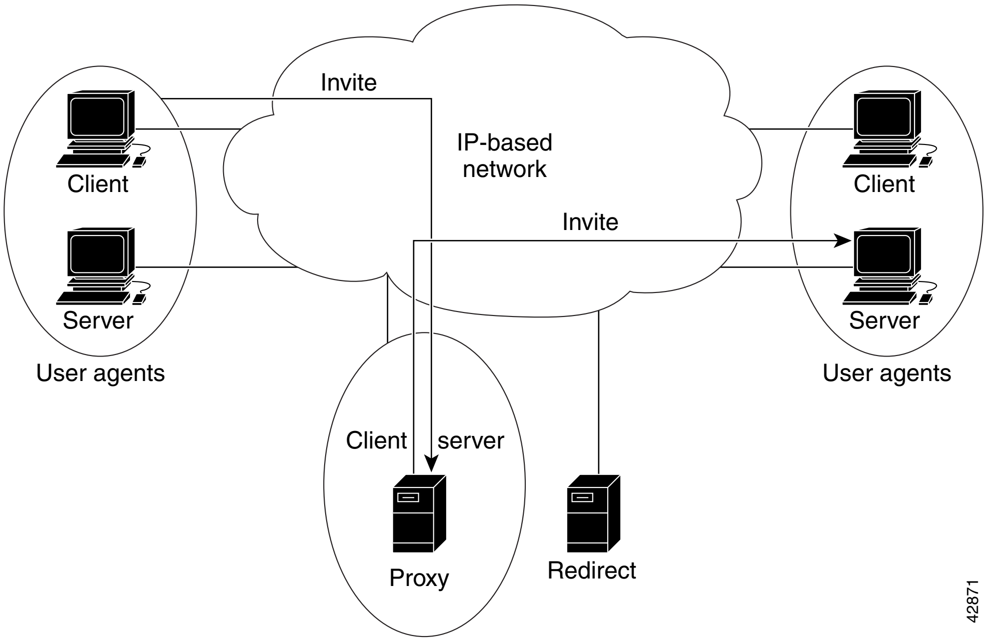
SIP servers include the following:

- Proxy server—Receives SIP messages and forwards them to the next SIP server in the network. The proxy server is an intermediate device that receives SIP requests from a client and then forwards the requests on behalf of the client. Proxy servers can provide functions such as authentication, authorization, network access control, routing, reliable request retransmission, and security.
- Redirect server—Provides the client with information about the next hop or hops that a message should take. The client then contacts the next hop server or UAS directly.
- Registrar server—Processes requests from UACs for registration of their current location. Registrar servers are often located near a redirect or proxy server.

Using a Proxy Server

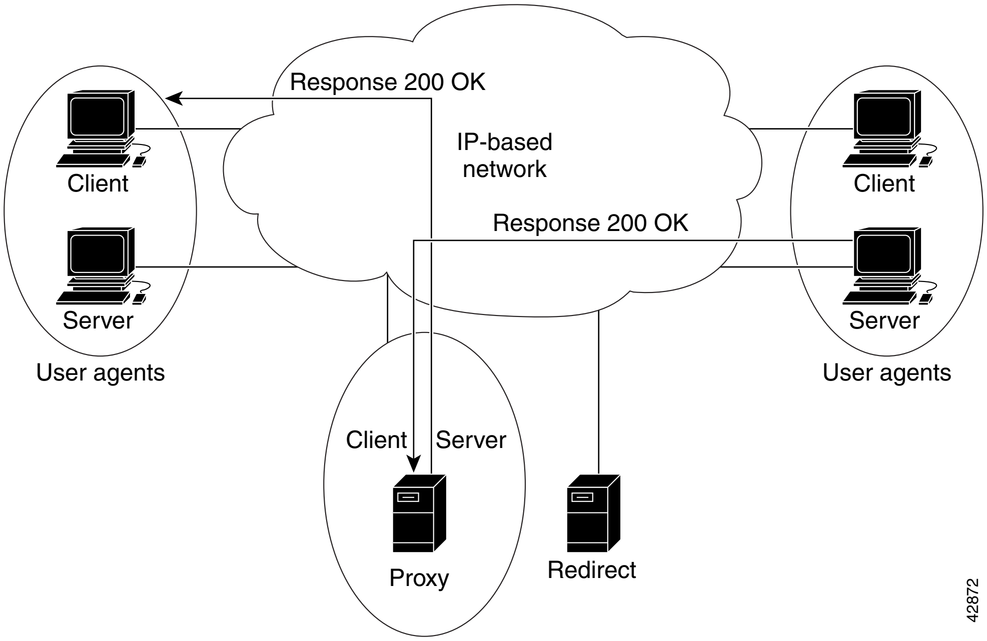
If a proxy server is used, the caller UA sends an INVITE request to the proxy server. The proxy server determines the path and then forwards the request to the callee, as shown in Figure 2.

Figure 2 SIP Request Through a Proxy Server



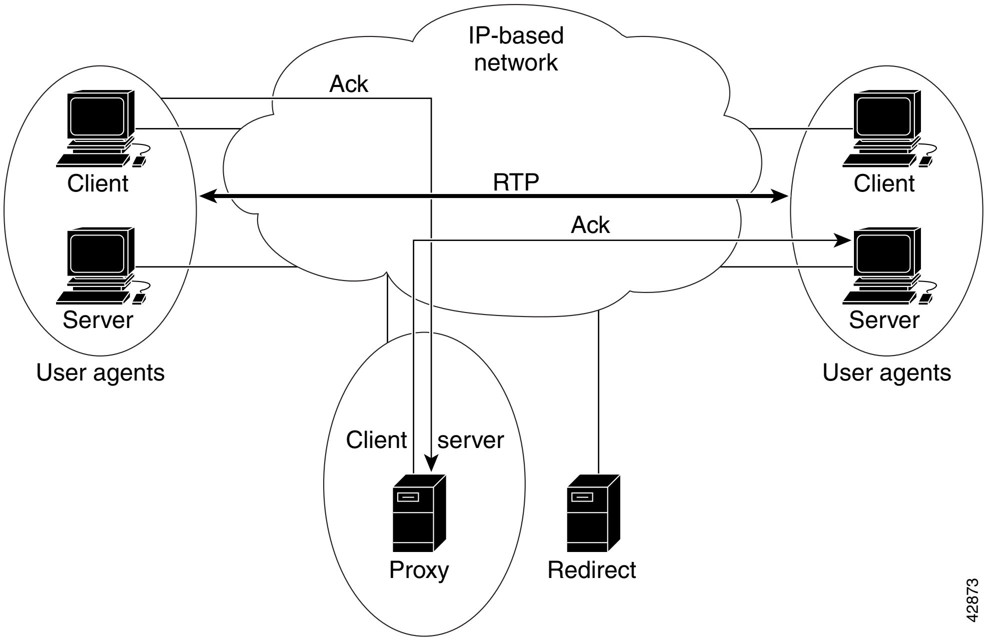
The callee responds to the proxy server, which in turn forwards the response to the caller, as shown in Figure 3.

Figure 3 SIP Response Through a Proxy Server



The proxy server forwards the acknowledgments of both parties. A session is then established between the caller and callee. Real-Time Transfer Protocol (RTP) is used for the communication between the caller and the callee, as shown in Figure 4.

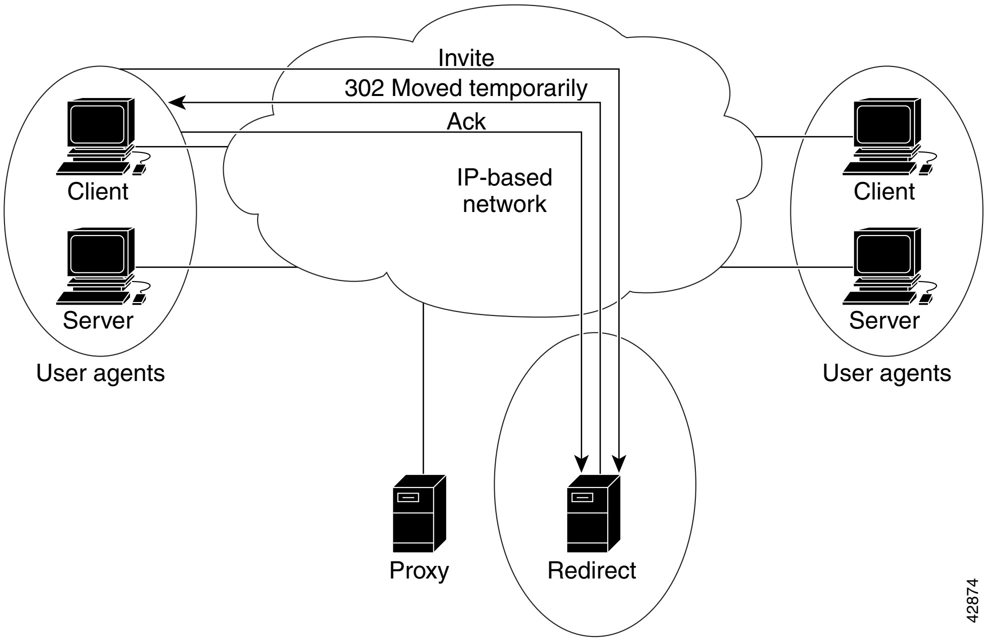
Figure 4 SIP Session Through a Proxy Server



Using a Redirect Server

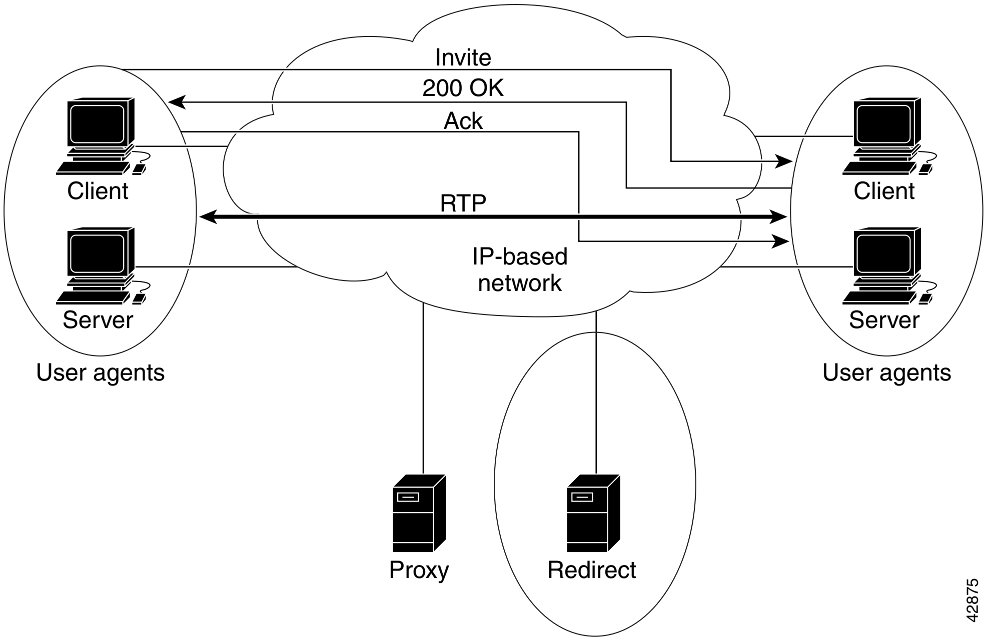
If a redirect server is used, the caller UA sends an INVITE request to the redirect server. The redirect server contacts the location server to determine the path to the callee, and the redirect server sends that information back to the caller. The caller then acknowledges receipt of the information, as shown in Figure 5.

Figure 5 SIP Request Through a Redirect Server



The caller then sends a request to the device indicated in the redirection information (which could be the callee or another server that will forward the request). Once the request reaches the callee, it sends back a response, and the caller acknowledges the response. RTP is used for the communication between the caller and the callee, as shown in Figure 6.

Figure 6 SIP Session Through a Redirect Server



SIP Enhancements

SIP provides the following feature enhancements over other voice signaling protocols:

- Ability to specify the maximum number of SIP redirects.
- Ability to specify SIP or H.323 on a dial-peer basis.
- Configurable SIP message timers and retries.
- Interoperability with unified call services (UCS).
- Support for a variety of signaling protocols, including Integrated Services Digital Network (ISDN), Primary Rate Interface (PRI), and channel-associated signaling (CAS).
- Support for a variety of interfaces, including
  - Analog interfaces: Foreign Exchange Station (FXS)/Foreign Exchange Office (FXO)/recEive and transMit (E&M) analog interfaces.
  - Digital interfaces: T1 CAS, T1 PRI, E1 CAS, E1 PRI, and E1 R2
- Support for SIP redirection messages and interaction with SIP proxies. The gateway can redirect an unanswered call to another SIP gateway or SIP-enabled IP phone. In addition, the gateway supports proxy-routed calls.
- Interoperability with Domain Name System (DNS) servers, including support for DNS SRV and “A” records to look up SIP URLs according to RFC 2052 formatting.
- Support for SIP over TCP and User Datagram Protocol (UDP).

- Support RTP/RTCP for media transport in VoIP networks.
- Support for the following codecs:
  - G711 u-law
  - G711 a-law
  - G723r63
  - G726r32
  - G728
  - G729r8
- Support for record-route headers.
- Support for IP quality of service (QoS) and IP precedence.
- Support for IP Security (IPSec) for SIP signaling messages.
- Authentication, authorization, and accounting (AAA) support. For accounting, the gateway device generates call data record (CDR) accounting records for export. For authentication, the SIP gateway sends validation requests to the AAA server. For authorization, the existing access lists are used.
- Support for call hold and call transfer features. The call hold sends a midcall INVITE message, which requests that the remote endpoint stop sending media streams. The call transfer is done without consultation (blind transfer). The transfer can be initiated by a remote SIP endpoint.
- Support for configurable expiration time for SIP INVITEs and maximum number of proxies or redirect servers that can forward a SIP request.
- Ability to hide the identity of the calling party by setting the ISDN presentation indicator.

Benefits

The SIP feature provides nonproprietary advantages in the following areas:

- Protocol extensibility
- System scalability
- Personal mobility services
- Interoperability with different vendors

Related Documents

For a complete description of the commands used in this chapter, refer to the Cisco IOS Voice, Video, and Fax Command Reference, Release 12.2.

For more information on cable-specific commands, see the Cisco IOS CMTS Cable Command Reference.

Supported Platforms

- Cisco uBR925
- Cisco CVA122

Determining Platform Support Through Cisco Feature Navigator

Cisco IOS software is packaged in feature sets that support specific platforms. To get updated information regarding platform support for this feature, access the Cisco Feature Navigator. The Cisco Feature Navigator dynamically updates the list of supported platforms as new platform support is added for the feature.

The Cisco Feature Navigator is updated regularly when major Cisco IOS software releases and technology releases occur. For the most current information, go to the Cisco Feature Navigator home page at the following URL:

http://www.cisco.com/go/fn

Supported Standards, MIBs, and RFCs

Standards

DOCSIS 1.0 specification SP-RFI-I05-991105

DOCSIS 1.1 specification SP-RFIv1.1-IO3-991105

MIBs

No new or modified MIBs are supported by this feature.

To obtain lists of supported MIBs by platform and Cisco IOS release, and to download MIB modules, go to the Cisco MIB website on Cisco.com at the following URL:

http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFC 2543, *SIP: Session Initiation Protocol*

Prerequisites

Before you configure your router with the SIP feature, you must perform the following tasks:

- Configure your gateway to support voice functionality for SIP or H.323.
- Establish a working IP network.
- Configure VoIP.
- Ensure that your router has a minimum of 16 MB Flash memory and 64 MB DRAM memory.

Configuration Tasks

See the following sections for configuration tasks for the Session Initiation Protocol for Voice over IP feature. Each task in the list is identified as either required or optional.

To configure SIP functions on the Cisco uBR925 and the Cisco CVA122, perform the following tasks:

- Configuring SIP Support for VoIP Dial Peers, page 10 (Required)
- Enabling the SIP User Agent, page 11 (Optional)
- Making a Simple SIP Call, page 12 (Optional)

- Configuring SIP Call Transfer, page 13 (Optional)
- Configuring Gateway Accounting, page 13 (Optional)

Configuring SIP Support for VoIP Dial Peers

To configure SIP support for a VoIP dial peer, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)#` **`dial-peer voice`** *`number`* **`voip`**	Enters dial-peer configuration mode to configure a VoIP dial peer.
Step 2	`Router(config-dial-peer)#` **`session transport`** `{`**`udp`** `|` **`tcp`**`}`	(Optional) Enters the session transport type for the SIP user agent. The default is **udp**. The transport protocol (**udp** or **tcp**) specified with the **session transport** command must be identical to the protocol specified with the **transport** command.
Step 3	`Router(config-dial-peer)#` **`session protocol`** `{`**`cisco`** `|` **`sipv2`**`}`	Enters the session protocol type. The keywords are as follows: • **cisco**—Configures the dial peer to use proprietary Cisco VoIP session protocol. • **sipv2**—Configures the dial peer to use IETF SIP. SIP users should use this option.

	Command	Purpose
Step 4	`Router(config-sip-ua)# sip-server {dns:[hostname] | ipv4:ip_addr:[port-num]}`	In SIP user agent (sip-ua) configuration mode. Enters the host name or IP address of the SIP server interface. If you use this command, you can then specify **session target sip-server** for each dial peer instead of repeatedly entering the SIP server interface address for each dial peer. The keywords and arguments are as follows: • **dns:***hostname*—Sets the global SIP server interface to a Domain Name Server (DNS) host name. A valid DNS host name takes the following format: *name.gateway.xyz.* • **ipv4:***ip_addr:*—Sets the IP address. • *port-num*—(Optional) Sets the UDP port number for the SIP server.
Step 5	`Router(config-dial-peer)# session target {sip-server | dns:[$s$.] | $d$. | $e$. | $u$. [hostname]| ipv4:ip_addr:[port-num]}`	In dial-peer configuration mode. Specifies a network-specific address for a dial peer. The keywords and arguments are as follows: • **sip-server**—Sets the session target to the global SIP server. Used when the **sip-server** command has already specified the host name or IP address of the SIP server interface. • **dns:***hostname*—Sets the global SIP server interface to a Domain Name Server (DNS) host name. A valid DNS host name takes the following format: *name.gateway.xyz.* • **ipv4:***ip_addr:*—Sets the IP address. • *port-num*—(Optional) Sets the UDP port number for the SIP server. **Note** Wildcards can be used when defining the session target for VoIP peers.

Enabling the SIP User Agent

To place a call, you must enable a SIP user agent (UA) with the **sip-ua** command in global configuration mode. When in sip-ua configuration mode, you can optionally adjust any of the user agent configuration settings. Some of the optional adjustment settings are described in the steps below.

	Command	Purpose
Step 1	`Router(config)# sip-ua`	Enables a SIP user agent (UA). Also enters the SIP user agent (sip-ua) configuration mode to allow you to configure optional sip-ua related commands.
Step 2	`Router(config-sip-ua)# transport {udp | tcp}`	(Optional) Configures the SIP user agent (sip-ua) for SIP signaling messages. The default is **udp**. The transport protocol (**udp** or **tcp**) specified with the **session transport** command must be identical to the protocol specified with the **transport** command.

	Command	Purpose
Step 3	Router(config-sip-ua)# **timers** {**trying** *number* | **connect** *number* | **disconnect** *number* | **expires** *number*}	(Optional) Configures the SIP signaling timers. The keywords are as follows: • **trying**—Sets the time to wait for a 100 response to an INVITE request. The default is 500. • **connect**—Sets the time to wait for a 200 response to an ACK request. The default is 500. • **disconnect**—Sets the time to wait for a 200 response to a BYE request. The default is 500. • **expires**—Limits the time duration (in milliseconds) for which an INVITE is valid. The default is 180000.
Step 4	Router(config-sip-ua)# **retry** {**invite** *number* | **response** *number* | **bye** *number* | **cancel** *number*}	(Optional) Configures the SIP signaling timers for retry attempts. The keywords are as follows: • **invite**—Number of INVITE retries. The default is 6. • **respons**e—Number of RESPONSE retries. The default is 6. • **bye**—Number of BYE retries. The default is 10. • **cancel**—Number of cancel retries. The default is 10.
Step 5	Router(config-sip-ua)# **max-forwards** *number*	(Optional) Limits the number of proxy or redirect servers that can forward a request. The default is 6.
Step 6	Router(config-sip-ua)# **max-redirects** *number*	(Optional) Sets the maximum number of redirect servers. The default is 1.
Step 7	Router(config-sip-ua)# **default** {**max-forwards** | **retry** {**invite** | **response** | **bye** | **cancel**} | **sip-server** | **timers** {**trying** | **connect** | **disconnect** | **expires**} | **transport**}	(Optional) Resets the value of a SIP user agent command to its default.

Making a Simple SIP Call

To make a simple SIP call, you must use the **dial-peer voice pots** command in global configuration mode.

	Command	Purpose
Step 1	Router(config)# **dial-peer voice** *number* **pots**	Enters dial-peer configuration mode to make a simple SIP call.
Step 2	Router(config-dial-peer)# **destination-pattern** *pattern*	Specifies the telephone number associated with the dial peer.
Step 3	Router(config-dial-peer)# **port** {*slot-number/subunit-number/port*} | {*slot/port:ds0-group-no*}	Specifies the local voice port through which incoming VoIP calls will be received.

Configuring SIP Call Transfer

To configure SIP call transfer for a POTS dial peer, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# dial-peer voice number pots`	Enters dial-peer configuration mode to configure a POTS dial peer.
Step 2	`Router(config-dial-peer)# application session`	Specifies that the standard session application will be invoked for this dial peer.
Step 3	`Router(config-dial-peer)# destination-pattern pattern`	Specifies the telephone number associated with the dial peer.
Step 4	`Router(config-dial-peer)# port number`	Specifies the local voice port through which incoming VoIP calls will be received. The port number may be 0 or 1.

To configure SIP call transfer for a VoIP dial peer, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# dial-peer voice number voip`	Enters dial-peer mode to configure a VoIP dial peer.
Step 2	`Router(config-dial-peer)# destination-pattern pattern`	Specifies the telephone number associated with the dial peer.
Step 3	`Router(config-dial-peer)# session protocol sipv2`	Enters the session protocol type. Select **sipv2** as the keyword. • **sipv2**—Configures the dial peer to use IETF SIP. SIP users should use this option.
Step 4	`Router(config-dial-peer)# session target ipv4:x.x.x.x`	Specifies the IP address of the destination gateway for outbound dial peers. It is useful when you make a call from one cable modem to others. This is optional for making a simple SIP call.

Configuring Gateway Accounting

Three keywords configure gateway accounting for SIP:

- The **voip** keyword sends the call data record (CDR) to the RADIUS server. Use this keyword with the SIP feature.
- The **H323** keyword sends the CDR to the RADIUS server.
- The **syslog** keyword uses the system logging facility to record the CDRs.

To enable gateway-specific accounting for SIP, use the following command in global configuration mode:

Command	Purpose
`Router(config)# gw-accounting {voip	syslog	h323 [syslog]}`	(Optional) Enables gateway-specific accounting in global configuration mode.

Configuration Examples

This section provides the following configuration examples:

Basic SIP Configuration Example

The following shows a basic SIP configuration. This output was created by using the **show running-config** command.

```
router# show running-config

Building configuration...
Current configuration : 1241 bytes
!
! Last configuration change at 15:38:15 - Fri Feb 22 2002
!
version 12.2
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname router
!
!
!
!
clock timezone - 0 1
ip subnet-zero
no ip routing
ip domain-name cisco.com
!
!
!
!
!
!
!
!
!
!
!
interface Ethernet0
ip address 188.199.0.34 255.255.0.0
no ip route-cache
bridge-group 59
bridge-group 59 spanning-disabled
!
interface cable-modem0
no ip route-cache
cable-modem boot admin 2
cable-modem boot oper 5
bridge-group 59
bridge-group 59 spanning-disabled
!
interface usb0
ip address 188.199.0.34 255.255.0.0
```

```
no ip route-cache
arp timeout 0
bridge-group 59
bridge-group 59 spanning-disabled
!
ip classless
ip pim bidir-enable
ip http server
no ip http cable-monitor
!
snmp-server manager
call rsvp-sync
!
voice-port 0
input gain -2
ren 0
!
voice-port 1
input gain -2
ren 0
!
!
mgcp profile default
!
dial-peer voice 100 pots
destination-pattern 8886618
port 0
!
dial-peer voice 101 voip
destination-pattern 888662.
session protocol sipv2
session target ipv4:188.199.0.35
!
dial-peer voice 102 pots
destination-pattern 8886619
port 1
!
sip-ua
!
!
line con 0
line vty 0 4
!
scheduler max-task-time 5000
end
```

Verifying SIP Configuration Example

Enter the **show running-config** command to verify your configuration, or use the **show sip-ua** subcommands to verify the SIP configurations.

The following example shows sample output for the **show sip-ua statistics** command:

```
Router# show sip-ua statistics

SIP Response Statistics (Inbound/Outbound)
    Informational:
      Trying 0/0, Ringing 0/0,
      Forwarded 0/0, Queued 0/0,
      SessionProgress 0/0
    Success:
       OkInvite 0/0, OkBye 0/0,
         OkCancel 0/0, OkOptions 0/0
      Redirection (Inbound only):
        MultipleChoice 0, MovedPermanently 0,
        MovedTemporarily 0, SeeOther 0,
        UseProxy 0, AlternateService 0
      Client Error:
        BadRequest 0/0, Unauthorized 0/0,
        PaymentRequired 0/0, Forbidden 0/0,
        NotFound 0/0, MethodNotAllowed 0/0,
        NotAcceptable 0/0, ProxyAuthReqd 0/0,
        ReqTimeout 0/0, Conflict 0/0, Gone 0/0,
        LengthRequired 0/0, ReqEntityTooLarge 0/0,
        ReqURITooLarge 0/0, UnsupportedMediaType 0/0,
        BadExtension 0/0, TempNotAvailable 0/0,
        CallLegNonExistent 0/0, LoopDetected 0/0,
        TooManyHops 0/0, AddrIncomplete 0/0,
        Ambiguous 0/0, BusyHere 0/0
      Server Error:
        InternalError 0/0, NotImplemented 0/0,
        BadGateway 0/0, ServiceUnavail 0/0,
        GatewayTimeout 0/0, BadSipVer 0/0
      Global Failure:
        BusyEverywhere 0/0, Decline 0/0,
        NoExistAnywhere 0/0, NotAcceptable 0/0

SIP Total Traffic Statistics (Inbound/Outbound)
    Invite 0/0, Ack 0/0, Bye 0/0,
    Cancel 0/0, Options 0/0

Retry Statistics
    Invite 0, Bye 0, Cancel 0, Response 0
```

The following example shows sample output for the **show sip-ua status** command:

```
Router# show sip-ua status

SIP User Agent Status
SIP User Agent for UDP : ENABLED
SIP User Agent for TCP : ENABLED
SIP max-forwards :6
```

The following example shows sample output for the **show sip-ua timers** command:

```
Router# show sip-ua timers

SIP UA Timer Values (millisecs)
trying 500, expires 180000, connect 500, disconnect 500
```