



Call Transfer Capabilities Using the Refer Method

Document Update Alert

This document was originally produced for Cisco IOS Release 12.2(11)T. This feature has been updated in subsequent releases, and more recent documentation is available.

If you are using Cisco IOS Release 12.2(11)T or higher, refer to the following chapter of the *Cisco IOS SIP Configuration Guide* in the Cisco IOS Voice Configuration Library, Release 12.3:

- [Configuring SIP Call Transfer](#)
-

Feature History

Release	Modification
12.2(2)XB	This feature was introduced on the Cisco 2600 series, Cisco 3600 series, Cisco 7200 series, Cisco AS5300, Cisco AS5350, and Cisco AS5400 platforms.
12.2(2)XB2	This feature was implemented on the Cisco AS5850 platform.
12.2(8)T	This feature was integrated into Cisco IOS Release 12.2(8)T. The Cisco AS5300, Cisco AS5350, Cisco AS5850, and Cisco AS5400 platforms were not supported in this release.
12.2(11)T	This feature was integrated into Cisco IOS Release 12.2(11)T and support was added for the Cisco AS5300, Cisco AS5350, Cisco AS5400, and Cisco AS5850 platforms.

This document describes the Call Transfer Capabilities Using the Refer Method feature. The Refer method provides call transfer capabilities to supplement the Bye and Also methods already implemented on Cisco IOS Session Initiation Protocol (SIP) gateways.

- [Feature Overview, page 2](#)
- [Supported Platforms, page 12](#)
- [Supported Standards, MIBs, and RFCs, page 13](#)
- [Prerequisites, page 14](#)
- [Configuration Tasks, page 14](#)

- [Configuration Examples, page 18](#)
- [Command Reference, page 21](#)
- [Glossary, page 34](#)

Feature Overview

This document describes Call Transfer Capabilities Using the Refer Method in Cisco IOS Release 12.2(11)T. The Refer method provides call transfer capabilities to supplement the Bye and Also methods already implemented on Cisco IOS Session Initiation Protocol (SIP) gateways.

Call transfer allows a wide variety of decentralized multiparty call operations. These decentralized call operations form the basis for third-party call control, and thus are important features for Voice over IP (VoIP) and SIP. Call transfer is also critical for conference calling, where calls can transition smoothly between multiple point-to-point links and IP level multicasting.

The following components described in this document are aspects of call transfer:

- [Refer Method, page 2](#)
- [Refer-To Header, page 3](#)
- [Referred-By Header, page 4](#)
- [Notify Method, page 4](#)
- [Using the Refer Method to Achieve Call Transfer, page 5](#)
- [Blind Transfer, page 5](#)
- [Attended Transfer, page 9](#)

Refer Method

The SIP Refer method provides call transfer capabilities to supplement the Bye and Also methods already implemented on Cisco IOS Session Initiation Protocol (SIP) gateways. The Refer method has three main roles:

- **Originator**—User agent that initiates the transfer or Refer request.
- **Recipient**—User agent that receives the Refer request and is transferred to the final-recipient.
- **Final-Recipient**—User agent introduced into a call with the recipient.



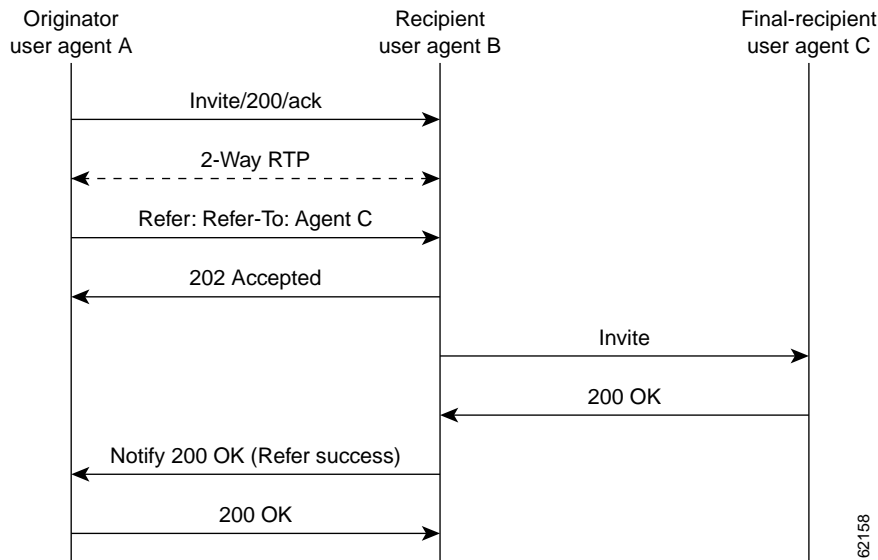
Note

A gateway can be a recipient or final-recipient; but not an originator.

The Refer method always begins within the context of an existing call and starts with the *originator*. The originator sends a Refer request to the *recipient* (user agent receiving the Refer request) to initiate a triggered Invite request. The triggered Invite request uses the SIP URL contained in the Refer-To header as the destination of the Invite request. The recipient then contacts the resource in the Refer-To header (*final-recipient*), and returns a SIP 202 (Accepted) response to the originator. The recipient also must notify the originator of the outcome of the Refer transaction—whether the final-recipient was successfully or unsuccessfully contacted. The notification is accomplished using the [Notify Method](#), SIP's event notification mechanism. A Notify message with a message body of SIP 200 OK indicates a successful transfer, while a body of SIP 503 Service Unavailable indicates an unsuccessful transfer. If the call was successful, a call between the recipient and the final-recipient results.

Figure 1 represents the call flow of a successful Refer transaction initiated within the context of an existing call.

Figure 1 Successful Refer Transaction



Refer-To Header

The recipient receives from the originator a Refer request that always contains a single Refer-to header. The Refer-to header includes a SIP URL that indicates the party to invite and must be in SIP URL format.



Note

The TEL URL format cannot be used in a Refer-to header, because it does not provide a host portion, and without one, the triggered Invite request cannot be routed.

The Refer-To header may contain three additional overloaded headers to form the triggered Invite request. If any of these three headers are present, they are included in the triggered Invite request. The three headers are:

- **Accept-Contact**—Optional in a Refer request. A SIP IOS gateway that receives an Invite request with an Accept-Contact does not act upon this header. This header is defined in draft-ietf-sip-callerprefs-03.txt and may be used by user agents that support caller preferences.
- **Proxy-Authorization**—A nonstandard header that SIP gateways do not act on. It is echoed in the triggered Invite request because proxies occasionally require it for billing purposes.
- **Replaces**—The Replaces header is used by SIP gateways to indicate whether the originator of the Refer request is requesting a blind or attended transfer. It is required if the originator is performing an attended transfer, and not required for a blind transfer. See the [“Replaces Header” section on page 9](#) for more information.

All other headers present in the Refer-To are ignored, and are not sent in the triggered invite.

**Note**

The Refer-To and Contact headers are required in the Refer request. The absence of these headers results in a 4xx class response to the Refer request. Also, the Refer request must contain exactly one Refer-To header. Multiple Refer-To headers result in a 4xx class response.

Referred-By Header

The Referred-By header is required in a Refer request. It identifies the originator and may also contain a signature (included for security purposes). SIP gateways echo the contents of the Referred-By header in the triggered Invite request, but on receiving an Invite request with this header, gateways do not act on it.

**Note**

The Referred-By header is required in a Refer request. The absence of this header results in a 4xx class response to the Refer request. Also, the Refer request must contain exactly one Referred-By header. Multiple Referred-By headers result in a 4xx class response.

Notify Method

Once the outcome of the Refer transaction is known, the recipient of the Refer request must notify the originator of the outcome of the Refer transaction—whether the final-recipient was successfully or unsuccessfully contacted. The notification is accomplished using the Notify method, SIP's event notification mechanism. The notification contains a message body with a SIP response status line and the response class in the status line indicates the success or failure of the Refer transaction.

The Notify message must:

- Reflect the same To, From, and Call-ID headers that were received in the Refer request.
- Contain an Event header refer.
- Contain a message body with a SIP response line. For example: SIP/2.0 200 OK to report a successful Refer transaction, or SIP/2.0 503 Service Unavailable to report a failure. To report that the recipient disconnected before the transfer finished, it must use SIP/2.0 487 Request Canceled.

Two new Cisco IOS commands pertain to the Notify method.

- The **timers notify** command sets the amount of time that the recipient should wait before retransmitting a Notify message to the originator.
- The **retry notify** command configures the number of times a Notify message is retransmitted to the originator.

Using the Refer Method to Achieve Call Transfer

This section discusses how the Refer method facilitates call transfer. There are two types of call transfers:

- [Blind Transfer](#) (or Unattended transfer)
- [Attended Transfer](#)

The main difference between the two types of call transfers is that the Replaces header is used in Attended call transfers. The Replaces header is interpreted by the final-recipient and contains a Call-ID header, indicating that the initial call leg is to be replaced with the incoming Invite request.

As outlined in the Refer method, there are three main roles:

- Originator—User agent that initiates the transfer or Refer request.
- Recipient —User agent that receives the Refer request and is transferred to the final-recipient.
- Final-Recipient —User agent introduced into a call with the recipient.



Note

A gateway can be a recipient or final-recipient; but not an originator.

Blind Transfer

The basic process of blind transfers works as described in the [“Refer Method” section on page 2](#). In summary, the originator sets up a call with the recipient, and issues a Refer request to the recipient. The recipient, triggered by the Refer request, sends an Invite request to the final-recipient. The recipient returns a SIP 202 (Accepted) response to the originator, and then notifies the originator of the outcome of the Refer transaction—if the final-recipient was successfully (SIP 200 OK) or unsuccessfully (SIP 503 Service Unavailable) contacted.

If successful, a call is established between the recipient and the final-recipient.

In addition, with Blind Transfer, the gateways send:

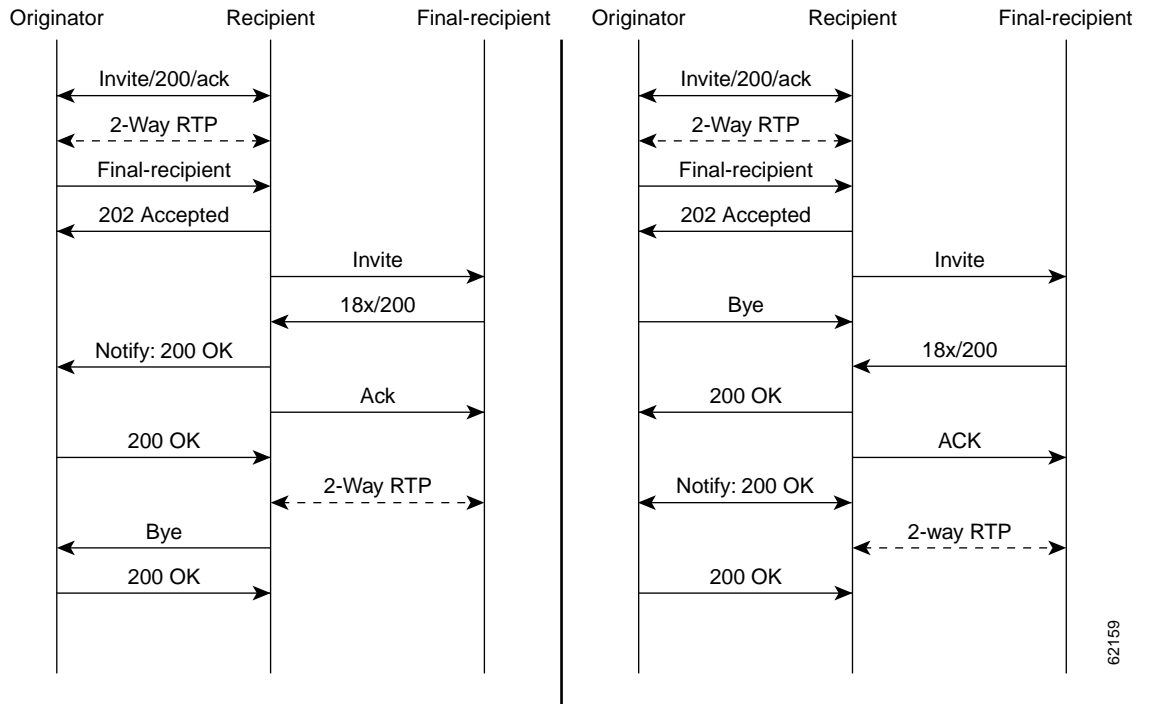
- A Bye request

The original signaling relationship between the originator and recipient is not terminated until a Bye request is sent by one of the parties. The recipient sends a Bye request to the originator if the transfer succeeds, as shown in [Figure 2](#), left. Occasionally the originator may send a Bye request immediately after sending the Refer request, before the outcome of the transfer is complete. In this case, because the Bye request has already been received, the recipient does not send a Bye request, as shown in [Figure 2](#), right.

- A Notify request

The recipient always sends a Notify message, even if a Bye request is received before the outcome of the transfer is determined. See [Figure 2](#).

Figure 2 Successful Blind Transfer. Bye Sent by Recipient (Left) and Bye Sent by Originator (Right)



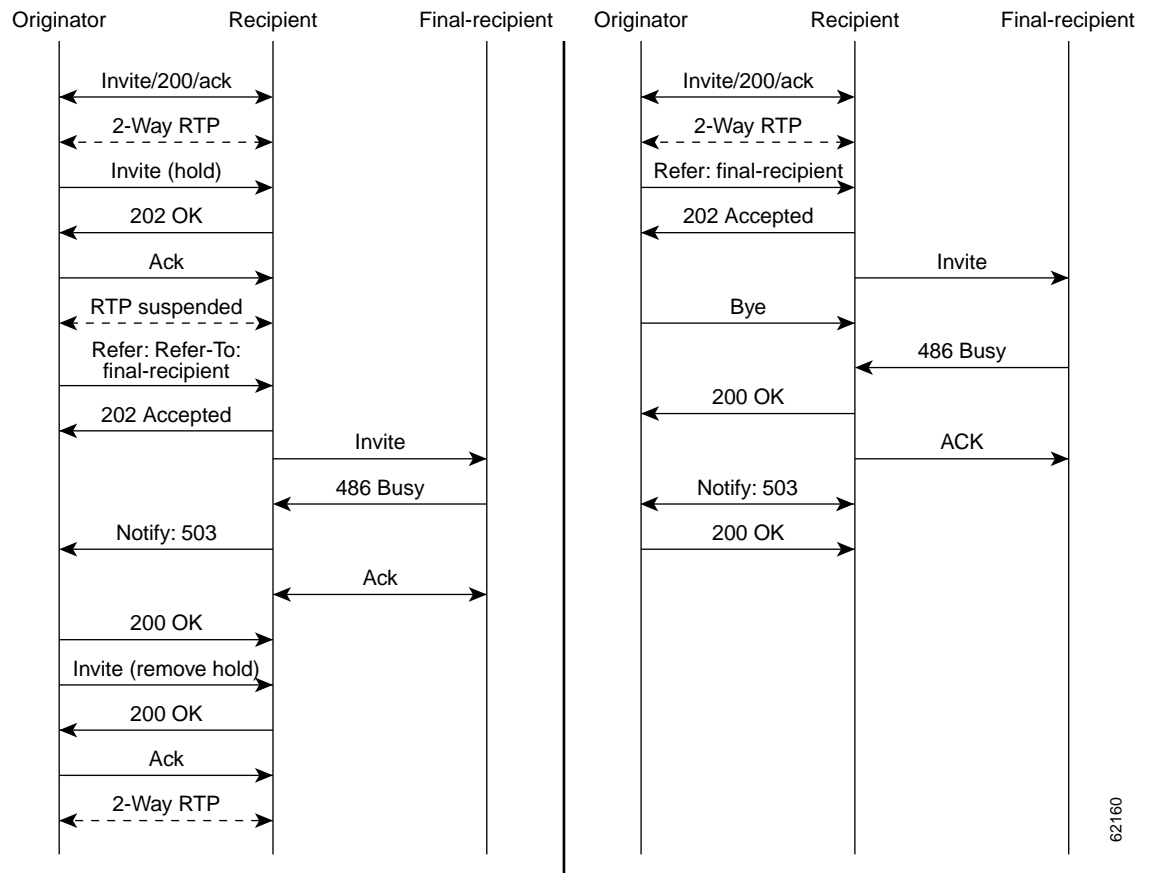
62159

- Termination

On a failed transfer, the original call between the originator and recipient may or may not be terminated. The termination is at the discretion of the originator. On a failed transfer the recipient does not send a Bye request unless the recipient goes on hook. The recipient waits for a Bye request from the originator. The original call remains active, and media is still sent from the recipient to the originator. See [Figure 3](#), left.

From the caller’s perspective, the recipient hears a busy tone when the transfer fails and the originator sends a Bye request. A busy tone is not generated on failed transfers when the Bye request has not been received, because it is at the discretion of the originator to resume the call. In other words, if the originator placed the recipient on hold, the originator can remove the recipient from hold and then resume the existing session. The recipient then reverts back to the original call between itself and the originator. If the call was on hold before the transfer, the call will still be on hold after the transfer fails.

Figure 3 Failed Blind Transfer—Original Call Remains Active (Left) and Call Is Terminated (Right)

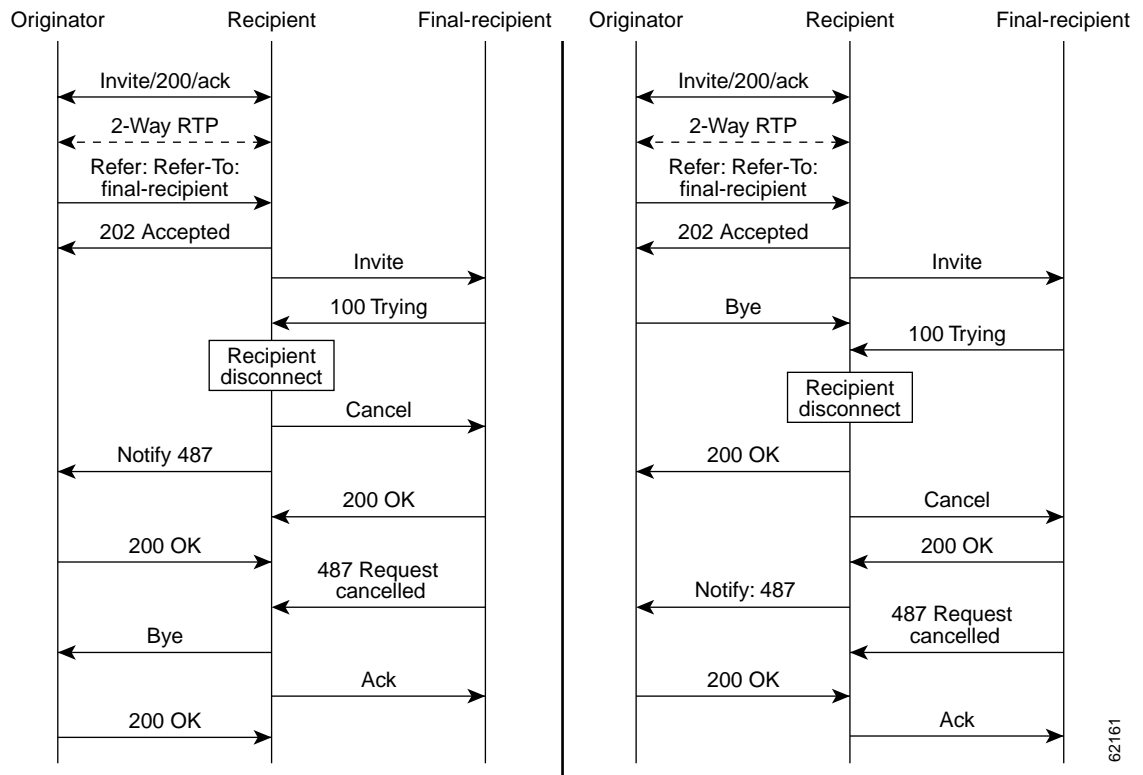


62160

- Recipient goes on-hook

Before the transfer finishes, the recipient goes on-hook. This causes a Cancel request to be sent to the final-recipient and a Bye request to be sent to the originator. Again, the Bye request sent to the originator is contingent upon the fact that the originator has not sent a Bye request immediately after the Refer request. See [Figure 4](#).

Figure 4 *Blind Transfer Recipient Disconnects Before Transfer Finishes—Recipient Sends Bye (Left) and Originator Sends Bye (Right)*



It is possible for the call state of the originator to be destroyed after sending a Bye request. In such cases, the notification is likely to be a 4xx class response instead of a 2xx class response.

62161

Attended Transfer

In attended transfers, the Replaces header is inserted by the initiator of the Refer request as an overloaded header in the Refer-To and is copied into the triggered Invite request sent to the final-recipient. The header has no affect on the recipient, but is interpreted by the final-recipient as a way to distinguish between blind transfer and attended transfer.

The process of attended transfers is as follows:

Step	Description or Detail
The originator sets up a call with the recipient.	Once the call is set-up, the originator places the recipient on hold.
The originator establishes a call to the final-recipient.	
The originator sends the recipient a Refer request with an overloaded Replaces header in the Refer-To header.	
Upon receipt of the Refer request, the recipient sends a triggered Invite request to the final-recipient.	The Invite request received by the final-recipient includes the Replaces header, identifying the call leg between the originator and final-recipient. See the “Replaces Header” section on page 9 for more information.
The recipient returns a SIP 202 (Accepted) response to the originator.	The SIP 202 (Accepted) acknowledges that the Invite has been sent.
The final-recipient establishes a direct signaling relationship with the recipient.	The receipt of the Replaces header is what indicates that the initial call leg is to be shut down and replaced by the incoming Invite request.
The recipient notifies the originator of the outcome of the Refer transaction.	The recipient notifies the originator if the final-recipient was successfully or unsuccessfully contacted.
The recipient terminates the session with the originator by sending a Bye request.	

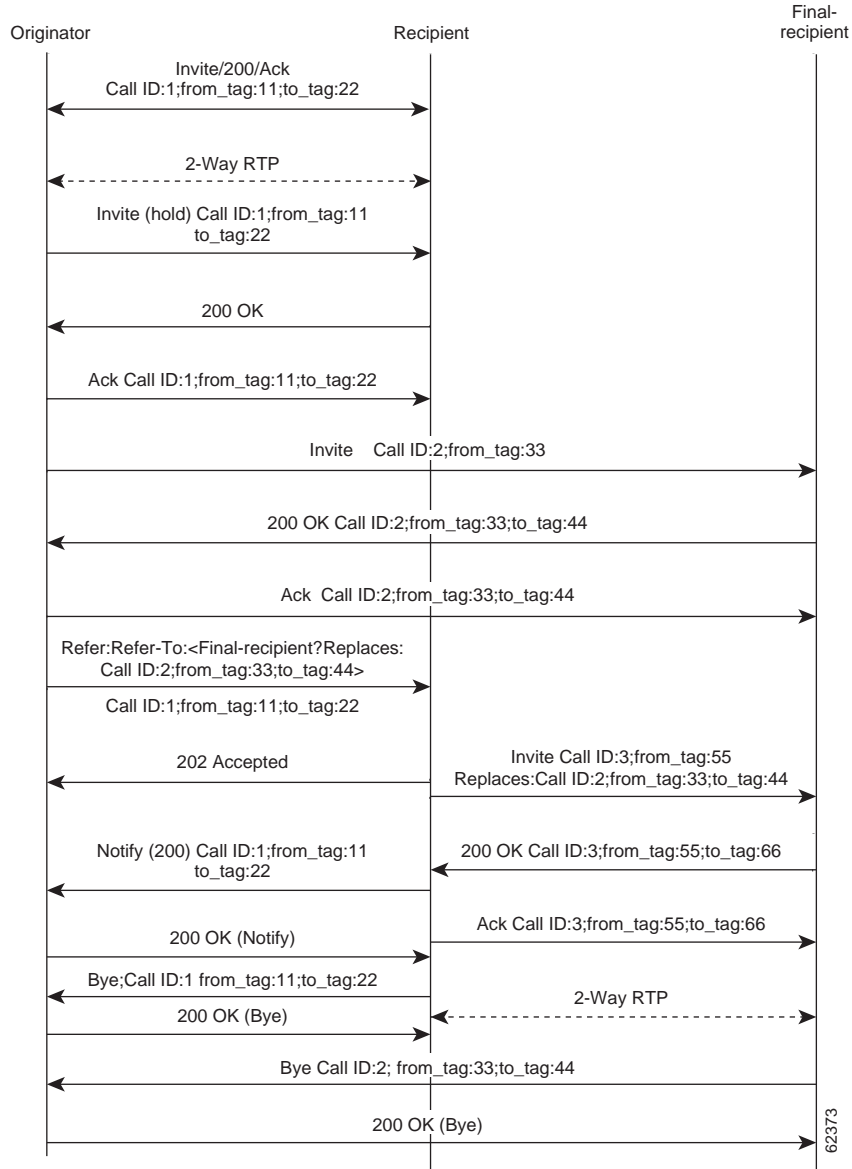
Replaces Header

The Replaces header is required in attended transfers. It indicates to the final-recipient that the initial call leg (identified by the Call-ID header and tags) is to be shut down and replaced by the incoming Invite request. The final recipient sends a Bye request to the originator to terminate its session.

If the information provided by the Replaces header does not match an existing call leg, or if the information provided by the Replaces header matches a call leg but the call leg is not active (a Connect, 200 OK to the Invite request has not been sent by the final-recipient), the triggered Invite does not replace the initial call leg and the triggered Invite request is processed normally.

Any failure resulting from the triggered Invite request from the recipient to final-recipient does not destroy the call between the originator and the final-recipient. In these scenarios, all calls that are active (originator to recipient and originator to final-recipient) remain active after the failed attended transfer attempt. [Figure 5](#) illustrates a call flow for a successful attended transfer.

Figure 5 Successful Attended Transfer



62373

Benefits

SIP Call Transfer Using Refer

SIP Call Transfer Using the Refer Method supports attended transfer and blind transfer in accordance with emerging SIP standards.

Restrictions

- Although SIP IOS gateways currently support SIP URLs and TEL URLs, the Refer-To header must be in SIP URL format to be valid. The TEL URL format cannot be used, because it does not provide a host portion, and without one, the triggered Invite request cannot be routed.
- Only three overloaded headers in the Refer-to header are accepted: Accept-Contact, Proxy-Authorization, and Replaces. All other headers present in the Refer-To are ignored.
- The Refer-To and Contact headers are required in the Refer request. The absence of either header results in a 4xx class response to the Refer request. Also, the Refer request must contain exactly one Refer-To header. Multiple Refer-To headers result in a 4xx class response.
- The Referred-By header is required in a Refer request. The absence of this header results in a 4xx class response to the Refer request. Also, the Refer request must contain exactly one Referred-By header. Multiple Referred-By headers result in a 4xx class response.
- As with the Bye and Also call transfer methods, the dial peers must be configured for correct functioning of the Refer method. See the [“Configuring SIP Call Transfer” section on page 15](#) for the complete configuration steps.

Related Features and Technologies

- Cisco SIP Proxy Server
- Cisco VoIP

Related Documents

The following documents contain information related to the Cisco SIP functionality:

- *Cisco IOS Voice, Video, and Fax Configuration Guide*, Release 12.2
- *Cisco IOS Voice, Video, and Fax Command Reference*, Release 12.2
- *Cisco IOS IP Configuration Guide*, Release 12.2
- *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services*, Release 12.2
- *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols*, Release 12.2
- *Cisco IOS IP Command Reference, Volume 3 of 3: Multicast*, Release 12.2
- Retry and Timer commands are described in:
[SIP Gateway Support of RSVP and TEL URL](#), Release 12.2(2)XB
- SIP call flows are described in: *SIP Call Flows*, Release 12.2(4)T

Supported Platforms

- Cisco 2600 series
- Cisco 3600 series
- Cisco AS5300 universal access server
- Cisco AS5350 universal gateway
- Cisco AS5400 universal gateway
- Cisco AS5850 universal gateway
- Cisco 7200 series

Table 1 Cisco IOS Release and Platform Support for this Feature

Platform	12.2(2)XB	12.2(2)XB2	12.2(8)T	12.2(11)T
Cisco 2600 series	X	X	X	X
Cisco 3600 series	X	X	X	X
Cisco 7200 series	X	X	X	X
Cisco AS5300	X	X	Not supported	X
Cisco AS5350	X	X	Not supported	X
Cisco AS5400	X	X	Not supported	X
Cisco AS5850 universal gateway	Not supported	X	Not supported	X

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 Cisco Feature Navigator. Cisco Feature Navigator dynamically updates the list of supported platforms as new platform support is added for the feature.

Cisco Feature Navigator is a web-based tool that enables you to determine which Cisco IOS software images support a specific set of features and which features are supported in a specific Cisco IOS image. You can search by feature or release. Under the release section, you can compare releases side by side to display both the features unique to each software release and the features in common.

To access Cisco Feature Navigator, you must have an account on Cisco.com. If you have forgotten or lost your account information, send a blank e-mail to cco-locksmith@cisco.com. An automatic check will verify that your e-mail address is registered with Cisco.com. If the check is successful, account details with a new random password will be e-mailed to you. Qualified users can establish an account on Cisco.com by following the directions at <http://www.cisco.com/register>.

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>

Availability of Cisco IOS Software Images

Platform support for particular Cisco IOS software releases is dependent on the availability of the software images for those platforms. Software images for some platforms may be deferred, delayed, or changed without prior notice. For updated information about platform support and availability of software images for each Cisco IOS software release, refer to the online release notes or, if supported, Cisco Feature Navigator.



Note

As of Cisco IOS Release 12.2(2)XB, Cisco Feature Navigator does not support features included in this limited-lifetime release.

Supported Standards, MIBs, and RFCs

Standards

No new or modified standards are supported by this feature.

MIBs

- CISCO-SIP-UA-MIB

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

The following are general prerequisites for SIP deployment.

- Ensure that your Cisco 2600 series, Cisco 3600 series, or Cisco 7200 series router has 16-MB Flash memory and 64-MB DRAM memory, minimum. A Cisco AS5300 must have a minimum of 16-MB Flash memory and 128-MB DRAM memory. A Cisco AS5400 must have a minimum of 32-MB Flash memory and 256-MB DRAM memory.
- Ensure that the gateway has voice functionality that is configurable for SIP.
- Establish a working IP network.

For more information about configuring IP, refer to: *Cisco IOS IP Configuration Guide*, Release 12.2

- Configure VoIP.

For more information about configuring VoIP, refer to: *Cisco IOS Voice, Video, and Fax Command Reference*, Release 12.2

- Configure the SIP dial peers for call transfer.
- As with the Bye and Also call transfer methods, the dial peers must be configured for correct functioning of the Refer method. See the [“Configuring SIP Call Transfer” section on page 15](#) for the complete configuration steps.

Configuration Tasks

See the following sections for configuration tasks for the Call Transfer Capabilities Using the Refer Method feature. Each task in the list is identified as either required or optional.

- [Configuring SIP Call Transfer](#) (required)
- [Configuring Notify Messages in the Refer Method](#) (required)

Configuring SIP Call Transfer

As with the Bye and Also call transfer methods, the dial peers must be configured for correct functioning of the Refer method.

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

	Command	Purpose
Step 1	Router(config)# dial-peer voice <i>number</i> pots	Enters the dial-peer 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 <i>[+]string</i> [T]	Specifies the telephone number associated with the dial peer.
Step 4	Router (config-dial-peer)# port <i>{slot-number/subunit-number/port}</i> <i>{slot/port:ds0-group-no}</i>	Specifies the local voice port through which incoming VoIP calls will be received. (Cisco 2600/3600).
Step 5	Router(config-dial-peer)# port <i>{controller number:D}</i>	Specifies the local voice port through which incoming VoIP calls will be received. (Cisco AS5300).
Step 6	Router (config-dial-peer)# port <i>{slot/port:ds0-group-no}</i> <i>{slot-number/subunit-number/port}</i>	Specifies the local voice port through which incoming VoIP calls will be received. (Cisco 7200).

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

	Command	Purpose
Step 1	Router(config)# dial-peer voice <i>number</i> voip	Enters the dial-peer mode to configure a VoIP 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 <i>[+]string</i> [T]	Specifies the telephone number associated with the dial peer.
Step 4	Router(config-dial-peer)# session target ipv4:destination-address	Specifies a network-specific address for a dial peer. ipv4:destination address: Sets the IP address of the dial peer. A valid IP address is in this format: xxx.xxx.xxx.xxx

Configuring Notify Messages in the Refer Method

Once the outcome of the Refer transaction is known, the recipient of the Refer request must notify the originator of the outcome of the Refer transaction—whether the final-recipient was successfully or unsuccessfully contacted. The notification is accomplished using the Notify method.

Complete these steps to configure the Notify method, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# sip-ua	Enters the SIP user agent configuration mode.
Step 2	Router(config-sip-ua)# timers notify number	Sets the amount of time for the user agent to wait before retransmitting the Notify message. The default amount of time to wait is 500 milliseconds.
Step 3	Router(config-sip-ua)# retry notify number	Sets the number of times the Notify message is retransmitted. The default is 10.
Step 4	Router(config-sip-ua)# exit	Exits SIP user agent configuration mode.

Verifying Notify in the Refer Method

Verifying SIP Timers

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

```
Router# show sip-ua timers
SIP UA Timer Values (milliseconds)
trying 500, expires 300000, connect 500, disconnect 500
comet 500, prack 500, rel1xx 500, notify 500
```

Verifying SIP Retries

The following example shows sample output for the **show sip-ua retry** command.

```
Router# show sip-ua retry
SIP UA Retry Values
invite retry count = 6 response retry count = 1
bye retry count = 1 cancel retry count = 1
prack retry count = 10 comet retry count = 10
reliable 1xx count = 6 notify retry count = 10
```

Verifying SIP Statistics

The following example shows sample output for the **show sip-ua statistics** command. The OKNotify1/0 field indicates a successful response to the Notify request. The 202Accepted 0/1 field indicates a successful response to the Refer request.

The Notify 0/1 and Refer 1/0 fields indicate status. Notify 0/1 indicates that no Notify requests were received from the gateway, and that one request was sent. Refer 1/0 indicates that one request was received, and no requests were sent.

The final Notify 0 field under Retry Statistics indicates that the Notify request was not retransmitted.

```
Router# show sip-ua statistics
SIP Response Statistics (Inbound/Outbound)
Informational:
Trying 4/2, Ringing 2/1,
Forwarded 0/0, Queued 0/0,
SessionProgress 0/0
Success:
OkInvite 1/2, OkBye 0/1,
OkCancel 1/0, OkOptions 0/0,
OkPrack 2/0, OkPreconditionMet 0/0,
OkNotify 1/0, 202Accepted 0/1
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
RequestCancel 1/0, NotAcceptableMedia 0/0
Server Error:
InternalError 0/1, NotImplemented 0/0,
BadGateway 0/0, ServiceUnavail 0/0,
GatewayTimeout 0/0, BadSipVer 0/0,
PreCondFailure 0/0
Global Failure:
BusyEverywhere 0/0, Decline 0/0,
NotExistAnywhere 0/0, NotAcceptable 0/0
SIP Total Traffic Statistics (Inbound/Outbound)
Invite 3/2, Ack 3/2, Bye 1/0,
Cancel 0/1, Options 0/0,
Prack 0/2, Comet 0/0,
Notify 0/1, Refer 1/0
Retry Statistics
Invite 0, Bye 0, Cancel 0, Response 0,
Prack 0, Comet 0, Reliablelxx 0, Notify 0
```

Troubleshooting Tips

To troubleshoot this feature, perform the following steps:

- Make sure that you can make a voice call.
- Use the **debug ccsip all** command to enable all SIP debugging capabilities, or use one of the following SIP debug commands:
 - **debug ccsip calls**
 - **debug ccsip error**
 - **debug ccsip events**
 - **debug ccsip messages**
 - **debug ccsip states**

Configuration Examples

Call Transfer Configuration

The following example illustrates how to configure call transfer. In [Figure 6](#), User A and User C are in an established call. User C then transfers the call to User B. This results in call establishment between User A and User B. User C is then disconnected with User A, regardless of whether the transfer fails or succeeds.

When a call originates or terminates on a gateway, either the calling party number, the called party number, or the port is used (depending on the scenario) to match a dial peer in order to determine the basic call characteristics. One of the characteristics to determine is which application to use for the call. For the call transfer to succeed, the matching dial peer must have application set to “session” on the gateway that is controlling the transfer. (This is the gateway that receives the Bye with an Also header).

There are two scenarios for dial-peer matching based on whether the call is coming from a POTS interface or from the IP network.

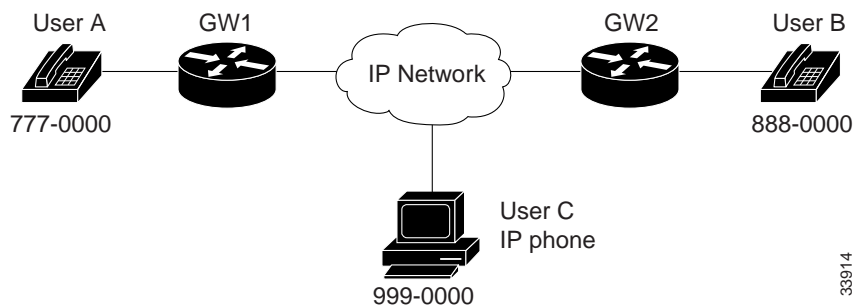
- For calls coming from a POTS interface, the port will be used to match a POTS dial peer with the port the call came in from. This dial peer should have “application session.”
- For calls coming from the IP network, a series of criteria is used (in the order listed below) to match dial peers. If the first criteria does not result in a match, the second criteria is used. If the second criteria does not result in a match, the third criteria is used. If a match does not occur, the default application, which does not support call transfer, is used.
 - a. The called number matches the “incoming called-number” on a VoIP dial peer.
 - b. The calling number matches the “answer-address” on a VoIP dial peer.
 - c. The calling number matches the “destination-pattern” on a VoIP dial peer.



Note

For calls coming from the IP network, it is possible for the calling number to be blocked based on privacy restrictions. In such cases, the “incoming called-number” be used for call transfers.

Figure 6 Call Transfer Example



In this example, Gateway 1 handles the transfer (recipient of the Bye with the Also header). User C invokes the transfer service (originator of the Bye with the Also header). There are two scenarios in which a dial peer match must have application set to “session” for the transfer to succeed:

- Incoming call from the PSTN—User A originates a call to User C. From the perspective of Gateway 1, this would be an incoming call from the POTS interface so Gateway 1 looks for a POTS dial-peer matching the port on which the call came in. Gateway 1 must have a POTS dial peer for User A with application set to “session” if transfer is later invoked by User C.
- Incoming call from IP network—User C calls User A. From the perspective of Gateway 1 this is an incoming call from the IP network. Gateway 1 uses the criteria previously discussed for a VoIP dial peer (match on incoming called-number, answer-address, or destination pattern). Gateway 1 must have one of the following:
 - A VoIP dial peer with an incoming called-number of User A
 - A VoIP dial peer with answer-address of User C
 - A VoIP dial peer with destination-pattern of User C.

The matching dial peer must have application set to “session” if transfer is later invoked by User C.

**Note**

To handle all call transfer situations, you should configure both POTS and VoIP dial peers.

To configure a POTS dial peer with the application of session, enter the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# dial-peer voice <i>number</i> pots	Enter the dial-peer 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.

To configure a VoIP dial peer with a destination pattern, enter the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# dial-peer voice <i>number</i> voip	Enter the dial-peer mode to configure a VoIP dial peer.
Step 2	Router(config-dial-peer)# destination-pattern [+]string[T]	Specifies the telephone number associated with the dial peer.

To configure a VoIP dial peer with an incoming called-number, enter the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# dial-peer voice <i>number</i> voip	Enter the dial-peer mode to configure a VoIP dial peer.
Step 2	Router(config-dial-peer)# incoming called-number <i>string</i>	Specifies an incoming called number of a dial peer.

To configure a VoIP dial peer with an incoming called-number, enter the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# dial-peer voice <i>number</i> voip	Enter the dial-peer mode to configure a VoIP dial peer.
Step 2	Router (config-dial-peer) # answer-address [+] <i>string</i> [T]	Specifies the full E.164 telephone number to be used to identify the dial peer of an incoming call.

Command Reference

This section documents new and modified commands. All other commands used with this feature are documented in the Cisco IOS Release 12.2 command reference publications.

New Commands

- [retry notify](#)
- [timers notify](#)

Modified Commands

- [show sip-ua retry](#)
- [show sip-ua statistics](#)
- [show sip-ua timers](#)

retry notify

To configure the number of times the Notify message is retransmitted to the user agent that initiated the transfer or Refer request, use the **retry notify** command in SIP user agent configuration mode. To achieve default capabilities, use the **no** form of this command.

retry notify *number*

no retry notify

Syntax Description	<i>number</i>	Number of notify message retries. Possible values are 1 through 10.
---------------------------	---------------	---

Defaults	The default number of retries is 10.
-----------------	--------------------------------------

Command Modes	SIP user agent configuration
----------------------	------------------------------

Command History	Release	Modification
	12.2(2)XB	This command was introduced.
	12.2(2)XB2	This command was implemented on the Cisco AS5850 platform.
	12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T. The Cisco AS5300, Cisco AS5350, Cisco AS5850, and Cisco AS5400 platforms were not supported in this release.
	12.2(11)T	This command was integrated into Cisco IOS Release 12.2(11)T and support was added for the Cisco AS5300, Cisco AS5350, Cisco AS5400, and Cisco AS5850 platforms.

Usage Guidelines	A Notify message informs the user agent that initiated the transfer or Refer request of the outcome of the SIP transaction.
-------------------------	---

When configuring the **retry notify** command, use the default number of 10, when possible. Lower values, such as 1, can lead to an increased chance of the message not being received by the other user agent.

Examples	The following example shows a Notify message being configured to be retransmitted 10 times:
-----------------	---

```
Router(config)# sip-ua
Router(config-sip-ua)# retry notify 10
```

Related Commands	Command	Description
	show sip-ua statistics	Displays response, traffic, timer, and retry statistics.

Command	Description
show sip-ua retry	Displays the SIP retry attempts.
timers notify	Sets the amount of time that the user agent (UA) should wait before retransmitting the Notify message.

show sip-ua retry

To display SIP retry statistics, use the **show sip-ua retry** command in privileged EXEC mode.

show sip-ua retry

Syntax Description This command has no arguments or keywords.

Defaults There are no default behaviors or values for this command.

Command Modes Privileged EXEC

Command History	Release	Modification
	12.1(3)T	This command was introduced.
	12.2(2)XB	Command output was enhanced to display: <ul style="list-style-type: none"> Reliable provisional responses (PRACK/reliable lxx). Conditions met (COMET) responses. Notify responses.
	12.2(2)XB2	This command was implemented on the Cisco AS5850 platform.
	12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T. The Cisco AS5300, Cisco AS5350, Cisco AS5850, and Cisco AS5400 platforms were not supported in this release.
	12.2(11)T	This command was integrated into Cisco IOS Release 12.2(11)T and support was added for the Cisco AS5300, Cisco AS5350, Cisco AS5400, and Cisco AS5850 platforms.

Usage Guidelines Use the **show sip-ua retry** command to verify SIP configurations

Examples The following example shows sample output for the **show sip-ua retry** command.

```
Router# show sip-ua retry
SIP UA Retry Values
invite retry count = 6 response retry count = 1
bye retry count = 1 cancel retry count = 1
prack retry count = 10 comet retry count = 10
reliable lxx count = 6 notify retry count = 10
```

Table 2 *show sip-ua retry Command Field Descriptions*

Field	Description
bye retry count	Indicates the number of times a Bye request is retransmitted.
cancel retry count	Indicates the number of times a Cancel request is retransmitted.

Table 2 *show sip-ua retry Command Field Descriptions (continued)*

Field	Description
comet retry count	Indicates the number of times a COMET request is retransmitted.
invite retry count	Indicates the number of times a Invite request is retransmitted.
notify retry count	Indicates the number of times a Notify message is retransmitted.
prack retry count	Indicates the number of times a PRACK request is retransmitted.
reliable 1xx count	Indicates the number of times a Reliable 1xx request is retransmitted.
response retry count	Indicates the number of times a Response request is retransmitted.
SIP UA Retry Values	Field header for SIP UA retry values.

Related Commands

Command	Description
retry notify	Configures the number of times a COMET request is retransmitted.
show sip-ua statistics	Displays response, traffic, and retry SIP statistics.
show sip-ua timers	Displays the current settings for SIP UA timers.

show sip-ua statistics

To display response, traffic, and retry SIP statistics, use the **show sip-ua statistics** command in privileged EXEC mode.

show sip-ua statistics

Syntax Description This command has no arguments or keywords.

Defaults There are no default behaviors or values for this command.

Command Modes Privileged EXEC

Command History

Release	Modification
12.1(3)T	This command was introduced.
12.2(2)XB	Command output was enhanced to display: <ul style="list-style-type: none"> • BadRequest counter (400 class) now counts Malformed Via entries. • Reliable provisional responses (PRACK/rel1xx). • Conditions met (COMET) • Notify responses.
12.2(2)XB2	This command was implemented on the Cisco AS5850 platform.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T. The Cisco AS5300, Cisco AS5350, Cisco AS5850, and Cisco AS5400 platforms were not supported in this release.
12.2(11)T	This command was integrated into Cisco IOS Release 12.2(11)T and support was added for the Cisco AS5300, Cisco AS5350, Cisco AS5400, and Cisco AS5850 platforms.

Usage Guidelines Use the **show sip-ua statistics** command to verify SIP configurations.

Examples

The following is sample output from 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,
OkPrack 0/0, OkPreconditionMet 0/0,
```

```

OkNotify 0/0, 202Accepted 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
RequestCancel 0/0, NotAcceptableMedia 0/0
Server Error:
InternalServerError 0/0, NotImplemented 0/0,
BadGateway 0/0, ServiceUnavail 0/0,
GatewayTimeout 0/0, BadSipVer 0/0,
PreCondFailure 0/0
Global Failure:
BusyEverywhere 0/0, Decline 0/0,
NotExistAnywhere 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,
Prack 0/0, Comet 0/0,
Notify 0/0, Refer 0/0
Retry Statistics
Invite 0, Bye 0, Cancel 0, Response 0,
Prack 0, Comet 0, Reliable1xx 0, Notify 0

```

The following table describes the fields displayed by the **show sip-ua statistics** command.

Table 3 *show sip-ua statistics Command Field Descriptions*

Field	Description
	When 0/0 is included in a field, the first number is an inbound count, the latter number is an outbound count.
Ack 0/0	Indicates a confirmed final response received/sent.
Accepted 0/0	202 Indicates a successful response to a Refer request received/sent.
AddrIncomplete 0/0	484 Address supplied is incomplete.
AlternateService 0	380 Unsuccessful call; however, an alternate service is available.
Ambiguous 0/0	485 Address supplied is ambiguous.
BadExtension 0/0	420 Server couldn't understand the protocol extension in the Require header.
BadGateway 0/0	502 Network is out of order.
BadRequest	400 Bad Request (includes the malformed Via header).
BadSipVer 0/0	505 SIP version requested is not supported.
BusyEverywhere 0/0	600 Called party is busy.
BusyHere 0/0	486 Called party is busy.

Table 3 *show sip-ua statistics Command Field Descriptions (continued)*

Field	Description
Bye 0	Indicates the number of times a Bye request is retransmitted to the other user agent.
Bye 0/0	Terminates the session.
CallLegNonExistent 0/0	481 Server is ignoring the request. It was either a Bye request and there wasn't a matching leg ID, or a Cancel request and there wasn't a matching transaction.
Cancel 0	Indicates the number of times a Cancel request is retransmitted to the other user agent.
Cancel 0/0	Terminates the pending request.
Client Error:	4xx Indicates a client error.
Comet 0	Indicates the number of times a COMET request is retransmitted to the other user agent.
Comet 0/0	Conditions have been met.
Conflict 0/0	409 Temporary failure.
Decline 0/0	603 Call rejected.
Forbidden 0/0	403 The SIP server has the request, but can not provide service.
Forwarded 0/0	181 Call has been forwarded.
GatewayTimeout 0/0	504 Indicates that the server or gateway did not receive a timely response from another server (such as a location server).
Global Failure:	6xx Called party does not exist anywhere.
Gone 0/0	410 Resource is no longer available at the server and no forwarding address is known.
Informational:	1xx Indicates an Informational response.
InternalError 0/0	500 Indicates that the server or gateway encountered an unexpected error that prevented it from processing the request.
Invite 0	Indicates the number of times an INVITE request is retransmitted to the other user agent.
Invite 0/0	Initiates a call.
LengthRequired 0/0	411 A content length is required.
LoopDetected 0/0	482 Indicates a loop—server received a request that included itself in the path.
MethodNotAllowed 0/0	405 Method specified in the request is not allowed.
MovedPermanently 0	301 User is no longer available at this location.
MovedTemporarily 0	302 User is temporarily unavailable.
MultipleChoice 0	300 Address resolves to more than one location.
NotAcceptable 0/0	406/606 Call contacted, but some aspect of the session description was unacceptable.
NotAcceptableMedia 0/0	406 Call contacted, but some aspect of the session description was unacceptable.

Table 3 *show sip-ua statistics Command Field Descriptions (continued)*

Field	Description
NotExistAnywhere 0/0	604 Server has authoritative information that the called party does not exist in the network.
NotFound 0/0	404 Called party does not exist in the specified domain.
Notify 0	Indicates the number of times a Notify is retransmitted to the other user agent.
Notify 0/0	Indicates the number of Notify messages received/sent.
NotImplemented 0/0	501 Service or option not implemented in the server or gateway.
OkBye 0/0	200 Indicates a successful response to a Bye request.
OkCancel 0/0	200 Indicates a successful response to a Cancel request.
OkInvite 0/0	200 Indicates a successful response to an INVITE request.
OkNotify 0/0	200 Indicates a successful response to an Notify request.
OkOptions 0/0	200 Indicates a successful response to an Options request.
OkPrack 0/0	200 Indicates a successful response to a PRACK request.
OkPreconditionMet 0/0	200 Indicates a successful response to a PreconditionMet request.
Options 0/0	Query the receiving/sending server as to its capabilities.
PaymentRequired 0/0	402 Payment is required to complete the call.
Prack 0	Indicates the number of times a PRACK request is retransmitted to the other user agent.
Prack 0/0	Provisional response received/sent.
PreCondFailure 0/0	580 Indicates the session could not be established due to failure to meet required preconditions.
ProxyAuthReqd 0/0	407 Rejected for proxy authentication.
Queued 0/0	182 Until the called party is available, the message is queued.
Redirection (Inbound only):	3xx Indicates that the called party is not available at the address used in the request; reissue.
Refer 0/0	Indicates the number of Refer requests received/sent.
Reliable1xx 0	Indicates the number of times the Reliable 1xx response is retransmitted to the other user agent.
ReqEntityTooLarge 0/0	413 Server refuses to process because the request is larger than is acceptable.
ReqURITooLarge 0/0	414 Server refuses to process, because the URI (URL) request is larger than is acceptable.
ReqTimeout 0/0	408 Server could not produce a response before the Expires time-out.
RequestCancel 0/0	Request has been cancelled.
Response 0	Indicates the number of Response retries.
Retry Statistics	One of the three categories of response statistics.
Ringing 0/0	180 Called party has been located and is being notified of the call.
SeeOther 0	303

Table 3 *show sip-ua statistics Command Field Descriptions (continued)*

Field	Description
Server Error:	5xx Server error.
ServiceUnavail 0/0	503 Service option is not available due to an overload or maintenance problem.
SessionProgress 0/0	183 Indicates inband alerting.
SIP Response Statistics (Inbound/Outbound)	One of the three categories of response statistics.
SIP Total Traffic Statistics (Inbound/Outbound)	One of the three categories of response statistics.
Success	2xx Request understood and performed.
TempNotAvailable 0/0	480 Called party didn't respond.
TooManyHops 0/0	483 Indicate that a server received a request that required more hops than is allowed by the Max-Forward header.
Trying 0/0	100 Action is being taken with no resolution.
Unauthorized 0/0	401 Indicates that the request requires user authentication.
UnsupportedMediaType 0/0	415 Server refuses to process because the service option is not available on the destination endpoint.
UseProxy 0	305 Caller must use a proxy to contact called party.

Related Commands

Command	Description
show sip-ua retry	Displays SIP retry statistics.
show sip-ua timers	Displays the current settings for SIP UA timers.

show sip-ua timers

To display the current settings for SIP UA timers, use the **show sip-ua timers** command in privileged EXEC mode.

show sip-ua timers

Syntax Description This command has no arguments or keywords.

Defaults There are no default behaviors or values for this command.

Command Modes Privileged EXEC

Release	Modification
12.1(1)T	This command was introduced.
12.1(3)T	The output of this command was changed to reflect the changes in the timers command.
12.2(2)XB	Command output was enhanced to display: <ul style="list-style-type: none"> Reliable provisional responses (PRACK/reliable 1xx). Conditions met (COMET) responses. Notify responses.
12.2(2)XB2	This command was implemented on the Cisco AS5850 platform.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T. The Cisco AS5300, Cisco AS5350, Cisco AS5850, and Cisco AS5400 platforms were not supported in this release.
12.2(11)T	This command was integrated into Cisco IOS Release 12.2(11)T and support was added for the Cisco AS5300, Cisco AS5350, Cisco AS5400, and Cisco AS5850 platforms.

Usage Guidelines Use the **show sip-ua timers** command to verify SIP configurations.

Examples The following is sample output from the **show sip-ua timers** command:

```
Router# show sip-ua timers
SIP UA Timer Values (milliseconds)
trying 500, expires 300000, connect 500, disconnect 500
comet 500, prack 500, rel1xx 500, notify 500
```

Table 4 *show sip-ua timers Command Field Descriptions*

Field	Description
SIP UA Timer Values (milliseconds)	Indicates SIP UA timer status.
trying	Indicates the time to wait before a Trying message is retransmitted.
expires	Indicates the time to wait before an Expires message is retransmitted.
connect	Indicates the time to wait before a Connect message is retransmitted.
disconnect	Indicates the time to wait before a Disconnect message is retransmitted.
comet	Indicates the time to wait before a COMET message is retransmitted.
prack	Indicates the time to wait before a PRACK acknowledgment is retransmitted.
rellxx	Indicates the time to wait before a Rel1xx response is retransmitted.
notify	Indicates the time to wait before a Notify response is retransmitted.

Related Commands

Command	Description
timers notify	Sets the amount of time that the user agent (UA) should wait before retransmitting the Notify message.
show sip-ua retry	Displays SIP retry statistics.
show sip-ua statistics	Displays response, traffic, and retry SIP statistics.

timers notify

To set the amount of time that the user agent (UA) should wait before retransmitting the Notify message, use the **timers notify** command in SIP user agent configuration mode. To achieve default capabilities, use the **no** form of this command.

timers notify *number*

no timers notify

Syntax Description	<i>number</i>	Time (in milliseconds). Possible values are between 100 through 1000.
--------------------	---------------	---

Defaults	The default amount of time to wait before retransmitting is 500 milliseconds.
----------	---

Command Modes	SIP user agent configuration
---------------	------------------------------

Command History	Release	Modification
	12.2(2)XB	This command was introduced.
	12.2(2)XB2	This command was implemented on the Cisco AS5850 platform.
	12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T. The Cisco AS5300, Cisco AS5350, Cisco AS5850, and Cisco AS5400 platforms were not supported in this release.
	12.2(11)T	This command was integrated into Cisco IOS Release 12.2(11)T and support was added for the Cisco AS5300, Cisco AS5350, Cisco AS5400, and Cisco AS5850 platforms.

Usage Guidelines	A Notify message informs the user agent that initiated the transfer or Refer request about the outcome of the SIP transaction.
------------------	--

Examples	The following example shows the timers notify command being set up with a retransmission time of 500 milliseconds:
----------	---

```
Router(config)# sip-ua
Router(config-sip-ua)# timers notify 500
```

Related Commands	Command	Description
	show sip-ua statistics	Displays response, traffic, timer, and retry statistics
	show sip-ua timers	Displays the current settings for SIP UA timers

Glossary

Call-ID—A general header field that uniquely identifies a particular invitation or all registrations of a particular client.

call leg— A logical connection between the router and another endpoint.

final-recipient—User agent introduced into a call with the recipient.

gateway—A gateway allows SIP or H.323 terminals to communicate with terminals configured to other protocols by converting protocols. A gateway is the point where a circuit-switched call is encoded and repackaged into IP packets.

INVITE—A method that initiates a session. It indicates that a user is invited to participate, provides a session description, indicates the type of media, and provides insight regarding the capabilities of the called and calling parties.

originator—User agent that initiates the transfer or Refer request with the recipient.

proxy—A SIP UAC or UAS that forwards requests and responses on behalf of another SIP UAC or UAS.

recipient—User agent that receives the Refer request from the originator and is transferred to the final-recipient.

SIP—Session Initiation Protocol. An application-layer protocol originally developed by the Multiparty Multimedia Session Control (MMUSIC) working group of the Internet Engineering Task Force (IETF). Their goal was to equip platforms to signal the setup of voice and multimedia calls over IP networks. SIP features are compliant with IETF RFC 2543, published in March 1999.

UA—user agent.

UAC—user agent client. A client application that initiates a SIP request.

UAS—user agent server (or user agent). A server application that contacts the user when a SIP request is received, then returns a response on behalf of the user. The response accepts, rejects or redirects the request.

VoIP—Voice over IP. The ability to carry normal telephone-style voice over an IP-based Internet with POTS-like functionality, reliability, and voice quality. VoIP is a blanket term that generally refers to the Cisco standards-based approach (for example, H.323) to IP voice traffic.