



## **Cisco SIP IP Phone Administrator Guide**

Release 6.0, 6.1, 7.0, 7.1, 7.2, 7.3, 7.4  
March, 2005

### **Corporate Headquarters**

Cisco Systems, Inc.  
170 West Tasman Drive  
San Jose, CA 95134-1706  
USA  
<http://www.cisco.com>  
Tel: 408 526-4000  
800 553-NETS (6387)  
Fax: 408 526-4100



THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

CCSP, CCVP, the Cisco Square Bridge logo, Follow Me Browsing, and StackWise are trademarks of Cisco Systems, Inc.; Changing the Way We Work, Live, Play, and Learn, and iQuick Study are service marks of Cisco Systems, Inc.; and Access Registrar, Aironet, ASIST, BPX, Catalyst, CCDA, CCDP, CCIE, CCIP, CCNA, CCNP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, Cisco Press, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Cisco Unity, Empowering the Internet Generation, Enterprise/Solver, EtherChannel, EtherFast, EtherSwitch, Fast Step, FormShare, GigaDrive, GigaStack, HomeLink, Internet Quotient, IOS, IP/TV, iQ Expertise, the iQ logo, iQ Net Readiness Scorecard, LightStream, Linksys, MeetingPlace, MGX, the Networkers logo, Networking Academy, Network Registrar, *Packet*, PIX, Post-Routing, Pre-Routing, ProConnect, RateMUX, ScriptShare, SlideCast, SMARTnet, StrataView Plus, TeleRouter, The Fastest Way to Increase Your Internet Quotient, and TransPath are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

All other trademarks mentioned in this document or Website are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0502R)

*Cisco SIP IP Phone Administrator Guide*

Copyright © 2004-2005 Cisco Systems, Inc. All rights reserved.



## **Preface vii**

Overview	<b>vii</b>
Who Should Use This Guide	<b>vii</b>
Objectives	<b>viii</b>
Document Organization	<b>viii</b>
Document Conventions	<b>viii</b>
Related Documentation	<b>ix</b>
Obtaining Documentation	<b>x</b>
Cisco.com	<b>x</b>
Ordering Documentation	<b>x</b>
Documentation Feedback	<b>xi</b>
Obtaining Technical Assistance	<b>xi</b>
Cisco TAC Website	<b>xi</b>
Opening a TAC Case	<b>xi</b>
TAC Case Priority Definitions	<b>xii</b>
Obtaining Additional Publications and Information	<b>xii</b>
	<b>xiii</b>

---

## **CHAPTER 1**

### **Product Overview 1-1**

New Information in This Release	<b>1-1</b>
Cisco IP 7960G/7940G Phone Overview	<b>1-2</b>
Session Initiation Protocol Overview	<b>1-6</b>
SIP Capabilities	<b>1-6</b>
SIP Components	<b>1-7</b>
SIP Clients	<b>1-8</b>
SIP Servers	<b>1-8</b>
BTXML Support	<b>1-8</b>
Cisco CallManager XML Support	<b>1-8</b>
Network Capabilities	<b>1-9</b>
Configuration Features	<b>1-9</b>
Signaling Support	<b>1-10</b>
Dial-Plan and Messaging Support	<b>1-10</b>
Routing and Proxy Support	<b>1-11</b>

Supported Languages and Character Set **1-12**  
 Supported Protocols **1-13**  
 Where to Go Next **1-14**

**CHAPTER 2**

**Installing Cisco IP 7960G/7940G Phone Hardware on the Desktop or Wall 2-1**

Placing the Phone on the Desktop **2-1**  
 Installing the Phone on the Wall **2-1**  
 Cabling the Phone Ports **2-2**  
 Using the Phone with a Cisco Catalyst Switch **2-3**  
 Connecting the Phone to Power **2-4**  
 Where to Go Next **2-5**

**CHAPTER 3**

**Initializing Cisco SIP IP Phones 3-1**

Prerequisites **3-1**  
 Overview of the Initialization Process **3-2**  
 Information About Configuration Files **3-4**  
 How to Customize the Default Configuration File **3-5**  
 How to Customize a Phone-Specific Configuration File **3-8**  
 How to Customize the Configuration from the Phone Menu **3-10**  
     Unlocking and Locking the Phone **3-11**  
     Setting and Restoring Network Parameters **3-11**  
     Setting and Restoring Phone-Specific Parameters **3-14**  
     Setting End-User Call Preferences **3-16**  
 How to Set the Date and Time **3-17**  
     Time-Zone Configuration Examples **3-21**  
 How to Create Dial Plans **3-22**  
     Dial-Plan Configuration Examples **3-25**  
 How to Verify Initialization **3-26**  
 Where to Go Next **3-26**

**CHAPTER 4**

**Managing Cisco SIP IP Phones 4-1**

How to Customize Cisco IP 7960G/7940G Phone Rings **4-1**  
 How to Access XML Cards **4-2**  
     Restrictions for XML Cards **4-2**  
 How to View Your Cisco SIP IP Phone Firmware Image Version **4-3**  
 How to Upgrade Your Cisco SIP IP Phone Firmware Image **4-3**  
     Upgrading to Version 7.0 and Later Releases **4-5**

How to Upgrade Your Cisco SIP IP Phone Firmware Image and Reboot Remotely	4-6
Where to Go Next	4-7

**CHAPTER 5****Monitoring Cisco SIP IP Phones 5-1**

How to Use the Command-Line Interface to Monitor Phones	5-1
Output Examples	5-7
How to Use the Phone Menus to Access Status Information	5-18
Viewing Status Messages	5-19
Viewing Network Statistics	5-19

**APPENDIX A****Compliance with RFC 3261 A-1**

SIP Functions	A-1
SIP Methods	A-2
SIP Responses	A-2
1xx Response—Information Responses	A-2
2xx Response—Successful Responses	A-3
3xx Response—Redirection Responses	A-3
4xx Response—Request Failure Responses	A-3
5xx Response—Server Failure Responses	A-5
6xx Response—Global Responses	A-6
SIP Header Fields	A-6
SIP Session Description Protocol Usage	A-7
Transport Layer Protocols	A-8
SIP Security Authentication	A-8
SIP DNS Records Usage	A-8
SIP DTMF Digit Transport	A-8

**APPENDIX B****SIP Call Flows B-1**

Call Flow Scenarios for Successful Calls	B-1
Gateway to Cisco SIP IP Phone in a SIP Network	B-2
Call Setup and Disconnect	B-2
Call Setup and Hold	B-4
Call to a Gateway Acting As an Emergency Proxy from a Cisco SIP IP Phone	B-6
Cisco SIP IP Phone to Cisco SIP IP Phone	B-7
Simple Call Hold	B-8
Call Hold with Consultation	B-10
Call Waiting	B-13
Call Transfer Without Consultation	B-17

- Call Transfer Without Consultation Using Failover **B-20**
- Call Transfer with Consultation **B-23**
- Call Transfer with Consultation Using Failover **B-27**
- Network Call Forwarding (Unconditional) **B-31**
- Network Call Forwarding (Busy) **B-33**
- Network Call Forwarding (No Answer) **B-36**
- Three-Way Calling **B-38**
- Call from a Cisco SIP IP Phone to a Gateway Acting As a Backup Proxy in a SIP Network **B-42**
- Call from a Cisco SIP IP Phone to a Cisco SIP IP Phone Using a SIP Backup Proxy **B-44**
- Call from a Cisco SIP IP Phone to a Cisco SIP IP Phone Using a SIP Emergency Proxy **B-45**
- Call Flow Scenarios for Failed Calls **B-47**
  - Gateway to Cisco SIP IP Phone in a SIP Network **B-47**
    - Called Number Is Busy **B-47**
    - Called Number Does Not Answer **B-49**
    - Client, Server, or Global Error **B-50**
  - Cisco SIP IP Phone to Cisco SIP IP Phone in a SIP Network **B-52**
    - Called Number Is Busy **B-53**
    - Called Number Does Not Answer **B-54**
    - Authentication Error **B-54**

**APPENDIX C**

**Technical Specifications of the Cisco IP 7960G/7940G Phone C-1**

- Physical and Operating Environment Specifications **C-1**
- Cable Specifications **C-2**
- Regulatory Safety Compliance **C-2**
- Connection Specifications **C-2**

**APPENDIX D**

**Configurable Parameters for the SIP IP Phone D-1**



## Preface

---

This administrator guide describes the Cisco IP 7960G/7940G phones in a Session Initiation Protocol (SIP) network. This preface describes the objectives and organization of the document and explains how to find additional information on related products and services. It contains the following sections:

- Overview, page vii
- Who Should Use This Guide, page vii
- Objectives, page viii
- Document Organization, page viii
- Document Conventions, page viii
- Related Documentation, page ix
- Obtaining Documentation, page x
- Obtaining Technical Assistance, page xi
- Obtaining Additional Publications and Information, page xii

## Overview

The *Cisco SIP IP Phone Administrator Guide* provides information about how to set up, cable, and configure a Cisco IP 7960G/7940G phone in a SIP network. It also provides information on how to configure the network and SIP parameters and change the settings and options of the Cisco IP phone. The appendixes include reference information such as RFC compliance and Cisco IP phone call flows.

## Who Should Use This Guide

Network engineers, system administrators, Cisco partners, and telecommunications engineers should use this guide to learn the tasks required to set up the Cisco IP phone in an SIP network. The described tasks are administration-level tasks and are not intended for the end users of the phones. Many of the tasks involve configuring network settings that could affect the ability of the phone to function in the network and require an understanding of IP networking and telephony concepts.

# Objectives

This guide provides necessary information to get the Cisco IP phone operational in a SIP network. It is not intended to provide information on how to implement a SIP or a VoIP network. For information on implementing SIP and VoIP networks, refer to the documents listed in the “Related Documentation” section on page ix.

## Document Organization

This document is organized into the following chapters and appendixes:

- Chapter 1, “Product Overview”—Describes the SIP protocol and the Cisco 7960G/7940G phone.
- Chapter 2, “Installing Cisco IP 7960G/7940G Phone Hardware on the Desktop or Wall”—Describes how to install phone hardware.
- Chapter 3, “Initializing Cisco SIP IP Phones”—Describes how to install firmware, customize configuration files, and connect the phone.
- Chapter 4, “Managing Cisco SIP IP Phones”—Describes how to upgrade firmware and perform other management tasks.
- Chapter 5, “Monitoring Cisco SIP IP Phones”—Describes how to debug and troubleshoot.
- Appendix A, “Compliance with RFC 3261”—Provides reference information about Cisco SIP IP phone compliance to RFC 3261.
- Appendix B, “SIP Call Flows”—Provides reference information about Cisco SIP IP phone call flows.
- Appendix C, “Technical Specifications of the Cisco IP 7960G/7940G Phone”—Provides physical and operating environment specifications, cable specifications, and connection specifications.
- Appendix D, “Configurable Parameters for the SIP IP Phone”—Lists and describes configurable parameters.

## Document Conventions

This document uses the following conventions:

- Commands and keywords are in **boldface** font.
- Arguments for which you supply values are in *italic* font.
- Elements in square brackets ([ ]) are optional.
- Required alternative keywords are grouped in braces and are separated by vertical bars (for example, {**x** | **y** | **z**}).
- Optional alternative keywords are grouped in brackets and are separated by vertical bars (for example, [**x** | **y** | **z**]).
- Terminal sessions and information that the system displays are in `screen` font.
- Information that you must enter is in **boldface screen** font.
- Buttons and menus that are called out in text in an imperative context are in **boldface** font.



**Note**

Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the publication.

**Caution**

Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

**Warning**

**This warning symbol means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. To see translations of the warnings that appear in this publication, refer to the translated safety warnings that accompanied this device.**

## Related Documentation

### Cisco IP Phone 7960 and Cisco IP Phone 7940

- *Cisco IP Phone 7960 and 7940 Series at a Glance*  
[http://www.cisco.com/univercd/cc/td/doc/product/voice/c\\_ipphon/english/ipp7960/6040atag.pdf](http://www.cisco.com/univercd/cc/td/doc/product/voice/c_ipphon/english/ipp7960/6040atag.pdf)
- *Installing the Wall Mount Kit on the Cisco IP Phone*  
[http://www.cisco.com/univercd/cc/td/doc/product/voice/c\\_ipphon/english/ipp7960/ippmount.htm](http://www.cisco.com/univercd/cc/td/doc/product/voice/c_ipphon/english/ipp7960/ippmount.htm)

### Session Initiation Protocol

- *Session Initiation Protocol Gateway Call Flows*  
[http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/re1\\_docs/sip-flo/](http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/re1_docs/sip-flo/)

### Implementing a VoIP Network

- *Cisco IOS Voice Configuration Library*  
<http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123cgcr/vcl.htm>
- *Cisco IOS Voice Command Reference*  
<http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123tcr/123tvr/>
- *Cisco IOS IP Configuration Guide*  
[http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123cgcr/ip\\_veg.htm](http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123cgcr/ip_veg.htm)
- *Cisco IOS IP Command Reference, Volume 1 of 4: Addressing and Services*  
<http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123tcr/123tip1r/>
- *Cisco IOS IP Command Reference, Volume 2 of 4: Routing Protocols*  
<http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123tcr/123tip2r/>
- *Cisco IOS IP Command Reference, Volume 3 of 4: Multicast*  
<http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123tcr/123tip3r/>
- *Cisco IOS IP Command Reference, Volume 4 of 4: IP Mobility*  
<http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123tcr/123tip4r/>

**Tones**

- Telcordia document GR-506-CORE, *Signaling for Analog Interfaces*  
<http://telecom-info.telcordia.com>
- *Cisco BTS 10200 Softswitch System Description*  
<http://www.cisco.com/univercd/cc/td/doc/pcat/bts10200.htm>




---

**Note** Access to the Cisco BTS 10200 Softswitch System technical documentation set is restricted. Contact your Cisco representative for access information.

---

**Upgrading Firmware**

- *Cisco 7940 and 7960 IP Phones Firmware Upgrade Matrix*  
[http://www.cisco.com/univercd/cc/td/doc/product/voice/c\\_ipphon/english/ipp7960/addprot/mgcp/rmwrup.htm](http://www.cisco.com/univercd/cc/td/doc/product/voice/c_ipphon/english/ipp7960/addprot/mgcp/rmwrup.htm)
- *Cisco IP Phone 7960/40 Release Notes for SIP*

## Obtaining Documentation

Cisco documentation and additional literature are available on Cisco.com. Cisco also provides several ways to obtain technical assistance and other technical resources. These sections explain how to obtain technical information from Cisco Systems.

### Cisco.com

You can access the most current Cisco documentation on the World Wide Web at this URL:

<http://www.cisco.com/univercd/home/home.htm>

You can access the Cisco website at this URL:

<http://www.cisco.com>

International Cisco websites can be accessed from this URL:

[http://www.cisco.com/public/countries\\_languages.shtml](http://www.cisco.com/public/countries_languages.shtml)

### Ordering Documentation

You can find instructions for ordering documentation at this URL:

[http://www.cisco.com/univercd/cc/td/doc/es\\_inpck/pdi.htm](http://www.cisco.com/univercd/cc/td/doc/es_inpck/pdi.htm)

You can order Cisco documentation in these ways:

- Registered Cisco.com users (Cisco direct customers) can order Cisco product documentation from the Ordering tool:  
<http://www.cisco.com/en/US/partner/ordering/index.shtml>
- Nonregistered Cisco.com users can order documentation through a local account representative by calling Cisco Systems Corporate Headquarters (California, USA) at 408 526-7208 or, elsewhere in North America, by calling 800 553-NETS (6387).

# Documentation Feedback

You can submit e-mail comments about technical documentation to [bug-doc@cisco.com](mailto:bug-doc@cisco.com).

You can submit comments by using the response card (if present) behind the front cover of your document or by writing to the following address:

Cisco Systems  
Attn: Customer Document Ordering  
170 West Tasman Drive  
San Jose, CA 95134-9883

We appreciate your comments.

## Obtaining Technical Assistance

For all customers, partners, resellers, and distributors who hold valid Cisco service contracts, the Cisco Technical Assistance Center (TAC) provides 24-hour-a-day, award-winning technical support services, online and over the phone. Cisco.com features the Cisco TAC website as an online starting point for technical assistance. If you do not hold a valid Cisco service contract, please contact your reseller.

### Cisco TAC Website

The Cisco TAC website provides online documents and tools for troubleshooting and resolving technical issues with Cisco products and technologies. The Cisco TAC website is available 24 hours a day, 365 days a year. The Cisco TAC website is located at this URL:

<http://www.cisco.com/tac>

Accessing all the tools on the Cisco TAC website requires a Cisco.com user ID and password. If you have a valid service contract but do not have a login ID or password, register at this URL:

<http://tools.cisco.com/RPF/register/register.do>

### Opening a TAC Case

Using the online TAC Case Open Tool is the fastest way to open P3 and P4 cases. (P3 and P4 cases are those in which your network is minimally impaired or for which you require product information.) After you describe your situation, the TAC Case Open Tool automatically recommends resources for an immediate solution. If your issue is not resolved using the recommended resources, your case will be assigned to a Cisco TAC engineer. The online TAC Case Open Tool is located at this URL:

<http://www.cisco.com/tac/caseopen>

For P1 or P2 cases (P1 and P2 cases are those in which your production network is down or severely degraded) or if you do not have Internet access, contact Cisco TAC by telephone. Cisco TAC engineers are assigned immediately to P1 and P2 cases to help keep your business operations running smoothly.

To open a case by telephone, use one of the following numbers:

Asia-Pacific: +61 2 8446 7411 (Australia: 1 800 805 227)

EMEA: +32 2 704 55 55

USA: 1 800 553-2447

For a complete listing of Cisco TAC contacts, go to this URL:  
<http://www.cisco.com/warp/public/687/Directory/DirTAC.shtml>

## TAC Case Priority Definitions

To ensure that all cases are reported in a standard format, Cisco has established case priority definitions.

Priority 1 (P1)—Your network is “down” or there is a critical impact to your business operations. You and Cisco will commit all necessary resources around the clock to resolve the situation.

Priority 2 (P2)—Operation of an existing network is severely degraded, or significant aspects of your business operation are negatively affected by inadequate performance of Cisco products. You and Cisco will commit full-time resources during normal business hours to resolve the situation.

Priority 3 (P3)—Operational performance of your network is impaired, but most business operations remain functional. You and Cisco will commit resources during normal business hours to restore service to satisfactory levels.

Priority 4 (P4)—You require information or assistance with Cisco product capabilities, installation, or configuration. There is little or no effect on your business operations.

## Obtaining Additional Publications and Information

Information about Cisco products, technologies, and network solutions is available from various online and printed sources.

- Cisco Marketplace provides a variety of Cisco books, reference guides, and logo merchandise. Go to this URL to visit the company store:  
<http://www.cisco.com/go/marketplace/>
- The *Cisco Product Catalog* describes the networking products offered by Cisco Systems, as well as ordering and customer support services. Access the Cisco Product Catalog at this URL:  
<http://cisco.com/univercd/cc/td/doc/pcat/>
- *Cisco Press* publishes a wide range of general networking, training and certification titles. Both new and experienced users will benefit from these publications. For current Cisco Press titles and other information, go to Cisco Press online at this URL:  
<http://www.ciscopress.com>
- *Packet* magazine is the Cisco quarterly publication that provides the latest networking trends, technology breakthroughs, and Cisco products and solutions to help industry professionals get the most from their networking investment. Included are networking deployment and troubleshooting tips, configuration examples, customer case studies, tutorials and training, certification information, and links to numerous in-depth online resources. You can access Packet magazine at this URL:  
<http://www.cisco.com/packet>
- *iQ Magazine* is the Cisco bimonthly publication that delivers the latest information about Internet business strategies for executives. You can access iQ Magazine at this URL:  
<http://www.cisco.com/go/iqmagazine>

- *Internet Protocol Journal* is a quarterly journal published by Cisco Systems for engineering professionals involved in designing, developing, and operating public and private internets and intranets. You can access the Internet Protocol Journal at this URL:  
<http://www.cisco.com/ipj>
- Training—Cisco offers world-class networking training. Current offerings in network training are listed at this URL:  
<http://www.cisco.com/en/US/learning/index.html>





# Product Overview

---

This chapter provides the following information about the Cisco IP 7960G/7940G phone:

- New Information in This Release, page 1-1
- Cisco IP 7960G/7940G Phone Overview, page 1-2
- Session Initiation Protocol Overview, page 1-6
- BTXML Support, page 1-8
- Cisco CallManager XML Support, page 1-8
- Network Capabilities, page 1-9
- Configuration Features, page 1-9
- Signaling Support, page 1-10
- Dial-Plan and Messaging Support, page 1-10
- Routing and Proxy Support, page 1-11
- Supported Languages and Character Set, page 1-12
- Supported Protocols, page 1-13
- Where to Go Next, page 1-14

## New Information in This Release

### Release 6.0

- New digital-signal-processor (DSP) firmware has been released.
- DSP alarm-status and error-message-reporting capabilities are available using the **debug dsp** and **show dsp** commands. You can log your session by means of console or Telnet and save the log to a file.
- New tones and ringers are added for interoperation with the Cisco BTS 10200 Softswitch.
- Selectable tones and ringing patterns already known to the phone are available to use as alternate ring tones.
- Local call forwarding capability is added so that you can forward calls to a desired phone number or URL.
- Autoconnect capability that operates like an intercom is added. The lines that are in autoconnect mode are displayed on the phone with an autoconnect icon.

- Speed-dial functionality is added.
- Multiple-call-appearance capability is available so that you can configure the same directory number (DN) on multiple lines using the `linex_name` parameter.
- DHCP option 60 is enabled so that the phone can identify itself with vendor-specific information.
- Personal directory functionality is added for storage of up to 32 unique numbers. When the Add soft key is pressed, a screen appears, allowing you to enter a name and number.
- Call-statistics data can be presented upon termination of a call.
- A new clear message waiting indicator (MWI) command has been added for use with a console and Telnet in privileged user mode.

**Release 6.1**

- New configurable parameters have been added. See Appendix D, “Configurable Parameters for the SIP IP Phone.”

**Release 7.0 and 7.1**

- Universal application loader functionality has been added.
- Microphone sensitivity has been increased.
- UDP Fragmentation has been increased to 2434 bytes.
- New upgrade procedures are provided for Version 7.0 and 7.1. See the “Upgrading to Version 7.0 and Later Releases” section on page 4-5 of Chapter 4, “Managing Cisco SIP IP Phones.”

**Release 7.3 and 7.4**

- Caveats can be found on the product release notes page at the following URL:  
[http://www.cisco.com/univercd/cc/td/doc/product/voice/c\\_ipphon/english/ipp7960/addprot/sip/relnote/](http://www.cisco.com/univercd/cc/td/doc/product/voice/c_ipphon/english/ipp7960/addprot/sip/relnote/)

## Cisco IP 7960G/7940G Phone Overview

Cisco IP 7960G/7940G phones are full-featured telephones that can be plugged directly into a SIP network and can be used very much like a standard PBX telephone. Phone terminals can attach to the existing data network infrastructure, using 10BASE-T and 100BASE-T interfaces on an Ethernet switch.

When used with a voice-capable Ethernet switch, one that understands type of service (ToS) bits and can prioritize VoIP traffic, the phones eliminate the need for a traditional proprietary telephone set and key system or PBX.

The Cisco IP 7960G/7940G phone also supports an adjustable ring tone, a hearing-aid compatible handset, and a headset.

The Cisco IP 7960G/7940G phone complies with RFC 3261, as described in Appendix A, “Compliance with RFC 3261.”

See Figure 1-1 and Figure 1-2 to identify the buttons and hardware on your Cisco IP phone.



Figure 1-1 Cisco IP Phone 7960



Figure 1-2 Cisco IP Phone 7940



1	Handset with indicator light	The light strip at the top of the handset blinks when the phone rings and can be set to remain lit when there is a voice message.
2	LCD screen	Displays information about the Cisco IP phone, such as the time, date, phone number, caller ID, line and call status, and the soft key tabs. The screen is 4.25 x 3 inches (10.79 x 7.62 cm) and has an adjustable contrast.
3	Cisco IP Phone model type	Indicates the Cisco IP phone model.












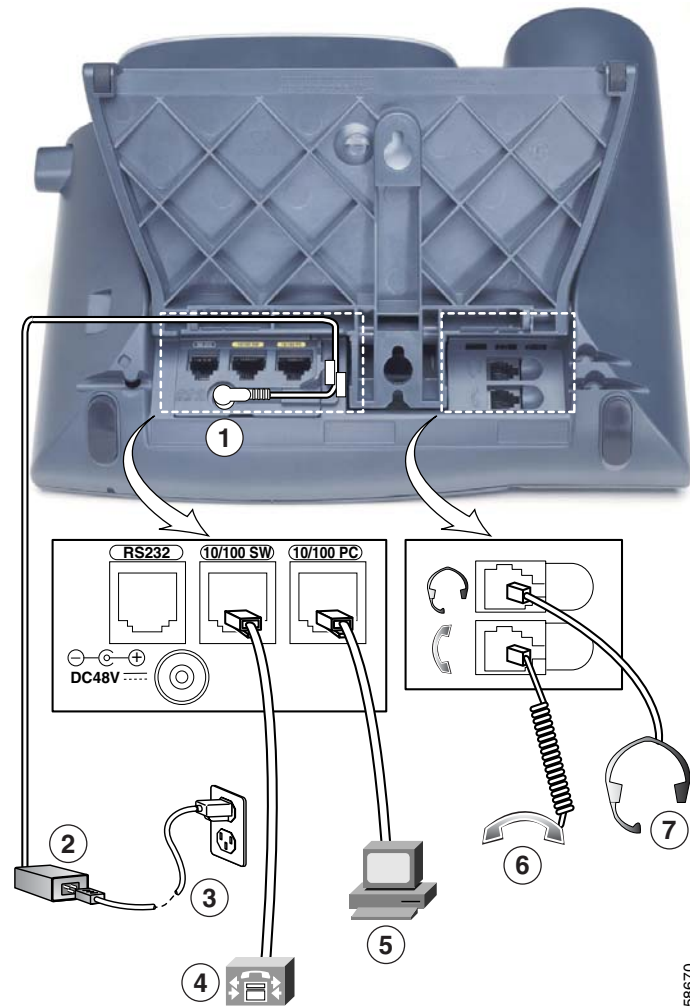
4	Line or speed-dial button 	Opens a new line or speed-dials the number on the LCD screen. Phones in the Cisco IP Phone 7960 series have six line or speed-dial buttons, and phones in the 7940 series have two.
5	Footstand adjustment	Allows adjustment of the angle of the phone base.
6	Directories button 	Provides access to call histories and directories.
7	<i>i</i> or ? button 	Provides online help for selected keys or features and network statistics about the active call. Pressing the button and then the up or down scroll key displays a descriptor of the key. For example, pressing the <i>i</i> or ? button and then the up or down scroll key displays a screen that instructs you how to scroll up and down on the LCD.
8	Settings button 	Provides access to phone settings such as contrast and ring sound, network configuration, and status information.
9	Speaker button 	Toggles the speaker on or off.
10	Mute button 	Toggles the mute on or off.
11	Headset button 	Toggles the headset on or off.
12	Volume button 	Increases or decreases the volume for the handset, headset, or speakerphone (depending upon which is currently active). Also controls the ringer volume (if the handset is in its cradle) and the LCD screen contrast.
13	Services button 	Provides access to any available phone services.
14	Messages button 	Provides access to a message system, if available.
15	Navigation button 	Allows scrolling through text and selection of features displayed on the LCD screen.
16	Dial pad	Works exactly like the dial pad on a traditional telephone.
17	Soft keys	Activates any functions displayed on the corresponding LCD screen tabs. Soft keys point to feature options displayed along the bottom of the LCD screen.

Figure 1-3 shows the connections on the back of the Cisco IP phone. Cisco IP 7960G/7940G phones have the same hardware configuration.

**Figure 1-3 Cisco IP Phone Cable Connections**



1	AC/DC adapter port (DC48V) for power connector. For redundancy, you can use the AC adapter even if you are using inline power from Cisco Catalyst switches. The Cisco IP 7960G/7940G phone can share the power being used from the inline power and external power source. If either the inline power or the external power goes down, the phone can switch entirely to the other power source.
2	Power supply with AC plug.
3	Power cable with wall socket plug for connecting to power.
4	Network port (10 and 100 SW) RJ-45 to connect the phone to the network supporting 10- or 100-Mbps half- or full-duplex Ethernet connections to external devices. You can use either Category 3 or Category 5 cabling for 10-Mbps connections, but use Category 5 for 100-Mbps connections. To avoid collisions, use full-duplex mode. You must use a straight-through cable on this port. The phone can also obtain inline power from the Cisco Catalyst switch over this connection.

5	Access port (10 and 100 PC) RJ-45 to connect a network device, such as a computer, to the phone supporting from 10- to 100- Mbps half- or full-duplex Ethernet connections to external devices. You can use either Category 3 or Category 5 cabling for 10-Mbps connections, but use Category 5 for 100-Mbps connections. To avoid collisions, use full-duplex mode. You must use a straight-through cable on this port.
6	Handset port for connecting a handset.
7	<p>Headset port for connecting a headset. Enables the headset. The phone supports a four- or six-wire headset jack. The volume and mute controls also adjust volume to the earpiece and mute the speech path of the headset. The headset activation key is located on the front of the Cisco IP 7960G/7940G phone.</p> <p>The phone supports the following Plantronics four- or six-wire headsets: Tristar Monaural, Encore Monaural H91, and Encore Binaural H101.</p> <p>When a headset is used, an amplifier is not required. However, a coil cord is required to connect the headset to the headset port on the back of your Cisco IP 7960G/7940G phone. For information on ordering compatible headsets and coil cords for the Cisco IP 7960G/7940G phone, go to <a href="http://cisco.getheadsets.com">http://cisco.getheadsets.com</a> or <a href="http://vxicorp.com/cisco">http://vxicorp.com/cisco</a>.</p>

## Session Initiation Protocol Overview

Session Initiation Protocol (SIP) is the Internet Engineering Task Force (IETF) standard for multimedia conferencing over IP. SIP is an ASCII-based, application-layer control protocol (defined in RFC 3261) that can be used to establish, maintain, and terminate calls between two or more endpoints.

Like other VoIP 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 Capabilities

SIP provides the capabilities to do the following:

- Determine the location of the target endpoint—SIP supports address resolution, name mapping, and call redirection.
- Determine the media capabilities of the target endpoint—Using Session Description Protocol (SDP), SIP determines the “lowest level” of common services between the endpoints. Conferences are established using only the media capabilities that can be supported by all endpoints.
- Determine 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 already on the phone or did not answer in the allotted number of rings. It then returns a message that indicates why the target endpoint was unavailable.
- Establish a session between the originating and target endpoint—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.
- Handle 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 between all parties.

Conferences can consist of two or more parties and can be established using multicast or multiple unicast sessions.

**Note**

The term *conference* means an established session (or *call*) between two or more endpoints. In this document, the terms conference and call are used interchangeably.

## SIP Components

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.

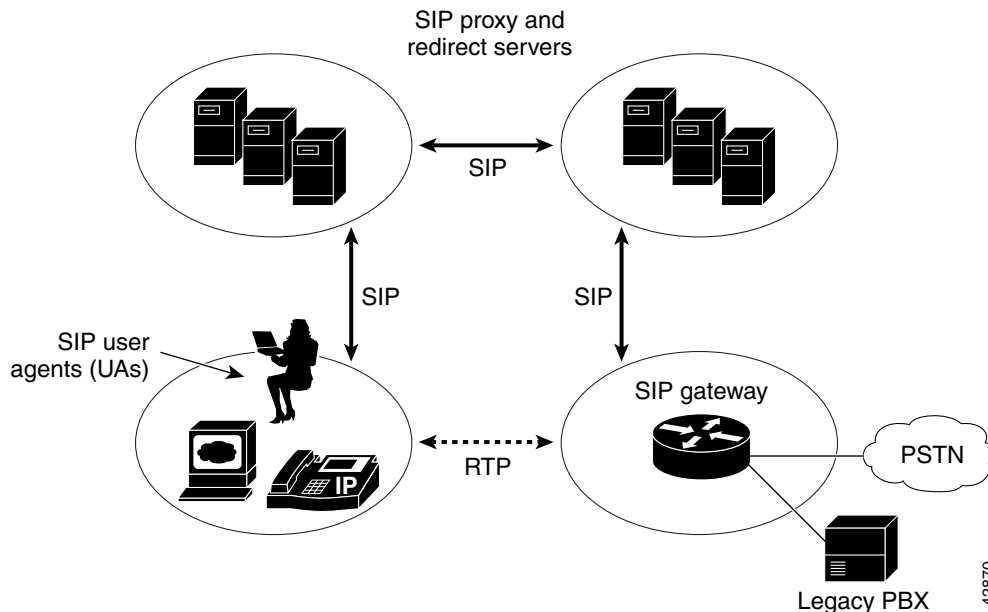
Typically, a SIP endpoint is capable of functioning as both a UAC and a UAS, but functions only as one or the other per transaction. Whether the endpoint functions as a UAC or a UAS depends on the UA that initiated the request.

From an architecture standpoint, the physical components of a SIP network can also be grouped into two categories: clients and servers. Figure 1-4 illustrates the architecture of a SIP network.

**Note**

In addition, the SIP servers can interact with other application services, such as Lightweight Directory Access Protocol (LDAP) servers, a database application, or an eXtensible Markup Language (XML) application. These provide back-end services such as directory, authentication, and billing.

**Figure 1-4 SIP Architecture**



## SIP Clients

SIP clients include the following:

- Phones

Phones act as either a UAS or a UAC. Softphones (PCs that have phone capabilities installed) and Cisco SIP IP phones can initiate SIP requests and respond to requests.

- Gateways

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 also translates between audio and video codecs and performs call setup and clearing on both the LAN side and the switched-circuit side.

## SIP Servers

SIP servers include the following:

- Proxy server

A proxy server receives SIP client messages and forwards them to the next SIP server in the network. Proxy servers can provide functions such as authentication, authorization, network access control, routing, reliable request retransmission, and security.

- Redirect server

A redirect server receives SIP requests, strips out the address in the request, checks its address tables for any other addresses that may be mapped to the one in the request, and then returns the results of the address mapping to the client. Basically, redirect servers provide the client with information about the next hop or hops that a message should take, and then the client contacts the next-hop server or UAS directly. Location services are often included with redirect servers and provide contact information and address bindings for called parties.

- Registrar server

A registrar server processes requests from UACs for registration of their current location. Registrar servers are often colocated with a redirect or proxy server.

## BTXML Support

The Cisco SIP IP phone supports Basic Telephony eXtensible Markup Language. BTXML defines XML elements for controlling the user interface of an IP telephone. It describes what information is displayed on the screen and how to provide input using soft keys and hard keys. User-interface control is internal to the phone; there is no external BTXML user interface control.

## Cisco CallManager XML Support

The Cisco SIP IP phone supports Cisco CallManager XML cards that you configure to provide data such as stock quotes, calendars, and directory lookups. Phone users access this information using phone buttons such as the Services or Directories buttons or soft keys. See Chapter 4, “Managing Cisco SIP IP Phones,” for information about configuring these cards.

For more information about using XML on your Cisco SIP IP phone, refer to the following:

- IP Telephony  
<http://www.hotdispatch.com/cisco-ip-telephony>
- Cisco Call Manager Services Developer Kit  
[http://www.cisco.com/cgi-bin/dev\\_support/access\\_level/product\\_support](http://www.cisco.com/cgi-bin/dev_support/access_level/product_support)
- *Developing Cisco IP Phone Services* by Darrick Deel, Mark Nelson, and Anne Smith, ISBN 1-58705-060-9

## Network Capabilities

The Cisco IP 7960G/7940G phone supports the following networking capabilities:

- Telnet support  
Allows you to use Telnet to connect directly to the phone to debug and troubleshoot the phone.
- Ping support  
Allows you to ping the phone to see if it is operational and assess how long the response time from the phone is.
- Traceroute support  
Allows you to see the path that the signal traverses in the route to its desired destination.

## Configuration Features

With the Cisco IP 7960G/7940G phone, you can do the following:

- Configure an Ethernet port mode and speed
- Register with or unregister from a proxy server or backup proxy server
- Specify a TFTP boot directory
- Configure a label for phone-identification display purposes
- Configure a name for caller identification purposes for each active line on a phone
- Configure a 12- or 24-hour user interface time display
- Lock and unlock the phone
- Configure local call forwarding
- Configure autoconnect (intercom)
- Configure speed dialing
- Configure the same directory number (DN) on several lines
- Configure the phone to collect terminating-call data for troubleshooting and billing purposes
- Clear the message-waiting indicator from the console (by means of the command-line interface or by using Telnet in privileged access mode)
- Configure voice activity detection (VAD) with the `enable_vad` parameter

# Signaling Support

The following signaling and transport features are supported:

- G.711 (mu-law and a-law) and G.729a audio compression
- In-band dual tone multifrequency (DTMF) support for codecs for touch-tone dialing
- Out-of-band DTMF signaling (RFC 2833)

# Dial-Plan and Messaging Support

The Cisco SIP IP phone supports a dial plan that enables automatic dialing and generation of a selectable secondary dial tone. Direct entries must be saved using the Keep soft key in order to be stored in the personal directory. The Cisco SIP IP phone also supports the following:

- Local directory configuration (save and recall) and automatic dial completion

Each time a call is successfully made or received, the number is stored in a local directory that is maintained on the phone. The maximum number of entries is 32. Entries are aged out according to usage and age. The oldest entry that has been called the least number of times is overwritten first. This feature cannot be changed by the phone user; however, up to 20 entries can be “locked” (using the Locked soft key) so that they will never be deleted.

- Message waiting indication using unsolicited NOTIFY

When a caller leaves a voice message, a light indicates that a new voice message is in your mailbox. If you listen to the message but do not save or delete the message, the light remains on. If you listen to the new message and save or delete it, the light goes off. The message waiting indicator is controlled by the voice-mail server. The indication is saved over a phone upgrade or reboot.

- Dial-plan support for the asterisk (\*) and pound (#) characters

In the current dial plan, the “\*” and “#” characters are not matched if specified as dialed digits. The “\*” character is a wildcard that equates to “match one or more characters.” Support was added for the “\*” character by adding “\” as an escape character and the “#” character as a digit in the dial plan. Backward compatibility for the “#” and “\*” characters is supported.

- Distinctive alerting

If the INVITE message from the server contains an Alert-Info header, distinctive ringing is invoked. The format of the header is “Alert-info: x.” The value of “x” can be any number. This header is received only by the phone and is not generated by the phone.

Distinctive ringing is supported when the phone is idle or during a call. In the idle mode, the phone rings with a different cadence. The selected ringing type plays twice with a short pause in between. In call-waiting mode, two short beeps are generated instead of one long beep.

- Configurable and user-selectable dial tone

In earlier releases, specifying a comma (,) in the dial plan caused the phone to play the default secondary dial tone. With this release, specific user-specified tones can play. You can specify up to three different secondary dial tones in a single dial-plan match template. Tones play in the order in which they are listed.

- Ring-back tone for calls on hold

The parameter is selectable from the Settings > Call Preferences menu. When the parameter is enabled, the phone rings if the handset is placed on-hook and there is also a call on hold.

- Dial-plan update support using the Event header in NOTIFY messages.



- Local (180 Ringing) or remote (183 Session Progress) call progress tone.
- Call redirection information support using the Diversion header.

## Routing and Proxy Support

The Cisco SIP IP phone supports the following routing and proxy features:

- Proxy routing

The route attribute in the dial-plan template file can be used to indicate to which proxy—default, emergency, fully qualified domain name (FQDN)—the call should be initially routed. For example, to configure an emergency proxy, specify the value of the route attribute as “emergency.”

- Backup SIP proxy

When the primary proxy does not respond to the INVITE message sent by the phone after the configured number of retries, the phone sends the INVITE to the backup proxy. This is independent from the proxy defined in the route attribute in the dial-plan template used.

The Cisco SIP IP phone attempts to register with the backup proxy. All interactions with the backup proxy, such as authentication challenges, are treated the same as the interactions with the primary proxy. The backup proxy is used only with new INVITE messages that fail to communicate with the primary proxy. Once the backup proxy is used, it is active for the duration of the call.

The location of the backup SIP proxy can be defined as an IP address in the default configuration file.

- Emergency SIP proxy

An optional emergency SIP proxy can be configured with the route attribute of the template tag in the dial-plan template file. When an emergency SIP proxy is configured and a call is initiated, the phone generates an INVITE message to the address specified in the proxy\_emergency parameter. The emergency proxy is used for the entire call.

The location of the emergency proxy can be defined as an IP address in the default configuration file.

- Support of the DNS SRV query

The Domain Name Server (DNS) SRV query is used to locate servers for a given service. SIP on Cisco IP phones uses a DNS SRV query to determine the IP address of the SIP proxy or redirect server. The query string generated is in compliance with RFC 2782 and prepends the protocol label with an underscore (\_), as in “\_protocol.\_transport.” The addition of the underscore reduces the risk of the same name being used for unrelated purposes.

In compliance with RFC 2782 and the draft-ietf-sip-srv-01 specification, the system can remember multiple IP addresses and use them properly. In the draft-ietf-sip-srv-01 specification, it is assumed that all proxies returned for the SRV record are equivalent such that the phone can register with any of the proxies and initiate a call using any other proxy.

- Network Address Translation (NAT) and outbound proxy

Enable or disable NAT with the nat\_enable parameter. You can configure the address of the NAT or firewall server using the nat\_address parameter.

You can configure the IP address and port number of the outbound proxy server. When outbound proxy is enabled, all SIP requests are sent to the outbound proxy server instead of to the proxyN\_address. All responses continue to reconcile the normal Using processing rules. The media stream is not routed through the outbound proxy.

You can enable or disable NAT and outbound proxy modes independently. The `received=` tag is added to the `Via` header of all responses if there is no `received=` tag in the uppermost `Via` header and the source IP address is different from the IP address in the uppermost `Via` header. Responses are sent back to the source under the following conditions:

- If a `received=` tag is in the uppermost `Via` header, the response is sent back to the IP address contained in the `received=` tag.
- If there is no `received=` tag and the IP address in the uppermost `Via` header is different from the source IP address, the response is sent back to the source IP address. Otherwise the response is sent back to the IP address in the uppermost `Via` header.

**Note**

For information on how to use the standard telephony features and URL dialing, refer to the documents listed in the “Related Documentation” section on page ix.

## Supported Languages and Character Set

The Cisco SIP IP phone supports the ISO 8859-1 Latin1 characters and the following languages: French (fr), Spanish (es), Catalan (ca), Basque (eu), Portuguese (pt), Italian (it), Albanian (sq), Rhaeto-Romanic (rm), Dutch (nl), German (de), Danish (da), Swedish (sv), Norwegian (no), Finnish (fi), Faroese (fo), Icelandic (is), Irish (ga), Scottish (gd), English (en), Afrikaans (af), and Swahili (sw).

It does not support the following languages: Zulu (zu) and other Bantu languages that use Latin Extended-B letters; Arabic in North Africa; and Guarani (gn), which is missing the letters G, E, I, U, and Y with tildes (~).

**Note**

The XML cards, information text, and menus are all in English. These items are built into the phone image and cannot be changed.

You can use ISO 8859-1 Latin1 characters in the following areas:

- Caller ID information. When a SIP message is received with ISO 8859-1 Latin1 characters in the caller ID strings, those caller ID strings are displayed on the phone LCD with the correct ISO 8859-1 Latin1 characters.
- Services menu applications written in Cisco CallManager XML. You can develop language-specific applications for a particular region. For example, an application that displayed the current weather in Sweden using Swedish characters can be displayed on the Cisco SIP IP phone. If you develop the same application for a Spanish locale, the application can be translated into Spanish.
- Line key labels. The line keys can be configured to support the Latin1 characters. You can specify the line key name in the configuration file, and it displays correctly. The Latin1 characters cannot be used in the `linex_name` parameter, but can be used in the `linex_shortcode` and `linex_displayname` parameters. If the proxy supports Latin1 characters in the `To/From` headers, they can be used in the `linex_name` parameter as well.

**Note**

The *i* button text and the Settings menu are in English. These items are built into the phone image and cannot be changed.

# Supported Protocols

The Cisco SIP IP phone supports the following protocols.

**Table 1-1 Supported Protocols**

Protocol	Description
AVT	Audio video transport. Handles payload negotiation.
DHCP	Dynamic Host Configuration Protocol. Dynamically allocates and assigns IP addresses. DHCP allows you to move network devices from one subnet to another without administrative attention. It allows connection of Cisco SIP IP phones to the network so that they become operational without having to manually assign an IP address and additional network parameters. DHCP option 60 allows the Cisco SIP IP phone to identify itself with vendor-specific information. The Cisco SIP IP phone complies with DHCP specifications documented in RFC 2131. By default, the phone is DHCP-enabled.
DNS	Domain Name System. Translates names of network nodes into addresses. SIP uses DNS to resolve the host names of endpoints into IP addresses.
Dynamic DNS and TFTP	You can configure additional DNS and TFTP servers. Upon bootup, the phone first goes to the default TFTP server to download the configuration files. If a new dynamic TFTP server is specified in the files, the phone requests a new set of files from the specified server. If new DNS addresses are specified in the files, the phone uses those addresses for lookups.
HTTP	Hypertext Transfer Protocol. The phone contains limited support for HTTP 1.1. The Cisco SIP IP phone uses HTTP to retrieve Cisco CallManager XML files.
ICMP	Internet Control Message Protocol. A network-layer Internet protocol that enables hosts to send error or control messages to other hosts. ICMP also provides other information relevant to IP packet processing. The Cisco SIP IP phone supports ICMP as defined in RFC 792.
IP	Internet Protocol. A network layer protocol that sends datagram packets between nodes on the Internet. IP also provides features for addressing, type-of-service (ToS) specification, fragmentation and reassembly, and security. The Cisco SIP IP phone supports IP as defined in RFC 791.
RTP	Real-Time Transport Protocol. Supports transport of real-time data (such as voice) over data networks. RTP also has the ability to obtain quality-of-service (QoS) information. The Cisco SIP IP phone supports RTP as a media channel.
SDP	Session Description Protocol. An ASCII-based protocol that describes multimedia sessions and their related scheduling information. Third-party call control is supported using delayed media negotiation, which is SDP data that is not completely advertised in the initial call setup. SDP also supports endpoints specified as fully qualified domain names (FQDNs). The Cisco SIP IP phone uses SDP for session description.

**Table 1-1 Supported Protocols (continued)**

Protocol	Description
SNTP	Simple Network Time Protocol. Synchronizes computer clocks on an IP network. Current date and time are supported using SNTP including time zone and daylight saving time. The Cisco SIP IP phone uses SNTP for date and time support.
TCP	Transmission Control Protocol. Provides a reliable byte-stream transfer service between endpoints on the Internet. The Cisco SIP IP phone supports TCP for Telnet sessions only.
TFTP	Trivial File Transfer Protocol. Allows files to be transferred from one computer to another over a network. The Cisco SIP IP phone uses TFTP to download configuration files and software updates.
ToS	Type of service. An indication of how an upper-layer protocol requires a lower-layer protocol to treat its messages. In SNA subarea routing, ToS definitions are used by subarea nodes to determine the optimal route to establish a given session. A ToS definition comprises a virtual route number and a transmission priority field. Also called class of service (CoS).
UDP	User Datagram Protocol. Exchanges data packets without acknowledgments or guaranteed delivery. A SIP network can use UDP as the underlying transport protocol. If UDP is used, retransmissions are used to ensure reliability. UDP fragmentation is supported. The Cisco SIP IP phone supports UDP as defined in RFC 768 for SIP signaling.
VAD	Voice activity detection. When enabled on a voice port or a dial peer, silence is not transmitted over the network, only audible speech. Sound quality is slightly degraded, but the connection monopolizes much less bandwidth.

## Where to Go Next

- See Chapter 2, “Installing Cisco IP 7960G/7940G Phone Hardware on the Desktop or Wall,” for placement of the phone on the desktop or wall and cabling information.
- See Chapter 3, “Initializing Cisco SIP IP Phones,” for information on installing firmware, customizing configuration files, and connecting the phone to power sources and the network.
- See Chapter 4, “Managing Cisco SIP IP Phones,” for information on upgrading firmware and performing other management tasks.
- See Chapter 5, “Monitoring Cisco SIP IP Phones,” for information on debugging and on viewing network statistics.



# Installing Cisco IP 7960G/7940G Phone Hardware on the Desktop or Wall

---

This chapter explains how to install the Cisco IP 7960G/7940G phone on the desktop or how to mount it to the wall. It provides information on the following:

- Placing the Phone on the Desktop, page 2-1
- Installing the Phone on the Wall, page 2-1
- Cabling the Phone Ports, page 2-2
- Using the Phone with a Cisco Catalyst Switch, page 2-3
- Connecting the Phone to Power, page 2-4
- Where to Go Next, page 2-5

## Placing the Phone on the Desktop

You can place the phone on the desktop or on the wall. The Cisco IP phone includes an adjustable footstand. When placing the phone on a desktop surface, you can adjust the tilt height to several different angles in 7.5-degree increments from flat to 60 degrees.

## Installing the Phone on the Wall

Mount the Cisco IP 7960G/7940G phone on the wall by using the footstand as a mounting bracket or by using the optional locking bracket. To use the optional locking bracket, refer to the *Installing the Wall Mount Kit for the Cisco IP Phone* document on Cisco.com.



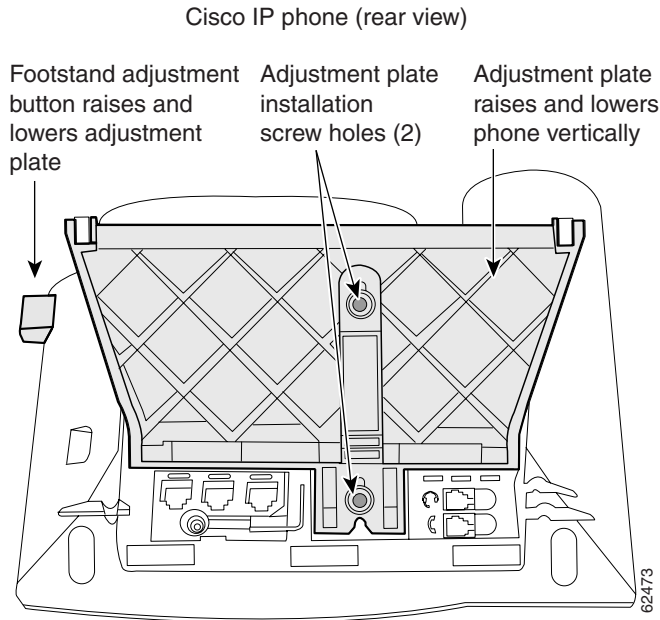
### Note

Mounting the Cisco IP 7960G/7940G phone on the wall requires some tools and equipment that are not provided. The tools and parts required for a typical Cisco IP phone installation are a screwdriver and screws to secure the phone to the wall.

### Procedure

---

- Step 1** Push in the footstand adjustment knob.
- Step 2** Adjust the footstand so that it is flat against the back of the phone, as shown in Figure 2-1.

**Figure 2-1 Adjusting the Footstand**

- Step 3** Modify the handset rest so that the handset remains on the earpiece rest when the phone is vertical.
- a. Remove the handset from the earpiece rest.
  - b. Locate the tab (handset wall hook) at the base of the earpiece rest.
  - c. Slide this tab out, rotate it 180 degrees, and reinsert it.
  - d. Place the handset on the earpiece rest.
- Step 4** Insert two screws into a wall stud, matching them to the two screw holes on the back of the footstand. The screw holes fit standard phone jack mounts.
- Step 5** Hang the phone on the wall.

## Cabling the Phone Ports

The Cisco MGCP IP phone has several ports on the back of the phone, shown in Figure 1-3 on page 1-5. Plug the appropriate equipment into the appropriate port.



### Caution

Consider use of an uninterruptible power supply (UPS). Without one, when you use either a local transformer or inline power on the LAN, the phone is inoperable during a power outage. This affects your ability to make emergency calls: 911 in USA and Canada, 999 in the UK, and 112 in mainland Europe.



### Note

Use of a headset does not require an amplifier. However, it does require a coil cord to connect the headset to the headset port. For information on ordering compatible headsets and coil cords, refer to <http://cisco.getheadsets.com> or <http://vxicorp.com/cisco>.

**Note**

Do not connect the phone to power or the network at this time. Doing so causes the phone to initialize and download firmware and configuration files. Initialization and modification of the configuration files are described in the Chapter 3, “Initializing Cisco SIP IP Phones.”

## Using the Phone with a Cisco Catalyst Switch

To function in the SIP telephony network, the Cisco IP 7960G/7940G phone must be connected to a networking device, such as a Cisco Catalyst switch, to obtain network connectivity. The Cisco IP 7960G/7940G phone has an internal Ethernet switch, which enables it to switch traffic that comes from the phone, access port, and network port.

If a computer is connected to the access port, packets that travel to and from the computer and to and from the phone share the same physical link to the switch and the same port on the switch. This configuration has the following implications for the VLAN configuration on the network:

- The current VLANs might be configured on an IP subnet basis, and additional IP addresses might not be available to assign the phone to the same subnet as other devices (PC) connected to the same port.
- Data traffic that is present on the VLAN that supports phones might reduce the quality of VoIP traffic.

You can resolve these issues by isolating the voice traffic onto a separate VLAN on each of the ports connected to a phone. The switch port configured for connecting a phone would have separate VLANs configured for carrying the following traffic:

- Voice traffic to and from the Cisco IP 7960G/7940G phone (auxiliary VLAN).
- Data traffic to and from the PC connected to the switch through the access port of the Cisco IP 7960G/7940G phone (native VLAN).

Isolating the phones on a separate, auxiliary VLAN increases the quality of the voice traffic and allows a large number of phones to be added to an existing network where there are not enough IP addresses.

For redundancy, you can use the AC adapter even if you are using inline power from the Cisco Catalyst switches. The Cisco IP 7960G/7940G phone can share the power load being used from the inline power and external power source. If either the inline power or the external power goes down, the phone can switch entirely to the other power source.

To use this redundancy feature you *must* set the inline power mode to “auto” on the Cisco Catalyst switch. Next, connect the Cisco IP 7960G/7940G phone to the network without connecting it to an external power source. After the phone powers up, connect the external power supply to the phone.

For more information, refer to the documentation included with the Cisco Catalyst switch.

# Connecting the Phone to Power



## Warning

**Read the installation instructions before you connect the system to its power source.**

**Waarschuwing** Raadpleeg de installatie-aanwijzingen voordat u het systeem met de voeding verbindt.

**Varoitus** Lue asennusohjeet ennen järjestelmän yhdistämistä virtalähteeseen.

**Attention** Avant de brancher le système sur la source d'alimentation, consulter les directives d'installation.

**Warnung** Lesen Sie die Installationsanweisungen, bevor Sie das System an die Stromquelle anschließen.

**Avvertenza** Consultare le istruzioni di installazione prima di collegare il sistema all'alimentatore.

**Advarsel** Les installasjonsinstruksjonene før systemet kobles til strømkilden.

**Aviso** Leia as instruções de instalação antes de ligar o sistema à sua fonte de energia.

**¡Advertencia!** Ver las instrucciones de instalación antes de conectar el sistema a la red de alimentación.

**Varning!** Läs installationsanvisningarna innan du kopplar systemet till dess strömförsörjningsenhet.



## Caution

When you connect the phone to power, the phone automatically begins startup and initialization. Use this section to determine how you will power your phones. Do not connect the phones to power, though, until you address the prerequisites and follow the configuration instructions in Chapter 3, “Initializing Cisco SIP IP Phones.”

You can connect the Cisco IP phone to the following power sources:

- External power source—Optional AC adapter and power cord for connecting to a standard wall receptacle.
- WS-X6348-RJ45V 10/100 switching module—Provides inline power to the phone when connected to a Cisco Catalyst 3500, 4000, or 6000 family 10/100BASE-TX switching module. This module sends power on pins 1 and 2, and 3 and 6.
- WS-PWR-PANEL—Power patch panel provides power to the phone, which allows the phone to be connected to existing Cisco Catalyst 4000, 5000, and 6000 family 10/100BASE-TX switching modules. This module sends power on pins 4, 5, 7, and 8.
- WS-X4148-RJ45V—48-port 10/100 Ethernet with inline power module for the Cisco Catalyst 4006 switch.
- WS-X4095-PEM—VoIP DC Power Entry module for the Cisco Catalyst 4006 switch.
- WS-X4608-2PSU and WS-X4608—External –48VDC power shelf common equipment for the Cisco Catalyst 4006 with two AC-to-DC power supply units (PSUs) and one empty bay for redundant option, and the 110-V 15-A AC-to-48VDC PSU redundant option for the power shelf.
- WS-C3524-PWR-XL-EN—Cisco Catalyst 3524-PWR XL switch.



## Note

- Only the network port (labeled 10/100 SW) supports inline power from the Cisco Catalyst switches.
- If you are using a Cisco Catalyst switch to provide power, see the “Using the Phone with a Cisco Catalyst Switch” section on page 2-3.



## Where to Go Next

- See Chapter 3, “Initializing Cisco SIP IP Phones,” for information on installing firmware, customizing configuration files, and connecting the phone to power sources and the network.
- See Chapter 4, “Managing Cisco SIP IP Phones,” for information on upgrading firmware and performing other management tasks.
- See Chapter 5, “Monitoring Cisco SIP IP Phones,” for information on debugging and on viewing network statistics.

■ Where to Go Next



## Initializing Cisco SIP IP Phones

This chapter describes the initial firmware installation tasks and configuration process for the Cisco IP 7960G/7940G phone in a Session Initiation Protocol (SIP) network. It provides information on the following:

- Prerequisites, page 3-1
- Overview of the Initialization Process, page 3-2
- Information About Configuration Files, page 3-4
- How to Customize the Default Configuration File, page 3-5
- How to Customize a Phone-Specific Configuration File, page 3-8
- How to Customize the Configuration from the Phone Menu, page 3-10
- How to Set the Date and Time, page 3-17
- How to Create Dial Plans, page 3-22
- How to Verify Initialization, page 3-26
- Where to Go Next, page 3-26

## Prerequisites

### Installation Strategy

Choose one of the following installation strategies:

- Download, then customize. Download the firmware image and configuration files to your TFTP server. Connect each phone to power, causing it to automatically download the image and default files. Configure each phone individually as needed.
- Customize, then download. Download the firmware image and configuration files to your TFTP server. Open the configuration files and customize parameters for all the phones at once. Save the customized file to the TFTP server. Connect each phone to power, causing it to automatically download the image and customized files.

### Network Functionality

Ensure that your network meets the following requirements:

- A working IP network is established and configured for SIP.

For information on configuring IP, refer to the *Cisco IOS IP Configuration Guide*.

[http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123cgcr/ip\\_vcg.htm](http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123cgcr/ip_vcg.htm)

- VoIP is configured on your Cisco routers.

For information on configuring VoIP, refer to the *Cisco IOS Voice Configuration Library*.  
<http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123cgr/vcl.htm>

VoIP gateways are configured for SIP.

- A TFTP server is configured on your network.

When the phone initializes, it requests the following from the TFTP server:

- Latest firmware image
- Dual-boot file (OS79XX.TXT)
- Phone-specific MAC-address configuration file
- Default configuration file
- Ring-list file
- Dial-plan file

For information about configuring your TFTP server, refer to your operating-system documentation.

- A DHCP server is configured on your network.

The phone can use DHCP to obtain IP addresses. Configuration options are as follows:

- dhcp option #1 (IP subnet mask)
- dhcp option #3 (default IP gateway)
- dhcp option #6 (DNS server IP address)
- dhcp option #15 (domain name)
- dhcp option #50 (IP address)
- dhcp option #66 (TFTP server IP address)

If you do not configure DHCP options on the DHCP server, you must manually configure them on the phone. For information on configuring a DHCP server, refer to your operating-system documentation.

- A proxy server is active and configured to receive and forward SIP messages.



**Note**

---

Refer to the *Cisco 7940 and 7960 IP Phones Firmware Upgrade Matrix* for additional prerequisites.

---

## Overview of the Initialization Process

The initialization process for the Cisco SIP IP phone establishes network connectivity and makes the phone operational in your SIP network. After you connect your phone to the network and to a power supply, the phone begins initialization, during which the following occurs:

1. The phone loads the firmware image.

The phone has nonvolatile flash memory that contains permanent factory information about the phone and, eventually, firmware images and user-defined preferences. During initialization, the phone runs a bootstrap loader that loads the firmware image.

2. The phone learns its VLAN membership.

If the phone is connected to a Cisco Catalyst switch, the switch notifies the phone of the voice VLAN defined on the switch. The phone needs to know its VLAN membership before it can send a DHCP request for its IP settings (if using DHCP).

3. The phone acquires its IP address.

If the phone uses DHCP to obtain IP settings, it queries the DHCP server. Otherwise, it uses IP settings that are stored in flash memory.

4. The phone contacts the TFTP server and downloads the following files (or uses settings that are stored in its flash memory):

- SEP<macaddress>.cnf.xml—Creates the filename SEP<macaddress>.cnf.xml on the TFTP server into which you can place one of the following:

```
<device>
<loadInformation>POS3-07-3-00</loadInformation>
</device>
```

The phone then checks the load information and either upgrades the phone firmware in FLASH memory to the version stated in the <LoadInformation> tag using the TFTP loader in the Universal Application Loader, or, if the version matches, exits the Universal Application Loader and executes the firmware already loaded in FLASH memory.

- <firmware-version>.loads—If the version matches, enables the phone to exit the Universal Application Loader and executes the firmware already loaded in FLASH memory, as defined in SEP<macaddress>.cnf.xml.
- OS79XX.TXT—Enables the phone to initialize and automatically determine the network in which it is being installed.


**Note**

The use of dual boot file OS79XX.TXT is deprecated in favor of individual XML configuration files for the phone using their SEP<macaddress>.cnf.xml style names. This allows Cisco CallManager and SIP-based configurations to share a common TFTP server, as the XML configuration is phone-specific and allows individual phones to be switched between SIP, SCCP, or MGCP images.

- SIPDefault.cnf—Contains parameters intended for all phones. For information on customizing the file, see the “How to Customize the Default Configuration File” section on page 3-5.
- SIP<mac-addr>.cnf—Contains parameters specific to a phone. Use this file as a template from which to create a file for each phone. Insert the MAC address of the phone in the filename.
- RINGLIST.DAT—Lists audio files that are the custom-ring-type options for the phones. The files must be in the root directory of the TFTP server.
- dialplan.xml—Contains the North American sample dial plan. You can push the file down to the phones with a notify (NTFY) message with a check-sync Event header.


**Note**

Refer to the *Cisco 7940 and 7960 IP Phones Firmware Upgrade Matrix* for additional information.

5. The phone verifies the firmware version.

If the phone determines that the image defined in a configuration file differs from the image that it has stored in flash memory, it performs a firmware upgrade. During upgrade, the phone downloads the firmware image from the TFTP server, programs the image into flash memory, and reboots.

**Note**

Upon startup, the phone attempts to download both configuration files. If neither file exists, a TFTP timeout occurs after approximately 9 seconds per file. If the files exist, they are parsed and processed. These files are not required for the phone to initialize; however, it takes longer (approximately 20 seconds) for the phone to boot because it is waiting for the timeout on the TFTP server. Both configuration files can use the same values, or they can contain empty values. If the files contain empty values, the phone boots using default values for some of the parameters.

**Note**

Values in the phone-specific configuration file take precedence over those in the default configuration file because the phone-specific file is processed last.

## Information About Configuration Files

Configuration files reside in a TFTP server subdirectory (you specify the location of this subdirectory with the `tftp_cfg_dir` parameter). For more information, refer to the *Cisco 7940 and 7960 IP Phones Firmware Upgrade Matrix*.

**Note**

Be sure to customize configuration files *before* you power up the phone. When powered up, the phone automatically loads parameters stored in flash memory and then requests configuration files from the TFTP server.

When modifying parameters, remember the following:

- Parameters in the configuration file override those stored in the phone's flash memory.
- Locally changed parameters are used until the next reboot.
- The name of each phone-specific configuration file is unique and is based on the MAC address of the phone.

The format of the filename must be `SIPXXXXYYYYZZZZ.cnf`, where `XXXXYYYYZZZZ` is the MAC address of the phone. The MAC address must be in uppercase; the `.cnf` extension must be in lowercase (for example, `SIP00503EFFD842.cnf`).

**Note**

You can find the MAC address of a phone on the middle sticker adhered to the base of the phone. You can also view it on the Network Configuration menu.

- Each line in a configuration file must use the following format and must adhere to the following rules:

```
variable-name : value ; optional comments
```

- Associate only one value with one variable.
- Separate variable names and values with colons.
- Set only one variable per line.
- Indicate the end of a line with `<lf>` or `<cr><lf>`.
- Put the variable and value on the same line, and do not break the line.

- You can include white space before or after a variable or value. You can include any character within them. However, if white spaces are needed within the value, you must enclose the value in single or double quotes. If the value is enclosed in quotes, the end quote must be the same as the start quote.
- You can include comments after the value. Use the semicolon (;) and pound (#) delimiters to distinguish the comments.
- You can include comment lines.
- You can include blank lines.
- You can use any case for variables; they are not case sensitive.

## How to Customize the Default Configuration File

You have the following initialization choices:

- Download, then customize. Download the default configuration file to your TFTP server, and then plug each phone into power and the network. The phones automatically download the default configuration file from the TFTP server. You can then customize parameters if required.
- Customize, then download. Download the default configuration file to your TFTP server, open the file, customize parameters for all the phones at once, save the customized file, and then plug the phones into power and the network. The phones automatically download the customized file from the TFTP server.

This section describes how to customize, then download. Maintaining parameters—such as whether phones must register with a proxy server and the codec that phones must use when initiating a call—in the default configuration file allows you to perform global changes, such as upgrading the image version, without having to customize the phone-specific configuration file for each phone.

**Note**

---

For a complete alphabetical list of configurable parameters, see Appendix D, “Configurable Parameters for the SIP IP Phone.”

---

**Prerequisites**

- If you have an existing system from a release earlier than Release 7.x, upgrade your system firmware as described in the “How to Upgrade Your Cisco SIP IP Phone Firmware Image” section on page 4-3 before proceeding.

**Procedure**

- 
- Step 1** Obtain the default configuration file as follows:
- a. Go to the Cisco.com SIP IP 7940/7960 phone software-download site at <http://www.cisco.com/cgi-bin/tablebuild.pl/sip-ip-phone7960>.
  - b. Download the SIPDefault.cnf file to the root directory of your TFTP server or to a subdirectory in which all phone-specific configuration files are stored.
- Step 2** Using an ASCII text editor such as vi, open the file.
- Step 3** Modify the following required parameters:
- line1\_name—Number or e-mail address for use when registering. Enter a number without dashes. For example, enter 555-0100 as 5550100. Enter an e-mail ID without the host name.

- proxy1\_address—IP address of the SIP proxy server that is used by the phones. Enter the address in IP dotted-decimal notation or use the FQDN.
- proxy1\_port—Port number of the SIP proxy server that is used by line 1.

**Step 4** Modify additional parameters as needed.

**Step 5** Save the file to the root directory of your TFTP server or to a subdirectory in which all phone-specific configuration files are stored.

### Configuration Example

The following is an example of the SIPDefault configuration file that you downloaded from Cisco.com:

```
# SIP Default Configuration File

# Image Version
image_version: POS3-06-0-00

# Proxy Server
proxy1_address: 172.16.255.255
proxy2_address: ""; Can be dotted IP or FQDN
proxy3_address: ""; Can be dotted IP or FQDN
proxy4_address: ""; Can be dotted IP or FQDN
proxy5_address: ""; Can be dotted IP or FQDN
proxy6_address: ""; Can be dotted IP or FQDN

# Proxy Server Port (default - 5060)
proxy1_port: 5060
proxy2_port: 5060
proxy3_port: 5060
proxy4_port: 5060
proxy5_port: 5060
proxy6_port: 5060

# Proxy Registration (0-disable (default), 1-enable)
proxy_register: 0

# Phone Registration Expiration [1-3932100 sec] (Default - 3600)
timer_register_expires: 3600

# Codec for media stream (g711ulaw (default), g711alaw, g729a)
preferred_codec: g711ulaw

# TOS bits in media stream [0-5] (Default - 5)
tos_media: 5

# Inband DTMF Settings (0-disable, 1-enable (default))
dtmf_inband: 1

# Out of band DTMF Settings
#(none-disable, avt-avt enable (default), avt_always-always avt)
dtmf_outofband: avt

# DTMF dB Level Settings
#(1-6dB down, 2-3db down, 3-nominal (default), 4-3db up, 5-6dB up)
dtmf_db_level: 3

# SIP Timers
timer_t1: 500; Default 500 msec
timer_t2: 4000; Default 4 sec
sip_retx: 10; Default 10
```



```

sip_invite_retx: 6; Default 6
timer_invite_expires: 180 ; Default 180 sec

##### New Parameters added in Release 2.0 #####

# Dialplan template (.xml format file relative to the TFTP root directory)
dial_template: dialplan

# TFTP Phone Specific Configuration File Directory
tftp_cfg_dir: ""; Example: ./sip_phone/

# Time Server
#(There are multiple values and configurations refer to Admin Guide for Specifics)
sntp_server: ""; Sntp Server IP Address
sntp_mode: anycast (default); unicast, multicast, or directedbroadcast
time_zone: EST; Time Zone Phone is in
dst_offset: 1; Offset from Phone's time when DST is in effect
dst_start_month: April; Month in which DST starts
dst_start_day: ""; Day of month in which DST starts
dst_start_day_of_week: Sun; Day of week in which DST starts
dst_start_week_of_month: 1; Week of month in which DST starts
dst_start_time: 02; Time of day in which DST starts
dst_stop_month: Oct; Month in which DST stops
dst_stop_day: ""; Day of month in which DST stops
dst_stop_day_of_week: Sunday; Day of week in which DST stops
dst_stop_week_of_month: 8; Week of month in which DST stops 8=last week of month
dst_stop_time: 2; Time of day in which DST stops
dst_auto_adjust: 1; Enable(1-Default)/Disable(0) DST automatic adjustment
time_format_24hr: 1; Enable(1 - 24Hr Default)/Disable(0 - 12Hr)

# Do Not Disturb Control
#(0-off (default), 1-on, 2-off with no user control, 3-on with no user control)
dnd_control: 0;

# Caller ID Blocking
#(0-disabled, 1-enabled, 2-disabled no user control, 3-enabled no user control)
callerid_blocking: 0; (Default is 0 - disabled and sending all calls as anonymous)

# Anonymous Call Blocking
#(0-disabled, 1-enabled, 2-disabled no user control, 3-enabled no user control)
anonymous_call_block: 0; (Default is 0 - disabled and blocking of anonymous calls)

# DTMF AVT Payload (Dynamic payload range for AVT tones - 96-127)
dtmf_avt_payload: 101; Default 101

# Sync value of the phone used for remote reset
sync: 1; Default 1

##### New Parameters added in Release 2.1 #####

# Backup Proxy Support
proxy_backup: ""; Dotted IP of Backup Proxy
proxy_backup_port: 5060; Backup Proxy port (default is 5060)

# Emergency Proxy Support
proxy_emergency: ""; Dotted IP of Emergency Proxy
proxy_emergency_port: 5060; Emergency Proxy port (default is 5060)

# Configurable VAD option
enable_vad: 0; VAD setting 0-disable (Default), 1-enable

##### New Parameters added in Release 2.2 #####

# NAT/Firewall Traversal

```

```

nat_enable: 0; 0-Disabled (default), 1-Enabled
nat_address: ""; WAN IP address of NAT box (dotted IP or DNS A record only)
voip_control_port: 5060; UDP port used for SIP messages (default - 5060)
start_media_port: 16384; Start RTP range for media (default - 16384)
end_media_port: 32766; End RTP range for media (default - 32766)
nat_received_processing: 0; 0-Disabled (default), 1-Enabled

# Outbound Proxy Support
outbound_proxy: ""; restricted to dotted IP or DNS A record only
outbound_proxy_port: 5060; default is 5060

##### New Parameter added in Release 3.0 #####

# Allow for the bridge on a 3way call to join remaining parties upon hangup
cnf_join_enable: 1; 0-Disabled, 1-Enabled (default)

##### New Parameters added in Release 3.1 #####

# Allow Transfer to be completed while target phone is still ringing
semi_attended_transfer: 1; 0-Disabled, 1-Enabled (default)

# Telnet Level (enable or disable the ability to Telnet into the phone)
telnet_level: 1; 0-Disabled (default), 1-Enabled, 2-Privileged

##### New Parameters added in Release 4.0 #####

# XML URLs
services_url: ""; URL for external Phone Services
directory_url: ""; URL for external Directory location
logo_url: ""; URL for branding logo to be used on phone display

# HTTP Proxy Support
http_proxy_addr: ""; Address of HTTP Proxy server
http_proxy_port: 80; Port of HTTP Proxy Server (80-default)

# Dynamic DNS/TFTP Support
dyn_dns_addr_1: ""; restricted to dotted IP
dyn_dns_addr_2: ""; restricted to dotted IP
dyn_tftp_addr: ""; restricted to dotted IP

# Remote Party ID
remote_party_id: 0; 0-Disabled (default), 1-Enabled

```

## How to Customize a Phone-Specific Configuration File

You can define parameters that are specific to a particular phone, such as the lines configured on a phone and the defined users for those lines, in a phone-specific configuration file.



### Note

- If you configure a line to use an e-mail address, that line can be called only by using the e-mail address. Similarly, if you configure a line to use a number, that line can be called only by using the number. Each line can have a different proxy configured.
- Define the dial\_template parameter in the default configuration file for maintenance and control purposes. Define the parameter in a phone-specific configuration file only if that phone needs to use a different dial plan than the one being used by the other phones in the same system.

- For a complete alphabetical list of configurable parameters, see Appendix D, “Configurable Parameters for the SIP IP Phone.”

---

## Procedure

- 
- Step 1** Obtain the phone-specific configuration file as follows:
- a. Go to the Cisco.com SIP IP 7940/7960 phone software-download site at <http://www.cisco.com/cgi-bin/tablebuild.pl/sip-ip-phone7960>.
  - b. Download the SIP<mac-addr>.cnf file to the root directory of your TFTP server or to a subdirectory in which all phone-specific configuration files are stored.
- Step 2** Do the following for each phone that you plan to install:
- a. Using an ASCII text editor such as vi, create and open a SIP<mac-addr>.cnf file for the phone.
  - b. Modify the following end-user call-preference parameters as needed to permit or deny end-user use or customization:
    - anonymous\_call\_block
    - autocomplete
    - callerid\_blocking
    - call\_hold\_ringback
    - call\_waiting
    - dnd\_control
  - c. Modify additional parameters as needed.
  - d. Save the file to the root directory of your TFTP server or to a subdirectory that contains all the phone-specific configuration files.  
  
Name the file SIP<mac-addr>.cnf. Type the MAC address in uppercase and the extension, cnf, in lowercase (for example, SIP00503EFFF842.cnf).
- 

## Configuration Example

The following is an example of the phone-specific configuration file that you downloaded from Cisco.com.

```
# SIP Configuration Generic File

# Line 1 appearance
line1_name: 1234567

# Line 1 Registration Authentication
line1_authname: "UNPROVISIONED"

# Line 1 Registration Password
line1_password: "UNPROVISIONED"

# Line 2 appearance
line2_name: football

# Line 2 Registration Authentication
line2_authname: "UNPROVISIONED"
```

```

# Line 2 Registration Password
line2_password: "UNPROVISIONED"

##### New Parameters added in Release 2.0 #####

# Phone Label (Text desired to be displayed in upper right corner)
phone_label: ""; Has no effect on SIP messaging

# Line 1 Display Name (Display name to use for SIP messaging)
line1_displayname: "User ID"

# Line 2 Display Name (Display name to use for SIP messaging)
line2_displayname: ""

##### New Parameters added in Release 3.0 #####

# Phone Prompt (The prompt that will be displayed on console and Telnet)
phone_prompt: "SIP Phone"; Limited to 15 characters (Default - SIP Phone)

# Phone Password (Password to be used for console or Telnet login)
phone_password: "cisco"; Limited to 31 characters (Default - cisco)

# User classification used when Registering [ none (default), phone, ip ]
user_info: none

```

## How to Customize the Configuration from the Phone Menu

After the phone has been connected to power and initialized and the configuration files have been downloaded, you can modify your configuration using the phone menu.

This section contains the following procedures:

- Unlocking and Locking the Phone, page 3-11
- Setting and Restoring Network Parameters, page 3-11
- Setting and Restoring Phone-Specific Parameters, page 3-14
- Setting End-User Call Preferences, page 3-16



### Tip

- To select a parameter, press the down arrow to scroll to and highlight the parameter, or press the number that represents the parameter (located to the left of the parameter on the LCD).
- During configuration, use \* for dots (periods) or press the “.” soft key when available on the LCD.
- During configuration:
  - To enter a number, press the **Number** soft key. To enter a name, press the **Alpha** soft key.
  - To enter a new value, use the buttons on the dial pad.

If entering letters, use the numbers on the dial pad that are associated with a particular letter. For example, the 2 key has the letters A, B, and C. For a lowercase *a*, press the 2 key once. To scroll through the available letters and numbers, press the key repeatedly.

  - To delete any mistakes, press the << soft key.
  - To cancel all changes and exit a menu during configuration, press **Cancel**.

- After editing a parameter, press the **Validate** soft key to save the value that you have entered and exit the Edit panel.

---

Modifying your configuration using the phone menus requires that you unlock and relock the phone. A padlock icon in the upper-right corner of your LCD displays on the phone when the phone is locked. By default, the phone is locked.

**Note**

If the Network Configuration or SIP Configuration menu is displayed, the lock icon in the upper-right corner of your LCD changes to an unlocked state. If you are located elsewhere in the Cisco IP 7960G/7940G phone menus, the next time you access the Network Configuration or SIP Configuration menu, the unlocked icon displays, and you can modify the network and SIP configuration settings.

## Unlocking and Locking the Phone

You must unlock and relock the phone to modify a configuration using the phone menus. Similarly, phone users must unlock and relock the phone to modify end-user parameters.

A padlock icon in the upper-right corner of your LCD displays on the phone when the phone is locked. By default, the phone is locked.

**Prerequisites**

- Set the phone password with the phone\_password parameter in the phone-specific configuration file.

**Procedure**

- 
- Step 1** To unlock the phone, do the following:
- a. Press **Settings > Unlock Config**. The password prompt displays.
  - b. Enter a phone password. The phone unlocks, and the unlock icon displays on the LCD.

**Note**

The Unlock Config menu choice changes to Lock Config and the configuration remains unlocked while you work within it. When you exit the configuration menu, the configuration automatically relocks.

- 
- Step 2** To relock the phone, select **Lock Config** or **Exit**.
- 

## Setting and Restoring Network Parameters

You can modify network parameters using the phone menus.

**Note**

- TFTPServer may be a required parameter, depending on how you intend for the phone to locate the TFTP server from which it downloads its configuration file. You can provide the TFTP server IP address in either of two ways:

- Provide it to the DHCP server. (For information, see the “Prerequisites” section on page 3-1.) Using normal Cisco Discovery Protocol (CDP) processes, the phone locates the DHCP server upon connection to the network; the server in turn provides the TFTP server address.
- Provide it directly to the phone by means of the TFTPServer parameter as described in this procedure. If you use this method, first select **DHCP Enabled > No**.
- Network parameters that can be modified are listed in Table 3-1. Those that cannot be modified are listed in Table 3-2.
- For a complete alphabetical list of configurable parameters, see Appendix D, “Configurable Parameters for the SIP IP Phone.”

---

### Procedure

- 
- Step 1** Unlock the phone (see the “Unlocking and Locking the Phone” section on page 3-11).
- Step 2** Select **Settings > Network Configuration**. The Network Configuration menu displays.
- Step 3** To set a parameter, select it and set it as desired.
- Step 4** To restore all parameters to their defaults, select **Erase Config > Yes**.




---

**Note** If DHCP is disabled on a phone, restoring default phone settings reenables DHCP.

---

- Step 5** Select **Save**. The phone programs the new information into flash memory and resets.
- Step 6** Relock the phone.
- 

**Table 3-1 Network Parameters That Can Be Modified**

Parameter	Description
Admin. VLAN Id <sup>1</sup>	Unique identifier of the VLAN to which the phone is attached (for use in switched networks that are not Cisco networks).
Alternate TFTP	Whether to use an alternate remote TFTP server rather than the local one.  Valid values are Yes and No. If you set this parameter to Yes, you must change the IP address in the TFTP server parameter to the address of the alternate TFTP server. Default is No.
Default Router 1 to 5 <sup>2</sup>	IP address (1) of the default gateway used by the phone and (2 to 5) of the gateways that the phone attempts to use as an alternate gateway if the default gateway is unavailable.

**Table 3-1 Network Parameters That Can Be Modified (continued)**

Parameter	Description
DHCP Address Released	<p>Whether the IP address of the phone can be released for reuse in the network.</p> <p>Valid values are Yes and No. When set to Yes, the phone sends a DHCP release message to the DHCP server and goes into a release state. The release state provides enough time to remove the phone from the network before the phone attempts to acquire another IP address from the DHCP server. When you move the phone to a new network segment, first release the DHCP address.</p>
DHCP Enabled	<p>Whether the phone uses DHCP to configure network settings (IP address, subnet mask, domain name, default router list, DNS server list, and TFTP address).</p> <p>Valid values are Yes and No. To manually configure your IP settings, you must set this parameter to No. Default is Yes.</p>
DNS Servers 1 to 5 <sup>2</sup>	IP address of the DNS server used by the phone to resolve names to IP addresses. The phone attempts to use DNS servers 2 to 5 if DNS server 1 is unavailable.
Domain Name	Name of the DNS domain in which the phone resides.
Erase Configuration	<p>Whether to erase all of the locally defined network settings on the phone and reset the values to the defaults.</p> <p>Valid values are Yes and No. Yes reenables DHCP. For information on erasing the local configuration, see the “Setting and Restoring Network Parameters” section on page 3-11.</p>
HTTP Proxy Address	IP address of the HTTP proxy server. You can use either a dotted IP address or a DNS name (a record only).
HTTP Proxy Port	Port number of the outbound proxy port. Default is 80.
IP Address <sup>2</sup>	IP address of the phone that is assigned by DHCP or that is locally configured.
Network Media Type	<p>Ethernet port negotiation mode. Valid values are as follows:</p> <ul style="list-style-type: none"> <li>• Auto—Port is autonegotiated.</li> <li>• Full-100—Port is configured to be a full-duplex, 100-MB connection.</li> <li>• Half-100—Port is configured to be a half-duplex, 100-MB connection.</li> <li>• Full-10—Port is configured to be a full-duplex, 10-MB connection.</li> <li>• Half-10—Port is configured to be a half-duplex, 10-MB connection.</li> </ul> <p>Default is Auto.</p>

**Table 3-1 Network Parameters That Can Be Modified (continued)**

Parameter	Description
Network Port 2 Device Type	Device type that is connected to port 2 of the phone. Valid values are Hub/Switch and PC. Default is Hub/Switch.  <b>Note</b> If the value is PC, port 2 can be connected only to a PC. If you are not sure about the connection, use the default value. Using a value of PC and connecting port 2 to a switch could result in spanning-tree loops and network confusion.
Subnet Mask <sup>2</sup>	IP subnet mask used by the phone. A subnet mask partitions the IP address into a network and a host identifier.
TFTP Server <sup>2</sup>	IP address of the TFTP server.

1. If you have an administrative VLAN setting assigned on the Cisco Catalyst switch, that setting overrides any changes made on the phone.
2. DHCP must be disabled.

**Table 3-2 Network Parameters That Cannot Be Modified**

Parameter	Description
DHCP Server	IP address of the DHCP server from which the phone received its IP address and additional network settings.
Dynamic DNS Server 1 and 2	IP address of a dynamic DNS server.
Dynamic TFTP Server	IP address of a dynamic TFTP server.
Host Name	Unique host name assigned to the phone.
MAC Address	Factory-assigned unique 48-bit hexadecimal MAC address of the phone.
Operational VLAN Id	Unique identifier of the VLAN of which the phone is a member.

## Setting and Restoring Phone-Specific Parameters

Phone users can modify the phone-specific configuration settings using the phone menus.



### Note

- Parameters defined in the default configuration file override those specified in the phone-specific configuration file.
- If a phone-specific configuration file exists, the phone uses parameters entered locally until the next reboot.
- If you do not configure the phone using a TFTP server, you must configure the phone locally.
- To configure the preferred codec and out-of-band DTMF parameters, press **Change** until the option displays and then press **Save**.
- If your system has been set up to have the phones retrieve the configuration file from a TFTP server, you must use the server's configuration file to change the parameter value to a null value "" or to "UNPROVISIONED." The phone uses the setting for that variable that it has stored in flash memory.



- If the `telnet_level` parameter is set to allow privileged commands to be executed, the entire SIP configuration can be erased. Use the **erase\_protflash** command so that the phone can retrieve its configuration files.

---

### Prerequisites

- Define the line parameters (those identified as `linex`) on the phone. If you configure a line to use an e-mail address, that line can be called only by using an e-mail address. Similarly, if you configure a line to use a number, that line can be called only by using the number.

### Procedure

---

- Step 1** Unlock the phone (see the “Unlocking and Locking the Phone” section on page 3-11).
- Step 2** Select **Settings > SIP Configuration**. The SIP Configuration menu displays.
- Step 3** To set a required parameter, select it and set it as desired. The following are required parameters that you must set now if you did not set them in the default configuration file as described in the “How to Customize the Default Configuration File” section on page 3-5:
- `line1_name`—Number or e-mail address for use when registering. Enter a number without dashes. For example, enter 555-0100 as 5550100. Enter an e-mail ID without the host name.
  - `proxy1_address`—IP address of the SIP proxy server that is used by the phones. Enter the address in IP dotted-decimal notation or use the FQDN. The “x” argument is representative of server addresses. If the parameter is provisioned with an FQDN, the phone sends REGISTER and INVITE messages by using the FQDN in the Req-URI, To, and From fields.
  - `proxy1_port`—Port number of the SIP proxy server that is used by line 1.
  - If the proxy server with which the phone communicates has authentication enabled, set the following parameters as well:
    - `line1_authname`—Name used by the phone for authentication if a registration is challenged by the proxy server during initialization. Default is UNPROVISIONED.
    - `line1_password`—Password used by the phone for authentication if a registration is challenged by the proxy server during initialization. Default is UNPROVISIONED.
- Step 4** To set an additional parameter, select it and set it as desired. Phone-specific parameters are listed in Table 3-3.
- Step 5** To restore a parameter to its default, do the following:
- a. One at a time, highlight the parameter whose setting you want to erase, and then select **Edit** followed by **<<**.
  - b. Select **Validate > Exit**.
  - c. If necessary, select **Back** to exit the menu.
- Step 6** Select **Save**. The phone programs the new information into flash memory and resets.
- Step 7** Relock the phone.
-

**Table 3-3 Phone-Specific Parameters**

Parameter	Description
Authentication Name <sup>1</sup>	Name used by the phone for authentication if a registration is challenged by the proxy server during initialization.
Authentication Password <sup>1</sup>	Password used by the phone for authentication if a registration is challenged by the proxy server during initialization. If a value is not configured for the Authentication Password parameter when registration is enabled, the default logical password is used. The default logical password is <code>SIPmac-address</code> , where <code>mac-address</code> is the MAC address of the phone.
Display Name	Identification as it should appear for caller identification. For example, instead of <code>jdoe@company.com</code> appearing on phones that have caller ID, you can specify John Doe in this parameter to have John Doe appear on the callee end instead. If a value is not specified for this parameter, the Name value is used.
Name	Description phone number or e-mail address used when registering. When entering a number, enter the number without any dashes. For example, enter 555-0100 as 5550100. When entering an e-mail address, enter the e-mail ID without the host name.
Proxy Address	IP address of the primary SIP proxy server that will be used by the phone. Enter this address in IP dotted-decimal notation.
Proxy Port	Port number of the primary SIP proxy server. This is the port that the SIP client will use. The default is 5060.
Short Name	Name or number associated with the <code>linex_name</code> as you want it to display on the phone LCD if the <code>linex_name</code> value exceeds the display area. For example, if the <code>linex_name</code> value is the phone number 111-222-333-4444, you can specify 34444 for this parameter to have 34444 display on the LCD instead. Alternatively, if the value for the <code>linex_name</code> parameter is the e-mail address “ <code>username@company.com</code> ,” you can specify the “ <code>username</code> ” to have just the username appear on the LCD instead. This parameter is used for display only. If a value is not specified for this parameter, the value in the Name variable is displayed.

1. Required when registration is enabled and the registrar challenges registration.

## Setting End-User Call Preferences

End users can modify call preferences from their own phones, according to how you set the associated parameters.

### Prerequisites

- Set configuration variables for call preferences as follows (see the “How to Customize a Phone-Specific Configuration File” section on page 3-8):
  - To enable end users to modify a preference, set to 0 or 1.
  - To prohibit end users from modifying a preference, set to 2 or 3.

### Procedure

- 
- Step 1** Unlock the phone (see the “Unlocking and Locking the Phone” section on page 3-11).
- Step 2** On the IP phone, select **Settings > Call Preferences**.
- Step 3** Set any of the following preferences to the desired setting:
- Anonymous Call Block
  - Auto-Complete Numbers
  - Caller ID Blocking
  - Call Hold Ringback
  - Call Waiting
  - Do Not Disturb
- Step 4** Select **Save**. The phone programs the new information into flash memory and resets.
- Step 5** Relock the phone.
- 

## How to Set the Date and Time

You can set date, time, and daylight savings time (DST) parameters. The current date and time is supported on the Cisco IP 7960G/7940G phone using Simple Network Time Protocol (SNTP) and is displayed on the LCD. DST and time-zone settings are also supported.

International time-zone abbreviations are supported and must be in all capital letters.



### Note

We recommend that you set date- and time-related parameters in the default file for all phones. Alternatively, you can set the time-zone parameter manually on the phone or in the phone-specific configuration files.

---

### Prerequisites

- Determine the type of DST that you want to configure:
  - Absolute DST (for example, starts on April 1 and ends on October 1)
  - Relative DST (for example, starts on the first Sunday in April and ends on the last Sunday of October)

Review the list of common and absolute DST parameters from Appendix D, “Configurable Parameters for the SIP IP Phone.”

- Review the information on SNTP in Table 3-4 on page 3-19. SNTP parameters specify how the phone obtains the current time from an SNTP server.
- Determine your time zone from Table 3-5 on page 3-20.

**Procedure**

---

**Step 1** Using an ASCII text editor such as vi, open the SIPDefault.cnf file.

**Step 2** Modify the following SNTP parameters as needed:

- sntp\_mode
- sntp\_server
- time\_zone

**Step 3** Modify the following common DST parameters as needed:

- dst\_offset
- dst\_auto\_adjust
- dst\_start\_month
- dst\_stop\_month
- dst\_start\_time
- dst\_stop\_time

**Step 4** Do one of the following:

- Modify the following absolute DST parameters as needed:
  - dst\_start\_day
  - dst\_stop\_day
- Modify the following relative DST parameters as needed:
  - dst\_start\_day\_of\_week
  - dst\_start\_week\_of\_month
  - dst\_stop\_day\_of\_week
  - dst\_stop\_week\_of\_month

**Step 5** Save the file to the root directory of your TFTP server.

**Note**

---

To adjust the phone display to European Day-Month-Year format, add the following entry to the SIPDefault.cnf file: date\_format:D/M/Y.

---

Table 3-4 describes the effects on SNTP mode when the SNTP server is null (not assigned an IP address) or when it is assigned a valid IP address.

**Table 3-4 Effects on SNTP Mode**

SNTP Server	SNTP Mode			
	Unicast	Multicast	Anycast <sup>1</sup>	Directed Broadcast
<b>SNTP server parameter is null</b>				
Sends	No known server with which to communicate.	SNTP requests are not sent.	SNTP packet to the local network broadcast address.  After the first SNTP response is received, the phone switches to unicast mode with the server being set as the one who first responded.	SNTP packet to the local network broadcast address.  After the first SNTP response is received, the phone switches to multicast mode.
Receives	No known server with which to communicate.	Multicast data using the SNTP/NTP multicast address from the local network broadcast address from any server on the network.	Unicast SNTP data from the SNTP server that first responded to the network broadcast request.	SNTP data from the SNTP/NTP multicast address and the local network broadcast address from any server on the network.
<b>SNTP server parameter is a valid IP address.</b>				
Sends	SNTP request to the SNTP server.	SNTP requests are not sent.	If the mode is anycast and the SNTP server parameter is a valid IP address, the phone sends the request to the broadcast address in version 7.4.	SNTP packet to the SNTP server.  After the first SNTP response is received, the phone switches to multicast mode.
Receives	SNTP response from the SNTP server and ignores responses from other SNTP servers.	SNTP data via the SNTP/NTP multicast address from the local network broadcast address.	SNTP response from the SNTP server and ignores responses from other SNTP servers.	SNTP data from the SNTP/NTP multicast address and the local network broadcast address and ignores responses from other SNTP servers.

1. If `sntp_mode` is set to anycast, the `sntp_server` address will be ignored and subsequent `sntp` requests will be sent to the first `sntp` server that responded (the first `sntp` request must be unconditionally sent to the broadcast address).

Table 3-5 includes the time-zone information that you need to configure the SNTP mode and server parameters.

**Table 3-5 Time-Zone Information**

Abbreviation	GMT Offset	Cities	Time-Zone Names
IDL	GMT-12:00	Eniwetok	IDL (International Date Line), IDLW (International Date Line West)
NT	GMT-11:00	Midway	BT (Bering Time), NT (Nome Time)
AHST	GMT-10:00	Hawaii	AHST (Alaska-Hawaii Standard Time), HST (Hawaiian Standard Time), CAT (Central Alaska Time)
IMT	GMT-09:30	Isle Marquises	Isle Marquises
YST	GMT-09:00	Yukon	YST (Yukon Standard Time)
PST	GMT-08:00	Los Angeles	PST (Pacific Standard Time)
MST	GMT-07:00	Phoenix	MST (Mountain Standard Time), PDT (Pacific Daylight Time)
CST	GMT-06:00	Dallas, Mexico City	CST (Central Standard Time), MDT (Mountain Daylight Time), Chicago
EST	GMT-05:00	New York	EST (Eastern Standard Time), CDT (Central Daylight Time), NYC
AST	GMT-04:00	La Paz	AST (Atlantic Standard Time), EDT (Eastern Daylight Time)
NST	GMT-03:30	Newfoundland	NST (Newfoundland Standard Time)
BST	GMT-03:00	Buenos Aires	BST (Brazil Standard Time), ADT (Atlantic Daylight Time), GST (Greenland Standard Time)
AT	GMT-02:00	Mid-Atlantic	AT (Azores Time)
WAT	GMT-01:00	Azores	WAT (West Africa Time)
GMT	GMT 00:00	London	GMT (Greenwich Mean Time), WET (Western European Time), UT (Universal Time)
CET	GMT+01:00	Paris	CET (Central European Time), MET (Middle European Time), BST (British Summer Time), MEWT (Middle European Winter Time), SWT (Swedish Winter Time), FWT (French Winter Time)
EET	GMT+02:00	Athens, Rome	EET (Eastern European Time), USSR-zone1, MEST (Middle European Summer Time), FST (French Summer Time)
BT	GMT+03:00	Baghdad, Moscow	BT (Baghdad Time), USSR-zone2
IT	GMT+03:30	Tehran	IT (Iran Time)

**Table 3-5 Time-Zone Information (continued)**

Abbreviation	GMT Offset	Cities	Time-Zone Names
ZP4	GMT+04:00	Abu Dhabi	USSR-zone3, ZP4 (GMT Plus 4 Hours)
AFG	GMT+04:30	Kabul	Afghanistan
ZP5	GMT+05:00	Islamabad	USSR-zone4, ZP5 (GMT Plus 5 Hours)
IST	GMT+05:30	Bombay, Delhi	IST (Indian Standard Time)
ZP6	GMT+06:00	Colombo	USSR-zone5, ZP6 (GMT Plus 6 Hours)
SUM	GMT+06:30	North Sumatra	NST (North Sumatra Time)
WAST	GMT+07:00	Bangkok, Hanoi	SST (South Sumatra Time), USSR-zone6, WAST (West Australian Standard Time)
HST	GMT+08:00	Beijing, Hong Kong	CCT (China Coast Time), HST (Hong Kong Standard Time), USSR-zone7, WADT (West Australian Daylight Time)
JST	GMT+09:00	Tokyo, Seoul	JST (Japan Standard Time/Tokyo), KST (Korean Standard Time), SSR-zone8
CAST	GMT+09:30	Darwin	SAST (South Australian Standard Time), CAST (Central Australian Standard Time)
EAST	GMT+10:00	Brisbane, Guam	GST (Guam Standard Time), USSR-zone9, EAST (East Australian Standard Time)
EADT	GMT+11:00	Solomon Islands	USSR-zone10, EADT (East Australian Daylight Time)
NZST	GMT+12:00	Auckland	NZT (New Zealand Time/Auckland), NZST (New Zealand Standard Time), IDLE (International Date Line East)

## Time-Zone Configuration Examples

### Absolute DST Configuration

The following is an example of an absolute DST configuration:

```
time_zone : PST
dst_offset : 01/00
dst_start_month : April
dst_start_day : 1
dst_start_time : 02/00
dst_stop_month : October
dst_stop_day : 1
dst_stop_time : 02/00
dst_stop_autoadjust : 1
```

**Relative DST Configuration**

The following is an example of a relative DST configuration:

```
time_zone : PST
dst_offset : 01/00
dst_start_month : April
dst_start_day : 0
dst_start_day_of_week : Sunday
dst_start_week_of_month : 1
dst_start_time : 02/00
dst_stop_month : October
dst_stop_day : 0
dst_stop_day_of_week : Sunday
dst_stop_week_of_month : 8
dst_stop_time : 02/00
dst_stop_autoadjust : 1
```

## How to Create Dial Plans

Dial plans enable the Cisco SIP IP phone to support automatic dialing and generation of a secondary dial tone. If a single dial plan is used for a system of phones, the dial plan is best specified in the default configuration file.

However, you can also create multiple dial plans and specify which phones are to use which dial plan by defining the `dial_template` parameter in the phone-specific configuration file. If one phone in a system of phones needs to use a different dial plan than the rest, you need to define a dial plan for that phone in its phone-specific configuration file.

**Prerequisites**

- Ensure that your dial plans adhere to the following:
  - They are written with the understanding that rules are matched from start to finish with the longest matching rule taken as the one to use. Matches against a period are not counted as part of the longest length.
  - They are in XML format.
  - They are stored on your TFTP server.
- Specify which dial plan a phone is to use by specifying the path to the dial plan in the `dial_template` parameter. Define the `dial_template` parameter in either the default configuration file or a phone-specific configuration file.

**Note**


---

To simplify maintenance and control, define this parameter in the default configuration file. Define it in a phone-specific configuration file only if that phone needs to use a different dial plan than the one being used by the other phones in the same system.

---

**Procedure**

- 
- Step 1** Using an ASCII text editor such as `vi`, open a new file.
- Step 2** Type the following to indicate the start of the dial-plan template:

```
<DIALTEMPLATE>
```



**Step 3** For each of the numbering schemes that you require, add the following string to the template, each starting on a separate line:

```
TEMPLATE MATCH="pattern" Timeout="sec" User="type" Rewrite="xxx" Route="route" Tone="tone"
```

Arguments are as follows:

Argument	Description																																							
TEMPLATE MATCH="pattern"	Dial-pattern string to match. You can use the following wildcards: <ul style="list-style-type: none"> <li>• A period (.) matches any character.</li> <li>• An asterisk (*) matches one or more characters.</li> <li>• A comma (,) causes the phone to generate a secondary dial tone.</li> </ul>																																							
Timeout="sec"	Time, in seconds, before the system times out and dials the number as entered by the user. To have the number dial immediately, specify 0.																																							
User="type"	Tag to be added automatically to the dialed number. Valid values are IP and Phone. This tag is not case sensitive.																																							
Rewrite="xxx"	<p>Alternate string to be dialed instead of the number that the user dials. The following apply:</p> <ul style="list-style-type: none"> <li>• Rewrite rules are matched from start to finish with the longest matching rule taken as the one to use.</li> <li>• Matches against a period are not counted as part of the length.</li> <li>• A complete rule is matched only if it has more nonwildcard matches than an incomplete rule.</li> <li>• Comments are allowed with the following syntax: <pre>&lt;!-- comment --&gt;</pre> </li> <li>• Rules allow for substitution of up to five replacement strings as well as picking off of replaced digits one at a time.</li> </ul> <p>For example, a match string of <code>ab..cd..ef*</code> and an input string of <code>ab12cd34ef5678</code> result in the following.</p> <table border="1"> <thead> <tr> <th>Rewrite</th> <th>Output</th> <th>Notes</th> </tr> </thead> <tbody> <tr> <td><code>%s</code></td> <td><code>ab12cd34ef5678</code></td> <td>—</td> </tr> <tr> <td><code>%0</code></td> <td><code>ab12cd34ef5678</code></td> <td>—</td> </tr> <tr> <td><code>%1</code></td> <td><code>12</code></td> <td>—</td> </tr> <tr> <td><code>%2</code></td> <td><code>34</code></td> <td>—</td> </tr> <tr> <td><code>%3</code></td> <td><code>5678</code></td> <td>—</td> </tr> <tr> <td><code>%4</code></td> <td></td> <td>Null output.</td> </tr> <tr> <td><code>%5</code></td> <td></td> <td>Null output.</td> </tr> <tr> <td><code>XYZ....</code></td> <td><code>XYZ1234</code></td> <td>—</td> </tr> <tr> <td><code>X.Y.Z...</code></td> <td><code>X1Y2Z345</code></td> <td>—</td> </tr> <tr> <td><code>919%1%2%3</code></td> <td><code>91912345678</code></td> <td>—</td> </tr> <tr> <td><code>AB...X%1X..</code></td> <td><code>AB123X12X45</code></td> <td>12 appears twice.</td> </tr> <tr> <td><code>X%1X%1X%1</code></td> <td><code>X12X12X12</code></td> <td>You can reuse the string.</td> </tr> </tbody> </table>	Rewrite	Output	Notes	<code>%s</code>	<code>ab12cd34ef5678</code>	—	<code>%0</code>	<code>ab12cd34ef5678</code>	—	<code>%1</code>	<code>12</code>	—	<code>%2</code>	<code>34</code>	—	<code>%3</code>	<code>5678</code>	—	<code>%4</code>		Null output.	<code>%5</code>		Null output.	<code>XYZ....</code>	<code>XYZ1234</code>	—	<code>X.Y.Z...</code>	<code>X1Y2Z345</code>	—	<code>919%1%2%3</code>	<code>91912345678</code>	—	<code>AB...X%1X..</code>	<code>AB123X12X45</code>	12 appears twice.	<code>X%1X%1X%1</code>	<code>X12X12X12</code>	You can reuse the string.
Rewrite	Output	Notes																																						
<code>%s</code>	<code>ab12cd34ef5678</code>	—																																						
<code>%0</code>	<code>ab12cd34ef5678</code>	—																																						
<code>%1</code>	<code>12</code>	—																																						
<code>%2</code>	<code>34</code>	—																																						
<code>%3</code>	<code>5678</code>	—																																						
<code>%4</code>		Null output.																																						
<code>%5</code>		Null output.																																						
<code>XYZ....</code>	<code>XYZ1234</code>	—																																						
<code>X.Y.Z...</code>	<code>X1Y2Z345</code>	—																																						
<code>919%1%2%3</code>	<code>91912345678</code>	—																																						
<code>AB...X%1X..</code>	<code>AB123X12X45</code>	12 appears twice.																																						
<code>X%1X%1X%1</code>	<code>X12X12X12</code>	You can reuse the string.																																						

Argument	Description	
	Xs% 919 .....	Xab12cd34ef5678 919 12345678
Route="route"	Proxy to which to route the call. Valid values are default, emergency, and FQDN. FQDN is treated the same as default proxy. This entry is not case sensitive.	
Tone="tone"	User-specific dial tone. If no tone is specified, the default secondary dial tone plays. If a comma (,) is specified followed by a tone, the phone plays the indicated tone instead of the secondary dial tone. If a tone is specified but there is no comma, the tone is ignored.  You can specify up to three different secondary dial tones in a single dial-plan template. The tones play in the order in which they appear in the template. Multiple tone entries are condensed into a single comma. For example, the phone interprets the match string 9,,234 as 9,234 and treats the three commas as a single comma.	

**Step 4** Specify the pound sign (#) and asterisk (\*) as dialed digits if required.

- The # is processed as a “dial now” event by default. You can override this by specifying # in the dial-plan template, in which case the phone does not dial immediately when the # is pressed but does continue to match the dial-plan template that specifies the #. The # is not matched by the wildcard character \* or the period (.).
- The \* is processed as a wildcard character. You can override this by preceding the \* with the backward slash (\) escape sequence, resulting in the sequence \\*. The phone automatically strips the \ so that it does not appear in the outgoing dial string. When \* is received as a dialed digit, it is matched by the wildcard characters \* and period (.).

**Step 5** Specify the comma (,) as a secondary dial tone if required.

In earlier releases, a comma in the dial-plan template caused the phone to play the default secondary dial tone (Bellcore-Outside). With this release, you can specify which tones are played. All tone names should begin with a common prefix. Tone names, which are case insensitive, are as follows:

• Bellcore-Alerting	• Bellcore-dr5	• Bellcore-Reorder
• Bellcore-Busy	• Bellcore-dr6	• Bellcore-Stutter
• Bellcore-BusyVerify	• Bellcore-Hold	• CallWaiting-2
• Bellcore-CallWaiting	• Bellcore-Inside	• CallWaiting-3
• Bellcore-Confirmation	• Bellcore-None	• CallWaiting-4
• Bellcore-dr1	• Bellcore-Outside (default)	• Cisco-BeepBonk
• Bellcore-dr2	• Bellcore-Permanent	• Cisco-Zip
• Bellcore-dr3	• Bellcore-Reminder	• Cisco-ZipZip
• Bellcore-dr4		

If desired, specify `<!--comment-->` at the end of each string to denote the type of plan (for example, `<!-- Long Distance -->` or `<!-- Corporate Dial Plan -->`).

**Note**

For more information on Bellcore tones, refer to *Bellcore GR-506-CORE*. For more information on tones in BTS 10200 Softswitch features, refer to the Cisco BTS 10200 Softswitch website at <http://www.cisco.com/en/US/partner/products/hw/vcallcon/ps531/index.html>.

**Step 6** Type the following to indicate the end of the dial-plan template:

```
</DIALTEMPLATE/>
```

**Step 7** Give the file a unique name specific to the dial plan that it defines and save it with a .xml extension to your TFTP server.

**Step 8** If the dial plan applies to a specific phone, add the path to the dial plan (without specifying the file type of .xml) via the dial\_template parameter in the phone-specific configuration file. If the dial plan applies to a system of phones, add the path to the dial plan via the dial\_template parameter in the default configuration file.

## Dial-Plan Configuration Examples

### Using the Pound-Sign (#) Character

The following example uses the pound sign (#) as a dialed digit:

```
</DIALTEMPLATE>
  <TEMPLATE MATCH="123#45#6" TIMEOUT="0" User="Phone"/> <!-- Match '#' -->
  <TEMPLATE MATCH="34#..." TIMEOUT="0" User="Phone"/> <!-- Match '#' -->
  <TEMPLATE MATCH="*" TIMEOUT="15" User="Phone"/>
</DIALTEMPLATE/>
```

In the example above, the 123#45#6 string is matched if the user dials 123#45#6. Pressing the pound sign (#) does not cause the phone to dial immediately because # is explicitly specified. However, dialing 1# or 123#4# causes the phone to dial immediately.

### Using the Backward-Slash (\) and Asterisk (\*) Characters

The following example uses the backward slash (\) and asterisk (\*) as a dialed digit:

```
</DIALTEMPLATE>
  <TEMPLATE MATCH="12\*345" TIMEOUT="0" User="Phone"/> <!-- Match * Char -->
  <TEMPLATE MATCH="*" TIMEOUT="10" User="Phone"/> <!-- Wildcard -->
</DIALTEMPLATE/>
```

If you use the backslash (\) on a character other than the asterisk (\*), the \ is ignored and the \ character is matched. If you need to explicitly specify the \ character in a dial plan, use \\. The \ is not sent out as part of the dialed digit string because the phone removes it before sending the dial string.

**Note**

The \\* character is matched by the “.” character.

**Specifying a Secondary Dial Tone**

The following example specifies two different tones:

```
</DIALTEMPLATE>
  <TEMPLATE MATCH="7,..." TIMEOUT="0" /> <!-- Default Secondary Dial Tone -->
  <TEMPLATE MATCH="9,..." TIMEOUT="0" Tone="Zip" /> <!-- Play Zip Tone -->
  <TEMPLATE MATCH="8,..." TIMEOUT="0" Tone="Hold" /> <!-- Play Hold Tone -->
  <TEMPLATE MATCH="8,123,..." TIMEOUT="0" Tone="Hold" Tone="Zip" /> <!--Play Hold Tone
                                after 8, Play Zip Tone after 123-->
</DIALTEMPLATE>
```

## How to Verify Initialization

The initialization process establishes network connectivity and makes the phone operational in your IP network.

**Procedure**

- 
- Step 1** After the phone has power connected to it, ensure that the phone cycles through the following steps:
- a. The following flash on and off in sequence: Headset button, Mute button, and Speaker button.
  - b. The Cisco Systems, Inc. copyright appears on the LCD.
  - c. The following messages appear:
    - Configuring VLAN—The phone configures the Ethernet connection.
    - Configuring IP—The phone contacts the DHCP server to obtain network parameters and the IP address of the TFTP server.
    - Requesting Configuration—The phone contacts the TFTP server to request its configuration files and compares firmware images.
    - Upgrading Software—The phone displays this message only if it determines that an image upgrade is required. After upgrading the image, the phone automatically reboots to run the new image.
  - d. The main LCD displays the following:
    - Primary directory number
    - Soft keys

If the phone successfully cycles through these steps, it has started up properly.

---

## Where to Go Next

- See Chapter 4, “Managing Cisco SIP IP Phones,” for information on upgrading firmware and performing other management tasks.
- See Chapter 5, “Monitoring Cisco SIP IP Phones,” for information on debugging and on viewing network statistics.



## Managing Cisco SIP IP Phones

---

This chapter provides information on the following:

- How to Customize Cisco IP 7960G/7940G Phone Rings, page 4-1
- How to Access XML Cards, page 4-2
- How to View Your Cisco SIP IP Phone Firmware Image Version, page 4-3
- How to Upgrade Your Cisco SIP IP Phone Firmware Image, page 4-3
- How to Upgrade Your Cisco SIP IP Phone Firmware Image and Reboot Remotely, page 4-6
- Where to Go Next, page 4-7

### How to Customize Cisco IP 7960G/7940G Phone Rings

The Cisco IP 7960G/7940G phone ships with two ring types: Chirp1 and Chirp2. However, you can create and add custom rings.

#### Procedure

---

**Step 1** Create a pulse-code-modulation (PCM) file for each desired ring type and store it in the root directory of your TFTP server.

PCM files must contain no header information and must comply with the following format guidelines:

- 8000-Hz sampling rate
- 8 bits per sample
- mu-law compression

**Step 2** Using an ASCII text editor such as vi, open the RINGLIST.DAT file. For each ring type that you are adding, specify the name as you want it to appear on the Ring Type menu, press **Tab**, and then specify the filename of the ring type. Your RINGLIST.DAT file should appear similar to the following:

```
Ring Type 1    ringer1.pcm
Ring Type 2    ringer2.pcm
```

**Step 3** Save and close the file.

---

# How to Access XML Cards

You can access XML cards in any of the following ways:

- Select **Services** (configured by means of the `services_url` parameter).
- Select **External Directory** (configured by means of the `directory_url` parameter).
- Download a bitmap to be used as the phone logo (configured by means of the `logo_url` parameter).

## Restrictions for XML Cards

The phone supports Cisco CallManager XML up through version 3.0. It does not support the following XML objects in version 3.1 and later: `CiscoIPPhoneIconMenu`, `CiscoIPPhoneExecute`, `CiscoIPPhoneError`, and `CiscoIPPhoneResponse`.

The following restrictions apply to phones that use XML cards:

- External directories cannot be appended to the main list of directories under the Directories button. If external directories are provisioned for the phone, access them by pressing the Directories button and selecting the External Directory option.
- The phone removes white space when Cisco CallManager XML cards are displayed. Multiple spaces are consolidated into a single space.
- The phone does not allow setting of *x* and *y* coordinates for the `CiscoIPPhoneImage` object. The image always appears at location 0,0. Centering of the image is not supported if *x* and *y* are set to -1.
- The phone displays any valid title that it receives. In contrast, with Cisco CallManager phones, the `CiscoIPPhoneGraphicMenu` object does not display a title even if it receives one; the `CiscoIPPhoneImage` object displays the previous menu item or service rather than received titles.
- The phone discards carriage returns and line feeds. In contrast, Cisco CallManager phones allow embedded carriage returns and line feeds.
- The phone always displays the full set of directory soft keys. In contrast, Cisco CallManager phones change the soft keys depending on what type of object the phone receives and the version of Cisco CallManager.
- The phone sends a parameter along with the initial request for a Services or Directory URL. Other types of phones do not.

For more information about using XML on your Cisco SIP IP phone, refer to the following:

- IP Telephony  
<http://www.hotdispatch.com/cisco-ip-telephony>
- Cisco CallManager Services Developer Kit  
[http://www.cisco.com/pcgi-bin/dev\\_support/access\\_level/product\\_support](http://www.cisco.com/pcgi-bin/dev_support/access_level/product_support)
- *Developing Cisco IP Phone Services* by Darrick Deel, Mark Nelson, and Anne Smith, ISBN 1-58705-060-9

# How to View Your Cisco SIP IP Phone Firmware Image Version

You can determine your firmware image version.

## Procedure

---

- Step 1** Select **Settings > Status**.
- Step 2** Select **Firmware Versions**. The Firmware Versions menu appears.
- Step 3** View information as needed. The following displays:
- Application Load ID—Current software image on the phone.
  - Boot Load ID—Bootstrap loader image version that is manufactured on the phone. This image name does not change.
  - DSP Load ID—Current DSP version on the phone.
- Step 4** Select **Exit**.
- 

# How to Upgrade Your Cisco SIP IP Phone Firmware Image



## Note

Refer to the *Cisco 7940 and 7960 IP Phones Firmware Upgrade Matrix* for additional upgrade information for all releases.

---

At some point after you install hardware and the initial firmware image, you could require an upgrade. You can upgrade by means of either of two methods:

- Upgrade one phone at a time using a phone-defined configuration file.
- Upgrade a system of phones using the default configuration file.

You can upgrade to follow-on releases from any earlier release. For example:

-- > Release 2.3 --> Release 5.3 --> Release 7.4

Each intermediate releases adds key functionality that is necessary for a subsequent upgrade. Each upgrade provides the following:

- Prevents digital signal processor (DSP) mismatches in the upgrade to a higher release.
- Addresses user-interface responsiveness and voice-clipping issues.
- Adds image authentication to IP phone protocols, so that tampering with the binary image before the image is downloaded to the phone is not allowed. Any tampering with the image causes the phone to fail the authentication process and reject the image.

It is possible during the upgrade procedure to configure your system to enable dual-booting from either SIP or Media Gateway Control Protocol (MGCP). If it is so configured, during initialization the phone automatically determines the network in which it is being installed.

**Procedure**

- 
- Step 1** Determine your current firmware image release. (For instructions, see the “How to View Your Cisco SIP IP Phone Firmware Image Version” section on page 4-3.)
- Step 2** To enable dual-booting, use an ASCII text editor such as vi, open the OS79XX.TXT dual-boot file that resides in the root directory of your TFTP server, and add the following:
- ```
POS30203
```
- Step 3** If your current release is earlier than Release 2.3, upgrade to Release 2.3 as follows:
- a. From the Cisco.com website, do the following:
    1. Go to the Cisco.com SIP IP 7940/7960 phone software-download site at <http://www.cisco.com/cgi-bin/tablebuild.pl/sip-ip-phone7960>.
    2. Download the following to the root directory of your TFTP server: Release 2.3 ZIP archive, default configuration file, and release notes.
  - b. Read the release notes and any readme.txt files. Unzip the archive and extract the binary image files to the root directory of the TFTP server.
  - c. Using an ASCII text editor such as vi, do the following:
    1. Open the default configuration file.
    2. Update the image version specified in the image\_version variable. The version name in the image\_version variable should match the version name (without the .sbn or .bin extension) of the latest firmware image; in this case the version is POS30203.
    3. Save the modified file to the TFTP server.
  - d. Reinitialize each phone and verify that it does the following:
    1. Contacts the TFTP server and requests its configuration files.
    2. Compares the image defined in the file to the image that it has stored in flash memory.
    3. If the images differ, downloads the image defined in the configuration file (which is stored in the root directory on the TFTP server).
    4. Programs that image into flash memory.
    5. Reboots.
- Step 4** If your current release is between Release 2.3 and Release 5.2, upgrade to Release 5.3 as follows:




---

**Note** If your current release is Release 5.2, you need not upgrade to Release 5.3. Upgrade directly to Release 6.0 or 6.1 as described in Step 5 below.

---

- a. From the Cisco.com website listed in Step 3a above, download the following to the root directory of your TFTP server: Release 5.3 ZIP archive, default configuration file, and release notes.
- b. Read the release notes and any readme.txt files. Unzip the archive and extract the binary image files to the root directory of the TFTP server.
- c. Using an ASCII text editor such as vi, do the following:
  1. Open the configuration file.
  2. Update the image version specified in the image\_version variable. The version name in the image\_version variable should match the version name (without the .sbn or .bin extension) of the latest firmware image.



3. Save the modified file to the TFTP server.
- d. Reinitialize each phone as described in Step 3d above.

**Step 5** If your current release is Release 5.2 or higher, upgrade to Release 6.0 or 6.1 as follows:

- a. From the Cisco.com website listed in Step 3a above, download the following to the root directory of your TFTP server: Release 6.0 (or 6.1) ZIP archive, default configuration file, and release notes.
- b. Read the release notes and any readme.txt files. Unzip the archive and extract the binary image files to the root directory of the TFTP server.
- c. Using an ASCII text editor such as vi, do the following:
  1. Open the configuration file.
  2. Update the image version specified in the image\_version variable. The version name in the image\_version variable should match the version name (without the .sbn or .bin extension) of the latest firmware image.
  3. Save the modified file to the TFTP server.
- d. Reinitialize each phone as described in Step 3d above.

**Step 6** If you are upgrading from any previous release to Release 7.0, 7.1, or a later release, see the “Upgrading to Version 7.0 and Later Releases” section on page 4-5.



**Note**

If you do not define the image\_version parameter in the default configuration file, only phones that have an updated phone-specific configuration file with the new image version and that have been restarted use the latest firmware image. All other phones use the older version until their configuration files have been updated with the new image version.

## Upgrading to Version 7.0 and Later Releases



**Note**

Refer to the *Cisco 7940 and 7960 IP Phones Firmware Upgrade Matrix* for additional upgrade information for all releases.

Release 7.x is delivered in a ZIP archive. The ZIP archive contains 6 files:

- P003-07-0-00.bin—New universal loader application file
- P003-07-0-00.sbn—New universal loader application file
- P0S3-07-0-00.sb2—New SIP 7.x release image.
- P0S3-07-0-00.loads—New loader file. The Universal Loader uses the P0S3-07-0-00.loads file to determine which images to put on the phone. This file specifies the Loader application version and the SIP application version. The P0S3-07-0-00.sbn, P0S3-07-0-00.loads, P003-07-0-00.sb2 are all signed files and cannot be altered.
- OS79XX.TXT—Contains the upgrade image of the universal loader used by pre-release 7.0 images.

### Procedure

**Step 1** Unzip the archive contents into your root TFTP directory.

**Step 2** Edit your SIPDefault.cnf and/or SIP<mac addr>.cnf file to have the correct image version. For example:

```
image_version: P0S3-07-0-00
```

**Step 3** Reboot the phone.

The phone automatically upgrades to the Universal Loader application (P003-07-0-00) upon reboot. Once the Universal Loader application starts, it will examine the SIP configuration file to determine which image to use (determined in Step 2 above).

After determining the image version, the Universal Loader application examines the .loads file to determine if the .loads file needs to be upgraded to the SIP .sb2 image (release 7.x). The upgrade will occur automatically, if needed. When upgrading is done, the release 7.x image is initiated.

## How to Upgrade Your Cisco SIP IP Phone Firmware Image and Reboot Remotely

If you have a SIP proxy server and a TFTP server in your phone network, you can upgrade your system firmware image and reboot remotely. You can also push the dialplan.xml file down to the phones.

### Procedure

**Step 1** Edit the default configuration file as follows:

- a. Using an ASCII text editor such as vi, open the SIPDefault.cnf file located in the root directory of your TFTP server.
- b. Change the image\_version parameter to Release 7.4.
- c. Save and close the file.

**Step 2** Edit the synchronization file as follows:

- a. Using a text editor, open the syncinfo.xml file located in the root directory of your TFTP server.
- b. Specify values for the image version and sync parameter as follows:

```
<IMAGE VERSION="image_version" SYNC="sync_number"/>
```

Where the variables are as follows:

- image\_version—Image version of the phone. You can use the asterisk (\*) as a wildcard character.
- sync\_number—Synchronization level of the phone. A valid value is a character string of up to 32 characters. Default is 1.

- c. Save and close the file.

**Step 3** Send a NOTIFY message to the phone. In the message, ensure that the Event header is equal to check-sync. The following is a sample NOTIFY message:

### Sample NOTIFY Message

```
NOTIFY sip:lineX_name@ipaddress:5060 SIP/2.0
Via: SIP/2.0/UDP ipaddress:5060;branch=1
Via: SIP/2.0/UDP ipaddress
From: <sip:webadim@ipaddress>
To: <sip:lineX_name@ipaddress>
Event: check-sync <<---->> Event header.
Date: Mon, 10 Jul 2000 16:28:53 -0700
```

```
Call-ID: 1349882@ipaddress  
CSeq: 1300 NOTIFY  
Contact: <sip:webadmin@ipaddress>  
Content-Length: 0
```

---

During a remote reboot, the phone does the following:

1. If it is idle, the phone waits 20 seconds and contacts the TFTP server for the syncinfo.xml file. Otherwise, it waits until it is idle for 20 seconds and then contacts the TFTP server.
2. The phone reads the syncinfo.xml file and does the following as appropriate:
  - a. Determines whether the current image is specified. If it is not specified, the phone proceeds to Step b. If it is specified, the phone proceeds to Step c.
  - b. Determines whether the image version parameter contains a wildcard entry (\*). If it does, the phone proceeds to Step c. If it does not, the phone proceeds to Step d.
  - c. Determines if the synchronization value is different from that stored on the phone. If it is, the phone proceeds to Step 3. If it is not, the phone proceeds to Step d.
  - d. Does nothing. The procedure is finished.
3. The phone performs a normal reboot.

During reboot, the phone sees the new image and upgrades to it with a synchronization value as specified in the syncinfo.xml file. The procedure is finished.

## Where to Go Next

See Chapter 5, “Monitoring Cisco SIP IP Phones,” for information on debugging and on viewing network statistics.





## Monitoring Cisco SIP IP Phones

---

This chapter provides information on the following:

- How to Use the Command-Line Interface to Monitor Phones, page 5-1
- How to Use the Phone Menus to Access Status Information, page 5-18

### How to Use the Command-Line Interface to Monitor Phones

You can use Telnet or a console to connect to your Cisco IP 7960G/7940G phone, and you can use the command-line interface (CLI) to debug or troubleshoot the phone. Table 5-1 shows the available CLI commands and their syntax.



**Note**

- You need the phone IP address to use the CLI in a Telnet session. To get the IP address, select **Settings > Network Configuration > IP Address**. The default Telnet password is “cisco.”
  - You can conduct only two Telnet sessions at any time.
  - The phone cannot originate a Telnet session to another address.
- 

Table 5-1 describes the available commands.

Table 5-1 CLI Commands

Command	Purpose
<pre>SIP Phone&gt; clear {arp   ethernet   ip   malloc   mwi   tcp-stats}</pre>	<p>Clears the following, depending on the keywords used:</p> <ul style="list-style-type: none"> <li>• <b>arp</b>—Address Resolution Protocol (ARP) cache.</li> <li>• <b>ethernet</b>—Network statistics.</li> <li>• <b>ip</b>—IP statistics.</li> <li>• <b>malloc</b>—Memory allocation.</li> <li>• <b>mwi</b>—Message-waiting indicator.</li> <li>• <b>tcp-stats</b>—TCP statistics.</li> </ul>
<pre>SIP Phone&gt; debug {arp   console-stall   strlib   malloc   malloctable   sk-platform   flash   dsp   vcm   dtmf   task-socket   lsm   fsm   auth   fim   gsm   cc   cc-msg   error   sip-task   sip-state   sip-messages   sip-reg-state   dns   config   sntp   sntp-packet   http   arp-broadcast   xml-events   xml-deck   xml-vars   xml-post}</pre>	<p>Shows detailed debug output for the following, depending on the keywords used:</p> <ul style="list-style-type: none"> <li>• <b>arp</b>—ARP cache.</li> <li>• <b>console-stall</b>—Console-stall driver output mode.</li> <li>• <b>strlib</b>—String library.</li> <li>• <b>malloc</b>—Memory allocation.</li> <li>• <b>malloctable</b>—Memory allocation table. The table can be viewed with the <b>show malloctable</b> command.</li> <li>• <b>sk-platform</b>—Platform.</li> <li>• <b>flash</b>—Flash memory information.</li> <li>• <b>dsp</b>—Digital signal processor (DSP) accesses.</li> <li>• <b>vcm</b>—Voice Channel Manager (VCM), including tones, ringing, and volume.</li> <li>• <b>dtmf</b>—Dual-tone multifrequency (DTMF) relay.</li> <li>• <b>task-socket</b>—Socket task.</li> <li>• <b>lsm</b>—Line State Manager.</li> <li>• <b>fsm</b>—Feature State Manager.</li> <li>• <b>auth</b>—SIP authorization state machine.</li> <li>• <b>fim</b>—Feature Interaction Manager.</li> <li>• <b>gsm</b>—Global State Manager.</li> <li>• <b>cc</b>—Call control.</li> <li>• <b>cc-msg</b>—Call-control messages.</li> <li>• <b>error</b>—General error debug output.</li> <li>• <b>sip-task</b>—SIP task.</li> <li>• <b>sip-state</b>—SIP state machine.</li> <li>• <b>sip-messages</b>—SIP messaging.</li> <li>• <b>sip-reg-state</b>—SIP registration state machine.</li> </ul>

Table 5-1 CLI Commands (continued)

Command	Purpose
<b>debug</b> command keywords (continued)	<ul style="list-style-type: none"> <li>• <b>dns</b>—DNS command-line interface (CLI) configuration; allows you to clear the cache and set servers.</li> <li>• <b>config</b>—Output for the <b>config system</b> command.</li> <li>• <b>sntp</b>—Simple Network Time Protocol (SNTP).</li> <li>• <b>sntp-packet</b>—Full SNTP packet data.</li> <li>• <b>http</b>—HTTP requests and responses.</li> <li>• <b>arp-broadcast</b>—ARP broadcast messages.</li> <li>• <b>xml-events</b>—XML events that are posted to the XML application chain.</li> <li>• <b>xml-deck</b>—XML requests for XML cards and decks.</li> <li>• <b>xml-vars</b>—XML content variables.</li> <li>• <b>xml-post</b>—XML post strings.</li> </ul> <p><b>Note</b> Do not use the <b>debug all</b> command because it can cause the phone to become inoperable. This command is for use only by Cisco TAC personnel.</p> <p><b>Note</b> To turn the debugging off, use the <b>undebug</b> command (works just as does the <b>no debug</b> command).</p>
SIP Phone> <b>dns</b> {-p   -c   -s ip-address   -b ip-address}	Manipulates the DNS system. Keywords and arguments are as follows: <ul style="list-style-type: none"> <li>• <b>-p</b>—Prints the DNS cache table.</li> <li>• <b>-c</b>—Clears the DNS cache table.</li> <li>• <b>-s ip-address</b>—Sets the primary DNS server.</li> <li>• <b>-b ip-address</b>—Sets the first backup server.</li> </ul>
SIP Phone> <b>erase protflash</b>	Erases the protocol area of flash memory. Forces the phone to reset its IP stack and request its configuration files again. This command can be used only if the telnet_level parameter is set to allow privileged commands to be executed.
SIP Phone> <b>exit</b>	Exits the Telnet or console session.
SIP Phone> <b>ping</b> ip-address number packet-size timeout	Sends an Internet Control Message Protocol (ICMP) ping to a network address. The arguments are as follows: <ul style="list-style-type: none"> <li>• <b>ip-address</b>—Dotted IP address or alphanumeric address host name to ping.</li> <li>• <b>number</b>—How many pings to send. Default is 5.</li> <li>• <b>packet size</b>—Size of the packet, in bytes. Range is 1 to 1480. Default is 100.</li> <li>• <b>timeout</b>—How long, in seconds, to wait before a request times out. Default is 2.</li> </ul>

Table 5-1 CLI Commands (continued)

Command	Purpose
SIP Phone> <b>register</b> { <b>option value</b>   <b>line value</b> }	<p>Instructs the Cisco IP 7960G/7940G phone to register with the proxy server. The keywords and argument are as follows:</p> <ul style="list-style-type: none"> <li>• <b>option value</b>—Whether each line is registered. Valid values are 0 (unregistered) and 1 (registered).</li> <li>• <b>line value</b>—Registers the number of lines or specifies a backup proxy. Valid values are 1 to 6 and backup (0). For example, if you enter 0, the phone registers to the backup proxy.</li> </ul>
SIP Phone> <b>reset</b>	<p>Resets the phone line. This command can be used only if the <code>telnet_level</code> parameter is set to allow privileged commands to be executed.</p>
SIP Phone> <b>show</b> { <b>arp</b>   <b>debug</b>   <b>ethernet</b>   <b>ip</b>   <b>strpool</b>   <b>memorymap</b>   <b>dump</b>   <b>malloctable</b>   <b>stacks</b>   <b>status</b>   <b>abort_vector</b>   <b>flash</b>   <b>dspstate</b>   <b>rtp</b>   <b>tcp</b>   <b>lsm</b>   <b>fsm</b>   <b>fsmdef</b>   <b>fsmcnf</b>   <b>fsmxfr</b>   <b>fim</b>   <b>gsm</b>   <b>register</b>   <b>network</b>   <b>config</b>   <b>personaldir</b>   <b>dialplan</b>   <b>timers</b> } [ <b>running</b>   <b>all</b> ]	<p>Shows information about the SIP IP phone, depending on the keywords used:</p> <ul style="list-style-type: none"> <li>• <b>arp</b>—Contents of the ARP cache.</li> <li>• <b>debug</b>—Which debug modes are activated.</li> <li>• <b>ethernet</b>—Network statistics.</li> <li>• <b>ip</b>—IP packet statistics.</li> <li>• <b>strpool</b>—String library pool of strings. This command can be used only if the <code>telnet_level</code> parameter is set to allow privileged commands to be executed.</li> <li>• <b>memorymap</b>—Memory mapping table, including free, used, and wasted blocks.</li> <li>• <b>dump</b>—Dump of the memory contents. This command can be used only if the <code>telnet_level</code> parameter is set to allow privileged commands to be executed.</li> <li>• <b>malloctable</b>—Memory allocation table.</li> <li>• <b>stacks</b>—Tasks and buffer lists.</li> <li>• <b>status</b>—Current phone status, including errors.</li> <li>• <b>abort_vector</b>—Address of the last recorded abort vector.</li> <li>• <b>flash</b>—Flash memory information.</li> <li>• <b>dspstate</b>—DSP status, including whether the DSP is ready, the audio mode, whether keepalive pending is turned on, and the ringer state.</li> <li>• <b>rtp</b>—Packet statistics for the RTP streams.</li> <li>• <b>tcp</b>—Status of TCP ports, including the state (listen or closed) and the port number.</li> <li>• <b>lsm</b>—Current status of the Line State Manager control blocks.</li> </ul>



Table 5-1 CLI Commands (continued)

Command	Purpose
show command keywords (continued)	<ul style="list-style-type: none"> <li>• <b>fsm</b>—Current status of the Feature State Manager function control blocks.</li> <li>• <b>fsmdef</b>—Current status of the Default Feature State Manager data control blocks.</li> <li>• <b>fsmcnf</b>—Current status of the Conference Feature State Manager call control blocks.</li> <li>• <b>fsmxfr</b>—Current status of the Transfer Feature State Manager transfer control blocks.</li> <li>• <b>fim</b>—Current status of the Feature Interaction Manager control blocks (interface control blocks and state control blocks).</li> <li>• <b>gsm</b>—Global State Manager status that includes these parameters: vcm, lsm, fim, fsm, and gsm.</li> <li>• <b>register</b>—Current registration status of SIP lines.</li> <li>• <b>network</b>—Network information, such as phone platform, DHCP server, phone IP address and subnet mask, default gateway, address of the TFTP server, phone MAC address, domain name, and phone name.</li> <li>• <b>config</b>—Current flash memory configuration, including network information, phone label and password, SNTP server address, DST information, time and date format, and input and output port numbers.</li> <li>• <b>personaldir</b>—Current contents of the personal directory. This command can be used only if the telnet_level parameter is set to allow privileged commands to be executed.</li> <li>• <b>dialplan</b>—Phone dial plan.</li> <li>• <b>timers</b>—Current status of the platform timers.</li> <li>• (Optional) <b>running</b>—Shows the running configuration.</li> <li>• <b>all</b>—Shows all.</li> </ul>
<pre>SIP Phone&gt; test {open   close   key {k1 ... k12}   onhook   offhook   show   hide}</pre>	<p>Accesses the remote call test interface, allowing you to control the phone from a remote site. This command can be used only if the telnet_level parameter is set to allow privileged commands to be executed. Keywords are as follows:</p> <ul style="list-style-type: none"> <li>• <b>open</b>—Enables the use of the test functionality.</li> <li>• <b>close</b>—Disables the use of the test functionality.</li> </ul>

Table 5-1 CLI Commands (continued)

Command	Purpose
test command keywords (continued)	<ul style="list-style-type: none"> <li>• <b>key</b>—Simulates key presses. The arguments <i>k1</i> through <i>k12</i> are as follows: <ul style="list-style-type: none"> <li>– k1—voldn—Volume down</li> <li>– k2—volup—Volume up</li> <li>– k3—headset—Headset</li> <li>– k4—spkr—Speaker</li> <li>– k5—mute—Mute</li> <li>– k6—info—Info</li> <li>– k7—msgs—Messages</li> <li>– k8—serv—Services</li> <li>– k9—dir—Directories</li> <li>– k10—set—Settings</li> <li>– k11—navup—Navigate up</li> <li>– k12—navdn—Navigate down</li> </ul> </li> </ul> <p><b>Note</b> You can enter 0 through 9, #, and * in continuous strings to better express typical dialing strings. A typical command is <b>test key 23234</b>.</p> <ul style="list-style-type: none"> <li>• <b>onhook</b>—Simulates a handset on-hook event.</li> <li>• <b>offhook</b>—Simulates a handset off-hook event.</li> <li>• <b>show</b>—Shows test feedback.</li> <li>• <b>hide</b>—Hides test feedback.</li> </ul>

Table 5-1 CLI Commands (continued)

Command	Purpose
SIP Phone> <b>traceroute</b> <i>ip-address</i> [ <i>tth</i> ]	<p>Initiates a traceroute session from the console or from a Telnet session. Traceroute shows the route that IP datagrams follow from the SIP IP phone to the specified IP address. The arguments are as follows:</p> <ul style="list-style-type: none"> <li><i>ip-address</i>—Dotted IP address or alphanumeric address (host name) of the host to which you are sending the traceroute.</li> <li><i>tth</i>—(Optional) Time-to-live value or the number of routers (hops) through which the datagram can pass. Default is 30.</li> </ul>
SIP Phone> <b>tty</b> { <b>echo</b> { <b>on</b>   <b>off</b> }   <b>mon</b>   <b>timeout</b> <i>value</i>   <b>kill</b> <i>session</i>   <b>msg</b> }	<p>Controls the Telnet system. Arguments and keywords are as follows:</p> <ul style="list-style-type: none"> <li><b>echo</b>—Controls local echo. Valid values are on and off.</li> <li><b>mon</b>—Sends all debug output to both the console and the Telnet sessions.</li> <li><b>timeout</b> <i>value</i>—Sets the Telnet session timeout period, in seconds. Range is from 0 to 65535.</li> <li><b>kill</b> <i>session</i>—Tears down the Telnet session specified by the <i>session</i> argument.</li> <li><b>msg</b>—Sends a message to another terminal logged into the phone; for example, you can send a message telling everyone else that is logged in to log off.</li> </ul>

## Output Examples

### Phone Status

The following sample output shows that the proxy servers are not configured:

```
Phone1> show status

Current Phone Status
-----
W351 unprovisioned proxy_emergency
W350 unprovisioned proxy_backup
```

### Telnet Session

The following sample output shows the initial Telnet session using a UNIX server:

```
UNIX% telnet 10.18.10.10

Trying 10.18.10.10...
Connected to 10.18.10.10.

Escape character is '^]'.
Password :*****
```

```

Cisco Systems, Inc. Copyright 2000-2003
Cisco IP phone MAC: 0000:00c0:0d00
Loadid: SW: POS3-05-8-10 ARM: PAS3ARM1 Boot: PC13K030 DSP: PS03AT36

```

### TTY Status

The following sample output shows TTY status:

```

Phone1> tty echo on
Current States:
  echo is 1
  mon is 1
  timeout is 3600 seconds
  prompt is anyone>
  level is 2 - Privileged

```

### String Pool Configuration

The following sample output shows a string for each node:

```

Phone1> show strpool

node_id  refcount  string
-----  -
1         1         sip:48@10.18.192.230
2         1         sip:48@10.18.192.230
3         1         sip:47@10.18.192.230
4         1         sip:47@10.18.192.230
5         1         sip:46@10.18.192.230
6         1         sip:46@10.18.192.230
7         1         sip:duval@10.18.192.230
8         1         sip:duval@10.18.192.230
9         1         sip:44@10.18.192.230
10        1         sip:44@10.18.192.230
11        1         sip:43@10.18.192.230
12        1         sip:43@10.18.192.230
13        1         1234
14        1         25640
15        1         26295@10.18.192.230
16        1         3333
17        1         user33
18        1         3434
19        1         user34
20        1         3636
21        1         user36
22        1         3737
23        1         user37
24        1         3838
25        1         user38
26        1         53@10.18.192.230
27        1         user53
28        1         54
29        1         user54
30        1         5550100
31        1         user1
32        1         5550101
33        1         5550102
34        1         Fid Mantel
35        1         9195550103
36        1         ciscotest@abccompany
37        1         Fid Mantel
38        1         croquet@abc.sip.com
39        1         handball@abc.sip.com
40        1         PGA
41        1         9195550104
42        1         5550105

```

```

43      1  5550106
44      1  53@10.18.192.230
45      1  user53
46      1  3434
47      1  user34
48      1  3333@10.18.192.230
49      1  3333
50      1  mickelson
51      1  pga tour
52     -1

```

### Memory Map

The following sample output shows the memory usage:

```

Phone1> show memorymap

===== MEMORY MAP START =====
free blocks :      11, free block space: 83500, largest free block:  73908
used blocks  :     302, used block space: 34944, largest used block:   4016
wasted block:    1252, str_lib space   :  4864
used space excluding str_lib space : 30080

===== MEMORY MAP END =====

```

### Memory Stacks and Buffer Lists

The following sample output shows the stacks and buffer sizes:

```

Phone1> show stacks

Use show stacks N where N is the task number

Task: SOC      (26) stkhi=0048b174 stklo=0048b973 Size=2048 Unused=1320
Task: RTP      (25) stkhi=0048b974 stklo=0048c173 Size=2048 Unused=1924
Task: PHN      (24) stkhi=0048c174 stklo=0048d173 Size=4096 Unused=2060
Task: GSM      (23) stkhi=0048d174 stklo=0048e973 Size=6144 Unused=5972
Task: SIP      (22) stkhi=0048e974 stklo=00490973 Size=8192 Unused=3268
Task: GUI      (21) stkhi=00490974 stklo=00491973 Size=4096 Unused=1012
Task: NET      (19) stkhi=00491974 stklo=00492173 Size=2048 Unused=536
Task: CFG      (18) stkhi=00492174 stklo=00492973 Size=2048 Unused=912
Task: TTY      (17) stkhi=00492974 stklo=00493973 Size=4096 Unused=3036
Task: AUD      (16) stkhi=00493974 stklo=00494173 Size=2048 Unused=1724
Task: PTMR     (28) stkhi=00494174 stklo=00494973 Size=2048 Unused=1932
Task: TMR      (27) stkhi=004bb60c stklo=004bbe0b Size=2048 Unused=1652

Buffer Lists:

Lst: SOC          Length:000 Max:006 Head:0x00000000 Tail:0x004bed2c
Lst: PHN          Length:000 Max:001 Head:0x00000000 Tail:0x004beac0
Lst: GSM          Length:000 Max:000 Head:0x00000000 Tail:0x004bee6c
Lst: SIP          Length:000 Max:010 Head:0x00000000 Tail:0x004bee48
Lst: RTP          Length:000 Max:000 Head:0x00000000 Tail:0x004beae4
Lst: CFG          Length:000 Max:003 Head:0x00000000 Tail:0x004bed50
Lst: TTY          Length:000 Max:001 Head:0x00000000 Tail:0x004bea9c
Lst: NET          Length:000 Max:005 Head:0x00000000 Tail:0x004bed74
Lst: PktBuf       Length:010 Max:020 Head:0x0048a974 Tail:0x00486174
Lst: TcpBuf       Length:005 Max:006 Head:0x00480174 Tail:0x0047f174
Lst: SysBuf       Length:040 Max:040 Head:0x0047d374 Tail:0x00479d74
Lst: GuiBufTmr    Length:005 Max:005 Head:0x004950b4 Tail:0x00495094
Lst: DSPBuf       Length:030 Max:030 Head:0x00478bd4 Tail:0x00478b74
Lst: DSP_Msg      Length:000 Max:002 Head:0x00000000 Tail:0x004beb98
Lst: DSP_Msg (KA) Length:000 Max:001 Head:0x00000000 Tail:0x004bebbc

```

**Abort Vector**

The following sample output shows the last recorded abort:

```
Phone1> show abort_vector

The Last Abort Vector Recorded [0x0]
```

**Flash Memory**

The following sample output shows the image version that is loaded in flash memory:

```
Phone1> show flash

APP1: loadid:POS3-05-8-10 apid:PAS3ARM1 flags:80000001 Comp Chksum: OK
      chksum:00003371 applen:00078518 cmpchksum:00005e27 cmplen:0005ce54
APP2: loadid:POS3-05-8-10 apid:PAS3ARM1 flags:80000001 Comp Chksum: OK
      chksum:00003371 applen:00078518 cmpchksum:00005e27 cmplen:0005ce54
DSP:  id:PS03AT36 flags:00000000 Chksum: OK
      chksum:00001c7a applen:00016d90 cmpchksum:00000000 cmplen:00000000
```

**DSP Status**

The following sample output shows the status of the DSPs:

```
Phone1> show dspstate

DSP State                : READY
DSP Audio mode           : None
DSP IsStreaming flag     : False
Keep Alive Pending       : False
Ringer state             : Off
  number                  : 2
  volume (dB)             : -17
Progress tone state      : Off
Number of DSP resets since boot : 0
Times DSP was not able to get a buffer: 0
Volumes.. (Range 0 -> 248)
  Speaker - 128
  Headset - 144
  Handset - 144
  Ringer   - 56
```

**RTP Status**

The following sample output shows the status of RTP:

```
Phone1> show rtp

RTP Packets Rx: 0
RTP Packets Tx: 0
anyone> show ethernet ?
Ethernet Mib:
-----
ResErr 00000007, RcvCnt 00001831, RcvErr 00000000, DrpCnt 00000007
BrdCst 00001205, TooLng 00000002, TxCnt 00000398, TxQCnt 00000000
TxQue 00000000, TxQMax 00000000, TxXCol 00000000, TxFram 00000398

Overflow Counters...
UDP 00000000, ICMP 00000000, NonIP 00000000, TCP 00000000
CDP 00000000, Unknown 00000000, Arp 00000000

Use 'clear ethernet' to clear data
```

## TCP Status

The following sample output shows the status of TCP:

```

Phone1> show tcp

Current TCP status...
TCP MIB
Listeners
  Slot State      Port
   1 LISTEN      5888
   2 CLOSED      0000
   3 CLOSED      0000
   4 CLOSED      0000
   5 CLOSED      0000
   6 CLOSED      0000
   7 CLOSED      0000
   8 CLOSED      0000

Connections
  Conn State      Rem Address      RPort LPort
   1 ESTABLISHED  10.70.67.166    56455 00023
   2 LISTEN      10.70.67.166    56451 00023
   3 CLOSED      0.0.0.0         00000 00000
   4 CLOSED      0.0.0.0         00000 00000
   5 CLOSED      0.0.0.0         00000 00000
   6 CLOSED      0.0.0.0         00000 00000
   7 CLOSED      0.0.0.0         00000 00000
   8 CLOSED      0.0.0.0         00000 00000

Statistics
  ActOpens:00000001 PsvOpen:00000001 AttFail:00000000 EstRsts:00000000
  CurrEstab:00000001 InSegs:00000530 OutSegs:00000330 RetransSegs:00000000
  OutPeer:00000011 InErrs:00000000 OutRsts:00000001 PktBufErrs: 00000000

Telnet Stats
Conn#1      Throttles:00000000
Conn#2      Throttles:00000000

```

## Dial-Plan Configuration

The following sample output shows the dial plan:

```

Phone1> show dialplan

Dialplan is...
01. Pattern: 0 Rewrite:
   Timeout: 0001 UserMode: Phone RouteMode: Default
02. Pattern: 9,011* Rewrite:
   Timeout: 0006 UserMode: Phone RouteMode: Default
03. Pattern: 9,0 Rewrite:
   Timeout: 0008 UserMode: Phone RouteMode: Default
04. Pattern: 9,11 Rewrite:
   Timeout: 0000 UserMode: Phone RouteMode: Emergency
05. Pattern: w! Rewrite:
   Timeout: 0001 UserMode: Phone RouteMode: Emergency
06. Pattern: 9,.11 Rewrite:
   Timeout: 0000 UserMode: Phone RouteMode: Default
07. Pattern: 9,101..... Rewrite:
   Timeout: 0000 UserMode: Phone RouteMode: Default
08. Pattern: 9,10..... Rewrite:
   Timeout: 0000 UserMode: Phone RouteMode: Default
09. Pattern: 9,10* Rewrite:
   Timeout: 0006 UserMode: Phone RouteMode: Default

```

```

10. Pattern: 9,1..... Rewrite:
    Timeout: 0000 UserMode: Phone RouteMode: Default
11. Pattern: 9000 Rewrite:
    Timeout: 0000 UserMode: Phone RouteMode: Default
12. Pattern: 9,..... Rewrite:
    Timeout: 0000 UserMode: Phone RouteMode: Default
13. Pattern: ..... Rewrite: 91%s
    Timeout: 0015 UserMode: Phone RouteMode: Default
14. Pattern: * Rewrite:
    Timeout: 0015 UserMode: Unspecified RouteMode: Default

```

### Personal Directory Configuration

The following sample output shows the entries in the personal directory of the phone:

```

Phone1> show personaldir

0: 28 (L1/T45) 2003/2/27 17:04:29 "1234"
1: 1667 (L1/T45) 2003/7/8 15:10:15 "3333"
2: 42 (L2/T45) 2003/7/8 15:10:23 "3434"
3: 20 (L4/T45) 2003/7/8 15:08:42 "3636"
4: 8 (L5/T45) 2003/7/8 15:08:49 "3737"
5: 35 (L6/T45) 2003/7/8 15:09:02 "3838"
6: 326 (L1/T45) 2003/7/8 11:08:53 "53@10.10.10.0"
7: 65 (L1/T45) 2003/6/26 14:42:49 "54"
8: 53 (L1/T45) 2003/3/31 17:04:17 "5550100"
9: 6 (L1/T45) 2002/12/20 13:42:50 "5550110"
Kazoo-9 Phone
10: 13 (L1/T45) 2002/8/29 16:38:14 "9195550111"
11: 6 (L1/T45) 2002/3/1 12:37:29 "9195550111@abc.com"
Fid Mantel
12: 12 (L1/T45) 2002/1/7 17:42:10 "9195550111"
13: 6 (L1/T45) 2003/7/9 17:07:54 "5550111"
14: 5 (L1/T45) 2002/3/8 17:19:59 "ciscotest@abc.com"
Fid Mantel
15: 41 (L3/T45) 2000/1/5 15:56:17 "croquet@abc.sip.com"
16: 77 (L3/T45) 2000/1/5 15:55:48 "handball@abc.sip.com"
PGA
17: 4 (L1/T45) 2002/5/13 13:16:57 "5550111"
18: 25 (L1/T45) 2002/7/25 10:57:02 "5550111"
5550100
19: 153 (L1/T45) 2002/7/24 15:06:23 "53@10.10.10.10"
20: 8 (L1/T45) 2002/7/23 15:02:52 "3434"
21: 15 (L1/T45) 2002/7/16 10:09:15 "3333@10.10.10.10"
22: 601 (L1/T45) 2002/7/25 18:08:47 "3333"
23: 5 (L3/T45) 2003/7/8 15:10:44 "mickelson"
PGAtour

```

### LSM Parameters

The following sample output shows the LSM parameters:

```

Phone1> show lsm

----- LSM lcbs -----
i  call_id  line  state  lcb
-----
0  0         0     IDLE   0x004e1f00
1  0         0     IDLE   0x004e1f14
2  0         0     IDLE   0x004e1f28
3  0         0     IDLE   0x004e1f3c
4  0         0     IDLE   0x004e1f50
5  0         0     IDLE   0x004e1f64

```



**FSM Parameters**

The following sample output shows the FSM parameters:

```
Phone1> show fsm
```

```
----- FSM fcbs -----
i   call_id  fcb          type          state          dcb          cb
-----
0   0         0x004e2628  UNDEFINED    IDLE           0x00000000  0x00000000
1   0         0x004e2644  UNDEFINED    IDLE           0x00000000  0x00000000
2   0         0x004e2660  UNDEFINED    IDLE           0x00000000  0x00000000
3   0         0x004e267c  UNDEFINED    IDLE           0x00000000  0x00000000
4   0         0x004e2698  UNDEFINED    IDLE           0x00000000  0x00000000
5   0         0x004e26b4  UNDEFINED    IDLE           0x00000000  0x00000000
6   0         0x004e26d0  UNDEFINED    IDLE           0x00000000  0x00000000
7   0         0x004e26ec  UNDEFINED    IDLE           0x00000000  0x00000000
8   0         0x004e2708  UNDEFINED    IDLE           0x00000000  0x00000000
9   0         0x004e2724  UNDEFINED    IDLE           0x00000000  0x00000000
10  0         0x004e2740  UNDEFINED    IDLE           0x00000000  0x00000000
11  0         0x004e275c  UNDEFINED    IDLE           0x00000000  0x00000000
12  0         0x004e2778  UNDEFINED    IDLE           0x00000000  0x00000000
13  0         0x004e2794  UNDEFINED    IDLE           0x00000000  0x00000000
14  0         0x004e27b0  UNDEFINED    IDLE           0x00000000  0x00000000
15  0         0x004e27cc  UNDEFINED    IDLE           0x00000000  0x00000000
16  0         0x004e27e8  UNDEFINED    IDLE           0x00000000  0x00000000
17  0         0x004e2804  UNDEFINED    IDLE           0x00000000  0x00000000
```

**FSMDEF Parameters**

The following sample output shows the FSMDEF parameters:

```
Phone1> show fsmdef all
```

```
----- FSMDEF dcbs -----
i   call_id  dcb          line
-----
0   0         0x004e1f84  0
1   0         0x004e2084  0
2   0         0x004e2184  0
3   0         0x004e2284  0
4   0         0x004e2384  0
5   0         0x004e2484  0
```

**FSMXFR Parameters**

The following sample output shows the FSMXFR parameters:

```
Phone1> show fsmxfr
```

```
----- FSMXFR xcbs -----
i   xfr_id  xcb          type  method  xfr_call_id  cns_call_id
-----
0   0         0x004e25c4  0     0       0             0
1   0         0x004e25f0  0     0       0             0
```

**FIM Parameters**

The following sample output shows the FIM parameters:

```
Phone1> show fim
```

```
----- FIM icbs -----
i  call_id  type  icb          next_chn      next_icb      cb           scb
-----
0  0         HEAD  0x004e2878  0x004e28c8  0x004e288c  0x00000000  0x004e282c
1  0         CNF   0x004e288c  0x00000000  0x004e28a0  0x00000000  0x004e283c
2  0         XFR   0x004e28a0  0x00000000  0x004e28b4  0x00000000  0x004e284c
3  0         DEF   0x004e28b4  0x00000000  0x00000000  0x00000000  0x004e285c
4  0         HEAD  0x004e28c8  0x004e2918  0x004e28dc  0x00000000  0x004e282c
5  0         CNF   0x004e28dc  0x00000000  0x004e28f0  0x00000000  0x004e283c
6  0         XFR   0x004e28f0  0x00000000  0x004e2904  0x00000000  0x004e284c
7  0         DEF   0x004e2904  0x00000000  0x00000000  0x00000000  0x004e285c
8  0         HEAD  0x004e2918  0x004e2968  0x004e292c  0x00000000  0x004e282c
9  0         CNF   0x004e292c  0x00000000  0x004e2940  0x00000000  0x004e283c
10 0         XFR   0x004e2940  0x00000000  0x004e2954  0x00000000  0x004e284c
11 0         DEF   0x004e2954  0x00000000  0x00000000  0x00000000  0x004e285c
12 0         HEAD  0x004e2968  0x004e29b8  0x004e297c  0x00000000  0x004e282c
13 0         CNF   0x004e297c  0x00000000  0x004e2990  0x00000000  0x004e283c
14 0         XFR   0x004e2990  0x00000000  0x004e29a4  0x00000000  0x004e284c
15 0         DEF   0x004e29a4  0x00000000  0x00000000  0x00000000  0x004e285c
16 0         HEAD  0x004e29b8  0x004e2a08  0x004e29cc  0x00000000  0x004e282c
17 0         CNF   0x004e29cc  0x00000000  0x004e29e0  0x00000000  0x004e283c
18 0         XFR   0x004e29e0  0x00000000  0x004e29f4  0x00000000  0x004e284c
19 0         DEF   0x004e29f4  0x00000000  0x00000000  0x00000000  0x004e285c
20 0         HEAD  0x004e2a08  0x00000000  0x004e2a1c  0x00000000  0x004e282c
21 0         CNF   0x004e2a1c  0x00000000  0x004e2a30  0x00000000  0x004e283c
22 0         XFR   0x004e2a30  0x00000000  0x004e2a44  0x00000000  0x004e284c
23 0         DEF   0x004e2a44  0x00000000  0x00000000  0x00000000  0x004e285c
```

```
----- FIM scbs -----
i  type  scb          sm          get_cb       free_cb
-----
0  HEAD  0x004e282c  0x00000000  0x00000000  0x00000000
1  CNF   0x004e283c  0x004c0ca8  0x00457bcf  0x0044e291
2  XFR   0x004e284c  0x004c1414  0x00457bcf  0x0043f755
3  DEF   0x004e285c  0x004bf28c  0x00457bcf  0x0042896b
```

**Registration Assignments**

The following sample output shows the registration of the proxy ports:

```
Phone1> show register
```

```
LINE REGISTRATION TABLE
```

```
Proxy Registration: ENABLED, state: REGISTERED
```

```
line  APR  state      timer      expires    proxy:port
-----
1     .11  REGISTERED 3595      2539      10.18.192.230:5060
2     .11  REGISTERED 3595      2539      10.18.192.230:5060
3     .11  REGISTERED 3595      2539      10.18.192.230:5060
4     .11  REGISTERED 3595      2540      10.18.192.230:5060
5     .11  REGISTERED 3595      2543      10.18.192.230:5060
6     .11  REGISTERED 3595      2543      10.18.192.230:5060
1-BU .1x  NONE       0          0          undefined:0
```

Note: APR is Authenticated, Provisioned, Registered

### Network Flash Configuration

The following sample output shows the network configuration that resides in flash memory:

```
Phone1> show network

----- Network *FLASH* Configuration -----

Platform : Cisco IP Phone 7960
Elapsed Time: 00:17:57

dhcp_server : 10.18.192.230
my_ip_addr : 10.18.199.14
subnet_mask : 255.255.255.0
defaultgw : 10.18.199.1
dyn_dns_addr_1 : 0.0.0.0
dyn_dns_addr_2 : 0.0.0.0
dns_addr : 10.18.192.48
tftp_addr : 10.10.92.150
dyn_tftp_addr : 0.0.0.0
my_mac_addr : 0030:94c2:5d40
domain_name : sip.com
my_name : SIP003094C25D40
Status Flags : 12300000
```

### Running Network Configuration

The following sample output shows the running configuration:

```
Phone1> show network running

----- Network *RUNNING* Configuration -----

Platform : Cisco IP Phone 7960
Elapsed Time: 00:18:11

dhcp_server : 10.18.192.230
my_ip_addr : 10.18.199.14
subnet_mask : 255.255.255.0
defaultgw : 10.18.199.1
dyn_dns_addr_1 : 0.0.0.0
dyn_dns_addr_2 : 0.0.0.0
dns_addr : 10.18.192.48
tftp_addr : 10.102.92.150
dyn_tftp_addr : 0.0.0.0
my_mac_addr : 0030:94c2:5d40
domain_name : sip.com
my_name : SIP003094C25D40
Status Flags : 12300000
```

### ARP Table

The following sample output shows the ARP table by IP address:

```
Phone1> show arp

Arp Table:
[00] IPAddr: 10.18.199.14 PortCnt: 0001 MacAddr: 0030:94c2:5d40
     Type: 00000001 GTick:00001287 LastTry: 00000000
     Mode: 00000001 Update: 00000000

[01] IPAddr: 10.18.199.1 PortCnt: 0001 MacAddr: 0000:0c07:ac08
     Type: 00000001 GTick:00001287 LastTry: 00001287
     Mode: 00000001 Update: 00000000
```

**Flash Configuration**

The following sample output shows the flash memory configuration:

```
Phone1> show config

----- Current *FLASH* Configuration -----

Platform : Cisco IP Phone 7960
Elapsed Time: 00:18:32

dhcp_server : 10.18.192.230
my_ip_addr : 10.18.199.14
subnet_mask : 255.255.255.0
defaultgw : 10.18.199.1
dyn_dns_addr_1 : 0.0.0.0
dyn_dns_addr_2 : 0.0.0.0
dns_addr : 10.18.192.48
tftp_addr : 10.10.92.150
dyn_tftp_addr : 0.0.0.0
my_mac_addr : 0030:94c2:5d40
domain_name : sip.com
my_name : SIP003094C25D40
Status Flags : 12300000

image_version : "P0S3-05-8-10"
FirmLoadID : "PC13K030"
DSPLoadID : "PS03AT36"
network_media_type : Half10
network_port2_type : Hub/Switch
tos_media : 5
phone_label : "user4X"
tftp_cfg_dir : "/"
phone_password : *****
phone_prompt : "Phone1"
language : english
snmp_mode : DirectedBroadcast
snmp_server : 10.10.10.150
time_zone : EST
dst_offset : 1
dst_start_month : April
dst_start_day : 0
dst_start_day_of_week : Sun
dst_start_week_of_month : 1
dst_start_time : 02
dst_stop_month : Oct
dst_stop_day : 0
dst_stop_day_of_week : Sunday
dst_stop_week_of_month : 8
dst_stop_time : 2
dst_auto_adjust : 1
time_format_24hr : 1
date_format : M/D/Y
nat_enable : 0
nat_address : UNPROVISIONED
voip_control_port : 5060
start_media_port : 16384
end_media_port : 32766
sync : "1"
xml_card_dir : ""
xml_card_file : "CARD.XML"
telnet_level : 2
services_url : "http://10.10.149.2/ciscodir/directory.xml"
directory_url : "http://10.10.93.154/CiscoServices/Directory.asp"
```

```
logo_url : "http://10.10.207.20/projects/phone/company.bmp"  
http_proxy_addr : UNPROVISIONED  
http_proxy_port : 80  
enable_vad : 1  
dial_template : "dialplan"  
callerid_blocking : 0  
anonymous_call_block : 0  
autocomplete : 1  
messages_uri : "1234567"  
dnd_control : 0  
preferred_codec : g729a  
dtmf_outofband : avt  
dtmf_avt_payload : 101  
dtmf_db_level : 3  
dtmf_inband : 1  
line1_name : "43"  
line2_name : "44"  
line3_name : "duval"  
line4_name : "46"  
line5_name : "47"  
line6_name : "48"  
line1_authname : "UNPROVISIONED"  
line2_authname : "UNPROVISIONED"  
line3_authname : "UNPROVISIONED"  
line4_authname : "UNPROVISIONED"  
line5_authname : "UNPROVISIONED"  
line6_authname : "UNPROVISIONED"  
line1_password : *****  
line2_password : *****  
line3_password : *****  
line4_password : *****  
line5_password : *****  
line6_password : *****  
line1_shortcode : "UNPROVISIONED"  
line2_shortcode : "UNPROVISIONED"  
line3_shortcode : "UNPROVISIONED"  
line4_shortcode : "UNPROVISIONED"  
line5_shortcode : "UNPROVISIONED"  
line6_shortcode : "UNPROVISIONED"  
line1_displayname : "user43"  
line2_displayname : "user44"  
line3_displayname : "pgatour"  
line4_displayname : "user46"  
line5_displayname : "user47"  
line6_displayname : "user48"  
proxy1_address : "10.10.10.0"  
proxy2_address : "10.10.10.0"  
proxy3_address : "10.10.10.0"  
proxy4_address : "10.10.10.0"  
proxy5_address : "10.10.10.0"  
proxy6_address : "10.10.10.0"  
proxy1_port : 5060  
proxy2_port : 5060  
proxy3_port : 5060  
proxy4_port : 5060  
proxy5_port : 5060  
proxy6_port : 5060  
sip_retx : 10  
sip_invite_retx : 6  
timer_t1 : 2000  
timer_t2 : 4000  
timer_invite_expires : 180  
timer_register_expires : 3600  
proxy_register : 1
```

```

proxy_backup : ""
proxy_emergency : "UNPROVISIONED"
proxy_backup_port : 6060
proxy_emergency_port : 5060
outbound_proxy : UNPROVISIONED
outbound_proxy_port : 5060
nat_received_processing : 0
mwi_status : 0
call_waiting : 1
user_info : none
cnf_join_enable : 1
remote_party_id : 0
semi_attended_transfer : 1
call_hold_ringback : 0
cfwd_url : ""
call_stats : 0
auto_answer : 0
speed_line2 : ""
speed_label2 : ""
speed_line3 : ""
speed_label3 : ""
speed_line4 : ""
speed_label4 : ""
speed_line5 : ""
speed_label5 : ""
speed_line6 : ""
speed_label6 : ""

```

### IP Statistics

The following sample output shows the IP statistics:

```
Phone1> show ip
```

```
IP Statistics:
```

```

-----
Received      00002623, RxDrops      00000006
RxFrags       00000000, RxFragDrops 00000000, RxReassembled 00000000
Transmitted   00000869, TxDrops      00000000, TxFragments  00000000

```

```
Use 'clear ip' to clear data
```

## How to Use the Phone Menus to Access Status Information

You can access several types of status information using the Settings button. The information that you can obtain can aid in system management and diagnosis of network problems. This section contains the following:

- Viewing Status Messages, page 5-19
- Viewing Network Statistics, page 5-19

## Viewing Status Messages

You can view status messages that you can use to diagnose network problems.

### Procedure

- 
- Step 1** Select **Settings > Status > Status Messages**. The Status Messages menu displays.
- Step 2** View information as needed.
- Step 3** Select **Exit**.
- 

## Viewing Network Statistics

You can view statistics about the phone and network performance.

### Procedure

- 
- Step 1** Select **Settings > Status > Network Statistics**. The Network Statistics menu displays.
- Step 2** View the following information as needed:
- Rcv—Number of packets received by the phone, not through the switch.
  - Xmit—Number of packets sent by the phone, not through the switch.
  - REr—Number of packets received by the phone that contained errors.
  - BCast—Number of broadcast packets received by the phone.
  - Phone State Message—TCP messages that indicate the state of the phone. The following are possible messages:
    - Phone Initialized—TCP connection has not gone down since the phone was powered on.
    - Phone Closed TCP—TCP connection was closed by the phone.
    - TCP Timeout—TCP connection was closed because of a retry timeout.
    - Error Code—Error messages that indicate unusual reasons for which the TCP connection was closed.
  - Elapsed Time—Length of time (in days, hours, minutes, and seconds) since the last power cycle.
  - Port 0 Full, 100—Indication that the network is in a linked state and has autonegotiated a full-duplex 100-Mbps connection.
  - Port 0 Half, 100—Indication that the network is in a linked state and has autonegotiated a half-duplex 100-Mbps connection.
  - Port 0 Full, 10—Indicates that the network is in a linked state and has autonegotiated a full-duplex 10-Mbps connection.
  - Port 0 Half, 10—Indication that the network is in a linked state and has autonegotiated a half-duplex 10-Mbps connection.
  - Port 1 Full, 100—Indication that the network is in a linked state and has autonegotiated a full-duplex 100-Mbps connection.

- Port 1 Half, 100—Indication that the network is in a linked state and has autonegotiated a half-duplex 100-Mbps connection.
- Port 1 Full, 10—Indication that the network is in a linked state and has autonegotiated a full-duplex 10-Mbps connection.
- Port 1 Half, 10—Indication that the network is in a linked state and has autonegotiated a half-duplex 10-Mbps connection.

**Step 3** Select **Exit**.



---

**Note** To reset the values, power the phone off and on.

---





## Compliance with RFC 3261

---

This appendix describes how the Cisco IP 7960G/7940G phone complies with the IETF definition of SIP as described in RFC 3261. It contains compliance information on the following:

- SIP Functions, page A-1
- SIP Methods, page A-2
- SIP Responses, page A-2
- SIP Header Fields, page A-6
- SIP Session Description Protocol Usage, page A-7
- Transport Layer Protocols, page A-8
- SIP Security Authentication, page A-8
- SIP DNS Records Usage, page A-8
- SIP DTMF Digit Transport, page A-8

### SIP Functions

Function	Supported?
User agent client (UAC)	Yes
User agent server (UAS)	Yes
Proxy server	Third-party only
Redirect server	Third-party only

## SIP Methods

Method	Supported?	Comments
INVITE	Yes	The phone supports midcall changes such as putting a call on hold as signaled by a new INVITE that contains an existing call ID.
ACK	Yes	—
OPTIONS	Response only	
BYE	Yes	
CANCEL	Yes	
REGISTER	Yes	The phone supports both user and device registration.
REFER	Yes	—
NOTIFY	Yes	Used for REFER and remote reboot.

## SIP Responses

The Cisco IP 7960G/7940G phone supports the following SIP responses:

- 1xx Response—Information Responses, page A-2
- 2xx Response—Successful Responses, page A-3
- 3xx Response—Redirection Responses, page A-3
- 4xx Response—Request Failure Responses, page A-3
- 5xx Response—Server Failure Responses, page A-5
- 6xx Response—Global Responses, page A-6

### 1xx Response—Information Responses

1xx Response	Supported?	Comments
100 Trying	Yes	The phone generates this response for an incoming INVITE. Upon receiving this response, the phone waits for a 180 Ringing, 183 Session Progress, or 200 OK response.
180 Ringing	Yes	—

1xx Response	Supported?	Comments
181 Call Is Being Forwarded	See comments	The phone does not generate these responses; however, the phone does receive them. The phone processes these responses in the same way that it processes the 100 Trying response.
182 Queued		
183 Session Progress		The phone does not generate this message. Upon receiving this response, the phone provides early media cut-through and then waits for a 200 OK response.

## 2xx Response—Successful Responses

2xx Response	Supported?	Comments
200 OK	Yes	—
202 Accepted	Yes	—

## 3xx Response—Redirection Responses

3xx Response	Supported?	Comments
300 Multiple Choices	Yes	—
301 Moved Permanently	Yes	
302 Moved Temporarily	Yes	The phone generates this response when local call forwarding is enabled. Upon receiving this response, the phone sends an INVITE that contains the contact information received in the 302 Moved Temporarily response.
305 Use Proxy	Yes	The phone does not generate these responses. The gateway contacts the new address in the Contact header field.
380 Alternate Service	Yes	

## 4xx Response—Request Failure Responses

4xx Response	Supported?	Comments
400 Bad Request	Yes	The phone generates a 400 Bad Request response for an erroneous request. For an incoming response, the phone initiates a graceful call disconnect (during which the caller hears a fast busy tone).
401 Unauthorized	Yes	If a 401 Unauthorized response is received, the phone accepts the response and sends a new request that contains the user's authentication information in the format of the HTTP digest as modified by RFC 3261.

4xx Response	Supported?	Comments
402 Payment Required	Yes	The phone does not generate the 402 Payment Required response.
403 Forbidden	Yes	If the phone receives a 403 Forbidden response, it notifies the user of the response. This response indicates that the SIP server has the request but will not provide service.
404 Not Found	Yes	The phone generates this response if it is unable to locate the callee. Upon receiving this response, the phone notifies the user.
405 Method Not Allowed	Yes	If the phone receives a 405 Method Not Allowed response, it notifies the user of the response.
406 Not Acceptable	Yes	The phone does not generate a 406 Not Acceptable response. For an incoming response, the gateway initiates a graceful call disconnect (during which the caller hears a busy or fast busy tone) before clearing the call request.
407 Proxy Authentication Required	Yes	If a 407 Proxy Authorization Required response is received, the phone accepts the response and sends a new request that contains the user's authentication information in the format of the HTTP digest as modified by RFC 3261.
408 Request Timeout	Yes	For an incoming response, the phone initiates a graceful call disconnect (during which a caller hears a fast busy). The phone generates this response after an INVITE Expires timeout to avoid leaving the phone in a continuous ringing state (fail safe mechanism). This will occur only if the phone does not receive any messages from the network to tear down the call after the timer expires.
409 Conflict	Yes	The 409 Conflict response indicates that the INVITE request could not be processed because of a conflict with the current state of the resource. If this response is received, the user is notified.
410 Gone	Yes	The 410 Gone response indicates that a resource is no longer available at the server and no forwarding address is known.
411 Length Required	Yes	This response indicates that the user refuses to accept the request without a defined content length. If received, the phone resends the INVITE request if it can add a valid Content-Length header field.
413 Request Entity Too Large	Yes	If a retry-after header field is contained in this response, the user can attempt the call once again in the retry time provided.
414 Request—URL Too Long	Yes	The user is notified if this response is received.
415 Unsupported Media	Yes	The user is notified if this response is received.

4xx Response	Supported?	Comments
420 Bad Extension	Yes	The user is notified if this response is received. If the phone does not understand the protocol extension specified in the Require field, the 420 Bad Extension response is generated.
480 Temporarily Unavailable	Yes	The response is received only by the phone. The user is notified if this response is received.
481 Call Leg/Transaction Does Not Exist	Yes	The user is notified if this response is received.
482 Loop Detected		
483 Too Many Hops		
484 Address Incomplete		
485 Ambiguous	Yes	If a new contact is received, the phone might reinitiate the call.
486 Busy Here	Yes	The phone generates this response if the called party is off-hook and the call cannot be presented as a call-waiting call. Upon receiving this response, the phone notifies the user and generates a busy tone.
487 Request Canceled	Yes	This response indicates that the initial request is terminated with a BYE or CANCEL request.
488 Not Acceptable	Yes	When media cannot be negotiated properly, the phone receives and generates this response.

## 5xx Response—Server Failure Responses

5xx Response	Supported?	Comments
500 Internal Server Error	Yes	The phone initiates a graceful call disconnect, and the user is notified.
501 Not Implemented	Yes	
502 Bad Gateway	Yes	
503 Service Unavailable	Yes	
504 Gateway Timeout	Yes	
505 Version Not Supported	Yes	

## 6xx Response—Global Responses

6xx Response	Supported?	Comments
600 Busy Everywhere	Yes	The phone initiates a graceful call disconnect, and the user is notified.
603 Decline	Yes	
604 Does Not Exist Anywhere	Yes	
606 Not Acceptable	Yes	

## SIP Header Fields

Header Field	Supported?
Accept	Yes
Accept-Encoding	Yes
Accept-Language	Yes
Allow	Yes
Also	Yes
Authorization	Yes
Call-ID	Yes
Contact	Yes
Content-Encoding	Yes
Content-Length	Yes
Content-Type	Yes
Cseq	Yes
Date	Yes
Encryption	No
Expires	Yes
From	Yes
Hide	No
Max-Forwards	Yes
Organization	No
Priority	No
Proxy-Authenticate	Yes
Proxy-Authorization	Yes
Proxy-Require	Yes
Record-Route	Yes

Header Field	Supported?
Referred-By	Yes
Referred-To	Yes
Remote-Party-ID	Yes
Replaces	Yes
Requested-By	Yes
Require	Yes
Response-Key	No
Retry-After	Yes
Route	Yes
Server	Yes
Subject	No
Timestamp	Yes
To	Yes
Unsupported	Yes
User-Agent	Yes
Via	Yes
Warning	Yes
WWW-Authenticate	Yes

## SIP Session Description Protocol Usage

SDP Headers	Supported?
v—Protocol version	Yes
o—Owner or creator and session identifier	Yes
s—Session name	Yes
t—Time description	Yes
c—Connection information	Yes
m—Media name and transport address	Yes
a—Media attribute lines	Yes

## Transport Layer Protocols

Protocol	Supported?
Unicast UDP	Yes
Multicast UDP	No
TCP	No

## SIP Security Authentication

Basic Authentication	Supported?
Digest Authentication	Yes
Proxy Authentication	No
PGP	No

## SIP DNS Records Usage

DNS Resource Record Type	Supported?
Type A	Yes
Type SRV	Yes
NAPTR	No

## SIP DTMF Digit Transport

Transport Type	Supported?
RFC 2833	Yes
In-band tones	Yes





## SIP Call Flows

---

This appendix includes the following sections:

- Call Flow Scenarios for Successful Calls, page B-1
- Call Flow Scenarios for Failed Calls, page B-47

SIP uses the following request methods:

- INVITE—Indicates that a user or service is being invited to participate in a call session.
- ACK—Confirms that the client has received a final response to an INVITE request.
- BYE—Terminates a call and can be sent by either the caller or the called party.
- CANCEL—Cancels any pending searches but does not terminate a call that has already been accepted.
- OPTIONS—Queries the capabilities of servers.
- REGISTER—Registers the address listed in the To header field with a SIP server.
- REFER—Indicates that the user (recipient) should contact a third party for use in transferring parties.
- NOTIFY—Notifies the user of the status of a transfer using REFER. Also used for remote reboot and message waiting indication (MWI).

The following types of responses are used by SIP and generated by the Cisco SIP gateway:

- SIP 1xx—Informational Responses
- SIP 2xx—Successful Responses
- SIP 3xx—Redirection Responses
- SIP 4xx—Client Failure Responses
- SIP 5xx—Server Failure Responses
- SIP 6xx—Global Failure Responses

## Call Flow Scenarios for Successful Calls

This section describes successful call flow scenarios, which are as follows:

- Gateway to Cisco SIP IP Phone in a SIP Network, page B-2
- Cisco SIP IP Phone to Cisco SIP IP Phone, page B-7

## Gateway to Cisco SIP IP Phone in a SIP Network

The following scenarios describe and illustrate successful calls in a gateway to a Cisco SIP IP phone:

- Call Setup and Disconnect, page B-2
- Call Setup and Hold, page B-4
- Call to a Gateway Acting As an Emergency Proxy from a Cisco SIP IP Phone, page B-6

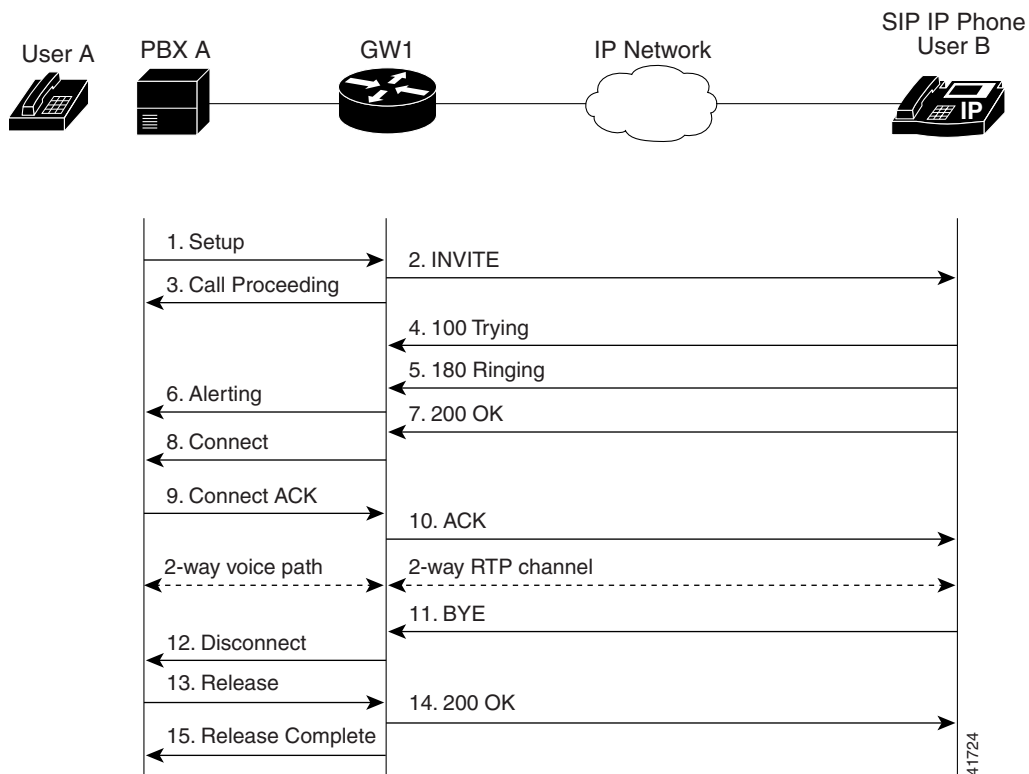
### Call Setup and Disconnect

Figure B-1 illustrates a successful phone-call setup and disconnect. In this scenario, the two end users are User A and User B. User A is located at PBX A. PBX A is connected to Gateway 1 (SIP gateway) via a T1/E1. User B is located at a Cisco SIP IP phone. Gateway 1 is connected to the Cisco SIP IP phone over an IP network.

The call flow is as follows:

1. User A calls User B.
2. User B answers the call.
3. User B hangs up.

**Figure B-1 Successful Setup and Disconnect**



Step	Action	Description
1.	Setup—PBX A to Gateway 1	Call setup is initiated between PBX A and Gateway 1. Setup includes the standard transactions that take place as User A attempts to call User B.
2.	INVITE—Gateway 1 to Cisco SIP IP phone	Gateway 1 maps the SIP URL phone number to a dial peer. The dial peer includes the IP address and the port number of the SIP-enabled entity to contact. Gateway 1 sends a SIP INVITE request to the address it receives as the dial peer, which, in this scenario, is the IP phone. In the INVITE request: <ul style="list-style-type: none"> <li>• The IP address of the phone is inserted in the Request-URI field.</li> <li>• PBX A is identified as the call session initiator in the From field.</li> <li>• A unique numeric identifier is assigned to the call and is inserted in the Call-ID field.</li> <li>• The transaction number within a single call leg is identified in the CSeq field.</li> <li>• The media capability that User A is ready to receive is specified.</li> <li>• The port on which the gateway is prepared to receive the RTP data is specified.</li> </ul>
3.	Call Proceeding—Gateway 1 to PBX A	Gateway 1 sends a Call Proceeding message to PBX A to acknowledge the Call Setup request.
4.	100 Trying—Cisco SIP IP phone to Gateway 1	The phone sends a SIP 100 Trying response to Gateway 1. The response indicates that the INVITE request has been received.
5.	180 Ringing—Cisco SIP IP phone to Gateway 1	The phone sends a SIP 180 Ringing response to Gateway 1. The response indicates that the user is being alerted.
6.	Alerting—Gateway 1 to PBX A	Gateway 1 sends an Alert message to User A. The message indicates that Gateway 1 has received a 180 Ringing response from the phone. User A hears the ringback tone that indicates that User B is being alerted.
7.	200 OK—Cisco SIP IP phone to Gateway 1	The phone sends a SIP 200 OK response to Gateway 1. The response notifies Gateway 1 that the connection has been made.
8.	Connect—Gateway 1 to PBX A	Gateway 1 sends a Connect message to the PBX A. The message notifies PBX A that the connection has been made.
9.	Connect ACK—PBX A to Gateway 1	PBX A acknowledges Gateway 1's Connect message.
10.	ACK—Gateway 1 to Cisco SIP IP phone	Gateway 1 sends a SIP ACK to the phone. The ACK confirms that Gateway 1 has received the 200 OK response. The call session is now active.
11.	BYE—Cisco SIP IP phone to Gateway 1	User B terminates the call session. The phone sends a SIP BYE request to Gateway 1. The request indicates that User B wants to release the call.
12.	Disconnect—Gateway 1 to PBX A	Gateway 1 sends a Disconnect message to PBX A.
13.	Release—PBX A to Gateway 1	PBX A sends a Release message to Gateway 1.

Step	Action	Description
14.	200 OK—Gateway 1 to Cisco SIP IP phone	Gateway 1 sends a SIP 200 OK response to the phone. The response notifies the phone that Gateway 1 has received the BYE request.
15.	Release Complete—Gateway 1 to PBX A	Gateway 1 sends a Release Complete message to PBX A, and the call session terminates.

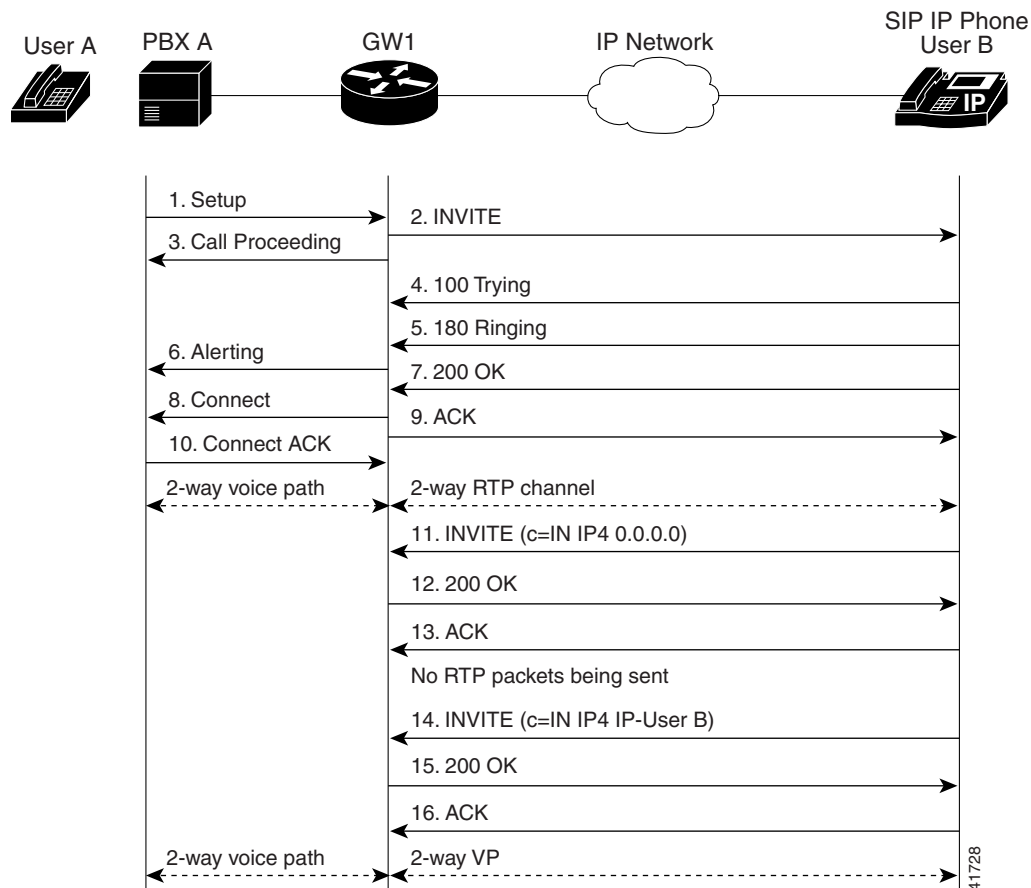
## Call Setup and Hold

Figure B-2 illustrates a successful phone-call setup and call hold. In this scenario, the two end users are User A and User B. User A is located at PBX A. PBX A is connected to Gateway 1 (SIP gateway) via a T1/E1. User B is located at a Cisco SIP IP phone. Gateway 1 is connected to the Cisco SIP IP phone over an IP network.

The call flow is as follows:

1. User A calls User B.
2. User B answers the call.
3. User B puts User A on hold.
4. User B takes User A off hold.

**Figure B-2 Successful Call Setup and Hold**



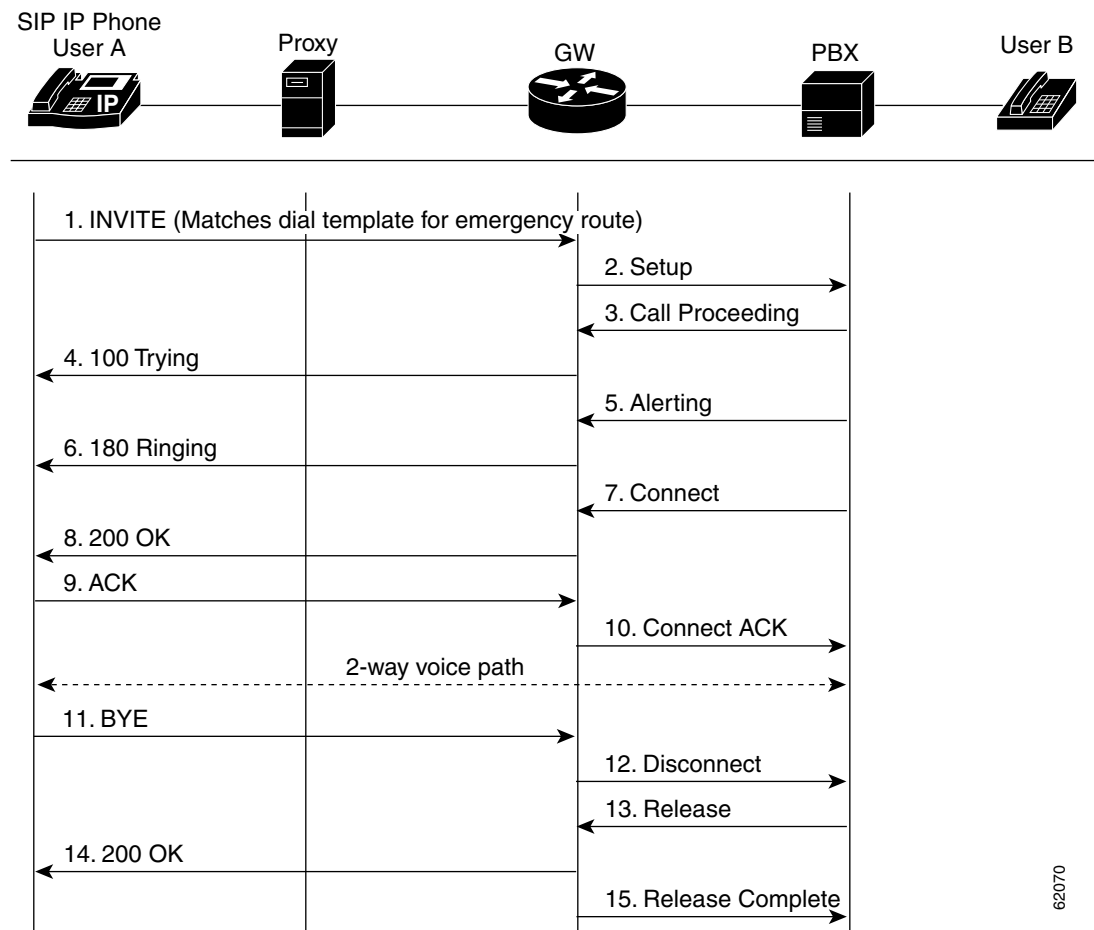
Step	Action	Description
1.	Setup—PBX A to Gateway 1	Call setup is initiated between PBX A and Gateway 1. Setup includes the standard transactions that take place as User A attempts to call User B.
2.	INVITE—Gateway 1 to Cisco SIP IP phone	Gateway 1 maps the SIP URL phone number to a dial peer. The dial peer includes the IP address and the port number of the SIP-enabled entity to contact. Gateway 1 sends a SIP INVITE request to the address it receives as the dial peer, which, in this scenario, is the IP phone. In the INVITE request: <ul style="list-style-type: none"> <li>• The IP address of the phone is inserted in the Request-URI field.</li> <li>• PBX A is identified as the call session initiator in the From field.</li> <li>• A unique numeric identifier is assigned to the call and is inserted in the Call-ID field.</li> <li>• The transaction number within a single call leg is identified in the CSeq field.</li> <li>• The media capability that User A is ready to receive is specified.</li> <li>• The port on which the gateway is prepared to receive the RTP data is specified.</li> </ul>
3.	Call Proceeding—Gateway 1 to PBX A	Gateway 1 sends a Call Proceeding message to the PBX A to acknowledge the Call Setup request.
4.	100 Trying—Cisco IP 7960G/7940G phone to Gateway 1	The phone sends a SIP 100 Trying response to Gateway 1. The response indicates that the INVITE request has been received.
5.	180 Ringing—Cisco SIP IP phone to Gateway 1	The phone sends a SIP 180 Ringing response to Gateway 1. The response indicates that the user is being alerted.
6.	Alerting—Gateway 1 to the PBX A	Gateway 1 sends an Alert message to User A. The message indicates that Gateway 1 has received a 180 Ringing response from the phone. User A hears the ringback tone that indicates that User B is being alerted.
7.	200 OK—Cisco SIP IP phone to Gateway 1	The phone sends a SIP 200 OK response to Gateway 1. The response notifies Gateway 1 that the connection has been made.
8.	Connect—Gateway 1 to PBX A	Gateway 1 sends a Connect message to PBX A. The message notifies PBX A that the connection has been made.
9.	ACK—Gateway 1 to Cisco SIP IP phone	Gateway 1 sends a SIP ACK to the phone. The ACK confirms that User A has received the 200 OK response. The call session is now active.
10.	Connect ACK—PBX A to Gateway 1	PBX A acknowledges Gateway 1's Connect message.
11.	INVITE—Cisco SIP IP phone to Gateway 1	User B puts User A on hold. The phone sends a SIP INVITE request to Gateway 1.
12.	200 OK—Gateway 1 to Cisco SIP IP phone	Gateway 1 sends a SIP 200 OK response to the phone. The response notifies the phone that the INVITE was successfully processed.
13.	ACK—Cisco SIP IP phone to Gateway 1	The phone sends a SIP ACK to Gateway 1. The ACK confirms that the phone has received the 200 OK response. The call session is now temporarily inactive. No RTP packets are being sent.

Step	Action	Description
14.	INVITE—Cisco SIP IP phone to Gateway 1	User B takes User A off hold. The phone sends a SIP INVITE request to Gateway 1.
15.	200 OK—Gateway 1 to Cisco SIP IP phone	Gateway 1 sends a SIP 200 OK response to the phone. The response notifies the phone that the INVITE was successfully processed.
16.	ACK—Cisco SIP IP phone to Gateway 1	The phone sends a SIP ACK to Gateway 1. The ACK confirms that the phone has received the 200 OK response. The call session is now active.

## Call to a Gateway Acting As an Emergency Proxy from a Cisco SIP IP Phone

Figure B-3 illustrates a successful call from a Cisco SIP IP phone to a gateway acting as an emergency proxy.

**Figure B-3 Successful Call from Cisco SIP IP Phone to SIP Gateway (Emergency Proxy)**



62070

Step	Action	Description
1.	INVITE—Cisco SIP IP phone to the gateway (emergency proxy)	The phone tries to connect to the gateway (emergency proxy) by sending out the INVITE message. The dial template for the emergency route is matched.
2.	Setup—Gateway to the PBX	Call setup is initiated between the gateway and the PBX. Setup includes the standard transactions that take place as User A attempts to call User B.
3.	Call Proceeding—PBX to the gateway	The PBX sends a Call Proceeding message to the gateway to acknowledge the Call Setup request.
4.	100 Trying—Gateway to Cisco SIP IP phone (User A)	The gateway sends a SIP 100 Trying response to User A. The response indicates that the INVITE request has been received.
5.	Alerting—PBX to the gateway	The PBX sends an Alert message to the gateway. The message indicates that the PBX has received a 100 Trying Ringing response from the gateway.
6.	180 Ringing—Gateway to Cisco SIP IP phone (User A)	The gateway sends a SIP 180 Ringing response to User A. The response indicates that the gateway is being alerted.
7.	Connect—PBX to the gateway	The PBX sends a Connect message to the gateway. The message notifies the gateway that the connection has been made.
8.	200 OK—Gateway to Cisco SIP IP phone (User A)	The gateway sends a SIP 200 OK response to User A. The response notifies User A that the connection has been made.
9.	ACK—Cisco SIP IP phone (User A) to the gateway	User A sends a SIP ACK to the gateway. The ACK confirms that User A has received the 200 OK response. The call session is now active.
10.	Connect ACK—Gateway to the PBX	The gateway acknowledges PBX's Connect message.
11.	BYE—Cisco SIP IP phone (User A) to the gateway	User A terminates the call session and sends a SIP BYE request to the gateway. The request indicates that User A wants to release the call.
12.	Disconnect—Gateway to the PBX	The gateway sends a Disconnect message to the PBX.
13.	Release—PBX to the gateway	The PBX sends a Release message to the gateway.
14.	200 OK—Gateway to Cisco SIP IP phone (User A)	The gateway sends a SIP 200 OK response to User A. The response notifies User A that the gateway has received the BYE request.
15.	Release Complete—Gateway to the PBX	The gateway sends a Release Complete message to the PBX and the call session terminates.

## Cisco SIP IP Phone to Cisco SIP IP Phone

The following sections describe and illustrate successful calls from a Cisco SIP IP phone to another Cisco SIP IP phone:

- Simple Call Hold, page B-8
- Call Hold with Consultation, page B-10
- Call Waiting, page B-13
- Call Transfer Without Consultation, page B-17
- Call Transfer Without Consultation Using Failover, page B-20
- Call Transfer with Consultation, page B-23
- Call Transfer with Consultation Using Failover, page B-27

- Network Call Forwarding (Unconditional), page B-31
- Network Call Forwarding (Busy), page B-33
- Network Call Forwarding (No Answer), page B-36
- Three-Way Calling, page B-38
- Call from a Cisco SIP IP Phone to a Gateway Acting As a Backup Proxy in a SIP Network, page B-42
- Call from a Cisco SIP IP Phone to a Cisco SIP IP Phone Using a SIP Backup Proxy, page B-44
- Call from a Cisco SIP IP Phone to a Cisco SIP IP Phone Using a SIP Emergency Proxy, page B-45

## Simple Call Hold

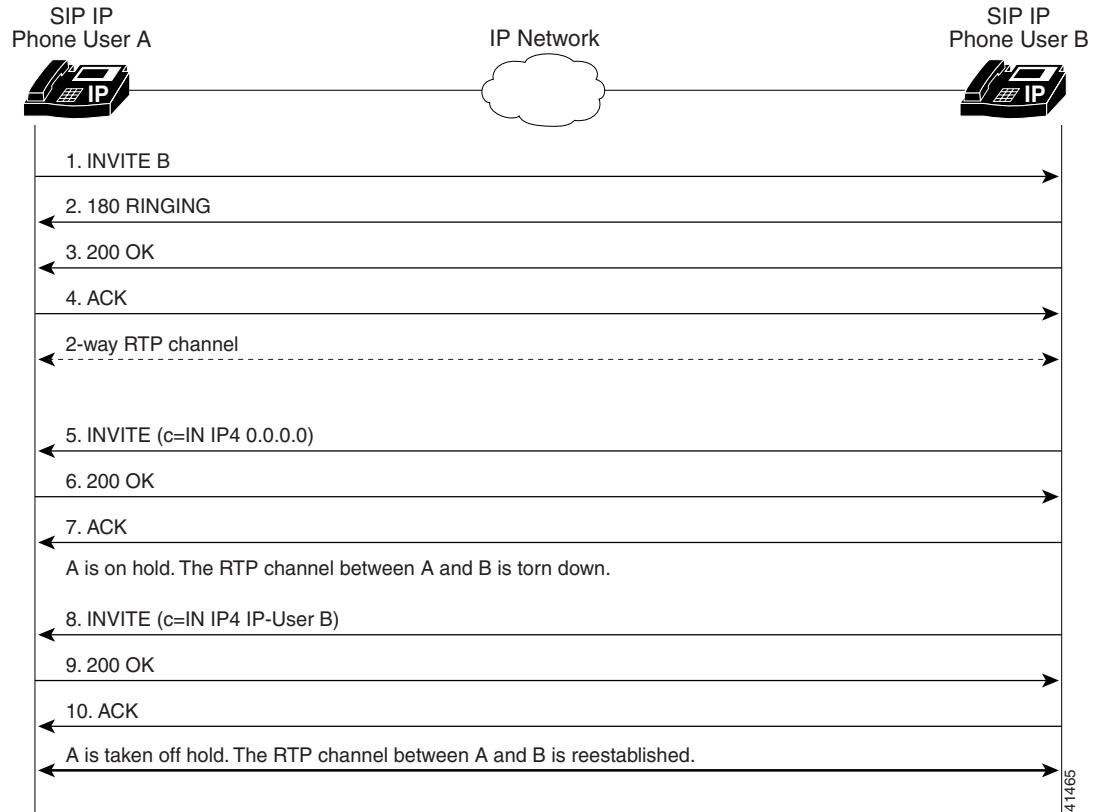
Figure B-4 illustrates a successful call between Cisco SIP IP phones in which one of the participants places the other on hold and then returns to the call. In this call flow scenario, the two end users are User A and User B. User A and User B are both using Cisco SIP IP phones, which are connected via an IP network.

The call flow scenario is as follows:

1. User A calls User B.
2. User B answers the call.
3. User B places User A on hold.
4. User B takes User A off hold.
5. The call continues.



Figure B-4 Simple Call Hold



41465

Step	Action	Description
1.	INVITE—Phone A to Phone B	<p>Phone A sends a SIP INVITE request to Phone B. The request is an invitation to User B to participate in a call session. In the INVITE request:</p> <ul style="list-style-type: none"> <li>The phone number of User B is inserted in the Request-URI field in the form of a SIP URL. The SIP URL identifies the address of User B and takes a form similar to an e-mail address (<i>user@host</i>, where <i>user</i> is the telephone number and <i>host</i> is either a domain name or a numeric network address). For example, the Request-URI field in the INVITE request to User B appears as “INVITE sip:555-0199@companyb.com; user=phone.” The “user=phone” parameter distinguishes that the Request-URI address is a telephone number rather than a username.</li> <li>Phone A is identified as the call session initiator in the From field.</li> <li>A unique numeric identifier is assigned to the call and is inserted in the Call-ID field.</li> <li>The transaction number within a single call leg is identified in the CSeq field.</li> <li>The media capability User A is ready to receive is specified.</li> </ul>
2.	180 Ringing—Phone B to Phone A	Phone B sends a SIP 180 Ringing response to Phone A.

Step	Action	Description
3.	200 OK—Phone B to Phone A	<p>Phone B sends a SIP 200 OK response to Phone A. The response notifies Phone A that the connection has been made.</p> <p>If Phone B supports the media capability advertised in the INVITE message sent by Phone A, it advertises the intersection of its own and Phone A's media capability in the 200 OK response. If Phone B does not support the media capability advertised by Phone A, it sends back a 400 Bad Request response with a 304 Warning header field.</p>
4.	ACK—Phone A to Phone B	<p>Phone A sends a SIP ACK to Phone B. The ACK confirms that Phone A has received the 200 OK response from Phone B.</p> <p>The ACK might contain a message body with the final session description to be used by Phone B. If the message body of the ACK is empty, Phone B uses the session description in the INVITE request.</p>

A two-way RTP channel is established between Phone A and Phone B.

5.	INVITE—Phone B to Phone A	<p>Phone B sends a midcall INVITE to Phone A with new Session Description Protocol (SDP) session parameters (IP address), which are used to place the call on hold.</p> <pre>Call_ID=1 SDP: c=IN IP4 0.0.0.0</pre> <p>The c= SDP field of the SIP INVITE contains an 0.0.0.0. This places the call in hold.</p>
6.	200 OK—Phone A to Phone B	Phone A sends a SIP 200 OK response to Phone B.
7.	ACK—Phone B to Phone A	Phone B sends a SIP ACK to Phone A. The ACK confirms that Phone B has received the 200 OK response from Phone A.

The RTP channel between Phone A and Phone B is torn down.

8.	INVITE—Phone B to Phone A	<p>Phone B sends a midcall INVITE to Phone A with the same call ID as the previous INVITE and new SDP session parameters (IP address), which are used to reestablish the call.</p> <pre>Call_ID=1 SDP: c=IN IP4 10.10.10.0</pre> <p>To reestablish the call between Phone A and Phone B, the IP address of Phone B is inserted into the c= SDP field.</p>
9.	200 OK—Phone A to Phone B	Phone A sends a SIP 200 OK response to Phone B.
10.	ACK—Phone B to Phone A	Phone B sends a SIP ACK to Phone A. The ACK confirms that Phone B has received the 200 OK response from Phone A.

A two-way RTP channel is reestablished between Phone A and Phone B.

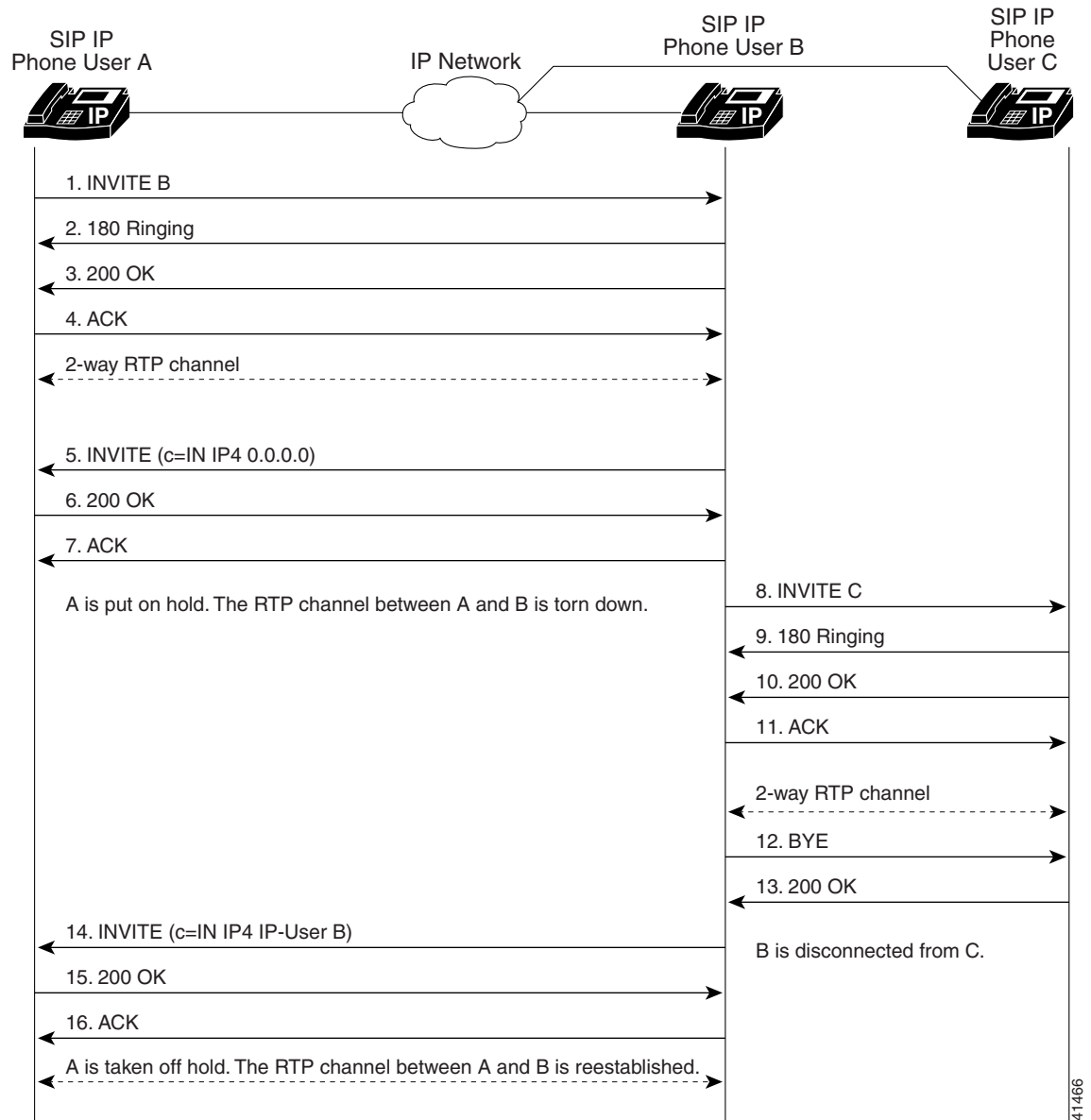
## Call Hold with Consultation

Figure B-5 illustrates a successful call between Cisco SIP IP phones in which one of the participants places the other on hold, calls a third party (consultation), and then returns to the original call. In this call flow scenario, the end users are User A, User B, and User C. They are all using Cisco SIP IP phones, which are connected via an IP network.

The call flow scenario is as follows:

1. User A calls User B.
2. User B answers the call.
3. User B places User A on hold.
4. User B calls User C.
5. User B disconnects from User C.
6. User B takes User A off hold.
7. The original call continues.

**Figure B-5 Call Hold with Consultation**



411466

Step	Action	Description
1.	INVITE—Phone A to Phone B	<p>Phone A sends a SIP INVITE request to Phone B. The request is an invitation to User B to participate in a call session. In the INVITE request:</p> <ul style="list-style-type: none"> <li>The phone number of User B is inserted in the Request-URI field in the form of a SIP URL. The SIP URL identifies the address of User B and takes a form similar to an e-mail address (<i>user@host</i>, where <i>user</i> is the telephone number and <i>host</i> is either a domain name or a numeric network address). For example, the Request-URI field in the INVITE request to User B appears as “INVITE sip:555-0199@companyb.com; user=phone.” The “user=phone” parameter distinguishes that the Request-URI address is a telephone number rather than a username.</li> <li>Phone A is identified as the call session initiator in the From field.</li> <li>A unique numeric identifier is assigned to the call and is inserted in the Call-ID field.</li> <li>The transaction number within a single call leg is identified in the CSeq field.</li> <li>The media capability User A is ready to receive is specified.</li> </ul>
2.	180 Ringing—Phone B to Phone A	Phone B sends a SIP 180 Ringing response to Phone A.
3.	200 OK—Phone B to Phone A	<p>Phone B sends a SIP 200 OK response to Phone A. The response notifies Phone A that the connection has been made.</p> <p>If Phone B supports the media capability advertised in the INVITE message sent by Phone A, it advertises the intersection of its own and Phone A’s media capability in the 200 OK response. If Phone B does not support the media capability advertised by Phone A, it sends back a 400 Bad Request response with a 304 Warning header field.</p>
4.	ACK—Phone A to Phone B	<p>Phone A sends a SIP ACK to Phone B. The ACK confirms that Phone A has received the 200 OK response from Phone B.</p> <p>The ACK might contain a message body with the final session description to be used by Phone B. If the message body of the ACK is empty, Phone B uses the session description in the INVITE request.</p>
A two-way RTP channel is established between Phone A and Phone B.		
5.	INVITE—Phone B to Phone A	<p>Phone B sends a midcall INVITE to Phone A with new SDP session parameters (IP address), which are used to place the call on hold.</p> <pre>Call_ID=1 SDP: c=IN IP4 0.0.0.0</pre> <p>The c= SDP field of the SIP INVITE contains 0.0.0.0. This places the call in hold.</p>
6.	200 OK—Phone A to Phone B	Phone A sends a SIP 200 OK response to Phone B.
7.	ACK—Phone B to Phone A	Phone B sends a SIP ACK to Phone A. The ACK confirms that Phone B has received the 200 OK response from Phone A.
The RTP channel between Phone A and Phone B is torn down.		
8.	INVITE—Phone B to Phone C	Phone B sends a SIP INVITE request to Phone C. The request is an invitation to User C to participate in a call session.

Step	Action	Description
9.	180 Ringing—Phone C to Phone B	Phone C sends a SIP 180 Ringing response to Phone B.
10.	200 OK—Phone C to Phone B	Phone C sends a SIP 200 OK response to Phone B. The response notifies Phone B that the connection has been made.  If Phone B supports the media capability advertised in the INVITE message sent by Phone A, it advertises the intersection of its own and Phone A's media capability in the 200 OK response. If Phone B does not support the media capability advertised by Phone A, it sends back a 400 Bad Request response with a 304 Warning header field.
11.	ACK—Phone B to Phone C	Phone B sends a SIP ACK to Phone C. The ACK confirms that Phone B has received the 200 OK response from Phone C.  The ACK might contain a message body with the final session description to be used by Phone C. If the message body of the ACK is empty, Phone C uses the session description in the INVITE request.
A two-way RTP channel is established between Phone B and Phone C.		
12.	BYE—Phone B to Phone C	The call continues and then User B hangs up. Phone B sends a SIP BYE request to Phone C. The request indicates that User B wants to release the call.
13.	200 OK—Phone C to Phone B	Phone C sends a SIP 200 OK response to Phone B. The response notifies Phone B that the BYE request has been received. The call session between User A and User B terminates.
The RTP channel between Phone B and Phone C is torn down.		
14.	INVITE—Phone B to Phone A	Phone B sends a midcall INVITE to Phone A with the same call ID as the previous INVITE and new SDP session parameters (IP address), which are used to reestablish the call.  <code>Call_ID=1</code> <code>SDP: c=IN IP4 10.10.10.0</code>  To reestablish the call between Phone A and Phone B, the IP address of Phone B is inserted into the c= SDP field.
15.	200 OK—Phone A to Phone B	Phone A sends a SIP 200 OK response to Phone B.
16.	ACK—Phone B to Phone A	Phone B sends a SIP ACK to Phone A. The ACK confirms that Phone B has received the 200 OK response from Phone A.
A two-way RTP channel is reestablished between Phone A and Phone B.		

## Call Waiting

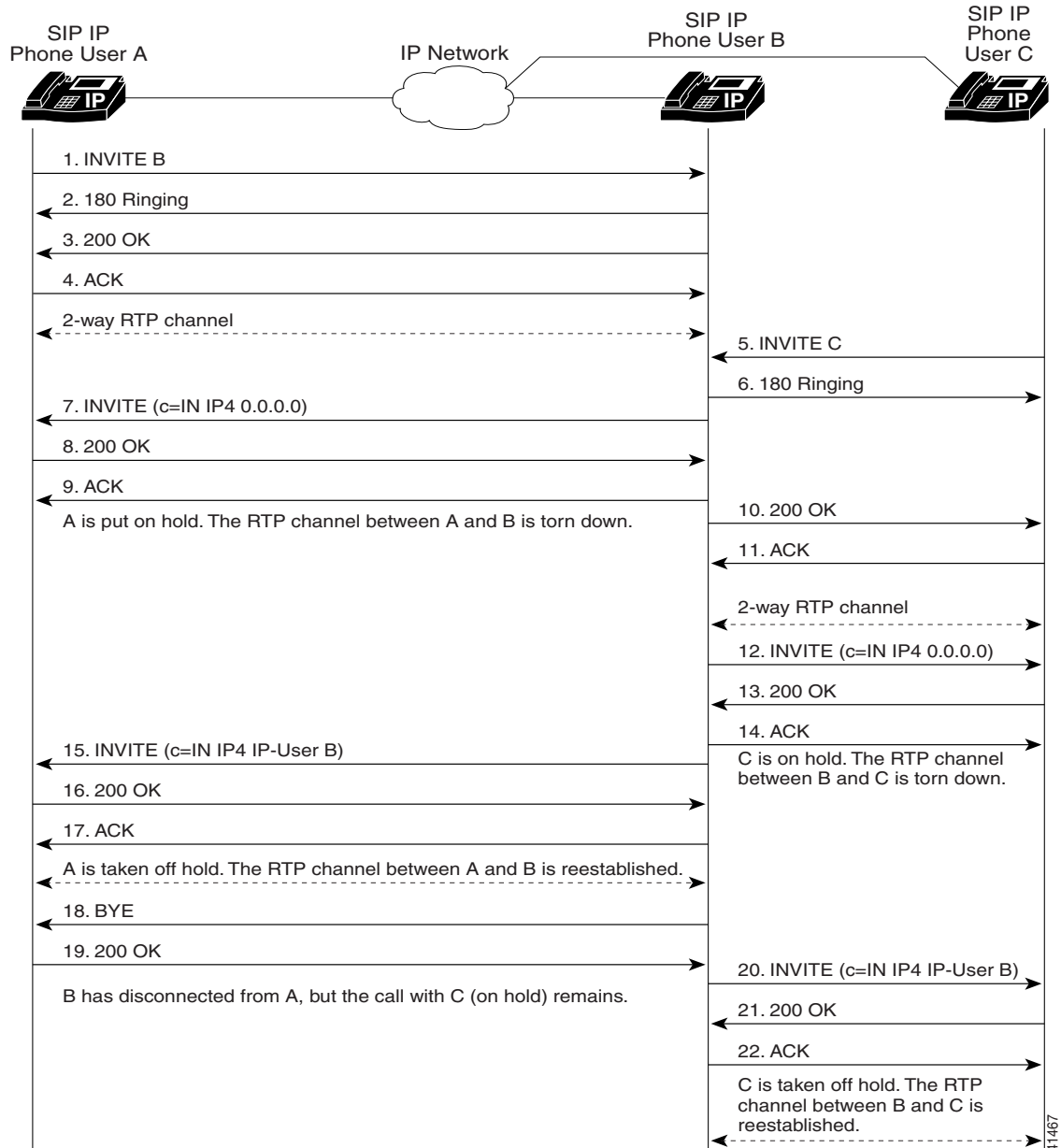
Figure B-6 illustrates a successful call between Cisco SIP IP phones in which two parties are in a call, and one of the participants receives a call from a third party and then returns to the original call. In this call flow scenario, the end users are User A, User B, and User C. They are all using Cisco IP 7960G/7940G phones, which are connected using an IP network.

The call flow scenario is as follows:

1. User A calls User B.
2. User B answers the call.
3. User C calls User B.

4. User B accepts the call from User C.
5. User B switches back to User A.
6. User B hangs up, ending the call with User A.
7. User B is notified of the remaining call with User C.
8. User B answers the notification and continues the call with User C.

Figure B-6 Call Waiting



Step	Action	Description
1.	INVITE—Phone A to Phone B	<p>Phone A sends a SIP INVITE request to Phone B. The request is an invitation to User B to participate in a call session. In the INVITE request:</p> <ul style="list-style-type: none"> <li>The phone number of User B is inserted in the Request-URI field in the form of a SIP URL. The SIP URL identifies the address of User B and takes a form similar to an e-mail address (<i>user@host</i>, where <i>user</i> is the telephone number and <i>host</i> is either a domain name or a numeric network address). For example, the Request-URI field in the INVITE request to User B appears as “INVITE sip:555-0199@companyb.com; user=phone.” The “user=phone” parameter distinguishes that the Request-URI address is a telephone number rather than a username.</li> <li>Phone A is identified as the call session initiator in the From field.</li> <li>A unique numeric identifier is assigned to the call and is inserted in the Call-ID field.</li> <li>The transaction number within a single call leg is identified in the CSeq field.</li> <li>The media capability User A is ready to receive is specified.</li> </ul>
2.	180 Ringing—Phone B to Phone A	Phone B sends a SIP 180 Ringing response to Phone A.
3.	200 OK—Phone B to Phone A	<p>Phone B sends a SIP 200 OK response to Phone A. The response notifies Phone A that the connection has been made.</p> <p>If Phone B supports the media capability advertised in the INVITE message sent by Phone A, it advertises the intersection of its own and Phone A media capability in the 200 OK response. If Phone B does not support the media capability advertised by Phone A, it sends back a 400 Bad Request response with a 304 Warning header field.</p>
4.	ACK—Phone A to Phone B	<p>Phone A sends a SIP ACK to Phone B. The ACK confirms that Phone A has received the 200 OK response from Phone B.</p> <p>The ACK might contain a message body with the final session description to be used by Phone B. If the message body of the ACK is empty, Phone B uses the session description in the INVITE request.</p>
A two-way RTP channel is established between Phone A and Phone B.		
5.	INVITE—Phone C to Phone B	Phone C sends a SIP INVITE request to Phone B. The request is an invitation to User B to participate in a call session.
6.	180 Ringing—Phone B to Phone C	Phone B sends a SIP 180 Ringing response to Phone C.
7.	INVITE—Phone B to Phone A	<p>Phone B sends a midcall INVITE to Phone A with new SDP session parameters (IP address), which are used to place the call on hold.</p> <pre>Call_ID=1 SDP: c=IN IP4 0.0.0.0</pre> <p>The c= SDP field of the SIP INVITE contains 0.0.0.0. This places the call in hold.</p>
8.	200 OK—Phone A to Phone B	Phone A sends a SIP 200 OK response to Phone B.
9.	ACK—Phone B to Phone A	Phone B sends a SIP ACK to Phone A. The ACK confirms that Phone B has received the 200 OK response from Phone A.

## Call Flow Scenarios for Successful Calls

Step	Action	Description
The RTP channel between Phone A and Phone B is torn down.		
10.	200 OK—Phone B to Phone C	Phone B sends a SIP 200 OK response to Phone C. The response notifies Phone C that the connection has been made.
11.	ACK—Phone C to Phone B	Phone C sends a SIP ACK to Phone B. The ACK confirms that Phone C has received the 200 OK response from Phone B.  The ACK might contain a message body with the final session description to be used by Phone B. If the message body of the ACK is empty, Phone B uses the session description in the INVITE request.
A two-way RTP channel is established between Phone B and Phone C.		
12.	INVITE—Phone B to Phone C	Phone B sends a midcall INVITE to Phone C with new SDP session parameters (IP address), which are used to place the call on hold.  <code>Call_ID=2</code> <code>SDP: c=IN IP4 0.0.0.0</code>  To establish the call between Phone B and Phone C, the IP address of Phone B is inserted into the c= SDP field.
13.	200 OK—Phone C to Phone B	Phone C sends a SIP 200 OK response to Phone B.
14.	ACK—Phone B to Phone C	Phone B sends a SIP ACK to Phone C. The ACK confirms that Phone B has received the 200 OK response from Phone C.
The RTP channel between Phone B and Phone C is torn down.		
15.	INVITE—Phone B to Phone A	Phone B sends a midcall INVITE to Phone A with the same call ID as the previous INVITE (sent to Phone A) and new SDP session parameters (IP address), which are used to reestablish the call.  <code>Call_ID=1</code> <code>SDP: c=IN IP4 10.10.10.0</code>
16.	200 OK—Phone A to Phone B	Phone A sends a SIP 200 OK response to Phone B.
17.	ACK—Phone B to Phone A	Phone B sends a SIP ACK to Phone A. The ACK confirms that Phone B has received the 200 OK response from Phone A.
A two-way RTP channel is reestablished between Phone A and Phone B.		
18.	BYE—Phone B to Phone A	The call continues and then User B hangs up. Phone B sends a SIP BYE request to Phone A. The request indicates that User B wants to release the call.
19.	200 OK—Phone A to Phone B	Phone A sends a SIP 200 OK response to Phone B. The response notifies Phone B that the BYE request has been received. The call session between User A and User B terminates.
The RTP channel between Phone A and Phone B is torn down.		
20.	INVITE—Phone B to Phone C	Phone B sends a midcall INVITE to Phone C with the same call ID as the previous INVITE (sent to Phone C) and new SDP session parameters (IP address), which are used to reestablish the call.  <code>Call_ID=2</code> <code>SDP: c=IN IP4 10.10.10.0</code>
21.	200 OK—Phone C to Phone B	Phone C sends a SIP 200 OK response to Phone B.



Step	Action	Description
22.	ACK—Phone B to Phone C	Phone B sends a SIP ACK to Phone C. The ACK confirms that Phone B has received the 200 OK response from Phone A.

A two-way RTP channel is reestablished between Phone B and Phone C.

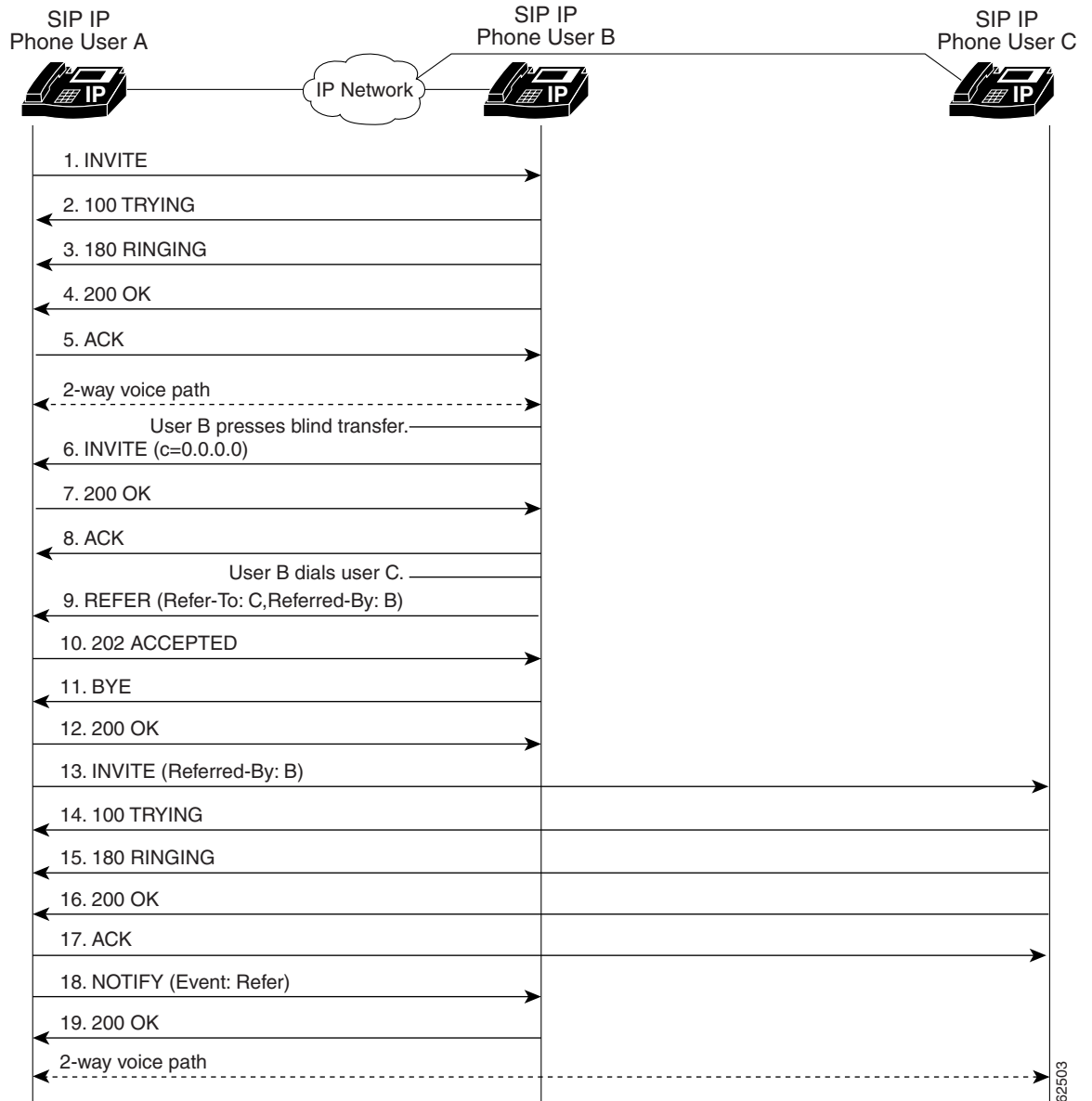
## Call Transfer Without Consultation

Figure B-7 illustrates a successful call between Cisco SIP IP phones in which two parties are in a call and then one of the participants transfers the call to a third party without first contacting the third party. This is called a blind or unattended transfer. In this call flow scenario, the end users are User A, User B, and User C. They are all using Cisco SIP IP phones, which are connected via an IP network.

The call flow scenario is as follows:

1. User A calls User B.
2. User B answers the call.
3. User B transfers the call to User C.

Figure B-7 Call Transfer Without Consultation



62503

Step	Action	Description
1.	INVITE—Phone A to Phone B	<p>Phone A sends a SIP INVITE request to Phone B. The request is an invitation to User B to participate in a call session. In the INVITE request:</p> <ul style="list-style-type: none"> <li>The phone number of User B is inserted in the Request-URI field in the form of a SIP URL. The SIP URL identifies the address of User B and takes a form similar to an e-mail address (<i>user@host</i>, where <i>user</i> is the telephone number and <i>host</i> is either a domain name or a numeric network address). For example, the Request-URI field in the INVITE request to User B appears as “INVITE sip:555-0199@companyb.com; user=phone.” The “user=phone” parameter distinguishes that the Request-URI address is a telephone number rather than a username.</li> <li>Phone A is identified as the call session initiator in the From field.</li> <li>A unique numeric identifier is assigned to the call and is inserted in the Call-ID field.</li> <li>The transaction number within a single call leg is identified in the CSeq field.</li> <li>The media capability that User A is ready to receive is specified.</li> </ul>
2.	100 Trying—Phone B to Phone A	Phone B sends a SIP 100 Trying response to Phone A. The response indicates that the INVITE request has been received.
3.	180 Ringing—Phone B to Phone A	Phone B sends a SIP 180 Ringing response to Phone A.
4.	200 OK—Phone B to Phone A	<p>Phone B sends a SIP 200 OK response to Phone A. The response notifies Phone A that the connection has been made.</p> <p>If Phone B supports the media capability advertised in the INVITE message sent by Phone A, it advertises the intersection of its own and Phone A’s media capability in the 200 OK response. If Phone B does not support the media capability advertised by Phone A, it sends back a 400 Bad Request response with a 304 Warning header field.</p>
5.	ACK—Phone A to Phone B	<p>Phone A sends a SIP ACK to Phone B. The ACK confirms that Phone A has received the 200 OK response from Phone B.</p> <p>The ACK might contain a message body with the final session description to be used by Phone B. If the message body of the ACK is empty, Phone B uses the session description in the INVITE request.</p>
<p>A two-way RTP channel is established between Phone A and Phone B. User B then selects the option to blind transfer the call to User C.</p>		
6.	INVITE—Phone B to Phone A	<p>Phone B sends a midcall INVITE to Phone A with new SDP session parameters (IP address), which are used to place the call on hold.</p> <pre>Call_ID=1 SDP: c=IN IP4 0.0.0.0</pre> <p>The c= SDP field of the SIP INVITE contains 0.0.0.0. This places the call on hold.</p>
7.	200 OK—Phone A to Phone B	Phone A sends a SIP 200 OK response to Phone B.
8.	ACK—Phone B to Phone A	Phone B sends a SIP ACK to Phone A. The ACK confirms that Phone B has received the 200 OK response from Phone A.

User B dials User C.

Step	Action	Description
9.	REFER—Phone B to Phone A	Phone B sends a REFER message to Phone A. The message contains the following information: <ul style="list-style-type: none"> <li>Refer-To: C</li> <li>Referred-By: B</li> </ul> The message indicates that Phone A should send an INVITE request to Phone C.
10.	202 ACCEPTED—Phone A to Phone B	Phone A sends a SIP 202 ACCEPTED message to Phone B. The message confirms that the REFER message has been received.
11.	BYE—Phone B to Phone A	Phone B sends a BYE message to Phone A. The message indicates that Phone B will disconnect from the call.
12.	200 OK—Phone A to Phone B	Phone A sends a SIP 200 OK response to Phone B. The response notifies Phone B that the BYE message was received.
13.	INVITE—Phone A to Phone C	Because of the REFER message from Phone B, Phone A sends a SIP INVITE request to Phone C. The request is an invitation to User C to participate in a call session. The request contains the following information: <ul style="list-style-type: none"> <li>Referred-By: B</li> </ul> The message indicates that the INVITE was referred by Phone B.
14.	100 Trying—Phone C to Phone A	Phone C sends a SIP 100 Trying response to Phone A. The response indicates that the INVITE request has been received.
15.	180 Ringing—Phone C to Phone A	Phone C sends a SIP 180 Ringing response to Phone A.
16.	200 OK—Phone C to Phone A	Phone C sends a SIP 200 OK response to Phone A. The response notifies Phone A that the connection has been made.
17.	ACK—Phone A to Phone C	Phone A sends a SIP ACK to Phone C. The ACK confirms that Phone A has received the 200 OK response from Phone C.
18.	NOTIFY—Phone A to Phone B	Phone A sends a NOTIFY message to Phone B. The message notifies Phone C of the REFER event.
19.	200 OK—Phone B to Phone A	Phone B sends a SIP 200 OK response to Phone A. The response notifies Phone A that the NOTIFY message was received.

A two-way RTP channel is established between Phone A and Phone C.

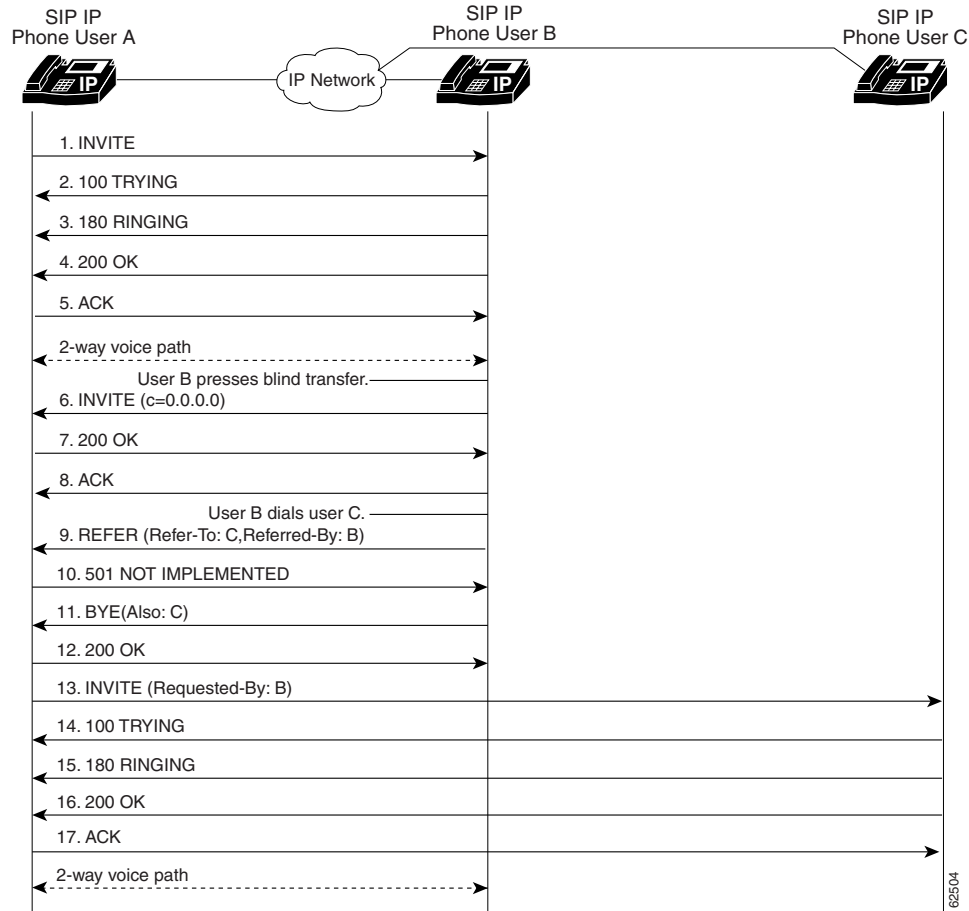
## Call Transfer Without Consultation Using Failover

Figure B-8 illustrates a successful call between Cisco SIP IP phones in which two parties are in a call and then one of the participants transfers the call to a third party without first contacting the third party. This is called a blind or unattended transfer. In this call flow scenario, the end users are User A, User B, and User C. They are all using Cisco SIP IP phones, which are connected using an IP network.

The call flow scenario is as follows:

1. User A calls User B.
2. User B answers the call.
3. User B transfers the call to User C.

**Figure B-8 Call Transfer Without Consultation Using Failover**



Step	Action	Description
1.	INVITE—Phone A to Phone B	<p>Phone A sends a SIP INVITE request to Phone B. The request is an invitation to User B to participate in a call session. In the INVITE request:</p> <ul style="list-style-type: none"> <li>• The phone number of User B is inserted in the Request-URI field in the form of a SIP URL. The SIP URL identifies the address of User B and takes a form similar to an e-mail address (<i>user@host</i>, where <i>user</i> is the telephone number and <i>host</i> is either a domain name or a numeric network address). For example, the Request-URI field in the INVITE request to User B appears as “INVITE sip:555-0199@companyb.com; user=phone.” The “user=phone” parameter distinguishes that the Request-URI address is a telephone number rather than a username.</li> <li>• Phone A is identified as the call session initiator in the From field.</li> <li>• A unique numeric identifier is assigned to the call and is inserted in the Call-ID field.</li> <li>• The transaction number within a single call leg is identified in the CSeq field.</li> <li>• The media capability User A is ready to receive is specified.</li> </ul>
2.	100 Trying—Phone B to Phone A	<p>Phone B sends a SIP 100 Trying response to Phone A. The response indicates that the INVITE request has been received.</p>

## Call Flow Scenarios for Successful Calls

Step	Action	Description
3.	180 Ringing—Phone B to Phone A	Phone B sends a SIP 180 Ringing response to Phone A.
4.	200 OK—Phone B to Phone A	Phone B sends a SIP 200 OK response to Phone A. The response notifies Phone A that the connection has been made.  If Phone B supports the media capability advertised in the INVITE message sent by Phone A, it advertises the intersection of its own and Phone A's media capability in the 200 OK response. If Phone B does not support the media capability advertised by Phone A, it sends back a 400 Bad Request response with a 304 Warning header field.
5.	ACK—Phone A to Phone B	Phone A sends a SIP ACK to Phone B. The ACK confirms that Phone A has received the 200 OK response from Phone B.  The ACK might contain a message body with the final session description to be used by Phone B. If the message body of the ACK is empty, Phone B uses the session description in the INVITE request.

A two-way RTP channel is established between Phone A and Phone B.

User B then selects the option to blind transfer the call to User C.

6.	INVITE—Phone B to Phone A	Phone B sends a midcall INVITE to Phone A with new SDP session parameters (IP address), which are used to place the call on hold.  Call_ID=1 SDP: c=IN IP4 0.0.0.0  The c= SDP field of the SIP INVITE contains an 0.0.0.0. This places the call in hold.
7.	200 OK—Phone A to Phone B	Phone A sends a SIP 200 OK response to Phone B.
8.	ACK—Phone B to Phone A	Phone B sends a SIP ACK to Phone A. The ACK confirms that Phone B has received the 200 OK response from Phone A.

User B dials User C.

9.	REFER—Phone B to Phone A	Phone B sends a REFER message to Phone A. The message contains the following information: <ul style="list-style-type: none"> <li>Refer-To: C</li> <li>Referred-By: B</li> </ul> The REFER message indicates that Phone A should send an INVITE request to Phone C.
10.	501 Not Implemented—Cisco SIP IP Phone A to Phone B	Phone A sends a 501 Not Implemented message to Phone B. The message indicates that the REFER message is not supported and that Phone B should failover to Bye/Also.
11.	BYE—Phone B to Phone A	Phone B sends a BYE message to Phone A. The message includes the following information: <ul style="list-style-type: none"> <li>Also: C</li> </ul> The message indicates that the 501 Not Implemented message was received in response to a REFER message.
12.	200 OK—Phone A to Phone B	Phone A sends a SIP 200 OK response to Phone B. The response notifies Phone B that the BYE message was received.

Step	Action	Description
13.	INVITE—Phone A to Phone C	Phone A sends a SIP INVITE request to Phone C. The request is an invitation to User C to participate in a call session. The request contains the following information: <ul style="list-style-type: none"> <li>Requested-By: B</li> </ul> The message indicates that the INVITE was requested by Phone B.
14.	100 Trying—Phone C to Phone A	Phone C sends a SIP 100 Trying response to Phone A. The response indicates that the INVITE request has been received.
15.	180 Ringing—Phone C to Phone A	Phone C sends a SIP 180 Ringing response to Phone A.
16.	200 OK—Phone C to Phone A	Phone C sends a SIP 200 OK response to Phone A. The response notifies Phone A that the connection has been made.
17.	ACK—Phone A to Phone C	Phone A sends a SIP ACK to Phone C. The ACK confirms that Phone A has received the 200 OK response from Phone C.

A two-way RTP channel is established between Phone A and Phone C.

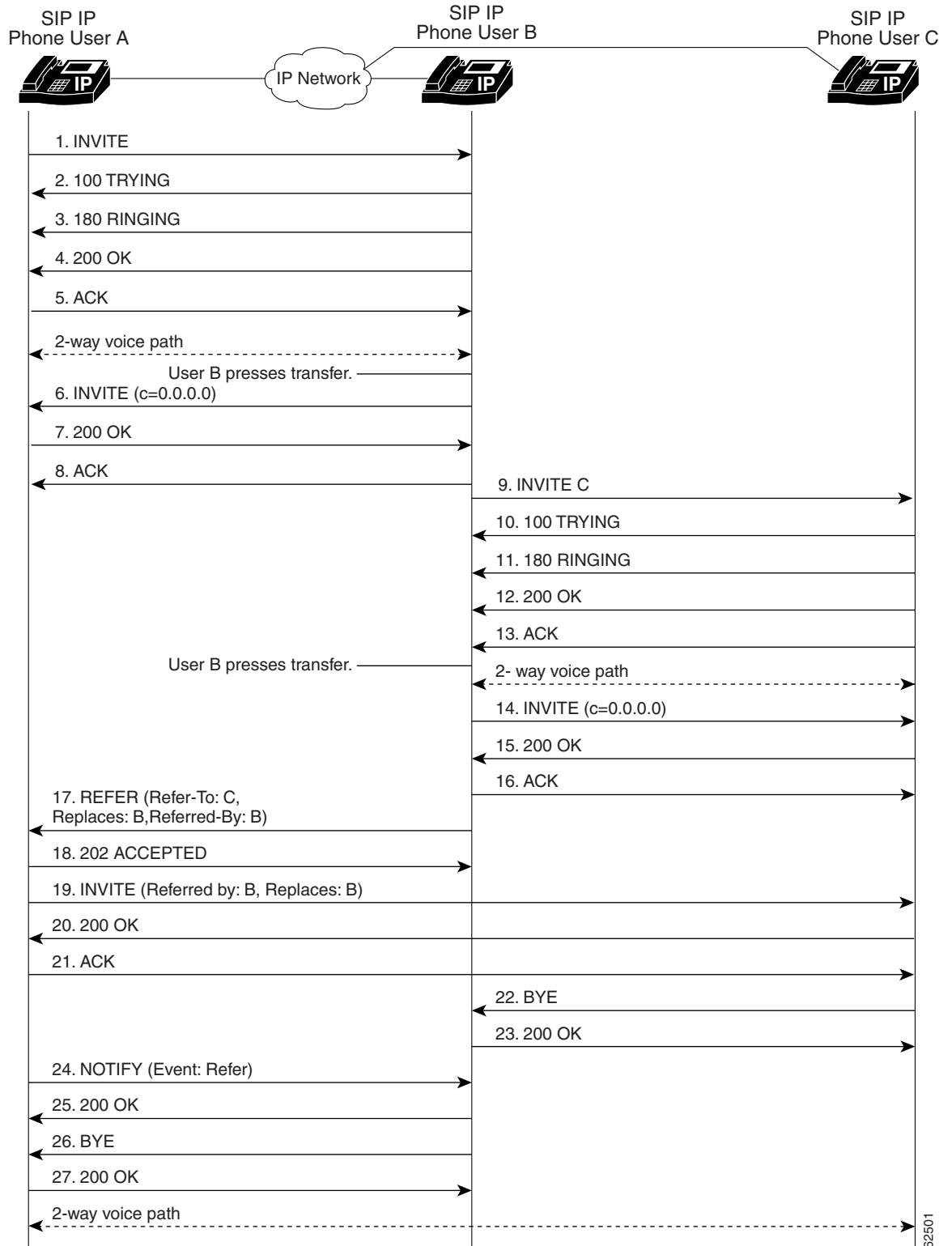
## Call Transfer with Consultation

Figure B-9 illustrates a successful call between Cisco SIP IP phones in which two parties are in a call, one of the participants contacts a third party, and then that participant transfers the call to the third party. This is called an attended transfer. In this call flow scenario, the end users are User A, User B, and User C. They are all using Cisco SIP IP phones, which are connected via an IP network.

The call flow scenario is as follows:

1. User A calls User B.
2. User B answers the call.
3. User B calls User C, and User C consents to take the call.
4. User B transfers the call to User C.
5. User B disconnects with User C.
6. User C and User A connect to each other.

Figure B-9 Call Transfer with Consultation



62501



Step	Action	Description
1.	INVITE—Phone A to Phone B	<p>Phone A sends a SIP INVITE request to Phone B. The request is an invitation to User B to participate in a call session. In the INVITE request:</p> <ul style="list-style-type: none"> <li>The phone number of User B is inserted in the Request-URI field in the form of a SIP URL. The SIP URL identifies the address of User B and takes a form similar to an e-mail address (<i>user@host</i>, where <i>user</i> is the telephone number and <i>host</i> is either a domain name or a numeric network address). For example, the Request-URI field in the INVITE request to User B appears as “INVITE sip:555-0199@companyb.com; user=phone.” The “user=phone” parameter distinguishes that the Request-URI address is a telephone number rather than a username.</li> <li>Phone A is identified as the call session initiator in the From field.</li> <li>A unique numeric identifier is assigned to the call and is inserted in the Call-ID field.</li> <li>The transaction number within a single call leg is identified in the CSeq field.</li> <li>The media capability User A is ready to receive is specified.</li> </ul>
2.	100 Trying—Phone B to Phone A	Phone B sends a SIP 100 Trying response to Phone A. The response indicates that the INVITE request has been received.
3.	180 Ringing—Phone B to Phone A	Phone B sends a SIP 180 Ringing response to Phone A.
4.	200 OK—Phone B to Phone A	<p>Phone B sends a SIP 200 OK response to Phone A. The response notifies Phone A that the connection has been made.</p> <p>If Phone B supports the media capability advertised in the INVITE message sent by Phone A, it advertises the intersection of its own and Phone A’s media capability in the 200 OK response. If Phone B does not support the media capability advertised by Phone A, it sends back a 400 Bad Request response with a 304 Warning header field.</p>
5.	ACK—Phone A to Phone B	<p>Phone A sends a SIP ACK to Phone B. The ACK confirms that Phone A has received the 200 OK response from Phone B.</p> <p>The ACK might contain a message body with the final session description to be used by Phone B. If the message body of the ACK is empty, Phone B uses the session description in the INVITE request.</p>
<p>A two-way RTP channel is established between Phone A and Phone B. User B then selects the option to transfer the call to User C.</p>		
6.	INVITE—Phone B to Phone A	<p>Phone B sends a midcall INVITE to Phone A with new SDP session parameters (IP address), which are used to place the call on hold.</p> <pre>Call_ID=1 SDP: c=IN IP4 0.0.0.0</pre>
7.	200 OK—Phone A to Phone B	Phone A sends a SIP 200 OK response to Phone B.
8.	ACK—Phone B to Phone A	Phone B sends a SIP ACK to Phone A. The ACK confirms that Phone B has received the 200 OK response from Phone A.

User B dials User C.

## Call Flow Scenarios for Successful Calls

Step	Action	Description
9.	INVITE—Phone B to Phone C	Phone B sends a SIP INVITE request to Phone C. The request is an invitation to User C to participate in a call session.
10.	100 Trying—Phone C to Phone B	Phone C sends a SIP 100 Trying response to Phone B. The response indicates that the INVITE request has been received.
11.	180 Ringing—Phone C to Phone B	Phone C sends a SIP 180 Ringing response to Phone B.
12.	200 OK—Phone C to Phone B	Phone C sends a SIP 200 OK response to Phone B. The response notifies Phone B that the connection has been made.  If Phone B supports the media capability advertised in the INVITE message sent by Phone A, it advertises the intersection of its own and Phone A's media capability in the 200 OK response. If Phone B does not support the media capability advertised by Phone A, it sends back a 400 Bad Request response with a 304 Warning header field.
13.	ACK—Phone B to Phone C	Phone B sends a SIP ACK to Phone C. The ACK confirms that Phone B has received the 200 OK response from Phone C.  The ACK might contain a message body with the final session description to be used by Phone C. If the message body of the ACK is empty, Phone C uses the session description in the INVITE request.
A two-way RTP channel is established between Phone B and Phone C. User B then selects the option to transfer the call to User C.		
14.	INVITE—Phone B to Phone A	Phone B sends a midcall INVITE to Phone A with new SDP session parameters (IP address), which are used to place the call on hold.  <code>Call_ID=1</code> <code>SDP: c=IN IP4 0.0.0.0</code>  The c= SDP field of the SIP INVITE contains an 0.0.0.0. This places the call in hold.
15.	200 OK—Phone A to Phone B	Phone A sends a SIP 200 OK response to Phone B.
16.	ACK—Phone B to Phone C	Phone B sends a SIP ACK to Phone C. The ACK confirms that Phone B has received the 200 OK response from Phone C.
17.	REFER—Phone B to Phone A	Phone B sends a REFER message to Phone A. The message contains the following information: <ul style="list-style-type: none"> <li>• Refer-To: C</li> <li>• Replaces: B</li> <li>• Referred-By: B</li> </ul> The message indicates that the user (recipient) should contact a third party for use in transferring parties.
18.	202 ACCEPTED—Phone A to Phone B	Phone A sends a SIP 202 ACCEPTED message to Phone B. The confirms that the REFER message has been received.
19.	INVITE—Phone A to Phone C	Phone A sends a SIP INVITE request to Phone C. The request contains the following information: <ul style="list-style-type: none"> <li>• Referred-By: B</li> <li>• Replaces: B</li> </ul>

Step	Action	Description
20.	200 OK—Phone C to Phone A	Phone C sends a SIP 200 OK response to Phone A. The response notifies Phone A that the INVITE request has been received.
21.	ACK—Phone A to Phone C	Phone A sends a SIP ACK to Phone C. The ACK confirms that Phone A has received the 200 OK response from Phone C.
22.	BYE—Phone C to Phone B	Phone C sends a SIP BYE request to Phone B.
23.	200 OK—Phone B to Phone C	Phone B sends a SIP 200 OK response to Phone C. The response notifies Phone C that the BYE request has been received.
24.	NOTIFY—Phone A to Phone B	Phone A sends a NOTIFY message to Phone B. The message notifies Phone B of the REFER event.
25.	200 OK—Phone B to Phone A	Phone B sends a SIP 200 OK response to Phone A. The response notifies Phone A that the NOTIFY request has been received.
26.	BYE—Phone B to Phone A	Phone B sends a SIP BYE request to Phone A.
27.	200 OK—Phone A to Phone B	Phone A sends a SIP 200 OK response to Phone B. The response notifies Phone A that the BYE request has been received.

A two-way RTP channel is established between Phone A and Phone C.

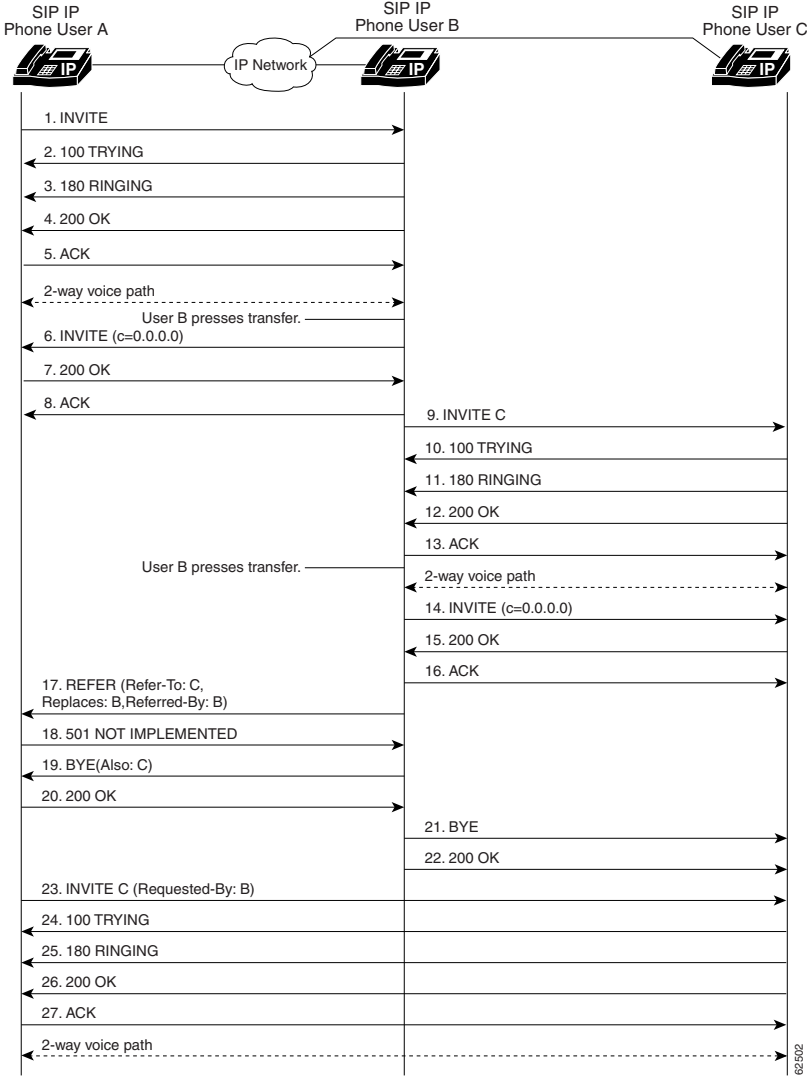
## Call Transfer with Consultation Using Failover

Figure B-10 illustrates a successful call between Cisco SIP IP phones in which two parties are in a call, one of the participants contacts a third party, and then that participant transfers the call to the third party. This is called an attended transfer. In this call flow scenario, the end users are User A, User B, and User C. They are all using Cisco SIP IP phones, which are connected via an IP network.

The call flow scenario is as follows:

1. User A calls User B.
2. User B answers the call.
3. User B calls User C, and User C consents to take the call.
4. User B transfers the call to User C.
5. User B disconnects with User C.
6. User C and User A connect to each other.

Figure B-10 Call Transfer with Consultation Using Failover



Step	Action	Description
1.	INVITE—Phone A to Phone B	<p>Phone A sends a SIP INVITE request to Phone B. The request is an invitation to User B to participate in a call session. In the INVITE request:</p> <ul style="list-style-type: none"> <li>The phone number of User B is inserted in the Request-URI field in the form of a SIP URL. The SIP URL identifies the address of User B and takes a form similar to an e-mail address (<i>user@host</i>, where <i>user</i> is the telephone number and <i>host</i> is either a domain name or a numeric network address). For example, the Request-URI field in the INVITE request to User B appears as “INVITE sip:555-0199@companyb.com; user=phone.” The “user=phone” parameter distinguishes that the Request-URI address is a telephone number rather than a username.</li> <li>Phone A is identified as the call session initiator in the From field.</li> <li>A unique numeric identifier is assigned to the call and is inserted in the Call-ID field.</li> <li>The transaction number within a single call leg is identified in the CSeq field.</li> <li>The media capability User A is ready to receive is specified.</li> </ul>
2.	100 Trying—Phone B to Phone A	Phone B sends a SIP 100 Trying response to Phone A. The response indicates that the INVITE request has been received.
3.	180 Ringing—Phone B to Phone A	Phone B sends a SIP 180 Ringing response to Phone A.
4.	200 OK—Phone B to Phone A	<p>Phone B sends a SIP 200 OK response to Phone A. The response notifies Phone A that the connection has been made.</p> <p>If Phone B supports the media capability advertised in the INVITE message sent by Phone A, it advertises the intersection of its own and Phone A’s media capability in the 200 OK response. If Phone B does not support the media capability advertised by Phone A, it sends back a 400 Bad Request response with a 304 Warning header field.</p>
5.	ACK—Phone A to Phone B	<p>Phone A sends a SIP ACK to Phone B. The ACK confirms that Phone A has received the 200 OK response from Phone B.</p> <p>The ACK might contain a message body with the final session description to be used by Phone B. If the message body of the ACK is empty, Phone B uses the session description in the INVITE request.</p>
<p>A two-way RTP channel is established between Phone A and Phone B. User B then selects the option to transfer the call to User C.</p>		
6.	INVITE—Phone B to Phone A	<p>Phone B sends a midcall INVITE to Phone A with new SDP session parameters (IP address), which are used to place the call on hold.</p> <pre>Call_ID=1 SDP: c=IN IP4 0.0.0.0</pre>
7.	200 OK—Phone A to Phone B	Phone A sends a SIP 200 OK response to Phone B.
8.	ACK—Phone B to Phone A	Phone B sends a SIP ACK to Phone A. The ACK confirms that Phone B has received the 200 OK response from Phone A.

User B dials User C.

Step	Action	Description
9.	INVITE—Phone B to Phone C	Phone B sends a SIP INVITE request to Phone C. The request is an invitation to User C to participate in a call session.
10.	100 Trying—Phone C to Phone B	Phone C sends a SIP 100 Trying response to Phone B. The response indicates that the INVITE request has been received.
11.	180 Ringing—Phone C to Phone B	Phone C sends a SIP 180 Ringing response to Phone B.
12.	200 OK—Phone C to Phone B	Phone C sends a SIP 200 OK response to Phone B. The response notifies Phone B that the connection has been made.  If Phone B supports the media capability advertised in the INVITE message sent by Phone A, it advertises the intersection of its own and Phone A's media capability in the 200 OK response. If Phone B does not support the media capability advertised by Phone A, it sends back a 400 Bad Request response with a 304 Warning header field.
13.	ACK—Phone B to Phone C	Phone B sends a SIP ACK to Phone C. The ACK confirms that Phone B has received the 200 OK response from Phone C.  The ACK might contain a message body with the final session description to be used by Phone C. If the message body of the ACK is empty, Phone C uses the session description in the INVITE request.
A two-way RTP channel is established between Phone B and Phone C. User B then selects the option to transfer the call to User C.		
14.	INVITE—Phone B to Phone A	Phone B sends a midcall INVITE to Phone A with new SDP session parameters (IP address), which are used to place the call on hold.  <code>Call_ID=1</code> <code>SDP: c=IN IP4 0.0.0.0</code>  The c= SDP field of the SIP INVITE contains an 0.0.0.0. This places the call in hold.
15.	200 OK—Phone A to Phone B	Phone A sends a SIP 200 OK response to Phone B.
16.	ACK—Phone B to Phone C	Phone B sends a SIP ACK to Phone C. The ACK confirms that Phone B has received the 200 OK response from Phone C.
17.	REFER—Phone B to Phone A	Phone B sends a REFER message to Phone A. The message contains the following information: <ul style="list-style-type: none"> <li>• Refer-To: C</li> <li>• Replaces: B</li> <li>• Referred-By: B</li> </ul> The message indicates that the user (recipient) should contact a third party for use in transferring parties.
18.	501 Not Implemented—Cisco SIP IP Phone A to Cisco SIP IP Phone B	Phone A sends a 501 Not Implemented message to Phone B. The message indicates that the REFER message is not supported and that Phone B should failover to Bye/Also.
19.	BYE—Phone B to Phone A	Phone B sends a BYE message to Phone A. The message includes the following information: <ul style="list-style-type: none"> <li>• Also: C</li> </ul> The message indicates that the 501 Not Implemented message was received in response to a REFER message.

Step	Action	Description
20.	200 OK—Phone A to Phone B	Phone A sends a SIP 200 OK response to Phone B. The response notifies Phone B that the BYE request has been received.
21.	BYE—Phone B to Phone C	Phone B sends a SIP BYE request to Phone C.
22.	200 OK—Phone C to Phone B	Phone C sends a SIP 200 OK response to Phone B. The response notifies Phone B that the BYE request has been received.
23.	INVITE—Phone A to Phone C	Phone A sends a SIP INVITE request to Phone C. The request contains the following information: <ul style="list-style-type: none"> <li>Requested-By: B</li> </ul> The message indicates that the INVITE was requested by Phone B.
24.	100 Trying—Phone C to Phone A	Phone C sends a SIP 100 Trying response to Phone A. The response indicates that the INVITE request has been received.
25.	180 Ringing—Phone C to Phone A	Phone C sends a SIP 180 Ringing response to Phone A.
26.	200 OK—Phone C to Phone A	Phone C sends a SIP 200 OK response to Phone A. The response notifies Phone A that the connection has been made.
27.	ACK—Phone A to Phone C	Phone A sends a SIP ACK to Phone C. The ACK confirms that Phone A has received the 200 OK response from Phone C.

A two-way RTP channel is established between Phone A and Phone C.

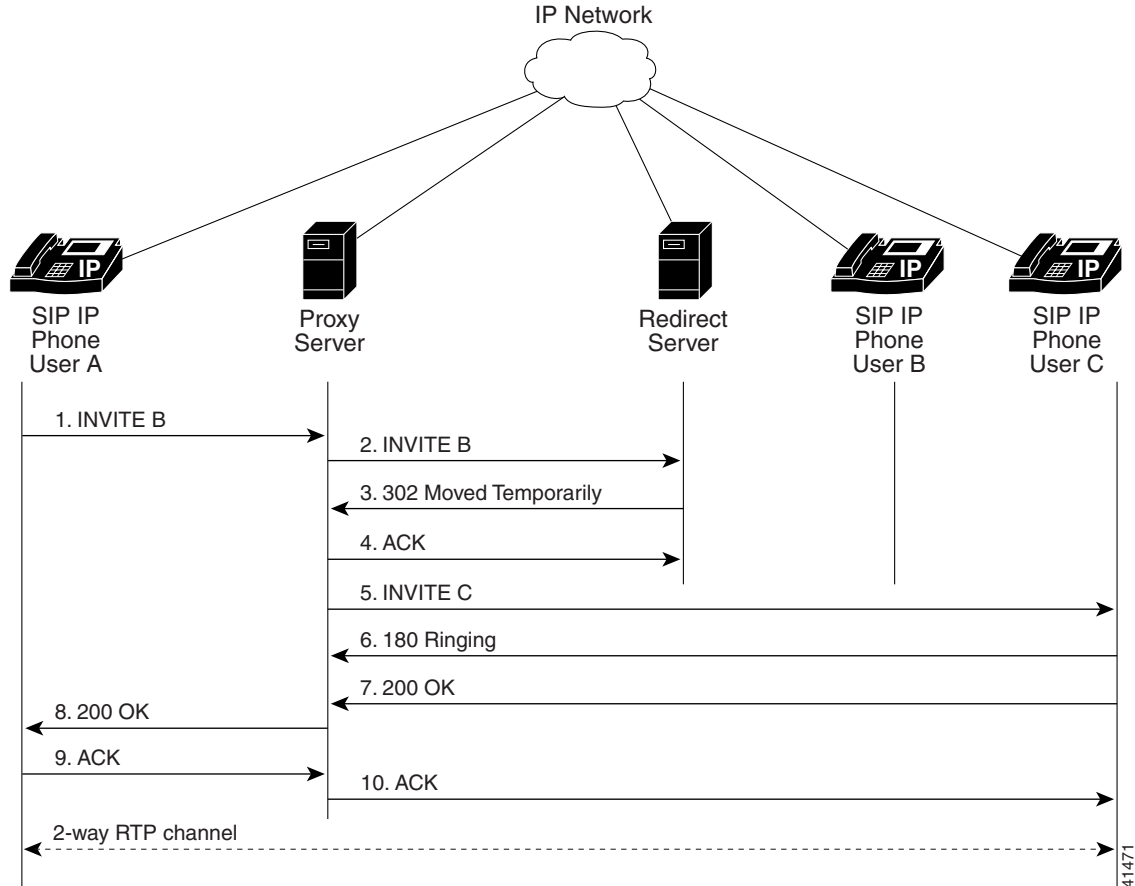
## Network Call Forwarding (Unconditional)

Figure B-11 illustrates successful call forwarding between Cisco SIP IP phones in which User B has requested unconditional call forwarding from the network. When User A calls User B, the call is immediately transferred to Phone C. In this call flow scenario, the end users are User A, User B, and User C. They are all using Cisco SIP IP phones, which are connected via an IP network.

The call flow scenario is as follows:

1. User B requests that the network forward all calls to Phone C.
2. User A calls User B.
3. The network transfers the call to Phone C.

Figure B-11 Network Call Forwarding (Unconditional)



Step	Action	Description
1.	INVITE—Phone A to SIP proxy server	<p>Phone A sends a SIP INVITE request to the SIP proxy server. The request is an invitation to User B to participate in a call session. In the INVITE request:</p> <ul style="list-style-type: none"> <li>• The phone number of User B is inserted in the Request-URI field in the form of a SIP URL. The SIP URL identifies the address of User B and takes a form similar to an e-mail address (<i>user@host</i>, where <i>user</i> is the telephone number and <i>host</i> is either a domain name or a numeric network address). For example, the Request-URI field in the INVITE request to User B appears as “INVITE sip:555-0199@companyb.com; user=phone.” The “user=phone” parameter distinguishes that the Request-URI address is a telephone number rather than a username.</li> <li>• Phone A is identified as the call session initiator in the From field.</li> <li>• A unique numeric identifier is assigned to the call and is inserted in the Call-ID field.</li> <li>• The transaction number within a single call leg is identified in the CSeq field.</li> <li>• The media capability User A is ready to receive is specified.</li> </ul>
2.	INVITE—SIP proxy server to SIP redirect server	The SIP proxy server sends the SIP INVITE request to the SIP redirect server.



Step	Action	Description
3.	302 Moved Temporarily—SIP redirect server to SIP proxy server	The SIP redirect server sends a SIP 302 Moved temporarily message to the SIP proxy server. The message indicates that User B is not available at Phone B and includes instructions to locate User B at Phone C.
4.	ACK—SIP proxy server to redirect server	The SIP proxy server sends the SIP ACK to the SIP redirect server.
5.	INVITE—SIP proxy server to Phone C	The SIP proxy server sends a SIP INVITE request to Phone C. The request is an invitation to User C to participate in a call session.
6.	180 Ringing—Phone C to SIP proxy server	Phone C sends a SIP 180 Ringing response to the SIP proxy server.
7.	200 OK—Phone C to SIP proxy server	Phone C sends a SIP 200 OK response to the SIP proxy server.
8.	200 OK—SIP proxy server to Phone A	The SIP proxy server forwards the SIP 200 OK response to Phone A.
9.	ACK—Phone A to SIP proxy server	Phone A sends a SIP ACK to the SIP proxy server. The ACK confirms that the SIP proxy server has received the 200 OK response from Phone C.
10.	ACK—SIP proxy server to Phone C	The SIP proxy server forwards the SIP ACK to the Phone C. The ACK confirms that Phone A has received the 200 OK response from Phone C.

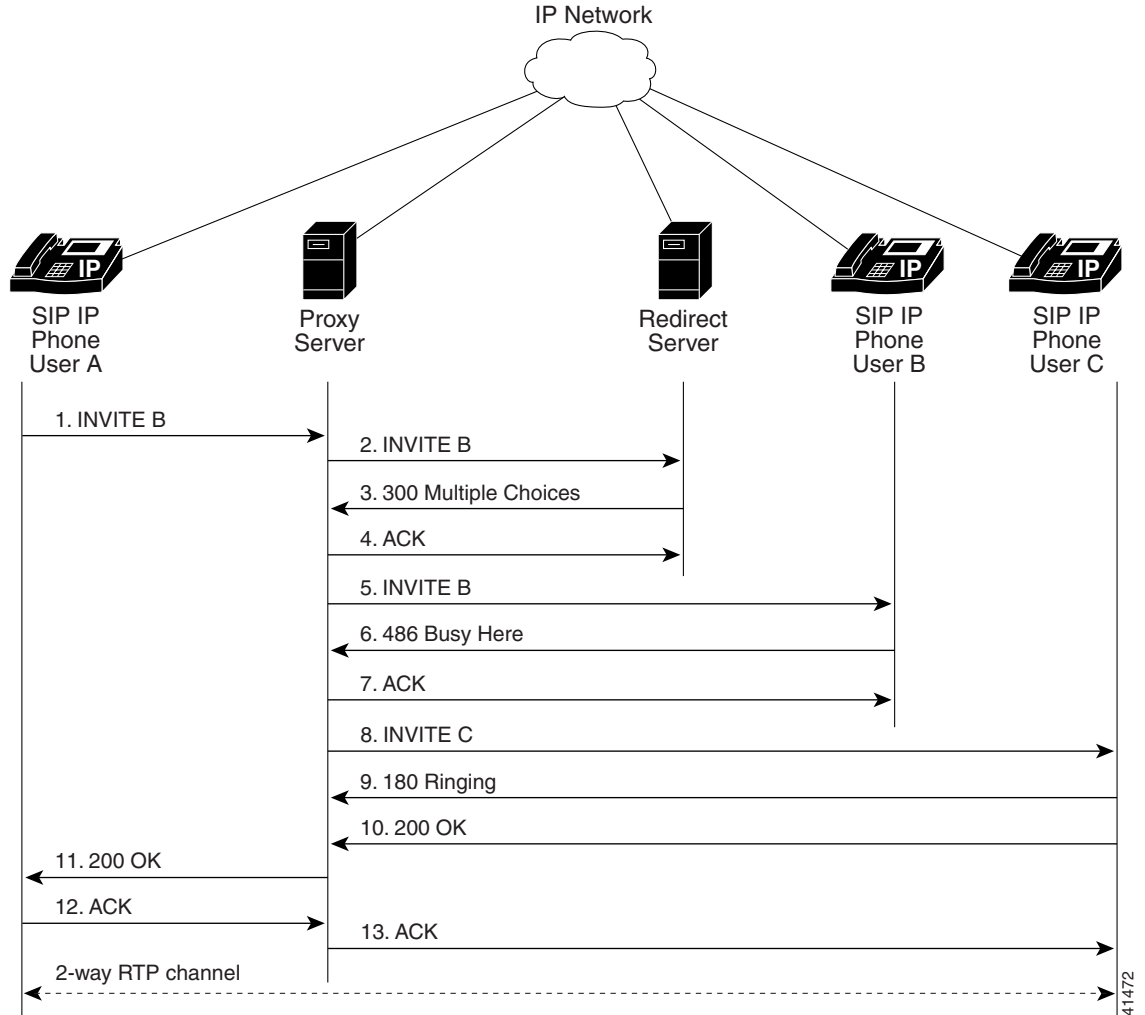
## Network Call Forwarding (Busy)

Figure B-12 illustrates successful call forwarding between Cisco SIP IP phones in which User B has requested call forwarding from the network in the event the phone is busy. When User A calls User B, the SIP proxy server tries to place the call to Phone B, and, if the line is busy, the call is transferred to Phone C. In this call flow scenario, the end users are User A, User B, and User C. They are all using Cisco SIP IP phones, which are connected via an IP network.

The call flow scenario is as follows:

1. User B requests that if Phone B is busy, the network should forward incoming calls to Phone C.
2. User A calls User B.
3. User B is busy.
4. The network transfers the call to Phone C.

Figure B-12 Network Call Forwarding (Busy)



Step	Action	Description
1.	INVITE—Phone A to SIP proxy server	<p>Phone A sends a SIP INVITE request to the SIP proxy server. The request is an invitation to User B to participate in a call session. In the INVITE request:</p> <ul style="list-style-type: none"> <li>The phone number of User B is inserted in the Request-URI field in the form of a SIP URL. The SIP URL identifies the address of User B and takes a form similar to an e-mail address (<i>user@host</i>, where <i>user</i> is the telephone number and <i>host</i> is either a domain name or a numeric network address). For example, the Request-URI field in the INVITE request to User B appears as “INVITE sip:555-0199@companyb.com; user=phone.” The “user=phone” parameter distinguishes that the Request-URI address is a telephone number rather than a username.</li> <li>Phone A is identified as the call session initiator in the From field.</li> <li>A unique numeric identifier is assigned to the call and is inserted in the Call-ID field.</li> <li>The transaction number within a single call leg is identified in the CSeq field.</li> <li>The media capability User A is ready to receive is specified.</li> </ul>
2.	INVITE—SIP proxy server to SIP redirect server	The SIP proxy server sends the SIP INVITE request to the SIP redirect server.
3.	300 Multiple Choices—SIP redirect server to SIP proxy server	The SIP redirect server sends a SIP 300 Multiple choices message to the SIP proxy server. The message indicates that User B can be reached either at Phone B or Phone C.
4.	ACK—SIP proxy server to redirect server	The SIP proxy server sends the SIP ACK to the SIP redirect server.
5.	INVITE—SIP proxy server to Phone B	The SIP proxy server sends a SIP INVITE request to Phone B. The request is an invitation to User B to participate in a call session.
6.	486 Busy Here—Phone B to SIP proxy server	Phone B sends a 486 Busy here message to the SIP proxy server. The message indicates that Phone B is in use and the user is not willing or able to take additional calls.
7.	ACK—SIP proxy server to Phone B	The SIP proxy server forwards the SIP ACK to the Phone B. The ACK confirms that the SIP proxy server has received the 486 Busy here response from Phone B.
8.	INVITE—SIP proxy server to Phone C	The SIP proxy server sends a SIP INVITE request to Phone C. The request is an invitation to User C to participate in a call session.
9.	180 Ringing—Phone C to SIP proxy server	Phone C sends a SIP 180 Ringing response to the SIP proxy server.
10.	200 OK—Phone C to SIP proxy server	Phone C sends a SIP 200 OK response to the SIP proxy server.
11.	200 OK—SIP proxy server to Phone A	The SIP proxy server forwards the SIP 200 OK response to Phone A.
12.	ACK—Phone A to SIP proxy server	Phone A sends a SIP ACK to the SIP proxy server. The ACK confirms that Phone A has received the 200 OK response from Phone C.
13.	ACK—SIP proxy server to Phone C	The SIP proxy server forwards the SIP ACK to the Phone C. The ACK confirms that Phone A has received the 200 OK response from Phone C.

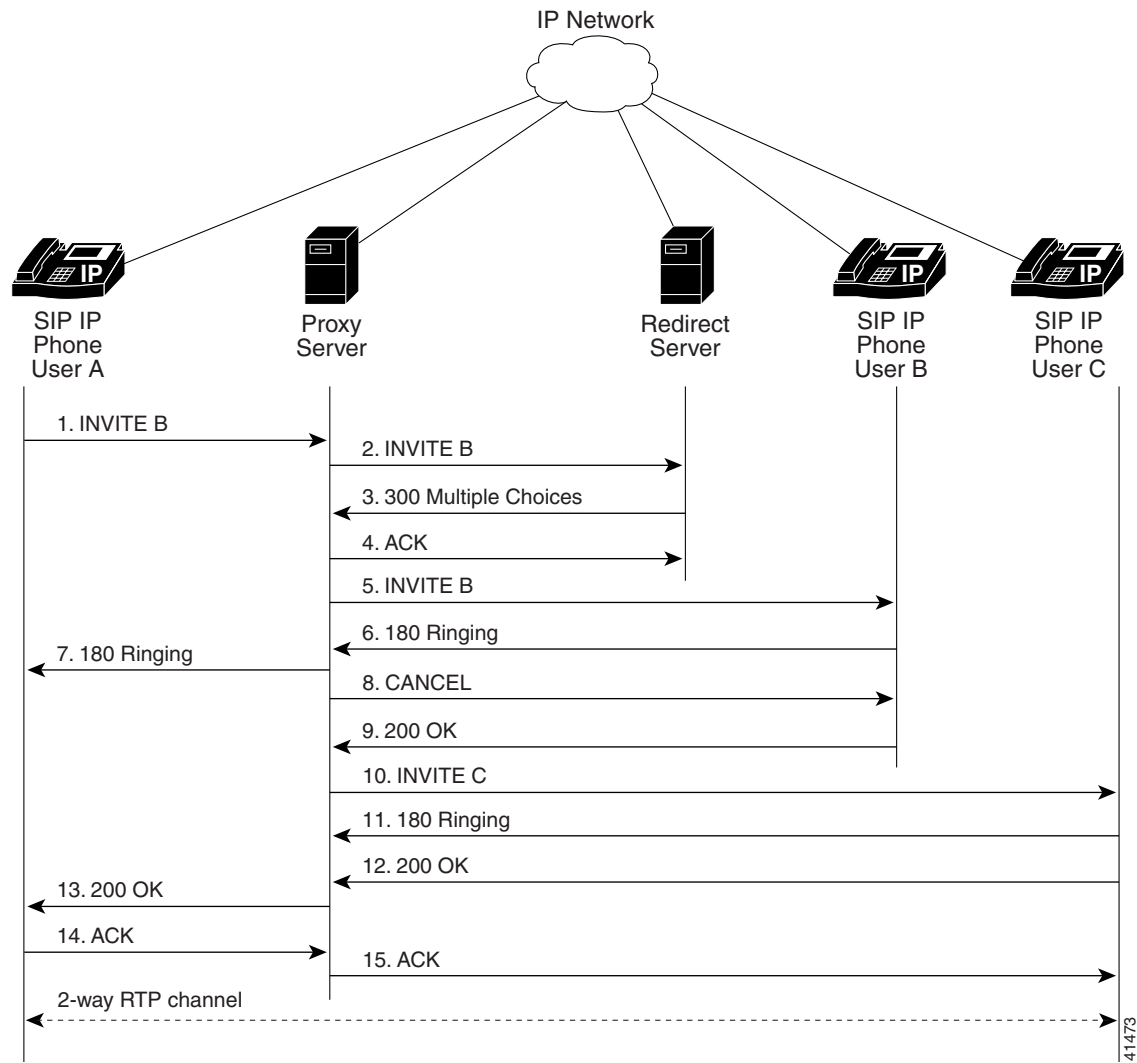
## Network Call Forwarding (No Answer)

Figure B-13 illustrates successful call forwarding between Cisco SIP IP phones in which User B has requested call forwarding from the network in the event that there is no answer. When User A calls User B, the proxy server tries to place the call to Phone B, and, if there is no answer, the call is transferred to Phone C. In this call flow scenario, the end users are User A, User B, and User C. They are all using Cisco SIP IP phones, which are connected via an IP network.

The call flow scenario is as follows:

1. User B requests that if the phone (Phone B) is not answered within a set amount of time, the network should forward incoming calls to Phone C.
2. User A calls User B.
3. User B does not answer.
4. The network transfers the call to Phone C.

**Figure B-13 Network Call Forwarding (No Answer)**



Step	Action	Description
1.	INVITE—Phone A to SIP proxy server	Phone A sends a SIP INVITE request to the SIP proxy server. The request is an invitation to User B to participate in a call session. In the INVITE request: <ul style="list-style-type: none"> <li>The phone number of User B is inserted in the Request-URI field in the form of a SIP URL. The SIP URL identifies the address of User B and takes a form similar to an e-mail address (<i>user@host</i>, where <i>user</i> is the telephone number and <i>host</i> is either a domain name or a numeric network address). For example, the Request-URI field in the INVITE request to User B appears as “INVITE sip:555-0199@companyb.com; user=phone.” The “user=phone” parameter distinguishes that the Request-URI address is a telephone number rather than a username.</li> <li>Phone A is identified as the call session initiator in the From field.</li> <li>A unique numeric identifier is assigned to the call and is inserted in the Call-ID field.</li> <li>The transaction number within a single call leg is identified in the CSeq field.</li> <li>The media capability User A is ready to receive is specified.</li> </ul>
2.	INVITE—SIP proxy server to SIP redirect server	The SIP proxy server sends the SIP INVITE request to the SIP redirect server.
3.	300 Multiple Choices—SIP redirect server to SIP proxy server	The SIP redirect server sends a SIP 300 Multiple choices message to the SIP proxy server. The message indicates that User B can be reached either at Phone B or Phone C.
4.	ACK—SIP proxy server to redirect server	The SIP proxy server sends the SIP ACK to the SIP redirect server.
5.	INVITE—SIP proxy server to Phone B	The SIP proxy server sends a SIP INVITE request to Phone B. The request is an invitation to User B to participate in a call session.
6.	180 Ringing—Phone B to SIP proxy server	Phone B sends a SIP 180 Ringing response to the SIP proxy server.
7.	180 Ringing—SIP proxy server to Phone A	The SIP proxy server sends a SIP 180 Ringing response to Phone A.
The timeout expires before the phone is answered.		
8.	CANCEL (Ring Timeout)—SIP proxy server to Phone B	The SIP proxy server sends a CANCEL request to Phone B to cancel the invitation.
9.	200 OK—Phone B to SIP proxy server	Phone B sends a SIP 200 OK response to the SIP proxy server. The response confirms receipt of the cancellation request.
10.	INVITE—SIP proxy server to Phone C	The SIP proxy server sends a SIP INVITE request to Phone C. The request is an invitation to User C to participate in a call session.
11.	180 Ringing—Phone C to SIP proxy server	Phone C sends a SIP 180 Ringing response to the SIP proxy server.
12.	200 OK—Phone C to SIP proxy server	Phone C sends a SIP 200 OK response to the SIP proxy server.
13.	200 OK—SIP proxy server to Phone A	The SIP proxy server forwards the SIP 200 OK response to Phone A.

Step	Action	Description
14.	ACK—Phone A to SIP proxy server	Phone A sends a SIP ACK to the SIP proxy server. The ACK confirms that Phone A has received the 200 OK response from Phone C.
15.	ACK—SIP proxy server to Phone C	The SIP proxy server forwards the SIP ACK to the Phone C. The ACK confirms that Phone A has received the 200 OK response from Phone C.

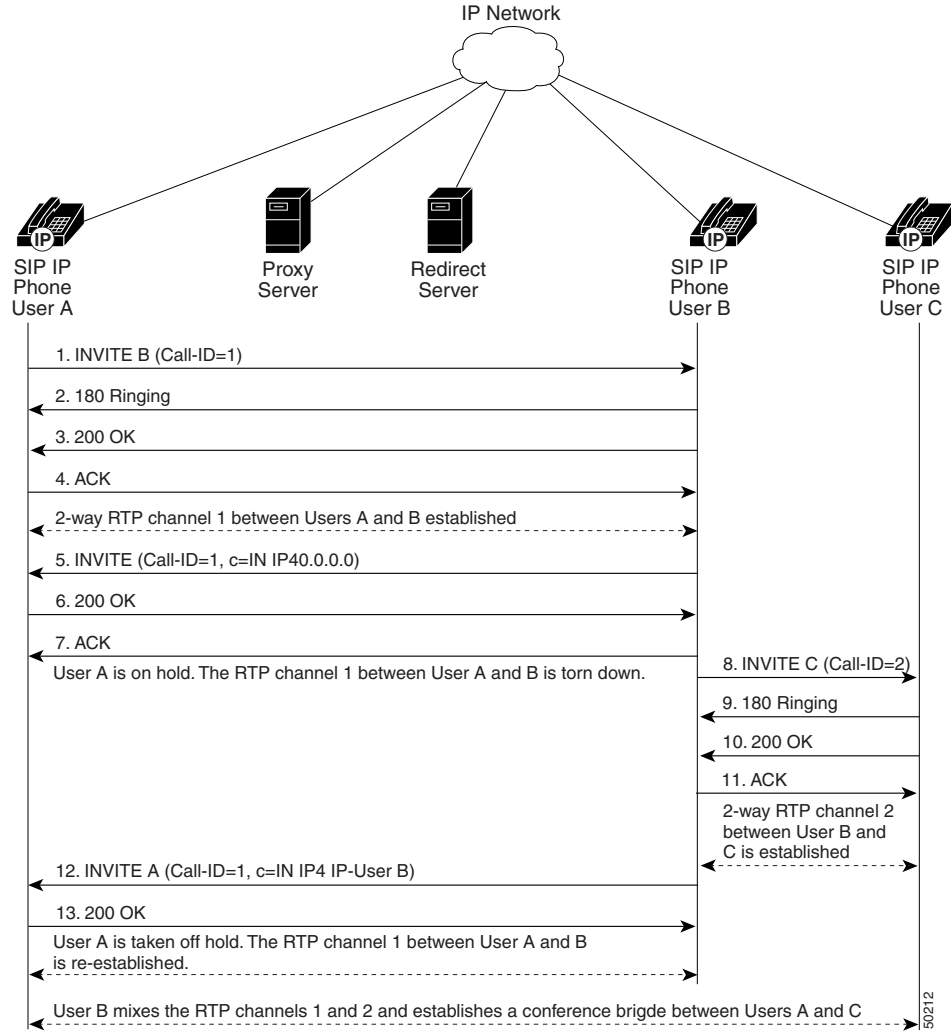
## Three-Way Calling

Figure B-14 illustrates successful three-way calling between Cisco SIP IP phones in which User B mixes two RTP channels and therefore establishes a conference bridge between User A and User C. In this call flow scenario, the end users are User A, User B, and User C. They are all using Cisco IP 7960G/7940G phones, which are connected using an IP network.

The call flow scenario is as follows:

1. User A calls User B.
2. User B answers the call.
3. User B puts User A on hold.
4. User B calls User C.
5. User C answers the call.
6. User B takes User A off hold.

Figure B-14 Three-Way Calling



Step	Action	Description
1.	INVITE—Phone A to Phone B	<p>Phone A sends a SIP INVITE request to Phone B. The request is an invitation to User B to participate in a call session. In the INVITE request:</p> <ul style="list-style-type: none"> <li>The phone number of User B is inserted in the Request-URI field in the form of a SIP URL. The SIP URL identifies the address of User B and takes a form similar to an e-mail address (<i>user@host</i>, where <i>user</i> is the telephone number and <i>host</i> is either a domain name or a numeric network address). For example, the Request-URI field in the INVITE request to User B appears as “INVITE sip:555-0199@companyb.com; user=phone.” The “user=phone” parameter distinguishes that the Request-URI address is a telephone number rather than a username.</li> <li>Phone A is identified as the call session initiator in the From field.</li> <li>A unique numeric identifier is assigned to the call and is inserted in the Call-ID field.</li> <li>The transaction number within a single call leg is identified in the CSeq field.</li> <li>The media capability A is ready to receive is specified.</li> </ul>
2.	180 Ringing—Phone B to Phone A	Phone B sends a SIP 180 Ringing response to Phone A.
3.	200 OK—Phone B to Phone A	<p>Phone B sends a SIP 200 OK response to Phone A. The response notifies Phone A that the connection has been made.</p> <p>If Phone B supports the media capability advertised in the INVITE message sent by Phone A, it advertises the intersection of its own and Phone A’s media capability in the 200 OK response. If Phone B does not support the media capability advertised by Phone A, it sends back a 400 Bad Request response with a 304 Warning header field.</p>
4.	ACK—Phone A to Phone B	<p>Phone A sends a SIP ACK to Phone B. The ACK confirms that Phone A has received the 200 OK response from Phone B.</p> <p>The ACK might contain a message body with the final session description to be used by Phone B. If the message body of the ACK is empty, Phone B uses the session description in the INVITE request.</p>
A two-way RTP channel is established between Phone A and Phone B.		
5.	INVITE—Phone B to Phone A	<p>Phone B sends a midcall INVITE to Phone A with new SDP session parameters (IP address), which are used to place the call on hold.</p> <pre>Call_ID=1 SDP: c=IN IP4 0.0.0.0</pre> <p>The c= SDP field of the SIP INVITE contains an 0.0.0.0. This places the call in hold.</p>
6.	200 OK—Phone A to Phone B	Phone A sends a SIP 200 OK response to Phone B.
7.	ACK—Phone B to Phone A	Phone B sends a SIP ACK to Phone A. The ACK confirms that Phone B has received the 200 OK response from Phone A.
The RTP channel between Phone A and Phone B is torn down. A is put on hold.		



Step	Action	Description
8.	INVITE—Phone B to Phone C	<p>Phone B sends a SIP INVITE request to Phone C. The request is an invitation to Phone B to participate in a call session. In the INVITE request:</p> <ul style="list-style-type: none"> <li>The phone number of Phone B is inserted in the Request-URI field in the form of a SIP URL. The SIP URL identifies the address of B and takes a form similar to an e-mail address (<i>user@host</i>, where <i>user</i> is the telephone number and <i>host</i> is either a domain name or a numeric network address). For example, the Request-URI field in the INVITE request to C appears as “INVITE sip:555-0199@companyb.com; user=phone.” The “user=phone” parameter distinguishes that the Request-URI address is a telephone number rather than a username.</li> <li>Phone B is identified as the call session initiator in the From field.</li> <li>A unique numeric identifier is assigned to the call and is inserted in the Call-ID field.</li> <li>The transaction number within a single call leg is identified in the CSeq field.</li> <li>The media capability B is ready to receive is specified.</li> </ul>
9.	180 Ringing—Phone C to Phone B	Phone C sends a SIP 180 Ringing response to Phone B.
10.	200 OK—Phone C to Phone B	<p>Phone C sends a SIP 200 OK response to Phone B. The response notifies Phone B that the connection has been made.</p> <p>If Phone C supports the media capability advertised in the INVITE message sent by Phone B, it advertises the intersection of its own and Phone B’s media capability in the 200 OK response. If Phone C does not support the media capability advertised by Phone B, it sends back a 400 Bad Request response with a 304 Warning header field.</p>
11.	ACK—Phone B to Phone C	<p>Phone B sends a SIP ACK to Phone C. The ACK confirms that Phone B has received the 200 OK response from Phone C.</p> <p>The ACK might contain a message body with the final session description to be used by Phone C. If the message body of the ACK is empty, Phone C uses the session description in the INVITE request.</p>

A two-way RTP channel is established between SIP IP Phone B and SIP IP Phone C.

12.	INVITE—Phone B to Phone A	<p>Phone B sends a midcall INVITE to Phone A with the same call ID as the previous INVITE and new SDP session parameters (IP address), which are used to reestablish the call.</p> <pre>Call_ID=1 SDP: c=IN IP4 10.10.10.0</pre> <p>To reestablish the call between Phone A and Phone B, the IP address of Phone B is inserted into the c= SDP field.</p>
13.	200 OK—Phone A to Phone B	Phone A sends a SIP 200 OK response to Phone B.

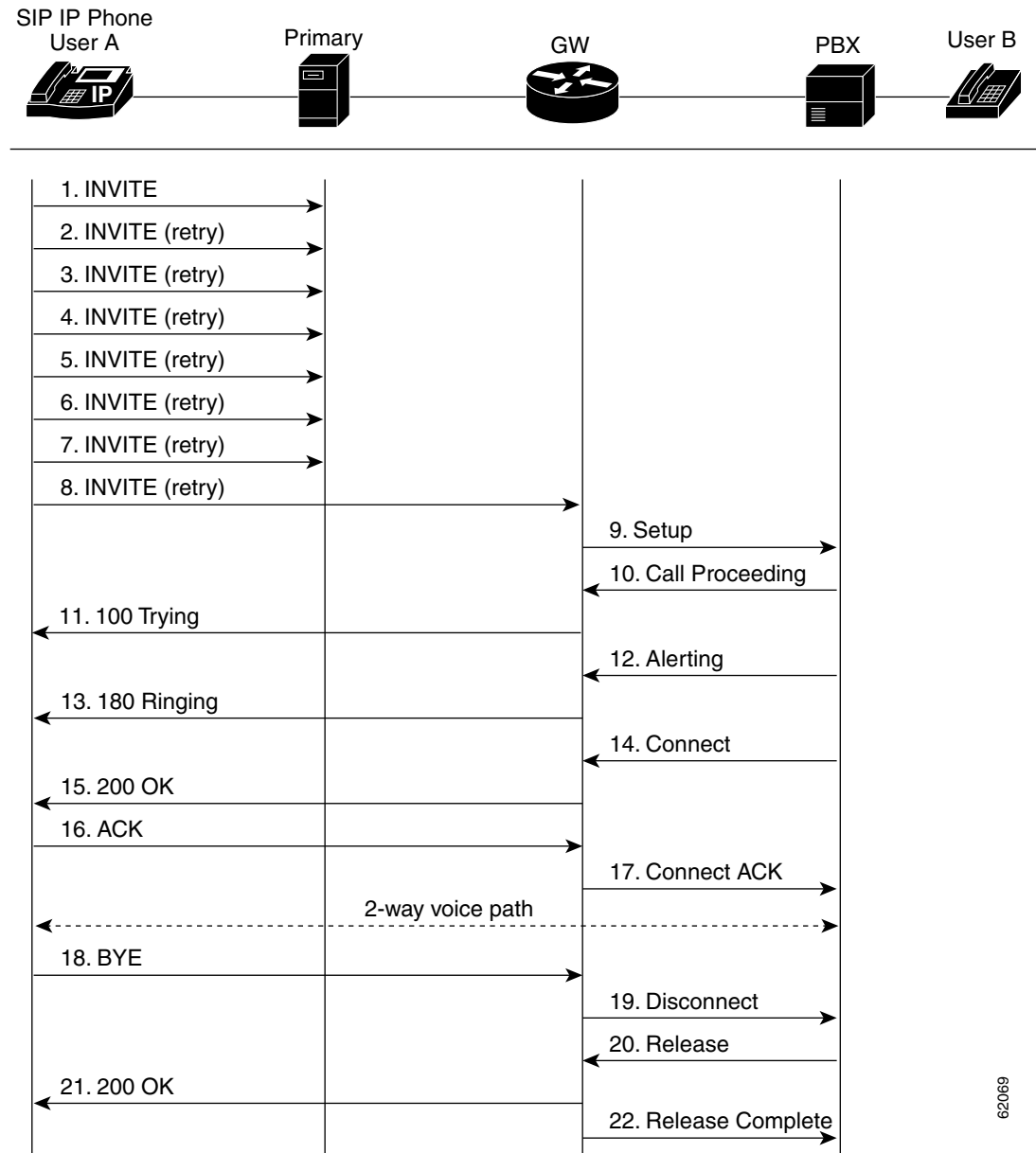
A is taken off hold. The RTP channel 1 between A and B is reestablished.

Phone B acts as a bridge mixing the RTP channel between A and B with the channel between B and C, establishing a conference bridge between A and C.

## Call from a Cisco SIP IP Phone to a Gateway Acting As a Backup Proxy in a SIP Network

Figure B-15 illustrates a successful call from a Cisco SIP IP phone to a gateway acting as a backup proxy.

**Figure B-15 Call from a Cisco SIP IP Phone to a Gateway Acting As a Backup Proxy**



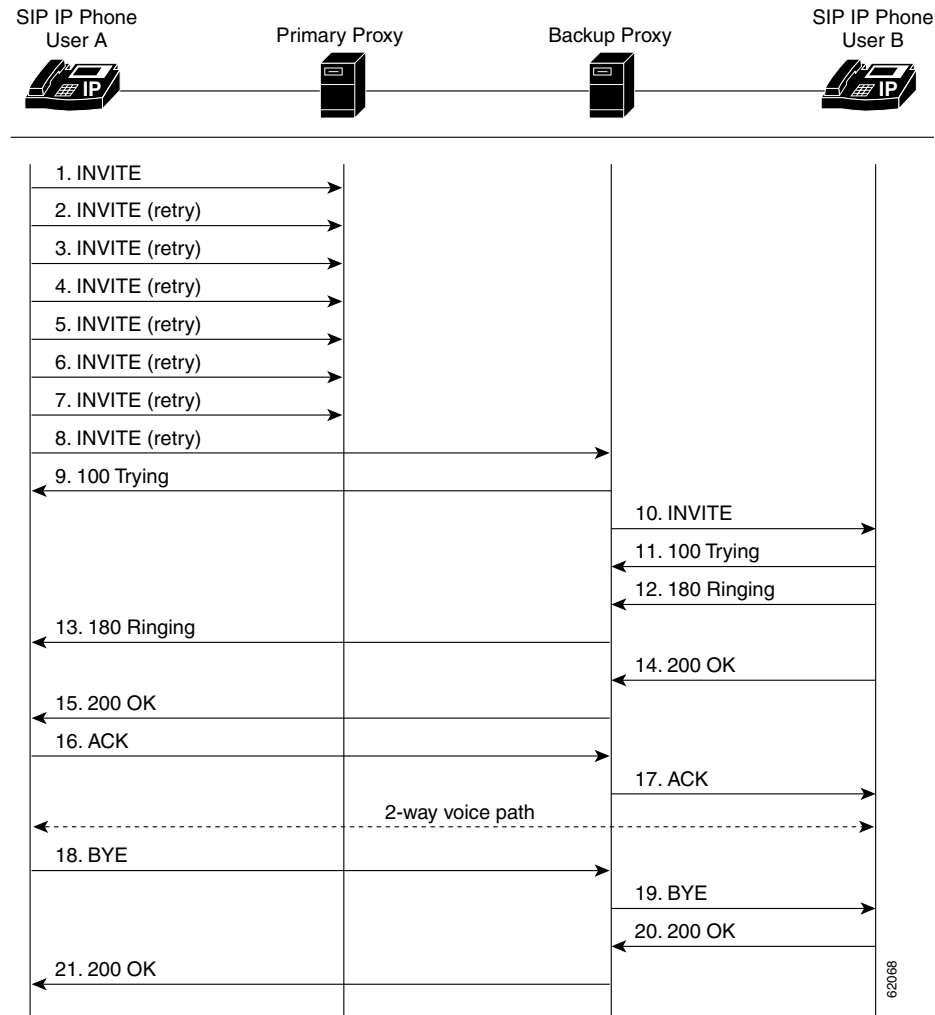
69029

Step	Action	Description
1.	INVITE—Cisco SIP IP phone to primary proxy	The phone tries to connect to the proxy by sending out the INVITE message.
2.	INVITE—Cisco SIP IP phone to primary proxy (second try)	The phone retries a second time to connect to the proxy by sending out the INVITE message.
3.	INVITE—Cisco SIP IP phone to primary proxy (third try)	The phone retries a third time to connect to the proxy by sending out the INVITE message.
4.	INVITE—Cisco SIP IP phone to primary proxy (fourth try)	The phone retries a fourth time to connect to the proxy by sending out the INVITE message.
5.	INVITE—Cisco SIP IP phone to primary proxy (fifth try)	The phone retries a fifth time to connect to the proxy by sending out the INVITE message.
6.	INVITE—Cisco SIP IP phone to primary proxy (sixth try)	The phone retries a sixth time to connect to the proxy by sending out the INVITE message.
7.	INVITE—Cisco SIP IP phone to primary proxy (seventh try)	The phone retries a seventh time to connect to the proxy. If the connection is not successful after this trial, the “Network Delay, Trying Backup” message displays on the phone.
8.	INVITE—Cisco SIP IP phone to the gateway (backup proxy)	The phone tries to connect to the gateway (backup proxy) by sending out the INVITE message.
9.	Setup—Gateway to the PBX	Call setup is initiated between the gateway and the PBX. The setup includes the standard transactions that take place as A attempts to call B.
10.	Call Proceeding—PBX to the gateway	The PBX sends a Call Proceeding message to the gateway to acknowledge the Call Setup request.
11.	100 Trying—Gateway to Cisco SIP IP phone (A)	The gateway sends a SIP 100 Trying response to A. The response indicates that the INVITE request has been received.
12.	Alerting—PBX to the gateway	The PBX sends an Alert message to the gateway. The message indicates that the PBX has received a 100 Trying Ringing response from the gateway.
13.	180 Ringing—Gateway to Cisco SIP IP phone (A)	The gateway sends a SIP 180 Ringing response to A. The response indicates that the gateway is being alerted.
14.	Connect—PBX to the gateway	The PBX sends a Connect message to the gateway. The message notifies the gateway that the connection has been made.
15.	200 OK—Gateway to Cisco SIP IP phone (A)	The gateway sends a SIP 200 OK response to A. The response notifies A that the connection has been made.
16.	ACK—Cisco SIP IP phone (A) to the gateway	A sends a SIP ACK to the gateway. The ACK confirms that A has received the 200 OK response. The call session is now active.
17.	Connect ACK—Gateway to the PBX	The gateway acknowledges PBX Connect message.
18.	BYE—Cisco SIP IP phone (A) to the gateway	A terminates the call session and sends a SIP BYE request to the gateway. The request indicates that A wants to release the call.
19.	Disconnect—Gateway to the PBX	The gateway sends a Disconnect message to the PBX.
20.	Release—PBX to the gateway	The PBX sends a Release message to the gateway.
21.	200 OK—Gateway to Cisco SIP IP phone (A)	The gateway sends a SIP 200 OK response to A. The response notifies A that the gateway has received the BYE request.
22.	Release Complete—Gateway to the PBX	The gateway sends a Release Complete message to the PBX, and the call session terminates.

## Call from a Cisco SIP IP Phone to a Cisco SIP IP Phone Using a SIP Backup Proxy

Figure B-16 illustrates a successful call from a Cisco SIP IP phone to a Cisco SIP IP phone that uses a SIP backup proxy.

**Figure B-16 A Successful Call from a Cisco SIP IP Phone to a Cisco SIP IP Phone Using a SIP Backup Proxy**



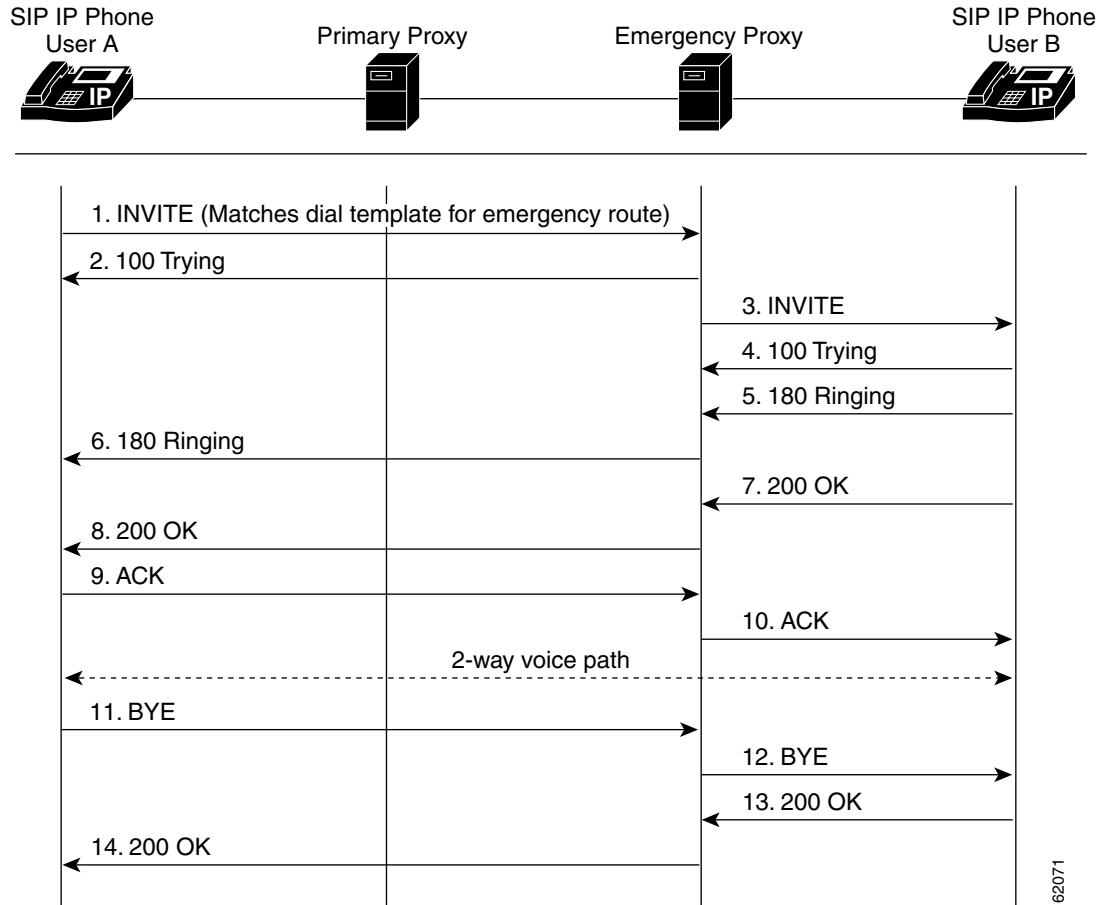
Step	Action	Description
1.	INVITE—Cisco SIP IP phone (A) to primary proxy	The phone tries to connect to the primary proxy by sending out the INVITE message.
2.	INVITE—Cisco SIP IP phone (A) to primary proxy (second try)	The phone retries a second time to connect to the primary proxy by sending out the INVITE message.
3.	INVITE—Cisco SIP IP phone (A) to primary proxy (third try)	The phone retries a third time to connect to the primary proxy by sending out the INVITE message.

Step	Action	Description
4.	INVITE—Cisco SIP IP phone (A) to primary proxy (fourth try)	The phone retries a fourth time to connect to the primary proxy by sending out the INVITE message.
5.	INVITE—Cisco SIP IP phone (A) to primary proxy (fifth try)	The phone retries a fifth time to connect to the primary proxy by sending out the INVITE message.
6.	INVITE—Cisco SIP IP phone (A) to primary proxy (sixth try)	The phone retries a sixth time to connect to the primary proxy by sending out the INVITE message.
7.	INVITE—Cisco SIP IP phone (A) to primary proxy (seventh try)	The phone retries a seventh time to connect to the primary proxy. If the connection is not successful after this trial, the “Network Delay, Trying Backup” message displays on the Phone.
8.	INVITE—Cisco SIP IP phone (A) to backup proxy	The phone tries to connect to the backup proxy by sending out the INVITE message.
9.	100 Trying—Backup proxy to Cisco SIP IP phone (A)	The backup proxy sends a SIP 100 Trying response to the phone. The response indicates that the INVITE request has been received.
10.	INVITE—Backup proxy to Cisco SIP IP phone (B)	The backup proxy tries to connect to B by sending out the INVITE message.
11.	100 Trying—Cisco SIP IP phone (B) to backup proxy	Phone B sends a SIP 100 Trying response to the backup proxy. The response indicates that the INVITE request has been received.
12.	180 Ringing—Cisco SIP IP phone (B) to backup proxy	Phone B sends a SIP 180 Ringing response to the backup proxy. The response indicates that B is being alerted.
13.	180 Ringing—Backup proxy to Cisco SIP IP phone (A)	The backup proxy sends a SIP 180 Ringing response to A. The response indicates that the backup proxy is being alerted.
14.	200 OK—Cisco SIP IP phone (B) to backup proxy	Phone B sends a SIP 200 OK response to the backup proxy. The response notifies the backup proxy that the connection has been made.
15.	200 OK—Backup proxy to Cisco SIP IP phone (A)	The backup proxy sends a SIP 200 OK response to Phone A. The response notifies Phone A that the connection has been made.
16.	ACK—Cisco SIP IP phone (A) to backup proxy	Phone A acknowledges the backup proxy Connect message.
17.	ACK—Backup proxy to Cisco SIP IP phone (B)	The backup proxy acknowledges the Phone B Connect message.
18.	BYE—Cisco SIP IP phone (A) to backup proxy	A terminates the call session and sends a SIP BYE request to the backup proxy. The request indicates that User A wants to release the call.
19.	BYE—Backup proxy to Cisco SIP IP phone (B)	The backup proxy terminates the call session and sends a SIP BYE request to User B. The request indicates that the backup proxy wants to release the call.
20.	200 OK—Cisco SIP IP phone (B) to backup proxy	User B sends a SIP 200 OK response to the backup proxy. The response notifies the backup proxy that B has received the BYE request.
21.	200 OK—Backup proxy to Cisco SIP IP phone (A)	The backup proxy sends a SIP 200 OK response to User A. The response notifies User A that the backup proxy has received the BYE request.

## Call from a Cisco SIP IP Phone to a Cisco SIP IP Phone Using a SIP Emergency Proxy

Figure B-17 illustrates a successful call from a Cisco SIP IP phone to a Cisco SIP IP phone via an emergency proxy. B is the extension of the dial template with the “Route” attribute as “emergency” in the dialplan.xml file.

Figure B-17 Successful Call from a Cisco IP Phone to a Cisco IP Phone Using a SIP Emergency Proxy



62071

Step	Action	Description
1.	INVITE—Cisco SIP IP phone (A) to emergency proxy	The phone tries to connect to the emergency proxy by sending out the INVITE message. The dial template for the emergency route is matched.
2.	100 Trying—Emergency proxy to Cisco SIP IP phone (A)	The emergency proxy sends a SIP 100 Trying response to A. The response indicates that the INVITE request has been received.
3.	INVITE—Emergency proxy to Cisco SIP IP phone (B)	The backup proxy tries to connect to B by sending out the INVITE message.
4.	100 Trying—Cisco SIP IP phone (B) to emergency proxy	User B sends a SIP 100 Trying response to the emergency proxy. The response indicates that the INVITE request has been received.
5.	180 Ringing—Cisco SIP IP phone (B) to emergency proxy	User B sends a SIP 180 Ringing response to the emergency proxy. The response indicates that User B is being alerted.
6.	180 Ringing—Emergency proxy to Cisco SIP IP phone (A)	The emergency proxy sends a SIP 180 Ringing response to User A. The response indicates that the emergency proxy is being alerted.
7.	200 OK—Cisco SIP IP phone (B) to emergency proxy	User B sends a SIP 200 OK response to the emergency proxy. The response notifies the emergency proxy that the connection has been made.
8.	200 OK—Emergency proxy to Cisco SIP IP phone (A)	The emergency proxy sends a SIP 200 OK response to User A. The response notifies User A that the connection has been made.

Step	Action	Description
9.	ACK—Cisco SIP IP phone (A) to emergency proxy	A acknowledges the emergency proxy's Connect message.
10.	ACK—Emergency proxy to Cisco SIP IP phone (B)	The emergency proxy acknowledges B's Connect message.
11.	BYE—Cisco SIP IP phone (A) to emergency proxy	A terminates the call session and sends a SIP BYE request to the emergency proxy. The request indicates that A wants to release the call.
12.	BYE—Emergency proxy to Cisco SIP IP phone (B)	The emergency proxy terminates the call session and sends a SIP BYE request to User B. The request indicates that the emergency proxy wants to release the call.
13.	200 OK—Cisco SIP IP phone (B) to emergency proxy	User B sends a SIP 200 OK response to the emergency proxy. The response notifies the emergency proxy that B has received the BYE request.
14.	200 OK—Emergency proxy to Cisco SIP IP phone (A)	The emergency proxy sends a SIP 200 OK response to A. The response notifies A that the emergency proxy has received the BYE request.

## Call Flow Scenarios for Failed Calls

This section describes call flows for the following scenarios, which illustrate unsuccessful calls:

- Gateway to Cisco SIP IP Phone in a SIP Network, page B-47
- Cisco SIP IP Phone to Cisco SIP IP Phone in a SIP Network, page B-52

### Gateway to Cisco SIP IP Phone in a SIP Network

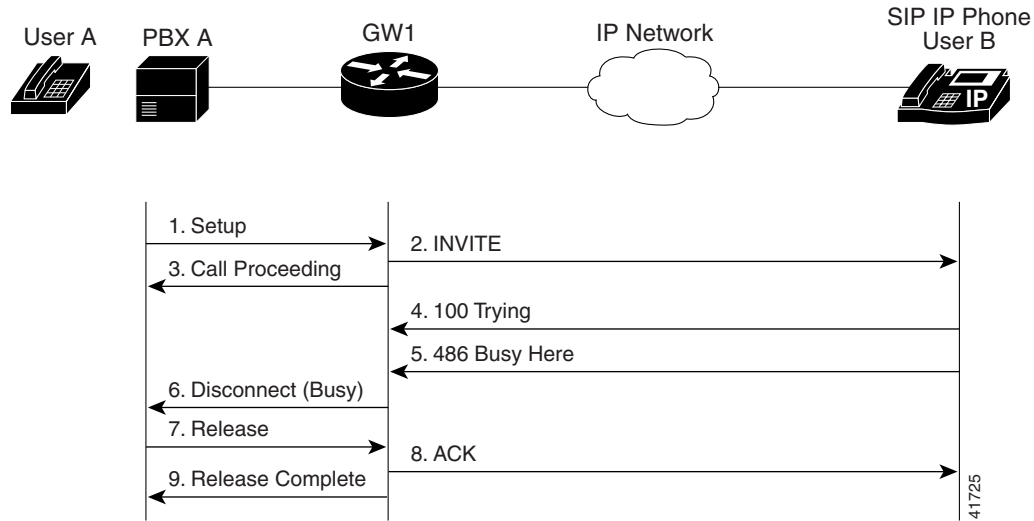
The following scenarios are failed calls in the gateway to a Cisco SIP IP phone:

- Called Number Is Busy, page B-47
- Called Number Does Not Answer, page B-49
- Client, Server, or Global Error, page B-50

#### Called Number Is Busy

Figure B-18 illustrates an unsuccessful call in which User A initiates a call to User B, but User B is on the phone and is unable or unwilling to take another call.

Figure B-18 Called Number is Busy



Step	Action	Description
1.	Setup—PBX A to Gateway 1	Call setup is initiated between PBX A and Gateway 1. Setup includes the standard transactions that take place as A attempts to call B.
2.	INVITE—Gateway 1 to Cisco SIP IP phone	Gateway 1 maps the SIP URL phone number to a dial peer. The dial peer includes the IP address and the port number of the SIP-enabled entity to contact. Gateway 1 sends a SIP INVITE request to the address it receives as the dial peer, which, in this scenario, is the IP phone. In the INVITE request: <ul style="list-style-type: none"> <li>The IP address of the phone is inserted in the Request-URI field.</li> <li>PBX A is identified as the call session initiator in the From field.</li> <li>A unique numeric identifier is assigned to the call and is inserted in the Call-ID field.</li> <li>The transaction number within a single call leg is identified in the CSeq field.</li> <li>The media capability A is ready to receive is specified.</li> <li>The port on which the gateway is prepared to receive the RTP data is specified.</li> </ul>
3.	Call Proceeding—Gateway 1 to PBX A	Gateway 1 sends a Call Proceeding message to PBX A to acknowledge the Call Setup request.
4.	100 Trying—Cisco SIP IP phone to Gateway 1	The phone sends a SIP 100 Trying response to Gateway 1. The response indicates that the INVITE request has been received.
5.	486 Busy Here—Cisco SIP IP phone to Gateway 1	The phone sends a SIP 486 Busy Here response to Gateway 1. The response is a client error response that indicates that B was successfully contacted but that B was not willing or was unable to take the call.
6.	Disconnect (Busy)—Gateway 1 to PBX A	Gateway 1 sends a Disconnect message to PBX A.
7.	Release—PBX A to Gateway 1	PBX A sends a Release message to Gateway 1.

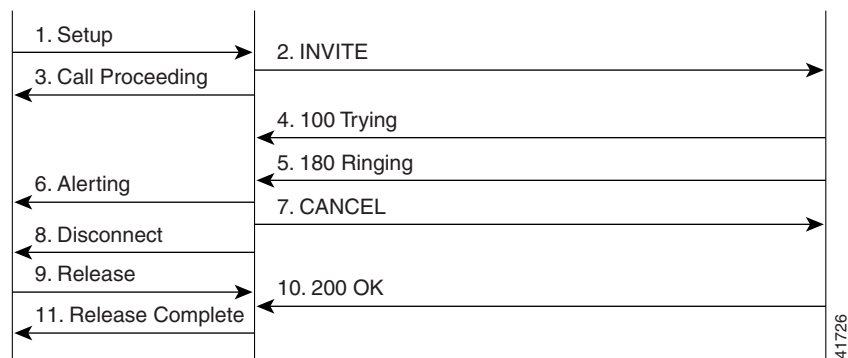


Step	Action	Description
8.	ACK—Gateway 1 to Cisco SIP IP phone	Gateway 1 sends a SIP ACK to the phone. The ACK confirms that A has received the 486 Busy Here response. The call session attempt terminates.
9.	Release Complete—Gateway 1 to PBX A	Gateway 1 sends a Release Complete message to PBX A, and the call session attempt terminates.

## Called Number Does Not Answer

Figure B-19 illustrates the call flow in which User A initiates a call to User B, but User B does not answer.

**Figure B-19 Called Number Does Not Answer**



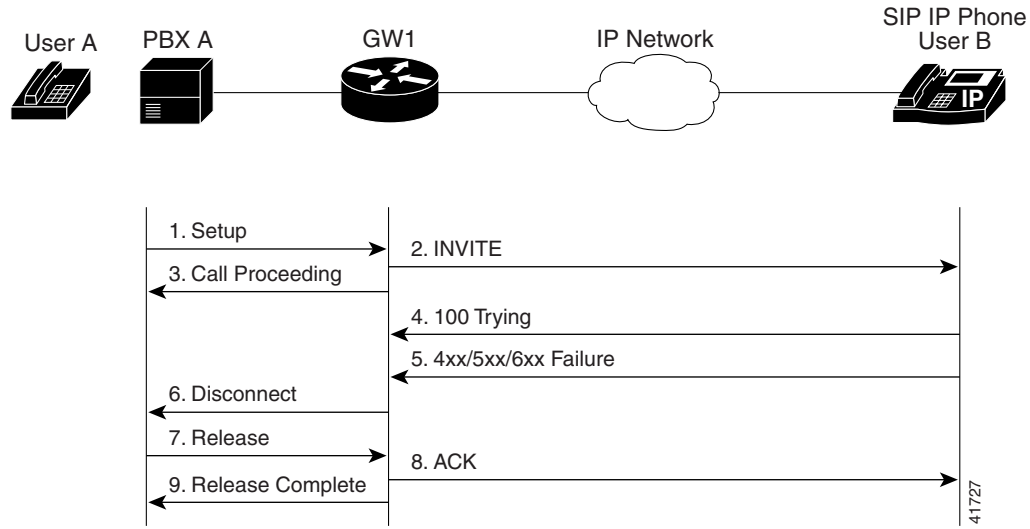
41726

Step	Action	Description
1.	Setup—PBX A to Gateway 1	Call setup is initiated between PBX A and Gateway 1. Setup includes the standard transactions that take place as A attempts to call B.
2.	INVITE—Gateway 1 to Cisco SIP IP phone	Gateway 1 maps the SIP URL phone number to a dial peer. The dial peer includes the IP address and the port number of the SIP-enabled entity to contact. Gateway 1 sends a SIP INVITE request to the address it receives as the dial peer, which, in this scenario, is the IP phone. In the INVITE request: <ul style="list-style-type: none"> <li>• The IP address of the phone is inserted in the Request-URI field.</li> <li>• PBX A is identified as the call session initiator in the From field.</li> <li>• A unique numeric identifier is assigned to the call and is inserted in the Call-ID field.</li> <li>• The transaction number within a single call leg is identified in the CSeq field.</li> <li>• The media capability A is ready to receive is specified.</li> <li>• The port on which the gateway is prepared to receive the RTP data is specified.</li> </ul>
3.	Call Proceeding—Gateway 1 to PBX A	Gateway 1 sends a Call Proceeding message to PBX A to acknowledge the Call Setup request.
4.	100 Trying—Cisco SIP IP phone to Gateway 1	The phone sends a SIP 100 Trying response to Gateway 1. The response indicates that the INVITE request has been received.
5.	180 Ringing—Cisco SIP IP phone to Gateway 1	The phone sends a SIP 180 Ringing response to Gateway 1. The response indicates that the user is being alerted.
6.	Alerting—Gateway 1 to PBX A	Gateway 1 sends an Alert message to PBX A.
7.	CANCEL (Ring Timeout)—Gateway 1 to Cisco SIP IP phone	Because Gateway 1 did not return an appropriate response within the time allocated in the INVITE request, Gateway 1 sends a SIP CANCEL request to the gateway 2. A CANCEL request cancels a pending request with the same Call-ID, To, From, and CSeq header field values.
8.	Disconnect—Gateway 1 to PBX A	Gateway 1 sends a Disconnect message to PBX A.
9.	Release Complete—Gateway 1 to PBX A	Gateway 1 sends a Release Complete message to PBX A, and the call session attempt terminates.
10.	200 OK—Cisco SIP IP phone to Gateway 1	The phone sends a SIP 200 OK response to Gateway 1. The response confirms that A has received the 486 Busy Here response. The call session attempt terminates.
11.	Release Complete—Gateway 1 to PBX A	Gateway 1 sends a Release Complete message to PBX A, and the call session terminates.

## Client, Server, or Global Error

Figure B-20 illustrates an unsuccessful call in which User A initiates a call to User B and receives a class 4xx, 5xx, or 6xx response.

Figure B-20 Client, Server, or Global Error



Step	Action	Description
1.	Setup—PBX A to Gateway 1	Call setup is initiated between PBX A and Gateway 1. Setup includes the standard transactions that take place as A attempts to call B.
2.	INVITE—Gateway 1 to Cisco SIP IP phone	Gateway 1 maps the SIP URL phone number to a dial peer. The dial peer includes the IP address and the port number of the SIP-enabled entity to contact. Gateway 1 sends a SIP INVITE request to the address it receives as the dial peer, which, in this scenario, is the IP phone. In the INVITE request: <ul style="list-style-type: none"> <li>• The IP address of the phone is inserted in the Request-URI field.</li> <li>• PBX A is identified as the call session initiator in the From field.</li> <li>• A unique numeric identifier is assigned to the call and is inserted in the Call-ID field.</li> <li>• The transaction number within a single call leg is identified in the CSeq field.</li> <li>• The media capability A is ready to receive is specified.</li> <li>• The port on which the gateway is prepared to receive the RTP data is specified.</li> </ul>
3.	Call Proceeding—Gateway 1 to PBX A	Gateway 1 sends a Call Proceeding message to PBX A to acknowledge the call setup request.
4.	100 Trying—Cisco SIP IP phone to Gateway 1	The phone sends a SIP 100 Trying response to Gateway 1. The response indicates that the INVITE request has been received.

Step	Action	Description
5.	4xx/5xx/6xx Failure—Cisco SIP IP phone to Gateway 1	<p>The phone sends a class 4xx, 5xx, or class 6xx failure response to the gateway 1. Depending on which class the failure response is, the call actions differ.</p> <p>If the phone sends a class 4xx failure response (a definite failure response that is a client error), the request will not be retried without modification.</p> <p>If the phone sends a class 5xx failure response (an indefinite failure that is a server error), the request is not terminated, but rather other possible locations are tried.</p> <p>If the phone sends a class 6xx failure response (a global error), the search for B terminates because the 6xx response indicates that a server has definite information about B, but not for the particular instance indicated in the Request-URI field. Therefore, all further searches for this user will fail.</p>
6.	Disconnect—Gateway 1 to PBX A	Gateway 1 sends a Release message to PBX A.
7.	Release—PBX A to Gateway 1	PBX A sends a Release message to Gateway 1.
8.	ACK—Gateway 1 to Cisco SIP IP phone	Gateway 1 sends a SIP ACK to the phone. The ACK confirms that PBX A has received the 486 Busy Here response. The call session attempt terminates.
9.	Release Complete—Gateway 1 to PBX A	Gateway 1 sends a Release Complete message to PBX A, and the call session attempt terminates.

## Cisco SIP IP Phone to Cisco SIP IP Phone in a SIP Network

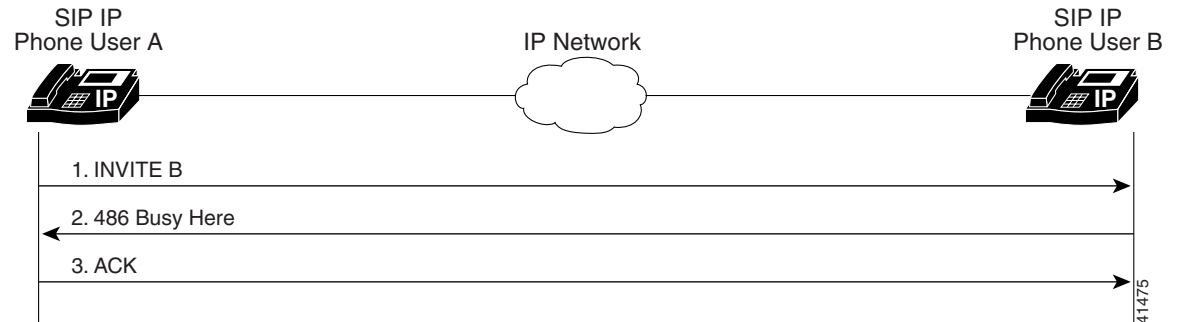
The following scenarios are Cisco SIP IP phone to Cisco SIP IP phone:

- Called Number Is Busy, page B-53
- Called Number Does Not Answer, page B-54
- Authentication Error, page B-54

## Called Number Is Busy

Figure B-21 illustrates an unsuccessful call in which User A initiates a call to User B, but User B is on the phone and is unable or unwilling to take another call.

**Figure B-21 Called Number Is Busy**

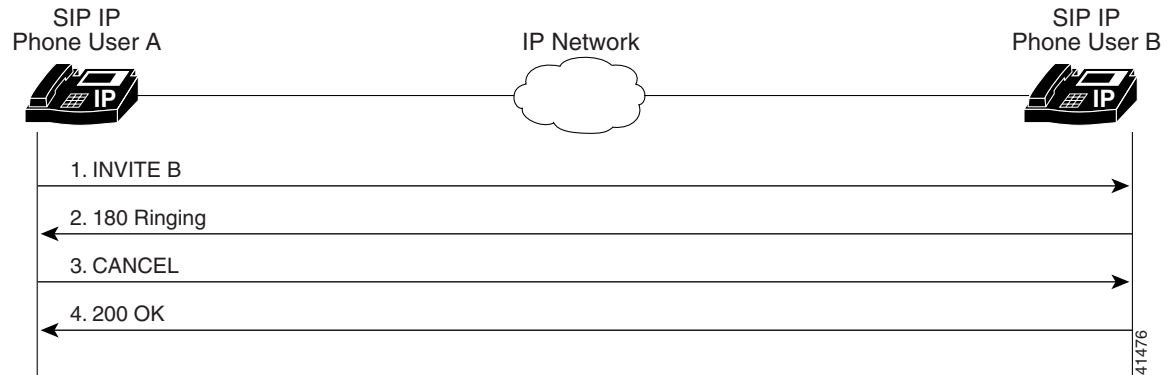


Step	Action	Description
1.	INVITE—Phone A to Phone B	Phone A sends a SIP INVITE request to Phone B. The request is an invitation to User B to participate in a call session. In the INVITE request: <ul style="list-style-type: none"> <li>• The phone number of User B is inserted in the Request-URI field in the form of a SIP URL. The SIP URL identifies the address of B and takes a form similar to an e-mail address (<i>user@host</i>, where <i>user</i> is the telephone number and <i>host</i> is either a domain name or a numeric network address). For example, the Request-URI field in the INVITE request to B appears as “INVITE sip:555-0199@companyb.com; user=phone.” The “user=phone” parameter distinguishes that the Request-URI address is a telephone number rather than a username.</li> <li>• Phone A is identified as the call session initiator in the From field.</li> <li>• A unique numeric identifier is assigned to the call and is inserted in the Call-ID field.</li> <li>• The transaction number within a single call leg is identified in the CSeq field.</li> <li>• The media capability A is ready to receive is specified.</li> </ul>
2.	486 Busy Here—Phone B to Phone A	Phone B sends a 486 Busy Here message to the Phone A. The message indicates that Phone B is in use and the user is not willing or able to take additional calls.
3.	ACK—Phone A to Phone B	Phone A sends a SIP ACK to the Phone B. The ACK confirms that Phone A has received the 486 Busy Here response from Phone B.

## Called Number Does Not Answer

Figure B-22 illustrates an unsuccessful call in which User A initiates a call to User B, but User B does not answer.

**Figure B-22 Called Number Does Not Answer**

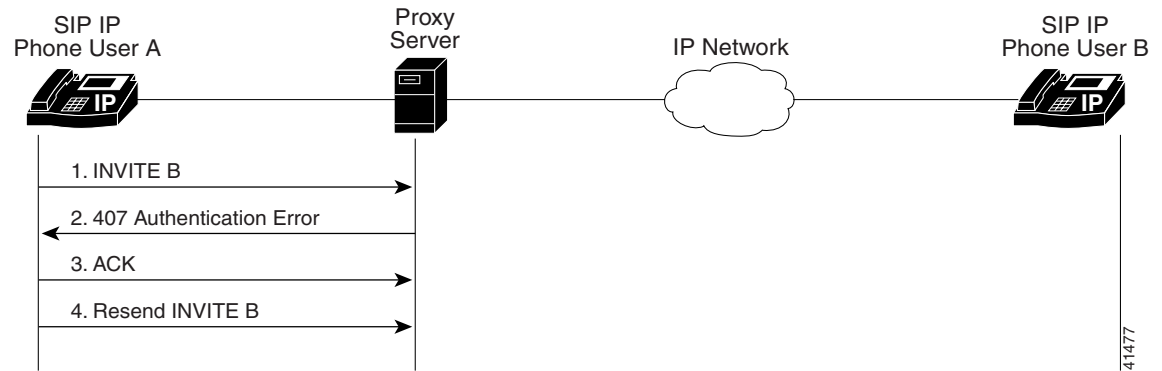


Step	Action	Description
1.	INVITE—Phone A to Phone B	Phone A sends a SIP INVITE request to Phone B. The request is an invitation to User B to participate in a call session. In the INVITE request: <ul style="list-style-type: none"> <li>• The phone number of B is inserted in the Request-URI field in the form of a SIP URL. The SIP URL identifies the address of B and takes a form similar to an e-mail address (<i>user@host</i>, where <i>user</i> is the telephone number and <i>host</i> is either a domain name or a numeric network address). For example, the Request-URI field in the INVITE request to B appears as “INVITE sip:555-0199@companyb.com; user=phone.” The “user=phone” parameter distinguishes that the Request-URI address is a telephone number rather than a username.</li> <li>• Phone A is identified as the call session initiator in the From field.</li> <li>• A unique numeric identifier is assigned to the call and is inserted in the Call-ID field.</li> <li>• The transaction number within a single call leg is identified in the CSeq field.</li> <li>• The media capability A is ready to receive is specified.</li> </ul>
2.	180 Ringing—Phone B to Phone A	Phone B sends a SIP 180 Ringing response to Phone A.
3.	CANCEL (Ring Timeout)—Phone A to Phone B	Phone A sends a CANCEL request to Phone B to cancel the invitation.
4.	200 OK—Phone B to Phone A	Phone B sends a SIP 200 OK response to Phone A. The response confirms receipt of the cancellation request.

## Authentication Error

Figure B-23 illustrates an unsuccessful call in which User A initiates a call to User B but is prompted for authentication credentials by the proxy server. User A’s SIP IP phone then reinitiates the call with a SIP INVITE request that includes authentication credentials.

Figure B-23 Authentication Error



41477

Step	Action	Description
1.	INVITE—Phone A to SIP proxy server	Phone A sends a SIP INVITE request to the SIP proxy server. The request is an invitation to B to participate in a call session. In the INVITE request: <ul style="list-style-type: none"> <li>• The phone number of B is inserted in the Request-URI field in the form of a SIP URL. The SIP URL identifies the address of User B and takes a form similar to an e-mail address (<i>user@host</i>, where <i>user</i> is the telephone number and <i>host</i> is either a domain name or a numeric network address). For example, the Request-URI field in the INVITE request to User B appears as “INVITE sip:555-0199@companyb.com; user=phone.” The “user=phone” parameter distinguishes that the Request-URI address is a telephone number rather than a username.</li> <li>• Phone A is identified as the call session initiator in the From field.</li> <li>• A unique numeric identifier is assigned to the call and is inserted in the Call-ID field.</li> <li>• The transaction number within a single call leg is identified in the CSeq field.</li> <li>• The media capability User A is ready to receive is specified.</li> </ul>
2.	407 Authentication Error—SIP proxy server to Phone A	The SIP proxy server sends a SIP 407 Authentication Error response to Phone A.
3.	ACK—Phone A to SIP proxy server	Phone A sends a SIP ACK to the SIP proxy server acknowledging the 407 error message.
4.	Resend INVITE—Cisco SIP IP phone A to SIP proxy server	Phone A resends a SIP INVITE to the SIP proxy server with authentication credentials.







## Technical Specifications of the Cisco IP 7960G/7940G Phone

The following sections describe the technical specifications for the Cisco IP 7960G/7940G phone:

- Physical and Operating Environment Specifications, page C-1
- Cable Specifications, page C-2
- Regulatory Safety Compliance, page C-2
- Connection Specifications, page C-2

### Physical and Operating Environment Specifications

Table C-1 lists the physical and operating specifications of the Cisco IP 7960G/7940G phone.

**Table C-1** Cisco IP 7960G/7940G Phone Operational and Physical Specifications

Specification	Value or Range
Operating temperature	32 to 104° F (0 to 40° C).
Operating relative humidity	10 to 95% (noncondensing).
Storage temperature	14 to 140° F (–10 to 60° C).
Height	8 in. (20.32 cm).
Width	10.5 in. (26.67 cm).
Depth	6 in. (15.24 cm).
Weight	3.5 lb (1.6 kg).
Power	100 to 240 VAC, 50 to 60 Hz, 0.5 A—when using the AC adapter. 48 VDC, 0.2 A—when using the in-line power over the network cable.
Cables	Two pair of Category 3 cables for 10 Mbps. Two pair of Category 5 cables for 100 Mbps.
Distance requirements	As supported by the Ethernet specification, it is assumed that most phones that are deployed in the field will be within 330 ft (100 m) of a phone closet.

## Cable Specifications

The following cables are required to connect the Cisco IP 7960G/7940G phone:

- RJ-11 for the handset connection.
- RJ-45 jack for the LAN connection (labeled “10/100 SW”).
- RJ-45 jack for a second 10BASE-T compliant connection (labeled “10/100 PC”).
- 48-volt power connector. The diameter of the center pin in the phone power jack (Switchcraft 712A) is 0.1 in (2.5 mm). The center pin is positive (+) voltage. The miniature power plug required to mate with the power jack on the phone is a Switchcraft 760 or equivalent.

## Regulatory Safety Compliance

For international agency compliance, safety, and statutory information on Cisco IP phone models in the 7900 series, refer to *Regulatory Compliance and Safety Information for the Cisco IP Phone 7900 Series* at [http://www.cisco.com/univercd/cc/td/doc/product/voice/c\\_ipphon/english/ipp7960/iphrcsi3.htm](http://www.cisco.com/univercd/cc/td/doc/product/voice/c_ipphon/english/ipp7960/iphrcsi3.htm).

## Connection Specifications

The Cisco IP 7960G/7940G phone has two RJ-45 ports that each support 10/100 Mbps half- or full-duplex connections to external devices—the network port and the access port. You can use either Category 3 or Category 5 cabling for 10-Mbps connections, but use Category 5 for 100-Mbps connections. On both the LAN-to-phone port (left RJ-45 port facing the back of the phone) and the PC-to-phone port (right port), use full-duplex to avoid collisions. Use the LAN-to-phone port to connect the phone to the network. Use the PC-to-phone port to connect a network device, such as a computer, to the phone.

For a diagram that identifies the different ports on the back of the Cisco IP 7960G/7940G phone, see the “Initializing Cisco SIP IP Phones” section on page 3-1.



## Configurable Parameters for the SIP IP Phone

This appendix describes configurable SIP parameters in the SIPDefault.cnf file and the SIP IP phone (see Table D-1). Parameters are in alphabetical order. Optional parameters are so noted.

**Table D-1 Configuration File Parameters**

Parameter	Description
anonymous_call_block	<p>(Optional) Configures anonymous call block.</p> <p>Valid values are as follows:</p> <ul style="list-style-type: none"> <li>• 0—Disabled by default, but can be turned on and off using the user interface. When disabled, anonymous calls are received.</li> <li>• 1—Enabled by default, but can be turned on and off using the user interface. When enabled, anonymous calls are rejected</li> <li>• 2—Disabled permanently and cannot be turned on and off locally using the user interface. Specify this parameter in the phone-specific configuration file.</li> <li>• 3—Enabled permanently and cannot be turned on and off locally using the user interface. Specify this parameter in the phone-specific configuration file.</li> </ul> <p>Default is 0.</p>
auto_answer	<p>(Optional) Configures the intercom functionality so that the user can define one or more of their lines for this feature. It is an integer field that represents a bit mask of on or off for each line key. The bit mask reads from the least significant bit of 0 equal to line 1 to the most significant bit of 5 for line 6.</p> <p>Valid values are as follows:</p> <ul style="list-style-type: none"> <li>• Cisco IP 7960 phone— 0 to 63</li> <li>• Cisco IP 7940 phone— 0 to 3</li> </ul> <p><b>Note</b> This parameter <i>cannot</i> be set in the configuration file.</p>
autocomplete	<p>(Phone-specific; optional) Configures automatic completion of numbers.</p> <p>Valid values are 0 (disable autocompletion) and 1 (enable autocompletion). Default is 1.</p>

Table D-1 Configuration File Parameters (continued)

Parameter	Description
call_hold_ringback	<p>(Phone-specific; optional) If you have a call on hold and are talking on another call, when you hang up the call, this parameter causes the phone to ring, letting you know that you still have another party on hold.</p> <p>Valid values are as follows:</p> <ul style="list-style-type: none"> <li>• 0—Off by default, but can be turned on and off locally using the user interface.</li> <li>• 1—On by default, but can be turned on and off locally using the user interface.</li> <li>• 2—Off permanently and cannot be turned on and off locally using the user interface. Specify this parameter in the phone-specific configuration file.</li> <li>• 3—On permanently and cannot be turned on and off locally using the user interface. Specify this parameter in the phone-specific configuration file.</li> </ul> <p>Default is 0.</p>
call_stats	<p>(Optional) Includes RTP statistics in BYE requests and responses.</p> <p>Valid values are 0 (disable) and 1 (enable). Default is 0.</p> <p>If this parameter is enabled, the phone inserts the headers RTP-RxStat and RTP-TxStat as follows:</p> <ul style="list-style-type: none"> <li>• RTP-RxStat: Dur=a,Pkt=b,Oct=c,LatePkt=d,LostPkt=e,AvgJit=f</li> <li>• RTP-TxStat: Dur=g,Pkt=h,Oct=i</li> </ul> <p>where the following apply:</p> <ul style="list-style-type: none"> <li>• Dur—Total number of seconds since the beginning of reception or transmission.</li> <li>• Pkt—Total number of RTP packets received or transmitted.</li> <li>• Oct—Total number of RTP payload octets received or transmitted (not including RTP header).</li> <li>• LatePkt—Total number of late RTP packets received.</li> <li>• LostPkt—Total number of lost RTP packets received (not including late RTP packets).</li> <li>• AvgJit—Average jitter, which is an estimate of the statistical variance of the RTP packet inter-arrival time, measured in timestamp unit and calculated according to RFC 1889.</li> <li>• a, b, c, d, e, f, g, h, and i—Integers.</li> </ul>

Table D-1 Configuration File Parameters (continued)

Parameter	Description
call_waiting	<p>(Phone-specific; optional) Configures call waiting.</p> <p>Valid values are as follows:</p> <ul style="list-style-type: none"> <li>• 0—Disabled by default, but can be turned on and off using the user interface. When disabled, call waiting calls are not received.</li> <li>• 1—Enabled by default, but can be turned on and off using the user interface. When enabled, call waiting calls are accepted.</li> <li>• 2—Disabled permanently and cannot be turned on and off locally using the user interface. Specify this parameter in the phone-specific configuration file.</li> <li>• 3—Enabled permanently and cannot be turned on and off locally using the user interface. Specify this parameter in the phone-specific configuration file.</li> </ul> <p>Default is 1.</p>
callerid_blocking	<p>(Phone-specific; optional) Configures caller ID blocking. When enabled, the phone blocks its own number or e-mail address from phones that have caller identification enabled.</p> <p>Valid values are as follows:</p> <ul style="list-style-type: none"> <li>• 0—Disabled by default, but can be turned on and off using the user interface. When disabled, the caller identification is included in the Request-URI header field.</li> <li>• 1—Enabled by default, but can be turned on and off using the user interface. When enabled, “Anonymous” is included in place of the user identification in the Request-URI header field.</li> <li>• 2—Disabled permanently and cannot be turned on and off locally using the user interface. Specify this parameter in the phone-specific configuration file.</li> <li>• 3—Enabled permanently and cannot be turned on and off locally using the user interface. Specify this parameter in the phone-specific configuration file.</li> </ul> <p>Default is 0.</p>
cfwd_url	<p>(Optional) Configures the call forwarding feature. The maximum allowable characters for the string is 128. The character can be a telephone number or a URL.</p> <p><b>Note</b> This parameter <i>cannot</i> be set in the configuration file.</p>
cnf_join_enable	<p>(Optional) Whether the conference bridge, when it hangs up, should attempt to join the two leaf nodes.</p> <p>Valid values are as follows:</p> <ul style="list-style-type: none"> <li>• 0—Do not join the two leaf nodes.</li> <li>• 1—Join the two leaf nodes.</li> </ul> <p>Default is 1.</p>

**Table D-1 Configuration File Parameters (continued)**

Parameter	Description
date_format	(Optional) Format for dates. Valid values are as follows: <ul style="list-style-type: none"> <li>• M/D/Y—Month/day/year</li> <li>• D/M/Y—Day/month/year</li> <li>• Y/M/D—Year/month/day</li> <li>• Y/D/M—Year/day/month</li> <li>• Y-M-D—Year-month-day</li> <li>• YY-M-D—4-digit year-month-day</li> </ul> Default is M/D/Y.
dial_template	Template with which you specify which file to download for your dial plan.
directory_url	(Optional) URL of the external directory server. This URL is accessed when you select Directory > External Directory. For example, use directory_url: “http://10.10.10.10/CiscoServices/Directory.asp”.
dnd_control	(Phone-specific; optional) Sets the Do Not Disturb (DND) feature. Valid values are as follows: <ul style="list-style-type: none"> <li>• 0—Off by default, but can be turned on and off locally using the user interface.</li> <li>• 1—On by default, but can be turned on and off locally using the user interface. The phone blocks all calls placed to the phone and logs those calls in the Missed Calls directory.</li> <li>• 2—Off permanently and cannot be turned on and off locally using the user interface. Specify this parameter in the phone-specific configuration file.</li> <li>• 3—On permanently and cannot be turned on and off locally using the user interface. This setting sets the phone to be a call-out phone only.</li> </ul> Default is 0.
Domain Name	Name of the DNS domain in which the phone resides.
dst_auto_adjust	(Optional) Whether daylight savings time (DST) is automatically adjusted on the phones. Valid values are 0 (do not adjust) and 1 (adjust). Default is 1.
dst_offset	(Optional) Offset from the phone time when DST is in effect. When DST is over, the specified offset is no longer applied to the phone time. Valid values are hour/minute, -hour/minute, +hour/minute, hour, -hour, and +hour.

**Table D-1 Configuration File Parameters (continued)**

Parameter	Description
dst_start_day	(Optional) Day of the month on which DST begins. Valid values are the following: <ul style="list-style-type: none"> <li>• 1 to 31 (for days of the month)</li> <li>• 0 (ignore this field and use the value in the dst_start_day_of_week parameter instead)</li> </ul> Default is 0.
dst_start_day_of_week	(Optional) Day of the week on which DST begins. Valid values are as follows: <ul style="list-style-type: none"> <li>• Any of the following: Sunday or Sun, Monday or Mon, Tuesday or Tue, Wednesday or Wed, Thursday or Thu, Friday or Fri, Saturday or Sat, Sunday or Sun</li> <li>• 1 to 7 (1 is Sunday; 7 is Saturday)</li> </ul> Name of the day is not case-sensitive. In the United States, the default is Sunday.
dst_start_month	(Optional) Month in which DST starts. Valid values are the following: <ul style="list-style-type: none"> <li>• January, February, March, April, May, June, July, August, September, October, November, and December</li> <li>• 1 to 12, with 1 being January and 12 being December</li> </ul> When the name of a month is specified, the value is not case-sensitive. In the United States, the default is April.
dst_start_time	(Optional) Time of day on which DST begins. Valid values are hour/minute (02/00) or hour:minute (02:00). In the United States, the default is 02:00.
dst_start_week_of_month	(Optional) Week of month in which DST begins. Valid values are 1 to 6 and 8 (1 is the first week; each subsequent number is a subsequent week; 8 is the last week in the month regardless of which week the last week is). In the United States, the default is 1.
dst_stop_day	(Optional) Day of the month on which DST ends. Valid values are as follows: <ul style="list-style-type: none"> <li>• 1 to 31 (for the days of the month)</li> <li>• 0 (ignore this field and use the value in the dst_stop_day_of_week parameter instead)</li> </ul> Default is 0.
dst_stop_day_of_week	(Optional) Day of the week on which DST ends. Valid values are Sunday or Sun, Monday or Mon, Tuesday or Tue, Wednesday or Wed, Thursday or Thu, Friday or Fri, Saturday or Sat, Sunday or Sun, and 1 to 7, with 1 being Sunday and 7 being Saturday. When the name of the day is specified, the value is not case-sensitive. In the United States, the default is Sunday.

**Table D-1 Configuration File Parameters (continued)**

Parameter	Description
dst_stop_month	(Optional) Month in which DST ends. Valid values are January, February, March, April, May, June, July, August, September, October, November, and December or 1 to 12 with 1 being January and 12 being December. When the name of a month is specified, the value is not case-sensitive. In the United States, the default is October.
dst_stop_time	(Optional) Time of day on which DST ends. Valid values are hour/minute (02/00) or hour:minute (02:00). In the United States, the default is 02:00.
dst_stop_week_of_month	(Optional) Week of month on which DST ends. Valid values are 1 to 6 and 8, with 1 being the first week, each number thereafter being subsequent weeks, and 8 being the last week in the month regardless of which week the last week is. In the United States, the default is 8.
dtmf_avt_payload	(Optional) Configures the payload type for Audio/Video Transport (AVT) packets. Range is from 96 to 127. If the value specified is null or invalid, default is 101.
dtmf_db_level	(Optional) In-band DTMF digit tone level. Valid values are as follows: <ul style="list-style-type: none"> <li>• 1—6 dB below nominal</li> <li>• 2—3 dB below nominal</li> <li>• 3—nominal</li> <li>• 4—3 dB above nominal</li> <li>• 5—6 dB above nominal</li> </ul> Default is 3.
dtmf_inband	(Optional) In-band signaling format. Valid values are 1 (generate DTMF digits in-band) and 0 (do not generate DTMF digits in-band). Default is 1.



**Table D-1 Configuration File Parameters (continued)**

Parameter	Description
dtmf_outofband	<p>(Optional) Configures the out-of-band signaling (for tone detection on the IP side of a gateway).</p> <p><b>Note</b> The Cisco SIP IP phone supports out-of-band signaling using the AVT tone method.</p> <p>Valid values are as follows:</p> <ul style="list-style-type: none"> <li>• none—Do not generate DTMF digits out-of-band.</li> <li>• avt—If requested by the remote side, generate DTMF digits out-of-band (and disable in-band DTMF signaling); otherwise, do not generate DTMF digits out-of-band.</li> <li>• avt_always—Always generate DTMF digits out-of-band. This option disables in-band DTMF signaling.</li> </ul> <p>Default is avt.</p>
dyn_dns_addr_1	<p>(Optional) IP address of a new dynamic DNS server. If a new DNS server address is specified, it is used for any further DNS requests after the phone uses the initial DNS address upon bootup. The DNS addresses are used in the following order:</p> <ol style="list-style-type: none"> <li>1. dyn_dns_addr_1 (if present)</li> <li>2. dyn_dns_addr_2 (if present)</li> <li>3. DNS Server 1</li> <li>4. DNS Server 2</li> <li>5. DNS Server 3</li> <li>6. DNS Server 4</li> <li>7. DNS Server 5</li> </ol> <p>The dynamic DNS address is not stored in flash memory. Only dotted IP addresses are accepted. This value can be cleared by removing it from the configuration file or by changing its value to a null value “ ” or to “UNPROVISIONED.”</p> <p><b>Note</b> The dynamic DNS address is not stored in flash memory.</p>
dyn_dns_addr_2	<p>(Optional) IP address of a second dynamic DNS to be used for DNS requests. See dyn_dns_addr_1 for more information.</p>
dyn_tftp_addr	<p>(Optional) IP address of a new dynamic TFTP server. After initially querying the default TFTP server, the phone rerequests the default and phone-specific configuration files from the new TFTP server. The dynamic TFTP server address is not stored in flash memory. The number of dyn_tftp_addr values supported by the phone is limited to prevent the phone configuration being downloaded repeatedly from multiple TFTP servers. Only dotted IP addresses are accepted. This value can be cleared by removing it from the configuration file or by changing its value to a null value “ ” or to “UNPROVISIONED.”</p>

**Table D-1 Configuration File Parameters (continued)**

Parameter	Description
enable_vad	(Optional) Enables voice activation detection (VAD). Valid values are 0 (disable) and 1 (enable). Default is 0.
end_media_port	(Optional) Configures the Real-Time Transport Protocol (RTP) end range for media. Valid values are 16384 to 32766. Default is 32766.
Host Name	Unique host name assigned to the phone. The value in this field is always SIP <i>mac</i> where <i>mac</i> is the MAC address of the phone.
http_proxy_addr	(Optional) IP address of the HTTP proxy server. You can use either a dotted IP address or a DNS name.
http_proxy_port	(Optional) Number of the HTTP proxy port. Default is 80.
image_version	Firmware version that the phone should use. Enter the name of the image version as it is released by Cisco. Do not enter the filename extension (.bin).  <b>Note</b> You cannot change the image version by changing the filename because the version is also built into the file header. Trying to change the image version by changing the filename causes the firmware to fail when it compares the version in the header against the filename.
language	(Optional) This parameter is for future use. English is the only value that is currently supported.
linex_authname	(Phone-specific; optional) Name used by the phone for authentication if a registration is challenged by the proxy server during initialization. It is required only if a proxy server requires authentication from phones. If a value is not configured for the linex_authname parameter when registration is enabled, the default name is used. Default is UNPROVISIONED. The <i>x</i> argument can be 1 or 2.
linex_displayname	(Phone-specific; optional) Identification as it should appear for caller-identification purposes. For example, instead of jdoe@company.com appearing on phones that have caller ID, you can specify User A in this parameter to have User A appear on the called party display. If a value is not specified for this parameter, nothing is used.  The <i>x</i> argument can be 1 or 2.
linex_name	(Phone-specific) Number or e-mail address for use when registering. Enter a number without dashes. For example, enter 555-0100 as 5550100. Enter an e-mail ID without the host name. The <i>x</i> argument can be 1 or 2.

**Table D-1 Configuration File Parameters (continued)**

Parameter	Description
linex_password	(Phone-specific; optional) Password used by the phone for authentication if a registration is challenged by the proxy server during initialization. If a value is not configured for the linex_password parameter when registration is enabled, the default password is used. Default is UNPROVISIONED.  Valid values for <i>x</i> are (Cisco IP 7969G phone) 1 to 6 and (Cisco IP 7940G phone) 1 to 2.
linex_shortcode	Labels a line key with a name other than the directory number.
local_cfwd_enable	Whether the phone can do local call forwarding. This is a boolean field; it is either enabled or disabled.  Valid values are 0 (disable) and 1 (enable). Default is 1.
logo_url	(Optional) Location of the company logo file. This logo appears on the phone display. The background space allocated for the image is 90 x 56 pixels. Images that are larger than this will automatically be scaled down to 90 x 56 pixels. The recommended file size for the image is from 5 to 15 Kb. For example, use logo_url: “http://10.10.10.10/companylogo.bmp”.  <b>Note</b> This parameter supports Windows 256 color bitmap format only. CMXML PhoneImage objects are not supported for this parameter. Using anything other than a Windows bitmap (.bmp) file can cause unpredictable results.
messages_uri	(Optional) Configures the voice-mail number that is dialed when the messages button is pressed. Value is typically a phone number but can be a URI.
mwi_status	(Optional) Displays the message waiting status.  <b>Note</b> You <i>cannot</i> set this parameter in the configuration file.
nat_address	(Optional) WAN IP address of the Network Address Translation (NAT) or firewall server. Value is either a dotted IP address or a DNS name.

Table D-1 Configuration File Parameters (continued)

Parameter	Description
nat_enable	<p>(Optional) Enables NAT.</p> <p>Valid values are 0 (disable) and 1 (enable). Default is 0.</p> <ul style="list-style-type: none"> <li>If NAT is enabled, the Contact header appears as follows:  <pre>Contact: sip:lineN_name@nat_address:voip_control_port</pre> <p>If the nat_address is invalid or UNPROVISIONED, the Contact header appears as follows:  <pre>Contact: sip:lineN_name@phone_ip_address:voip_control_port</pre> <p>and the Via header appears as follows:  <pre>Via: SIP/2.0/UDP phone_ip_address:voip_control_port</pre></p></p></li> <li>If NAT is enabled, the Session Description Protocol (SDP) message uses the nat_address and an RTP port between the start_media_port and the end_media_port range in the C and M fields. All RTP traffic is sourced from the port advertised in the SDP message.</li> </ul>
nat_received_processing	<p>(Optional) Enables NAT received processing.</p> <p>Valid values are 0 (disable) and 1 (enable). Default is 0.</p> <p>If nat_received_processing is enabled, and the received= tag is in the Via header of the 200 OK response from a REGISTER, the IP address in the received= tag is used instead of the nat_address in the Contact header. If this switch occurs, the phone unregisters the old IP address and reregisters with the new IP address.</p>
network_media_type	<p>(Optional) Ethernet port negotiation mode.</p> <p>Valid values are as follows:</p> <ul style="list-style-type: none"> <li>Auto—Port is autonegotiated.</li> <li>Full100—Port is configured to be a full-duplex, 100-MB connection.</li> <li>Half100—Port is configured to be a half-duplex, 100-MB connection.</li> <li>Full10—Port is configured to be a full-duplex, 10-MB connection.</li> <li>Half10—Port is configured to be a half-duplex, 10-MB connection.</li> </ul> <p>Default is Auto.</p>

**Table D-1 Configuration File Parameters (continued)**

Parameter	Description
network_port2_type	<p>(Optional) Configures the device type that is connected to port 2 of the phone.</p> <p>Valid values are Hub/Switch and PC. Default is Hub/Switch.</p> <p><b>Note</b> If the value is PC, port 2 can be connected only to a PC. If you are not sure about the connection, use the Hub/Switch default value. Specifying the PC option and then connecting port 2 to a switch results in spanning-tree loops and network confusion.</p>
outbound_proxy	(Optional) IP address of the outbound proxy server. You can use either a dotted IP address or a DNS name.
outbound_proxy_port	<p>(Optional) Port number of the outbound proxy server. Default is 5060.</p> <p>When an outbound proxy is enabled, all SIP requests are sent to the outbound proxy server instead of to the proxy<sub>x</sub>_address. All responses continue to reconcile the normal Via processing rules. The media stream is not routed through the outbound proxy.</p> <p>NAT and outbound proxy modes can be independently enabled or disabled. The received= tag is added to the Via header of all responses if there is no received= tag in the uppermost Via header and if the source IP address is different from the IP address in the uppermost Via header. Keep the following rules in mind:</p> <ul style="list-style-type: none"> <li>• If a received= tag is in the uppermost Via header, the response is sent back to the IP address contained in the received= tag.</li> <li>• If there is no received= tag and the IP address in the uppermost Via header is different from the source IP address, the response is sent back to the source IP address.</li> </ul>
phone_label	(Phone-specific; optional) Text to display on the top right status line of the LCD. This field is for end-user display only and has no effect on caller identification or messaging. For example, a phone label can display “User A’s phone.” Limited to 11 characters.
phone_password	(Phone-specific; optional) Password to be used for console or Telnet access. Limited to 31 characters. Default is cisco.
phone_prompt	(Phone-specific; optional) Prompt to display during Telnet or console access. Limited to 15 characters. Default is SIP Phone.
preferred_codec	<p>(Optional) Codec to use when a call is initiated.</p> <p>Valid values are g711alaw, g711ulaw, g729a, and none. Default is g711ulaw.</p>
proxy_backup	<p>(Optional) IP address of the backup proxy server or gateway. Enter this address in IP dotted-decimal notation.</p> <p><b>Note</b> You must specify at least one address or the phones cannot register.</p>
proxy_backup_port	(Optional) Port number of the backup proxy server. Default is 5060.

**Table D-1 Configuration File Parameters (continued)**

Parameter	Description
proxy_emergency	(Optional) IP address of the emergency proxy server or gateway. Enter this address in IP dotted-decimal notation.
proxy_emergency_port	(Optional) Port number of the emergency proxy server. Default is 5060.
proxy_register	<p>(Optional) The phone must register with a proxy server during initialization.</p> <p>Valid values are 0 (disable registration during initialization) and 1 (enable registration during initialization). Default is 0.</p> <p><b>Note</b> You can also use this parameter in a phone-specific configuration file. After a phone has initialized and registered with a proxy server, you can remove the registration by changing this value to 0 in the phone-specific configuration file. To reinitiate registration, change the value back to 1.</p> <p><b>Note</b> If you enable registration and authentication is required, you must specify values for the <code>linex_authname</code> and <code>linex_password</code> parameters in the phone-specific configuration file.</p>
proxyx_address	<p>IP address of the SIP proxy servers that are used by the phones. Enter the addresses in IP dotted-decimal notation or use the FQDN. The “x” argument is representative of server addresses.</p> <p>Valid values for “x” are 1 to 6. If the <code>proxyx_address</code> parameter is provisioned with an FQDN, the phone sends REGISTER and INVITE messages by using the FQDN in the Req-URI, To, and From fields.</p> <p>If the value of x is not specified in the <code>proxyx_address</code> parameter, the phone uses <code>proxy1_address</code> as the default value.</p>
proxyx_port	<p>Port number of the SIP proxy server that will be used by phone lines other than line 1. The x variable represents a phone line.</p> <p>Valid values are 2 to 6.</p> <p><b>Note</b> For additional phone lines, the <code>proxyx_port</code> parameter and the <code>proxyx_port</code> parameter can be used to assign different proxy addresses to different phone lines. The “x” in the parameters represents a phone line. The value of “x” can be from 1 to 6. If the value of the “x” is not specified in the <code>proxyx_address</code> parameter, the phone uses <code>proxy1_address</code> as the default.</p>

Table D-1 Configuration File Parameters (continued)

Parameter	Description
remote_party_id	<p>(Optional) The Remote-Party-ID header supports network verification and screening of a call participant's identity (for example, name and number) and provides privacy for call participants.</p> <p>Valid values are as follows:</p> <ul style="list-style-type: none"> <li>0—Remote party ID is disabled. The phone does not send or accept the remote party ID.</li> <li>1—Remote party ID is enabled. The phone sends the remote party ID, and can accept the remote party ID.</li> </ul> <p>Default is 0.</p>
semi_attended_transfer	<p>(Optional) Whether or not the caller can transfer the second leg of an attended transfer while the call is ringing.</p> <p>Valid values are as follows:</p> <ul style="list-style-type: none"> <li>0—Semi-attended transfer is disabled.</li> <li>1—Semi-attended transfer is enabled.</li> </ul> <p>Default is 1.</p>
services_url	<p>(Optional) URL of the services BTXML files. This URL is accessed when the Services button is pressed. For example, use services_url: "http://10.10.10.10/CiscoServices/Services.asp."</p>
sip_invite_retx	<p>(Optional) Maximum number of times that an INVITE request will be retransmitted.</p> <p>Valid value is any positive integer. Default is 6.</p>
sip_retx	<p>(Optional) Maximum number of times that a SIP message other than an INVITE request will be retransmitted.</p> <p>Valid value is any positive integer. Default is 10.</p>
sntp_mode	<p>(Optional) Mode in which the phone listens for the SNTP server.</p> <p>Valid values are unicast, multicast, anycast, or directedbroadcast. Default is anycast</p>
sntp_server	<p>IP address of the SNTP server from which the phone obtains time data.</p>
speed_labelx	<p>(Optional) Configures the speed-dial key label. The <i>x</i> variable is from label 2 to label 6. There are five possible labels that can be configured on the Cisco IP 7960G but only one on the Cisco IP 7940G. The <i>x</i> variable is a string of up to 15 characters.</p> <p><b>Note</b> This parameter <i>cannot</i> be set in the configuration file.</p>
speed_linex	<p>(Optional) Configures the speed-dial keys so that the user can set up one-touch dialing. There are five possible numbers that can be configured on the Cisco IP 7960G but only one on the Cisco IP 7940G. The <i>x</i> variable is a string of up to 128 bytes.</p> <p><b>Note</b> You <i>cannot</i> set this parameter in the configuration file.</p>

**Table D-1 Configuration File Parameters (continued)**

Parameter	Description
start_media_port	(Optional) Start RTP range for media. Range is from 16384 to 32766. Default is 16384.
stutter_msg_waiting	(Optional) Enables a stutter dial tone when there is a message waiting. It is disabled by default. Valid values are 0 (off) and 1 (on).
sync	(Optional) Value against which to compare the value in the syncinfo.xml file before a remote reboot is performed. Limited to 32 characters.
telnet_level	(Optional) Enables Telnet for the phone. Valid values are as follows: <ul style="list-style-type: none"> <li>• 0—Disabled.</li> <li>• 1—Enabled, and no privileged commands can be executed.</li> <li>• 2—Enabled, and privileged commands can be executed.</li> </ul> Default is 0.
tftp_cfg_dir	Path to the TFTP subdirectory in which phone-specific configuration files are stored. <b>Note</b> This parameter is only required if the phone-specific files are in a subdirectory and not in the root directory.
time_format_24hr	(Optional) Whether a 12- or 24-hour time format is displayed by default on the user interface. Valid values are as follows: <ul style="list-style-type: none"> <li>• 0—12-hour format is displayed by default but can be changed to a 24-hour format using the user interface.</li> <li>• 1—24-hour format is displayed by default but can be changed to a 12-hour format using the user interface.</li> <li>• 2—12-hour format is displayed and cannot be changed to a 24-hour format using the user interface.</li> <li>• 3—24-hour format is displayed and cannot be changed to a 12-hour format using the user interface.</li> </ul> Default is 1.
time_zone	(Optional) Time zone in which the phone is located. Valid values are the time-zone abbreviations shown in Table 3-5 on page 3-20. Abbreviations are case sensitive and must be in all capital letters. Default is PST.
timer_invite_expires	(Optional) Amount of time, in seconds, after which a SIP INVITE expires. This value is used in the Expire header field. Valid values are any positive number; however, we recommend 180. Default is 180.



**Table D-1 Configuration File Parameters (continued)**

Parameter	Description
timer_register_delta	Configures the time interval at which reregistration will occur. This is a numeric field in which the time interval is measured in seconds.  Valid values range from 32767 to 0. Default is 5 (phone will attempt to reregister 5 seconds before its registration period expires).
timer_register_expires	(Optional) Amount of time, in seconds, after which a REGISTRATION request expires. This value is inserted into the Expire header field.  Valid values are any positive number; however, we recommend 3600. Default is 3600.
timer_t1	(Optional) Lowest value, in milliseconds, of the retransmission timer for SIP messages.  Valid values are any positive integer. Default is 500.
timer_t2	(Optional) Highest value, in milliseconds, of the retransmission timer for SIP messages.  Valid values are any positive integer greater than timer_t1. Default is 4000.
tos_media	(Optional) Type of service (ToS) level for the media stream being used.  Valid values are as follows: <ul style="list-style-type: none"> <li>• 0—IP_ROUTINE</li> <li>• 1—IP_PRIORITY</li> <li>• 2—IP_IMMEDIATE</li> <li>• 3—IP_FLASH</li> <li>• 4—IP_OVERRIDE</li> <li>• 5—IP_CRITIC</li> </ul> Default is 5.
user_info	(Phone-specific; optional) Configures the “user=” parameter in the REGISTER message.  Valid values are as follows: <ul style="list-style-type: none"> <li>• none—No value is inserted.</li> <li>• phone—The value user=phone is inserted in the To, From, and Contact Headers for REGISTER.</li> <li>• ip—The value user=ip is inserted in the To, From, and Contact Headers for REGISTER.</li> </ul> Default is none.
voip_control_port	(Optional) UDP port used for SIP messages. All SIP REQUESTS use voip_control_port as the UDP source port when nat_enable = 1.  Range is from 1025 to 65535. Default is 5060.

Table 2 Network Parameters

Parameter	Description
Admin.VLAN Id	<p>Unique identifier of the VLAN to which the phone is attached, for use in switched networks that are not Cisco networks.</p> <p><b>Note</b> If you have an administrative VLAN setting assigned on the Catalyst switch, that setting overrides any changes made on the phone.</p>
Alternate TFTP	<p>Whether to use an alternate remote TFTP server rather than the local one.</p> <p>Valid values are Yes and No. If you set this parameter to Yes, you must change the IP address in the TFTP Server parameter to the address of the alternate TFTP server. Default is No.</p>
Default Router 1 to 5	<p>IP address (1) of the default gateway used by the phone and (2 to 5) of the gateways that the phone attempts to use as an alternate gateway if the default gateway is unavailable.</p> <p><b>Note</b> Default Router 1 always takes precedence. If Router 1 is unavailable, the phone moves down the list of available routers in order based on the configuration in Routers 2 through 5.</p> <p><b>Note</b> If you have an administrative VLAN setting assigned on the Catalyst switch, that setting overrides any changes made on the phone.</p>
DHCP Address Released	<p>Whether the IP address of the phone can be released for reuse in the network.</p> <p>Valid values are Yes and No. When set to Yes, the phone sends a DHCP release message to the DHCP server and goes into a release state. The release state provides enough time to remove the phone from the network before the phone attempts to acquire another IP address from the DHCP server. When you move the phone to a new network segment, first release the DHCP address.</p>
DHCP Enabled	<p>Whether the phone uses DHCP to configure network settings (IP address, subnet mask, domain name, default router list, DNS server list, and TFTP address).</p> <p>Valid values are Yes and No. Default is Yes. To manually configure your IP settings, you must set this parameter to No.</p>

**Table 2** Network Parameters (continued)

Parameter	Description
DHCP Server	IP address of the DHCP server from which the phone received its IP address and additional network settings.
DNS Servers 1 to 5	IP address of the DNS server used by the phone to resolve names to IP addresses. The phone attempts to use DNS servers 2 to 5 if DNS server 1 is unavailable. <b>Note</b> DHCP must be disabled.
Dynamic TFTP Server	IP address of a dynamic TFTP server. After initially querying the default TFTP server, the phone rerequests the default and phone-specific configuration files from the new TFTP server. The dynamic TFTP server address is not stored in flash memory.
Erase Configuration	Whether to erase all of the locally defined network settings on the phone and reset the values to the defaults. Valid values are Yes and No. Yes reenables DHCP.
HTTP Proxy Address	IP address of the HTTP proxy server. You can use either a dotted IP address or a DNS name (a record only).
HTTP Proxy Port	Number of the outbound proxy port. Default is 80.
IP Address	IP address of the phone that is assigned by DHCP or is locally configured. <b>Note</b> DHCP must be disabled.
MAC Address	Factory-assigned unique 48-bit hexadecimal MAC address of the phone.
Operational VLAN Id	Unique identifier of the VLAN of which the phone is a member. This identifier is obtained through Cisco Discovery Protocol (CDP).
Subnet Mask	IP subnet mask used by the phone. A subnet mask partitions the IP address into a network and a host identifier. <b>Note</b> DHCP must be disabled.
TFTP Server	IP address of the TFTP server. <b>Note</b> If you have an administrative VLAN setting assigned on the Catalyst switch, that setting overrides any changes made on the phone. <b>Note</b> DHCP must be disabled.

**Table 3 SIP Parameters**

Parameter	Description
Authentication Name	(Phone-specific) Name used by the phone for authentication if a registration is challenged by the proxy server during initialization.  <b>Note</b> Required when registration is enabled and the registrar challenges registration.
Authentication Password	(Phone-specific) Password used by the phone for authentication if a registration is challenged by the proxy server during initialization. If a value is not configured for the Authentication Password parameter when registration is enabled, the default logical password is used. The default logical password is <i>SIPmac-address</i> , where <i>mac-address</i> is the MAC address of the phone.  <b>Note</b> Required when registration is enabled and the registrar challenges registration.
Display Name	(Phone-specific) Identification as it should appear for caller identification. For example, instead of <i>jdoe@company.com</i> appearing on phones that have caller ID, you can specify <i>User A</i> in this parameter to have <i>User A</i> appear on the callee end instead. If a value is not specified for this parameter, the <i>Name</i> value is used.
Name	(Phone-specific) Description phone number or e-mail address used when registering. When entering a number, enter the number without any dashes. For example, enter 555-0100 as 5550100. When entering an e-mail address, enter the e-mail ID without the host name.
Proxy Address	(Phone-specific) IP address of the primary SIP proxy server that will be used by the phone. Enter this address in IP dotted-decimal notation.

**Table 3** *SIP Parameters (continued)*

<b>Parameter</b>	<b>Description</b>
Proxy Port	(Phone-specific) Port number of the primary SIP proxy server. This is the port that the SIP client will use. The default is 5060.
Short Name	(Phone-specific) Name or number associated with the <code>linex_name</code> as you want it to display on the phone LCD if the <code>linex_name</code> value exceeds the display area. For example, if the <code>linex_name</code> value is the phone number 111-222-333-4444, you can specify 34444 for this parameter to have 34444 display on the LCD instead. Alternatively, if the value for the <code>linex_name</code> parameter is the e-mail address “username@company.com”, you can specify the “username” to have just the username appear on the LCD instead. This parameter is used for display only. If a value is not specified for this parameter, the value in the Name variable is displayed.

