



## **Dial Peer Configuration Guide, Cisco IOS XE Release 3S**

### **Americas 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: 408 527-0883

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 NON-INFRINGEMENT 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.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at [www.cisco.com/go/offices](http://www.cisco.com/go/offices).

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <https://www.cisco.com/c/en/us/about/legal/trademarks.html>. Third-party trademarks mentioned 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. (1721R)

© 2021 Cisco Systems, Inc. All rights reserved.



## CONTENTS

---

### CHAPTER 1

#### **Dial Peer Overview 1**

- Call Legs 1
- POTS Dial Peers 4
- Voice-Network Dial Peers 4
- Data Dial Peers 5
- Creating a Dial Peer Configuration Table 5
- Codecs 6
  - Clear Channel (G.Clear) Codec 7
  - Adaptive Differential PCM Voice Codec G.726 8
  - iLBC Codec 8
- Toll Fraud Prevention 9

---

### CHAPTER 2

#### **Dial Planning 11**

- Fixed-Length Dial Plans 11
- Variable-Length Dial Plans 12

---

### CHAPTER 3

#### **Modes for Dial-Peer Configuration 15**

- How to Configure Modes for Dial-Peer Configuration 15
  - Entering Dial-Peer Configuration Mode 15
  - Entering Global VoIP Configuration Mode 16
  - Using Voice Classes to Modify Dial-Peer Behavior 17

---

### CHAPTER 4

#### **Dial Peer Features and Configuration 19**

- Common Practices 19
  - Voice Ports 19
    - Assigning Voice Ports 19

Session Targets	20
Configuring Session Targets	21
Destination Patterns	22
Configuring Destination Patterns	22
Digit Manipulation	23
Data Dial Peers	38
Configuring Data Dial Peers	38
Configuring a Search for Dial Peers by Type	39
Inbound and Outbound Dial Peers	40
Matching Inbound Dial Peers	41
Variable-Length Matching	42
Configuring the incoming called-number Command	43
answer-address Command	44
Configuring the destination-pattern Command	45
Configuring the port Command	45
Matching Outbound Dial Peers	45
Using Default Routes	45
Additional Features	46
One Stage and Two Stage Dialing	46
Direct Inward Dialing	47
Hunt Groups	49
Configuring Dial Peer Hunting Options	51
Configuring Dial Peer Hunting Options	52
Modem Pass Through	53
Configuring Modem Pass Through Capability for Individual Dial Peers	54
Dual Tone Multifrequency Relay	56
Configuring DTMF Relay and Payload Type	56
Connection PLAR	58
Configuring Connection PLAR	58
Connection PLAR Design Considerations	59
Connection Trunk	60
Configuring Connection Trunk	60
Class of Restrictions	62
Verifying Classes of Restrictions	64

Configuring an iLBC Codec	65
Configuring an iLBC Codec on a Dial Peer	65
Configuring an iLBC Codec in the Voice Class	67
Where to Go for Dial Peer Troubleshooting Information	68

**CHAPTER 5****PSTN Fallback 69**

Information About PSTN Fallback	69
Service Assurance Agent	70
Application of PSTN Fallback	70
Restrictions for PSTN Fallback	71
How to Configure PSTN Fallback	71
Configuring Call Fallback to Use MD5 Authentication for SAA Probes	71
Configuring Destination Monitoring without Fallback to Alternate Dial Peers	72
Configuring Call Fallback Cache Parameters	72
Configuring Call Fallback Jitter-Probe Parameters	73
Configuring Call Fallback Probe-Timeout and Weight Parameters	75
Configuring Call Fallback Threshold Parameters	75
Configuring Call Fallback Wait-Timeout	76
Configuring VoIP Alternate Path Fallback SNMP Trap	77
What to Do Next	78
Configuring Call Fallback Map Parameters	78
Configuring ICMP Pings to Monitor IP Destinations	79
Dial Peer Configuration of the call fallback icmp-ping and monitor probe Commands	79
Global Configuration of the call fallback icmp-ping Command	81
Voice Port Configuration of the busyout monitor probe icmp-ping Command	82
Voice Class Configuration of the busyout monitor probe icmp-ping Command	83
How to Verify and Monitor the PSTN Fallback Feature	84
Verifying PSTN Fallback Configuration	84
Monitoring and Maintaining PSTN Fallback	84
What To Do Next	84

**CHAPTER 6****Dial Peer Configuration Examples 87**

Two Analog Phones	87
Both Connected to the Same Voice Gateway Router	87

Each Connected to Their Own Voice Gateway Routers Using the G.711 Codec 88

Each Connected to Their Own Voice Gateway Routers Using the G.729r8 Codec 89

Two Fax Machines 89

An Analog Phone and an IP Phone Connected over an IP Network 90

Two IP Phones Connected via a Voice over Frame Relay Network 91

Using Digit Manipulation to Overcome the Obstacle of an IP Network Failure 93

---

**CHAPTER 7**

**Matching Inbound Dial Peers by URI 97**

Configuring an Inbound Dial Peer to Match on URI 97

Examples for Configuring an Inbound Dial Peer to Match on a URI 99

---

**CHAPTER 8**

**Multiple Pattern Support on a Voice Dial Peer 101**

Feature Information for Multiple Pattern Support on a Voice Dial Peer 101

Restrictions for Multiple Pattern Support on a Voice Dial Peer 102

Information About Multiple Pattern Support on a Voice Dial Peer 102

Configuring Multiple Pattern Support on a Voice Dial Peer 102

Verifying Multiple Pattern Support on a Voice Dial Peer 104

Configuration Examples for Multiple Pattern Support on a Voice Dial Peer 106

---

**CHAPTER 9**

**Outbound Dial-Peer Group as an Inbound Dial-Peer Destination 107**

Feature Information for Outbound Dial-Peer Group as an Inbound Dial-Peer Destination 107

Restrictions 108

Information About Outbound Dial-Peer Group as an Inbound Dial-Peer Destination 108

Configuring Outbound Dial-Peer Group as an Inbound Dial-Peer Destination 109

Verifying Outbound Dial-Peer Groups as an Inbound Dial-Peer Destination 111

Troubleshooting Tips 112

Configuration Examples for Outbound Dial Peer Group as an Inbound Dial-Peer Destination 113

---

**CHAPTER 10**

**Inbound Leg Headers for Outbound Dial-Peer Matching 117**

Feature Information for Inbound Leg Headers for Outbound Dial-Peer Matching 117

Prerequisites for Inbound Leg Headers for Outbound Dial-Peer Matching 118

Restrictions for Inbound Leg Headers for Outbound Dial-Peer Matching 118

Information About Inbound Leg Headers for Outbound Dial-Peer Matching 119

Configuring Inbound Leg Headers for Outbound Dial-Peer Matching 119

Verifying Inbound Leg Headers for Outbound Dial-Peer Matching	122
Configuration Example: Inbound Leg Headers for Outbound Dial-Peer Matching	124

---

**CHAPTER 11****Server Groups in Outbound Dial Peers 127**

Feature Information for Configuring Server Groups in Outbound Dial Peers	127
Information About Server Groups in Outbound Dial Peers	128
How to Configure Server Groups in Outbound Dial Peers	129
Configuring Server Groups in Outbound Dial Peers	129
Verifying Server Groups in Outbound Dial Peers	131
Configuration Examples for Server Groups in Outbound Dial Peers	133







# CHAPTER 1

## Dial Peer Overview

Configuring dial peers is the key to implementing dial plans and providing voice services over an IP packet network. Dial peers are used to identify call source and destination endpoints and to define the characteristics applied to each call leg in the call connection.

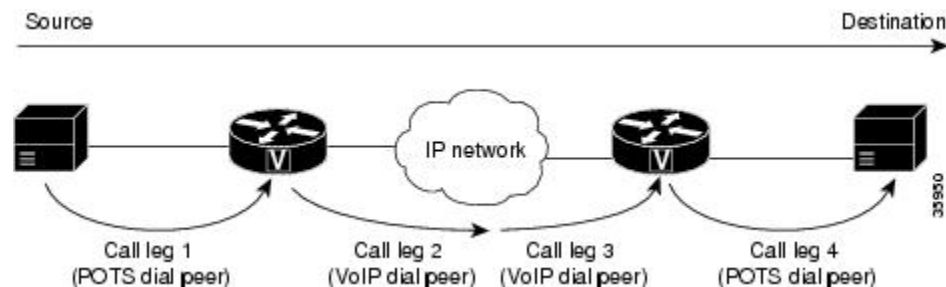
This chapter contains the following sections:

- [Call Legs, on page 1](#)
- [POTS Dial Peers, on page 4](#)
- [Voice-Network Dial Peers, on page 4](#)
- [Data Dial Peers, on page 5](#)
- [Creating a Dial Peer Configuration Table, on page 5](#)
- [Codecs, on page 6](#)
- [Toll Fraud Prevention, on page 9](#)

## Call Legs

A traditional voice call over the public switched telephone network (PSTN) uses a dedicated 64K circuit end to end. In contrast, a voice call over the packet network is made up of discrete segments or call legs. A call leg is a logical connection between two routers or between a router and a telephony device. A voice call comprises four call legs, two from the perspective of the originating router and two from the perspective of the terminating router, as shown in the figure below.

**Figure 1: Dial Peer Call Legs**



A dial peer is associated with each call leg. Attributes that are defined in a dial peer and applied to the call leg include the codec, quality of service (QoS), voice activity detection (VAD), and fax rate. To complete a voice call, you must configure a dial peer for each of the four call legs in the call connection.

Depending on the call leg, a call is routed using one of the two types of dial peers:

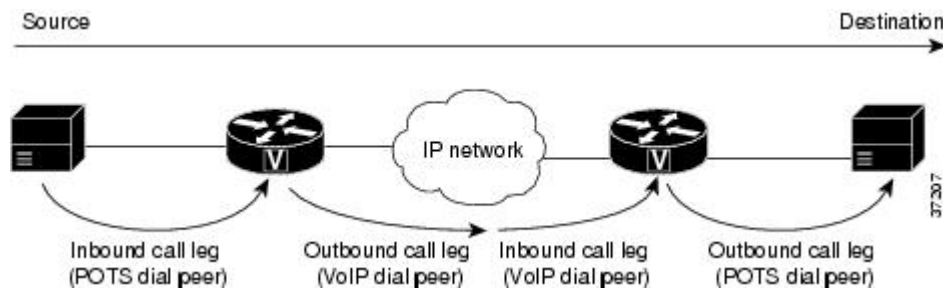
- Plain old telephone system (POTS)--Dial peer that defines the characteristics of a traditional telephony network connection. POTS dial peers map a dialed string to a specific voice port on the local router, normally the voice port connecting the router to the local PSTN, PBX, or telephone.
- Voice-network--Dial peer that defines the characteristics of a packet network connection. Voice-network dial peers map a dialed string to a remote network device, such as the destination router that is connected to the remote telephony device.

Both POTS and voice-network dial peers are needed to establish voice connections over a packet network.

When a voice call comes into the router, the router must match dial peers to route the call. For inbound calls from a POTS interface that are being sent over the packet network, the router matches a POTS dial peer for the inbound call leg and a voice-network dial peer for the outbound call leg. For calls coming into the router from the packet network, the router matches an outbound POTS dial peer to terminate the call and an inbound voice-network dial peer for features such as codec, VAD, and QoS.

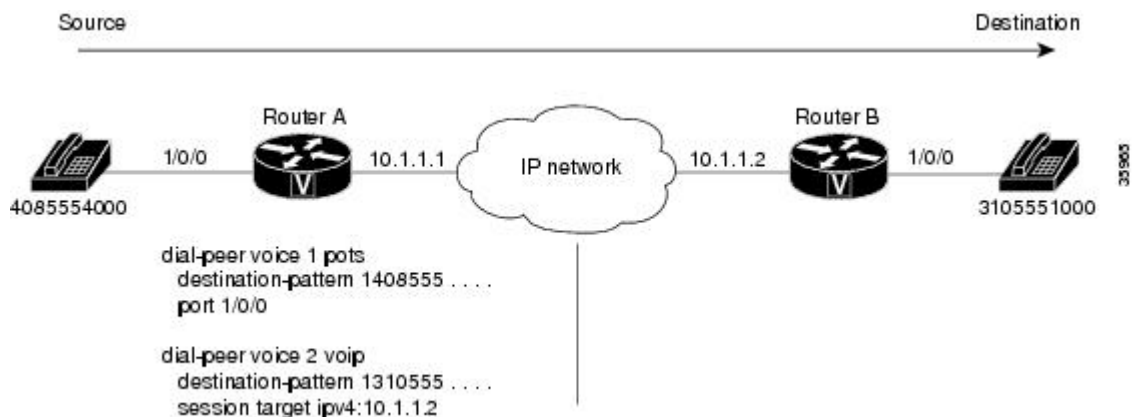
The figure below shows the call legs and associated dial peers necessary to complete a voice call.

**Figure 2: Matching Call Legs to Dial Peers**



The following configurations show an example of a call being made from 4085554000 to 3105551000. The figure below shows the inbound POTS dial peer and the outbound voice over IP (VoIP) dial peer that are configured on the originating router. The POTS dial peer establishes the source of the call (via the calling number or voice port), and the voice-network dial peer establishes the destination by associating the dialed number with the network address of the remote router.

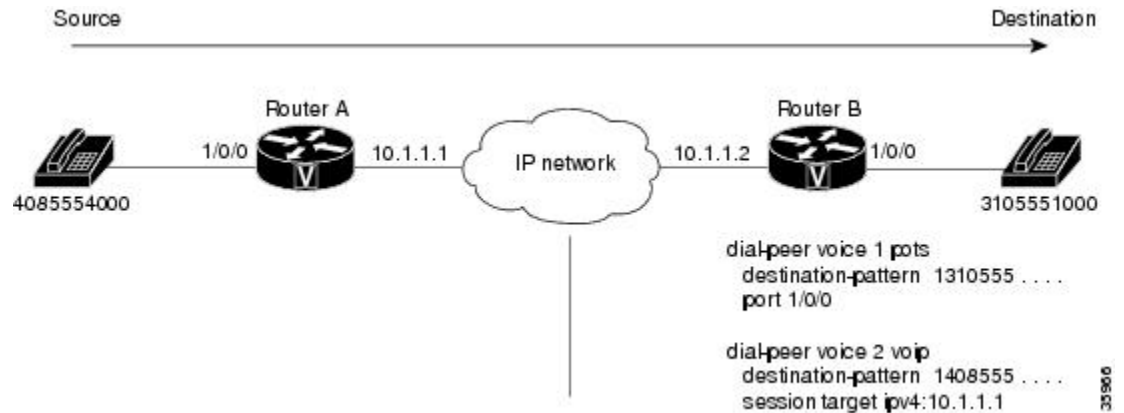
**Figure 3: Dial Peers from the Perspective of the Originating Router**



In this example, the dial string 14085554000 maps to telephone number 555-4000, with the digit 1 plus the area code 408 preceding the number. When you configure the destination pattern, set the string to match the local dialing conventions.

The figure below shows the inbound VoIP dial peer and outbound POTS dial peer that are configured on the terminating router to complete the call. Dial peers are of local significance only.

**Figure 4: Dial Peers from the Perspective of the Terminating Router**



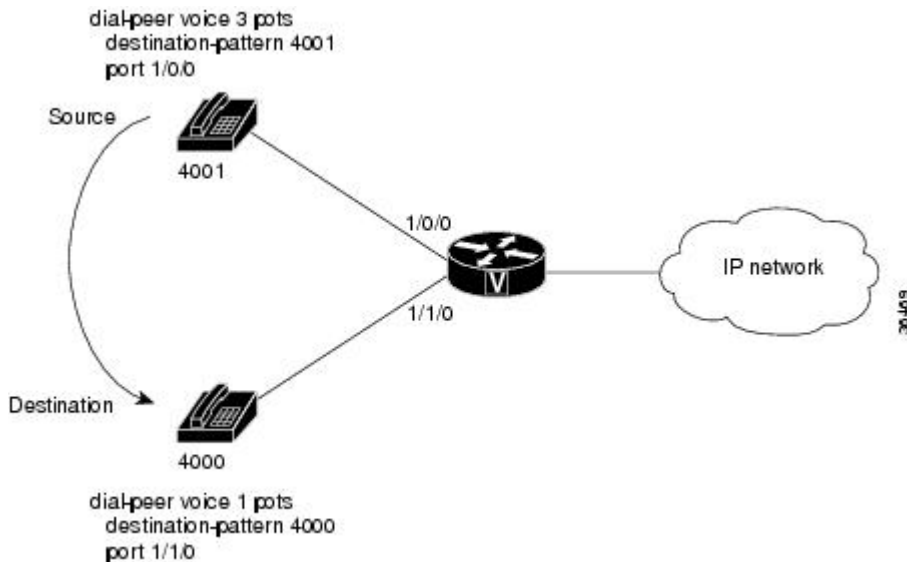
In the previous configuration examples, the last four digits in the VoIP dial peer's destination pattern were replaced with wildcards. Which means that from Router A, calling any telephone number that begins with the digits "1310555" will result in a connection to Router B. This behavior implies that Router B services all numbers beginning with those digits. From Router B, calling any telephone number that begins with the digits "1408555" will result in a connection to Router A. This behavior implies that Router A services all numbers beginning with those digits.



**Note** It is not always necessary to configure the inbound dial peers. If the router is unable to match a configured dial peer for the inbound call leg, it uses an internally defined default POTS or voice-network dial peer to match inbound voice calls. In the example shown in the figure above, dial peer 2 is required only when making a call from Router B to Router A.

The only exception to the previous example occurs when both POTS dial peers share the same router, as shown in the figure below. In this circumstance, you do not need to configure a voice-network dial peer.

Figure 5: Communication Between Dial Peers Sharing the Same Router



This type of configuration is similar to the configuration used for hairpinning, which occurs when a voice call destined for the packet network is instead routed back over the PSTN because the packet network is unavailable.

## POTS Dial Peers

POTS dial peers retain the characteristics of a traditional telephony network connection. POTS dial peers map a dialed string to a specific voice port on the local router, normally the voice port connecting the router to the local PSTN, PBX, or telephone.

## Voice-Network Dial Peers

Voice-network dial peers are components on an IP network to which a voice gateway router points via the component's IP address specified in the **session-target** command for a particular matching dial peer. The four types of voice-network dial peers (VoIP, voice over ATM (VoATM), voice over Frame Relay (VoFR), and multimedia mail over IP (MMoIP)) are determined according to the given packet network technology and are described as follows:

- VoIP--Points to the IP address of the destination router that terminates the call.
- VoFR--Points to the data-link connection identifier (DLCI) of the interface from which the call exits the router.
- VoATM--Points to the ATM virtual circuit for the interface from which the call exits the router.
- MMoIP--Points to the e-mail address of the simple mail transfer protocol (SMTP) server. This type of dial peer is used only for fax traffic.

## Data Dial Peers

Before Cisco IOS Release 12.2(11)T, a Cisco voice gateway would try to match a voice dial peer before matching and processing a modem call. If a voice dial peer was matched, the call was processed as voice. If there was no voice dial peer match, only then was a call considered to be a modem call. Voice calls always received preference over modem calls. Also, there was no way to assign a subset of addresses in the numbering plan for data calls.

In Cisco IOS Release 12.2(11)T, an interim solution in the form of application called "data\_dialpeer" was introduced to enable gateways to identify dial peers. The application enabled the handling of certain matched calls as modem calls. Refer to the Fine-Grain Address Segmentation in Dial Peers feature documentation in Cisco IOS Release 12.2(11)T for more information.

In Cisco IOS Release 12.2(13)T, formal support for data dial peers was released in the form of the Dial-Peer Support for Data Calls feature, which enables the configuration and order assignment of dial peers so that the gateway can identify incoming calls as voice or data (modem). You can use the **dial-peer data** and **dial-peer search** commands to perform this configuration. Refer to the "Data Dial Peers" section on page 33 for configuration steps and examples.

## Creating a Dial Peer Configuration Table

Before you can configure dial peers, you must obtain specific information about your network. One way to identify this information is to create a dial peer configuration table. This table should contain all the telephone numbers and access codes for each router that is carrying telephone traffic in the network. Because most installations require integrating equipment into an existing voice network, the telephone dial plans are usually preset.

The figure below shows an example of a network in which Router A, with an IP address of 10.1.1.1, connects a small sales branch office to the main office through Router B, with an IP address of 10.1.1.2.

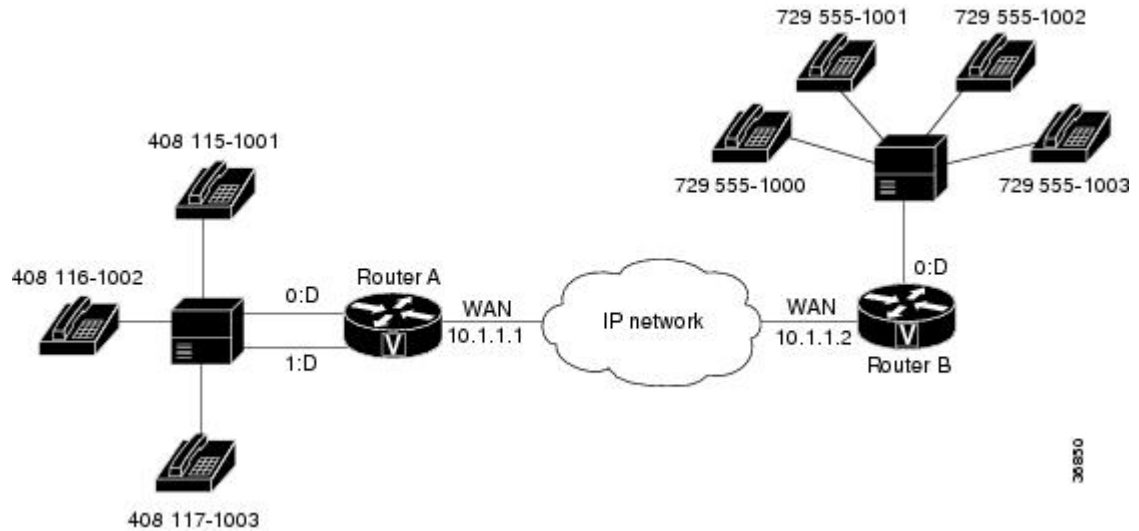


---

**Note** The example in the figure below shows a VoIP configuration. The same concepts also apply to VoFR and VoATM applications. The only change is in the format of the session target.

---

Figure 6: Sample VoIP Network



Three telephone numbers in the sales branch office need dial peers configured for them. Router B is the primary gateway to the main office; as such, it needs to be connected to the company's PBX. Four devices need dial peers, all of which are connected to the PBX, configured for them in the main office.

The table below shows the peer configuration table for the example in the figure above.

Table 1: Dial Peer Configuration Table for Sample Voice over IP Network

Dial Peer	Extension	Prefix	Destination Pattern	Type	Voice Port	Session Target
Router A						
1	51001	5	1408115....	POTS	0:D	--
2	61002	6	1408116....	POTS	0:D	--
3	71003	7	1408117....	POTS	0:D	--
10	--	--	1729555....	VoIP	--	10.1.1.2
Router B						
1	1000, 1001, 1002, 1003	--	1729555....	POTS	0:D	--
10	--	--	1408.....	VoIP	--	10.1.1.1

## Codecs

The term *codec* stands for *coder-decoder*. A codec is a particular method of transforming analog voice into a digital bit stream (and vice versa) and also refers to the type of compression used. Several different codecs have been developed to perform these functions, and each one is known by the number of the International Telecommunication Union Telecommunication Standardization Sector (ITU-T) standard in which it is defined. For example, two common codecs are the G.711 and the G.729 codecs.

Codecs use different algorithms to encode analog voice into digital bit streams and have different bit rates, frame sizes, and coding delays associated with them. Codecs also differ in the amount of perceived voice quality they achieve. Specialized hardware and software in the digital signal processors (DSPs) perform codec transformation and compression functions, and different DSPs may offer different selections of codecs.

Select the same type of codec at both ends of the call. For instance, if a call was coded with a G.729 codec, it must be decoded with a G.729 codec. Codec choice is configured on dial peers.

The table below lists the H.323, SIP, and MGCP codecs that are supported for voice.

**Table 2: Voice Codec/Signaling Support Matrix**

Codec	H.323	SIP	MGCP
g711ulaw	Yes	Yes	Yes
g711alaw	Yes	Yes	Yes
g729r8 <sup>1</sup>	Yes	Yes	Yes
g729br81	Yes	Yes	Yes
g723ar53	Yes	Yes	Yes
g723ar63	Yes	Yes	Yes
g723r53	Yes	Yes	Yes
g723r63	Yes	Yes	Yes
g726r16 <sup>2</sup>	Yes	Yes	Yes
g726r242	Yes	Yes	Yes
g726r32	Yes	Yes	Yes
clear-channel2	Yes	Yes	Yes
iLBC	Yes	Yes	No

<sup>1</sup> Annex A is used in the Cisco platforms that are supported in this software release.

<sup>2</sup> For dynamic payload types.

For more information, refer to the "Dial Planning" chapter in this document and see the Cisco IOS Voice Port Configuration Guide.

## Clear Channel (G.Clear) Codec

G.Clear guarantees bit integrity when transferring a DS-0 through a gateway server, supports the transporting of nonvoice circuit data sessions through a Voice over IP (VoIP) network, and enables the VoIP networks to transport ISDN and switched 56 circuit-switched data calls. With the availability of G.Clear, ISDN data calls that do not require bonding can be supported.

In a transit application, because it is possible to have a mix of voice and data calls, not supporting G.Clear limits the solution to voice-only calls. The end-user application is in charge of handling packet loss and error

recovery. This packet loss management precludes the use of clear channel with some applications unless the IP network is carefully engineered.

In an MGCP environment, the voice gateway backhauls the public switched telephony network (PSTN) signaling channel to the call agent. The call agent examines the bearer capability and determines when a G.Clear call should be established.



---

**Note** G.Clear codecs cannot be configured on a T1 channel associated signaling (CAS) trunk for incoming traffic. T1 CAS trunks use least significant bit-robbing for signaling, which causes the data to be incorrect and re-sent from high level protocols. Traffic on an incoming E1 R2 trunk can be configured.

---

## Adaptive Differential PCM Voice Codec G.726

Adaptive differential pulse code modulation (ADPCM) voice codec operates at bit rates of 16, 24, and 32 kbps. ADPCM provides the following functionality:

- Voice mail recording and playback, which is a requirement for Internet voice mail.
- Voice transport for cellular, wireless, and cable markets.
- High voice quality voice transport at 32 kbps.

## iLBC Codec

The internet Low Bitrate Codec (iLBC) Has the following benefits:

- Is designed specifically for packet-based communication.
- Is royalty free.
- Provides high-voice quality, even in conditions with high-packet loss.
- Has a sampling rate of 8 kHz, for narrow band speech.
- Supports two fixed bit-rate frame lengths:
  - Bit-rate of 13.3 kbps with an encoding frame length of 30 ms
  - Bit-rate of 15.2 kbps with an encoding frame length of 20 ms
- Is designed to be robust, even with packet loss. iLBC treats each packet independently and recovers from packet loss on the packet immediately following the lost one. By utilizing the entire available frequency band, this codec provides a high voice quality.

### Platforms that iLBC Supports

iLBC is supported on Cisco AS5350XM and Cisco AS5400XM Universal Gateways with Voice Feature Cards (VFCs) and IP-to-IP gateways with no transcoding and conferencing.

### Using iLBC with SIP

- For iLBC codecs using SIP, use RFC3952 as a reference for implementation.



- Mid-call codec parameter changes using SIP are supported. For example, iLBC 'mode' and 'ptime' changes are supported using SIP during the call.

### Using iLBC with H.323

- For iLBC codecs using H.323, a new proposal is written and submitted for approval to ITU. The new proposal is added as 'Annex S' in H245, Version 12 which is used as reference for implementation.

See H245, version 12 document at <http://www.packetizer.com/ipmc/h245/>

- Mid-call codec parameter changes using H.323 are not supported.

## Toll Fraud Prevention

When a Cisco router platform is installed with a voice-capable Cisco IOS software image, appropriate features must be enabled on the platform to prevent potential toll fraud exploitation by unauthorized users. Deploy these features on all Cisco router Unified Communications applications that process voice calls, such as Cisco Unified Communications Manager Express (CME), Cisco Survivable Remote Site Telephony (SRST), Cisco Unified Border Element (UBE), Cisco IOS-based router and standalone analog and digital PBX and public-switched telephone network (PSTN) gateways, and Cisco contact-center VoiceXML gateways. These features include, but are not limited to, the following:

- Disable secondary dial tone on voice ports--By default, secondary dial tone is presented on voice ports on Cisco router gateways. Use private line automatic ringdown (PLAR) for foreign exchange office (FXO) ports and direct-inward-dial (DID) for T1/E1 ports to prevent secondary dial tone from being presented to inbound callers.
- Cisco router access control lists (ACLs)--Define ACLs to allow only explicitly valid sources of calls to the router or gateway, and therefore to prevent unauthorized Session Initiation Protocol (SIP) or H.323 calls from unknown parties to be processed and connected by the router or gateway.
- Close unused SIP and H.323 ports--If either the SIP or H.323 protocol is not used in your deployment, close the associated protocol ports. If a Cisco voice gateway has dial peers configured to route calls outbound to the PSTN using either time division multiplex (TDM) trunks or IP, close the unused H.323 or SIP ports so that calls from unauthorized endpoints cannot connect calls. If the protocols are used and the ports must remain open, use ACLs to limit access to legitimate sources.
- Change SIP port 5060--If SIP is actively used, consider changing the port to something other than well-known port 5060.
- SIP registration--If SIP registration is available on SIP trunks, turn on this feature because it provides an extra level of authentication and validation that only legitimate sources can connect calls. If it is not available, ensure that the appropriate ACLs are in place.
- SIP Digest Authentication--If the SIP Digest Authentication feature is available for either registrations or invites, turn this feature on because it provides an extra level of authentication and validation that only legitimate sources can connect calls.
- Explicit incoming and outgoing dial peers--Use explicit dial peers to control the types and parameters of calls allowed by the router, especially in IP-to-IP connections used on CME, SRST, and Cisco UBE. Incoming dial peers offer additional control on the sources of calls, and outgoing dial peers on the

destinations. Incoming dial peers are always used for calls. If a dial peer is not explicitly defined, the implicit dial peer 0 is used to allow all calls.

- Explicit destination patterns--Use dial peers with more granularity than .T for destination patterns to block disallowed off-net call destinations. Use class of restriction (COR) on dial peers with specific destination patterns to allow even more granular control of calls to different destinations on the PSTN.
- Translation rules--Use translation rules to manipulate dialed digits before calls connect to the PSTN to provide better control over who may dial PSTN destinations. Legitimate users dial an access code and an augmented number for PSTN for certain PSTN (for example, international) locations.
- Tcl and VoiceXML scripts--Attach a Tcl/VoiceXML script to dial peers to do database lookups or additional off-router authorization checks to allow or deny call flows based on origination or destination numbers. Tcl/VoiceXML scripts can also be used to add a prefix to inbound DID calls. If the prefix plus DID matches internal extensions, then the call is completed. Otherwise, a prompt can be played to the caller that an invalid number has been dialed.
- Host name validation--Use the "permit hostname" feature to validate initial SIP Invites that contain a fully qualified domain name (FQDN) host name in the Request Uniform Resource Identifier (Request URI) against a configured list of legitimate source hostnames.
- Dynamic Domain Name Service (DNS)--If you are using DNS as the "session target" on dial peers, the actual IP address destination of call connections can vary from one call to the next. Use voice source groups and ACLs to restrict the valid address ranges expected in DNS responses (which are used subsequently for call setup destinations).

For more configuration guidance, see the [Cisco IOS Unified Communications Toll Fraud Prevention](#) paper.



## CHAPTER 2

# Dial Planning

A dial plan essentially describes the number and pattern of digits that a user dials to reach a particular telephone number. Access codes, area codes, specialized codes, and combinations of the number of digits dialed are all part of a dial plan. For instance, the North American public switched telephone network (PSTN) uses a 10-digit dial plan that includes a 3-digit area code and a 7-digit telephone number. Most PBXs support variable length dial plans that use 3 to 11 digits. Dial plans must comply with the telephone networks to which they connect. Only totally private voice networks that are not linked to the PSTN or to other PBXs can use any dial plan they choose.

Dial plans on Cisco routers are manually defined using dial peers. Dial peers are similar to static routes; they define where calls originate and terminate and what path the calls take through the network. Attributes within the dial peer configuration determine which dialed digits the router collects and forwards to telephony devices. Dial peer configuration allows you to implement both fixed- and variable-length dial plans for your existing voice network and enables you to adjust to future scalability needs that may arise as your voice network expands or contracts.

This chapter contains the following sections:

- [Fixed-Length Dial Plans, on page 11](#)
- [Variable-Length Dial Plans, on page 12](#)

## Fixed-Length Dial Plans

Fixed-length dialing plans, in which all the dial peer destination patterns have a fixed length, are sufficient for most voice networks because the telephone number strings are of known lengths. Some voice networks, however, require variable-length dial plans, particularly for international calls, which use telephone numbers of different lengths.

If you enter the timeout T-indicator at the end of the destination pattern in an outbound voice-network dial peer, the router accepts a fixed-length dial string and then waits for additional dialed digits. The timeout character must be an uppercase T. The following dial peer configuration shows how the T-indicator is set to allow variable-length dial strings:

```
dial-peer voice 1 voip
destination-pattern 2222T
session target ipv4:10.10.1.1
```

In the example, the router accepts the digits 2222, and then waits for an unspecified number of additional digits. The router can collect up to 31 additional digits, as long as the interdigit timeout has not expired. When the interdigit timeout expires, the router places the call.

The default value for the interdigit timeout is 10 seconds. Unless the default value is changed, using the T-indicator adds 10 seconds to each call setup because the call is not attempted until the timer has expired (unless the # character is used as a terminator). You should therefore reduce the voice-port interdigit timeout value if you use variable-length dial plans. You can change the interdigit timeout by using the **timeouts inter-digit** command in voice-port configuration mode.

The calling party can immediately terminate the interdigit timeout by entering the # character. If the # character is entered while the router is waiting for additional digits, the # character is treated as a terminator; it is not treated as part of the dial string or sent across the network. But if the # character is entered before the router begins waiting for additional digits (meaning that the # is entered as part of the fixed-length destination pattern), then the # character is treated as a dialed digit.

For example, if the destination pattern is configured as 2222...T, then the entire dialed string of 2222#9999 is collected, but if the dialed string is 2222#99#99, the #99 at the end of the dialed digits is not collected because the final # character is treated as a terminator. You can change the termination character by using the **dial-peer terminator** command.


**Note**

In most cases, you must configure the T-indicator only when the router uses two-stage dialing. If direct inward dialing (DID) is configured in the inbound plain old telephone system (POTS) dial peer, the router uses one-stage dialing, which means that the full dialed string is used to match outbound dial peers. The only exception is when the **isdn overlap-receiving** command is configured; the ISDN overlap-receiving feature requires the T-indicator.

## Variable-Length Dial Plans

In most voice configurations, fixed-length dialing plans, in which all the dial peer destination patterns have the same length, are sufficient because the telephone number strings are all the same length. However, in some voice network configurations, variable-length dial plans are required, especially if the network connects two or more countries where telephone number strings could be different lengths.

If you enter the "T" timer character in the destination pattern for your dial peer, the router can be configured to accept a fixed-length dial string, and then wait for additional dialed digits. For example, the following dial peer configuration shows how the T character can be set to allow variable-length dial strings:

```
dial peer voice 1 pots
destination-pattern 2222T
port 1/1
```

In this example, the router accepts the digits 2222, and then waits for an unspecified number of dialed digits. If digits continue to be entered before the interdigit timeout expires, then the router will continue to gather up to 31 additional digits. Once the interdigit timeout expires, however, the router places the call. You can configure the interdigit timeout value by using the **timeouts inter-digit** command in voice-port configuration mode.

The interdigit timeout timer can be terminated by entering the "#" character. If the # character is entered while the router is waiting to accept additional digits, the # character is treated as an end-dial accelerator. The # character is not treated as an actual digit in the destination pattern and is not sent as part of the dialed string across the network.

However, if the # character is entered before the router is ready to accept additional digits (meaning before the "T" character is entered in the destination pattern), then the # character is treated as a dialed digit. For

example, if a destination pattern is configured with the string 2222...T, then the digits 2222#####1234567 can be gathered, but the digits 2222###1234#67 cannot be gathered because the final # character is treated as a terminator.

The default value for the interdigit timeout is 10 seconds. If the duration is not changed, using the "T" timer adds 10 seconds to each call setup time because the call is not attempted until the timer expires (unless the # character is used as a terminator). Because of this dependency, if a variable-length dial plan is used, the interdigit timeout should be reduced to reduce the call setup time. For more information, refer to the "Variable-Length Matching" section on page 37 .





## CHAPTER 3

# Modes for Dial-Peer Configuration

You can manipulate a dial peer by configuring relevant commands globally, at a dial-peer level, or in a voice class that is then applied to a dial peer.

- [How to Configure Modes for Dial-Peer Configuration, on page 15](#)

## How to Configure Modes for Dial-Peer Configuration

### Entering Dial-Peer Configuration Mode

#### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dial-peer voice *number* voip**
4. Enter a command to configure dial-peer behavior.
5. **exit**

#### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>enable</b> <b>Example:</b>  Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	<b>configure terminal</b> <b>Example:</b>  Device# configure terminal	Enters global configuration mode.
Step 3	<b>dial-peer voice <i>number</i> voip</b> <b>Example:</b>	Enters dial-peer configuration mode for the specified VoIP dial peer.

	Command or Action	Purpose
	Device(config)# dial-peer voice 2 voip	
<b>Step 4</b>	Enter a command to configure dial-peer behavior.	
<b>Step 5</b>	<b>exit</b> <b>Example:</b> Device(config-dial-peer)# exit	Exits dial-peer configuration mode and returns to privileged EXEC mode.

## Entering Global VoIP Configuration Mode

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice service voip**
4. Enter a command to configure dial-peer behavior.
5. **exit**

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> Device# configure terminal	Enters global configuration mode.
<b>Step 3</b>	<b>voice service voip</b> <b>Example:</b> Device(config)# voice service voip	Enters global VoIP configuration mode.
<b>Step 4</b>	Enter a command to configure dial-peer behavior.	
<b>Step 5</b>	<b>exit</b> <b>Example:</b> Device(config-voi-serv)# exit	Exits global VoIP configuration current mode.



## Using Voice Classes to Modify Dial-Peer Behavior

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice class media tag**
4. Enter a command to configure dial-peer behavior.
5. **dial-peer voice tag voip**
6. **voice class media tag**
7. **exit**

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b>  Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b>  Device# configure terminal	Enters global configuration mode.
<b>Step 3</b>	<b>voice class media tag</b> <b>Example:</b>  Device(config)# voice class media 1	Enters voice-class configuration mode and assign an identification tag for a media voice class.
<b>Step 4</b>	Enter a command to configure dial-peer behavior.	
<b>Step 5</b>	<b>dial-peer voice tag voip</b> <b>Example:</b>  Device(config-class)# dial-peer voice 2 voip	Enters dial-peer configuration mode for the dial peer of your choice. You will apply the voice class on this dial peer.
<b>Step 6</b>	<b>voice class media tag</b> <b>Example:</b>  Device(config-dial-peer)# voice class media 1	Applies the voice class to this dial peer. The behavior defined in the voice class is now applicable to the dial peer.
<b>Step 7</b>	<b>exit</b> <b>Example:</b>  Device(config-dial-peer)# exit	Exit dial-peer configuration mode and returns to privileged EXEC mode.





## CHAPTER 4

# Dial Peer Features and Configuration

Establishing voice communication over a packet network is similar to configuring a static route: You are establishing a specific voice connection between two defined endpoints. Call legs define the discrete segments that lie between two points in the call connection. A voice call over the packet network comprises four call legs, two on the originating router and two on the terminating router; a dial peer is associated with each of these four call legs.



**Note** The example configurations in this section show voice over IP (VoIP) dial peers; the same concepts also apply to voice over Frame Relay (VoFR) and voice over ATM (VoATM) dial peers.

- [Common Practices, on page 19](#)
- [Data Dial Peers, on page 38](#)
- [Inbound and Outbound Dial Peers, on page 40](#)
- [Additional Features, on page 46](#)
- [Where to Go for Dial Peer Troubleshooting Information, on page 68](#)

## Common Practices

The following three sections cover the bare essential configuration steps necessary to support voice transmission and reception on a typical voice gateway router in your network:

## Voice Ports

Your dial peer configuration cannot function until you have logically assigned a voice port to one or more dial peers. Assigning voice ports to dial peers identifies the physical hardware in the router that will be employed to complete voice communication to and from associated voice network endpoints.

## Assigning Voice Ports

The purpose of this task is to assign a voice port to a plain old telephone system (POTS) dial peer.

### SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **dial-peer voice** *number* **pots**
4. **port** *string*

## DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> Router> enable	Enters privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> Router# configure terminal	Enters global configuration mode.
<b>Step 3</b>	<b>dial-peer voice</b> <i>number</i> <b>pots</b> <b>Example:</b> Router(config)# dial-peer voice 864 pots	Enters dial-peer voice configuration mode and defines a local POTS dial peer. <ul style="list-style-type: none"> <li>• The <i>number</i> argument identifies the dial peer. Valid entries are from 1 to 2147483647.</li> </ul>
<b>Step 4</b>	<b>port</b> <i>string</i> <b>Example:</b> Router(config-dialpeer)# port 1/0:0	Specifies the voice port associated with the given dial peer. The <b>port</b> command syntax is platform-specific. For more information about the syntax of this command, refer to the <b>port</b> command in the <i>Cisco IOS Voice, Video, and Fax Command Reference</i> .

### What to do next

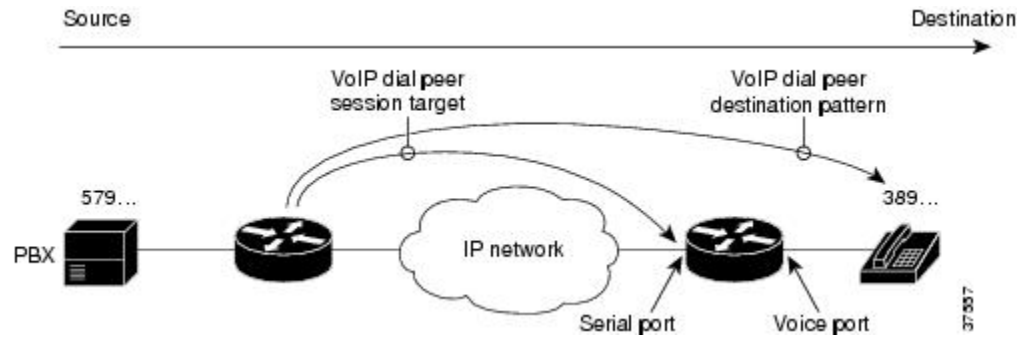


**Note** Voice port assignments are configured for POTS dial peers only.

## Session Targets

The session target is the network address of the remote router to which you want to send a call once a local voice-network dial peer is matched. It is configured in voice-network dial peers by using the **session target** command. For outbound dial peers, the destination pattern is the telephone number of the remote voice device that you want to reach. The session target represents the path to the remote router that is connected to that voice device. The figure below illustrates the relationship between the destination pattern and the session target, as shown from the perspective of the originating router.

Figure 7: Relationship Between Destination Pattern and Session Target



The address format of the session target depends on the type of voice-network dial peer:

- VoIP--IP address, host name of the Domain Name System (DNS) server that resolves the IP address, **ras** for registration, admission, and status (RAS) if an H.323 gatekeeper resolves the IP address, or **settlement** if the settlement server resolves the IP address
- VoFR--Interface type and number and the data link connection identifier (DLCI)
- VoATM--Interface number, and ATM virtual circuit
- MMoIP--E-mail address



**Note** For inbound dial peers, the session target is ignored.

## Configuring Session Targets

The purpose of this task is to assign a session target to a voice-network dial peer.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dial-peer voice** *number* **voip** | **vofr** | **voatm**
4. **session-target** *ip-address*

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>enable</b> <b>Example:</b> Router> enable	Enters privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
Step 2	<b>configure terminal</b> <b>Example:</b>	Enters global configuration mode.

	Command or Action	Purpose
	Router# configure terminal	
<b>Step 3</b>	<b>dial-peer voice</b> <i>number</i> <b>voip</b>   <b>vofr</b>   <b>voatm</b> <b>Example:</b> Router(config)# dial-peer voice 864 voip	Enters dial-peer voice configuration mode and defines a local dial peer. <ul style="list-style-type: none"> <li>The <i>number</i> argument identifies the dial peer. Valid entries are from 1 to 2147483647.</li> </ul>
<b>Step 4</b>	<b>session-target</b> <i>ip-address</i> <b>Example:</b> Router(config-dialpeer)# session-target 10.45.44.43	Defines the IP address identifying the next-hop location of the voice network component associated with this dial peer.

## Destination Patterns

The destination pattern associates a dialed string with a specific telephony device. It is configured in a dial peer by using the **destination-pattern** command. If the dialed string matches the destination pattern, the call is routed according to the voice port in POTS dial peers, or the session target in voice-network dial peers. For outbound voice-network dial peers, the destination pattern may also determine the dialed digits that the router collects and then forwards to the remote telephony interface, such as a PBX, a telephone, or the public switched telephone network (PSTN). You must configure a destination pattern for each POTS and voice-network dial peer that you define on the router.

## Configuring Destination Patterns

The purpose of this task is to configure a destination pattern for a dial peer.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dial-peer voice** *number* **pots** | **voip** | **vofr** | **voatm**
4. **destination-pattern** + ] *string*[T]

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> Router> enable	Enters privileged EXEC mode. <ul style="list-style-type: none"> <li>Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> Router# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	<b>dial-peer voice</b> <i>number</i> <b>pots</b>   <b>voip</b>   <b>vofr</b>   <b>voatm</b> <b>Example:</b> <pre>Router(config)# dial-peer voice 123 voip</pre>	Enters dial-peer voice configuration mode and defines a local dial peer. <ul style="list-style-type: none"> <li>• The <i>number</i> argument identifies the dial peer. Valid entries are from 1-to 2147483647.</li> </ul>
Step 4	<b>destination-pattern</b> + ] <i>string</i> [T] <b>Example:</b> <pre>Router(config-dialpeer)# destination-pattern 5551234</pre>	Defines the phone number that identifies the destination pattern associated with this dial peer. The keywords and argument are as follows: <ul style="list-style-type: none"> <li>• + --(Optional) Character indicating an E.164 standard number.               <p><b>Note</b> With CSCub65380, behavior of dial peers with destination-patterns configured with + symbol was rectified. The + symbol is no longer dropped from the dial peer and matching occurs as expected.</p> </li> <li>• <i>string</i> --A series of digits specifying the E.164 or private dial plan phone number. Valid entries are as follows:               <ul style="list-style-type: none"> <li>• Digits 0 through 9, letters A through F, pound sign (#), and asterisk (*), which represent specific digits that can be entered.</li> <li>• Comma (,), which inserts a pause between digits.</li> <li>• Period (.), which matches any entered digit.</li> </ul> </li> <li>• T --(Optional) Control character indicating that the <b>answer-address</b> value is a variable-length dial string.</li> </ul>

## Digit Manipulation

The router may need to manipulate digits in a dial string before it passes the dial string to the telephony device. Which can be necessary, for instance, when calling PBXs with different capabilities to accept digits, or for PSTN and international calls. You may need to consider different strategies for configuring digit manipulation within your dial peers depending on your existing dial plan, the digits users are expected to dial, and the capabilities of your PBX or key system unit (KSU). These digit-manipulation options, in conjunction with the destination pattern, determine the dial string that the router forwards to the telephony device.

### Wildcards

The destination pattern can be either a complete telephone number or a partial telephone number with wildcard digits, represented by a period (.) character. Each "." represents a wildcard for an individual digit that the originating router expects to match. For example, if the destination pattern for a dial peer is defined as "555....", then any dialed string beginning with 555, plus at least four additional digits, matches this dial peer.

In addition to the period (.), several other symbols can be used as wildcard characters in the destination pattern. These symbols provide additional flexibility in implementing dial plans and decrease the need for multiple dial peers in configuring telephone number ranges.

The table below shows the wildcard characters that are supported in the destination pattern.

**Table 3: Wildcard Symbols Used in Destination Patterns**

Symbol	Description
.	Indicates a single-digit placeholder. For example, 555.... matches any dialed string beginning with 555, plus at least four additional digits.
[ ]	Indicates a range of digits. A consecutive range is indicated with a hyphen (-); for example, [5-7]. A nonconsecutive range is indicated with a comma (,); for example, [5,8]. Hyphens and commas can be used in combination; for example, [5-7,9].  <b>Note</b> Only single-digit ranges are supported. For example, [98-102] is invalid.  <b>Note</b> Using a range of digits is the same as keying in each individual number in the dialed string.
( )	Indicates a pattern; for example, 408(555). It is used in conjunction with the symbol ?, %, or +.
?	Indicates that the preceding digit occurred zero or one time. Enter <b>ctrl-v</b> before entering ? from your keyboard.
%	Indicates that the preceding digit occurred zero or more times. This functions the same as the "*" used in regular expression.
+	Indicates that the preceding digit occurred one or more times.
T	Indicates the interdigit timeout. The router pauses to collect additional dialed digits.

The table below shows some examples of how these wildcard symbols are applied to the destination pattern and the dial string that results when dial string 4085551234 is matched to an outbound POTS dial peer. The wildcard symbols follow regular expression rules.

**Table 4: Dial Peer Matching Examples Using Wildcard Symbols**

Destination Pattern	Dial String Translation	String After Stripping <sup>3</sup>
408555.+	408555, followed by one or more wildcard digits. This pattern implies that the string must contain at least 7 digits starting with 408555.	1234
408555.%	408555, followed by zero or more wildcard digits. This pattern implies that the string must contain at least 408555.	1234
408555+	40855, followed by 5 repeated one or more times.	1234
408555%	40855, followed by 5 repeated one or more times. Any explicitly matching digit before the % symbol is not stripped off.	51234



Destination Pattern	Dial String Translation	String After Stripping <sup>3</sup>
408555?	40855, followed by 5. Any explicitly matching digit before the ? symbol is not stripped off.	51234
40855[5-7].+	40855, followed by 5, 6, or 7, plus any digit repeated one or more times.	51234
40855[5-7].%	40855, followed by 5, 6, or 7, plus any digit repeated one or more times.	51234
40855[5-7]+1234	40855, followed by 5, 6, or 7 repeated one or more times, followed by 1234.	51234
408(555)+1234	408, followed by 555, which may repeat one or more times, followed by 1234.	5551234

<sup>3</sup> These examples apply only to one-stage dialing, where direct inward dialing (DID) is enabled on the inbound POTS dial peer. If the router is using two-stage dialing and collecting digits one at a time as dialed, then the call is routed immediately after a dial peer is matched and any subsequent dialed digits are lost.

In addition to wildcard characters, the following characters can be used in the destination pattern:

- Asterisk (\*) and pound sign (#)--These characters on standard touch-tone dial pads can be used anywhere in the pattern. They can be used as the leading character (for example, \*650), except on the Cisco 3600 series.
- Dollar sign (\$)--Disables variable-length matching. It must be used at the end of the dial string.



**Note** If you have an exact match that needs preference or demands preference on the dial-peer, you can add a \$ at the end of that dial string. This method allows the exact match preferred over the other random wildcard, creating a consistent result wherein there are two dial-peers with an 'overlap' of destinations and one dial-peer being a range wildcard and the other being an exact match.

- Circumflex symbol (^)--When used within brackets, allows you to eliminate a digit from consideration for dial peer matching purposes. For example, a destination pattern including [^7] would not match any string beginning with 7.

Multiple digits can also be called out within brackets to eliminate more than one initial digit from dial peer matching. For example, a destination pattern including [^4^6^8] would not match any digit string beginning with 4, 6, or 8.



**Note** A destination pattern including [^752] would allow matching only for digit strings beginning with 5 or 2, but would not match any digit strings beginning with 7. This destination pattern entry essentially behaves the same way as if you had simply included [52] in the destination pattern.

To eliminate a multiple digit string from dial peer matching consideration, you must represent each digit in the string as a succession of individual exceptions. For example, if you wanted to eliminate matching any digit string beginning with 537 from consideration for dial peer matching, you must ensure that your destination pattern includes `[^5][^3][^7]`.

The same destination pattern can be shared across multiple dial peers to form hunt groups.

## Digit Stripping and Prefixes

When a terminating router receives a voice call, it selects an outbound POTS dial peer by comparing the called number (the full E.164 telephone number) in the call information with the number configured as the destination pattern in the POTS dial peer. The access server or router then strips off the left-justified digits that match the destination pattern. If you have configured a prefix, the prefix is added to the front of the remaining digits, creating a dial string, which the router then dials. If all numbers in the destination pattern are stripped out, the user receives a dial tone.

For example, consider a voice call whose E.164 called number is 1(408) 555-2222. If you configure a destination-pattern of "1408555" and a prefix of "9," the router strips off "1408555" from the E.164 telephone number, leaving the extension number of "2222." It then appends the prefix, "9," to the front of the remaining numbers, so that the actual numbers dialed are "9, 2222." The comma in this example means that the router will pause for 1 second between dialing the "9" and dialing the "2" to allow for a secondary dial tone.

When the terminating router matches a dial string to an outbound POTS dial peer, by default the router strips off the left-justified digits that explicitly match the destination pattern. Any remaining digits, called *excess digits*, are forwarded to the telephony interface, such as a PBX or the PSTN.

Some telephony interfaces require that any digits stripped from the dial string be recovered to support a particular dial plan. You can strip these digits either by using the **no digit-strip** dial-peer voice configuration command to disable the default digit-stripping behavior or by using the **prefix** dial-peer voice configuration command to add digits to the beginning of the dial string before it is forwarded to the telephony interface. These commands are supported only in POTS dial peers.

The **no digit-strip** command disables the automatic digit-stripping function so that matching digits are not stripped from the dialed string before it is passed to the telephony interface. For example, in the following dial peer configuration, the entire seven-digit dialed string is passed to the telephony interface:

```
dial-peer voice 100 pots
 destination-pattern 555....
 no digit-strip
 port 1/0:1
```

Disabling digit stripping is useful when the telephony interface requires the full dialed string. With some dial plans, however, the dialed digits must be manipulated according to specific rules. The **prefix** command can be used to add specific digits to the beginning of the dialed string before it is forwarded to the telephony interface.

For example, consider a telephone whose E.164 called number is 1(408)555-1234. This telephone can be reached within the company by dialing its extension number, 51234. If you configure a destination pattern of "1408555...." (the periods represent wildcards) for the associated outbound POTS dial peer, the terminating gateway will strip off the digits "1408555" when it receives a call for 1(408)555-1234. For the terminating gateway to forward the call to the appropriate destination, the digit "5" needs to be prepended to the remaining digits. In this case, you would configure a prefix of 5, as shown in the following dial peer configuration.

```
dial-peer voice 100 pots
 destination-pattern 1408555....
```

```
prefix 5
port 1/0:1
```

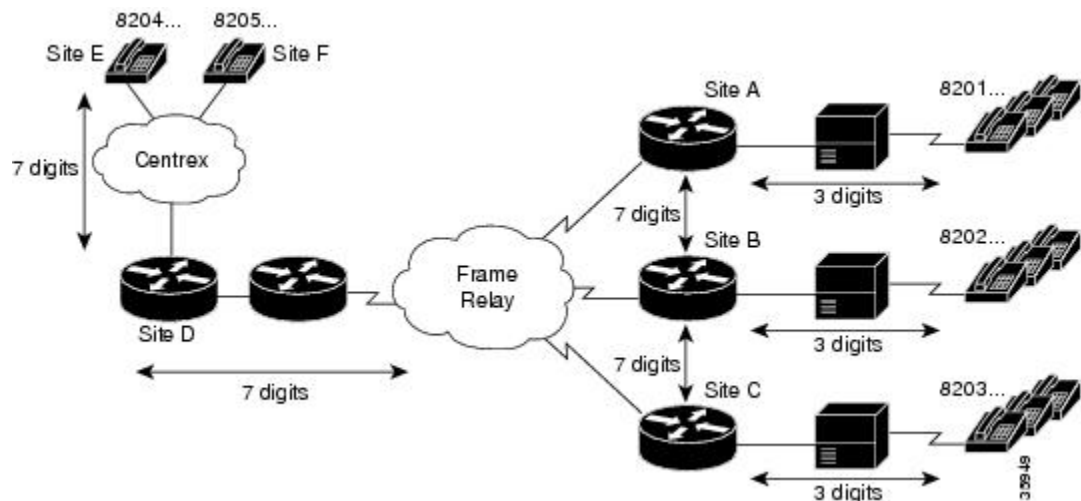
A prefix can also include commas (,). Each comma indicates a 1-second pause in dialing. For example, consider a telephone whose E.164 called number is 1(408)555-1234; to reach this device, you must dial "9." In this case, you might configure "1408....." as the destination pattern, and "9" as the prefix. In this example, the terminating router will strip the digits "1408" from the called number and append the digit "9" to the front of the remaining digits, so that the actual number dialed is "9,5551234." The router pauses for 1 second between dialing the "9" and the "5551234" to allow for a secondary dial tone. In this example, you would configure the router as follows:

```
dial-peer voice 100 pots
destination-pattern 1408.....
prefix 9,
port 1/0:1
```

Using a comma with the **prefix** command is useful when the router must allow for a secondary dial tone; otherwise the router does not wait for the dial tone before playing out excess digits. Putting commas in the prefix makes the router pause 1 second per comma, allowing for a dial tone to occur before the router transmits the remaining digits.

The figure below shows an example of a network using the **no digit-strip** command. In this example, a central site (Site D) is connected to remote sites through routers (Sites A, B, and C), and through a Centrex system for sites still using the PSTN (Sites E and F). The Centrex service requires the full 7-digit dial string to complete calls. The dial peers are configured with a fixed-length 7-digit dial plan.

**Figure 8: Network with Digit Stripping Disabled or Prefixes Enabled**



When Site E (8204...) dials 8201999, the full 7-digit dialed string is passed through the Centrex to the router at Site D. Router D matches the destination pattern 8201... and forwards the 7-digit dial string to Router A. Router A matches the destination pattern 8201..., strips off the matching 8201, and forwards the remaining 3-digit dial string to the PBX. The PBX matches the correct station and completes the call to the proper extension.

Calls in the reverse direction are handled similarly, but because the Centrex service requires the full 7-digit dial string to complete calls, the POTS dial peer at Router D is configured with digit stripping disabled. Alternatively, digit stripping could be enabled and the dial peer could instead be configured with a 4-digit prefix, in this case 8204, which would result in forwarding the full dial string to the Centrex service.

Router A	Router D
<pre>dial-peer voice 1 pots destination-pattern 8201...  port 1/0:1 ! dial-peer voice 4 vofr destination-pattern 8204...  session target s0 2 ! dial-peer voice 5 vofr destination-pattern 8205...  session target s0 2 !</pre>	<pre>dial-peer voice 4 pots destination-pattern 8204...  no digit-strip port 1/0:1 ! dial-peer voice 5 pots destination-pattern 8205...  no digit-strip port 1/0:1 ! dial-peer voice 1 vofr destination-pattern 8201...  session target s0 1 !</pre>

## Forwarding Digits

The **forward-digits** command controls the number of digits that are stripped before the dialed string is passed to the telephony interface. On outbound POTS dial peers, the terminating router normally strips off all digits that explicitly match the destination pattern in the terminating POTS dial peer. Only digits matched by the wildcard pattern are forwarded. The **forward-digits** command can be used to forward a fixed number of dialed digits, or all dialed digits, regardless of the number of digits that explicitly match the destination pattern.

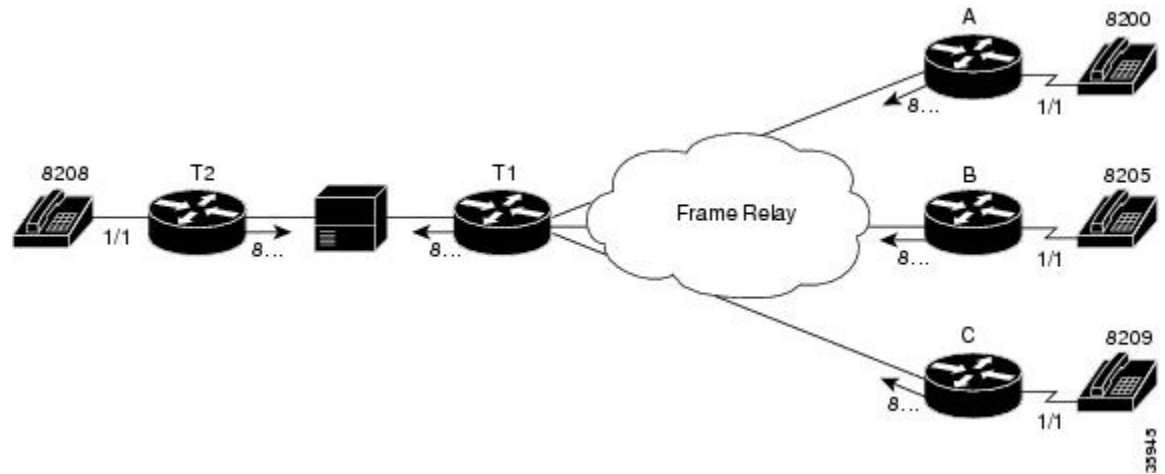
For example, the **forward-digits 4** command tells the router to forward the last four digits in the dialed string. The **forward-digits all** command instructs the router to forward the full dialed string. If the length of the dialed string is longer than the length of the destination pattern, the **forward-digits extra** command forwards the extra trailing digits. Extra digits are not forwarded, however, if the dial peer destination pattern is variable length; for example, 123T, 123...T.



**Note** The **forward-digits** command is supported only in POTS dial peers.

The figure below shows an example of routing voice calls through a PBX using forward digits. In this configuration, Routers T1 and T2 are tandem nodes that must support forward digits so that calls from Routers A, B, or C can make a call to extension 8208.

Figure 9: Routing Voice Calls Through a PBX Using Forward Digits



In this example, all digits matched with destination 8... are forwarded to the appropriate port. For a call from Router A to reach extension 8208, the call first terminates at Router T1, which transmits the digits 8208 to the voice port connected to the PBX. The PBX then routes the voice call to Router T2. The **forward-digits all** command is used here, but the **forward-digits 4** command could also be used in this example.

The following dial peer configurations are required on each router for this example:

Router T1	Router T2
<pre>dial-peer voice 1 vofr destination-pattern 8200  session-target s0 1 ! dial-peer voice 6 vofr destination-pattern 8205  session-target s0 6 ! dial-peer voice 10 vofr destination-pattern 8209  session-target s0 10 ! dial-peer voice 1 pots destination-pattern 8...  forward-digits all port 1/1</pre>	<pre>dial-peer voice 8 pots destination-pattern 8208  port 1/1 ! dial-peer voice 1000 pots destination-pattern 8...  forward-digits all port 1/1 ! dial-peer voice 9999 pots destination-pattern ....  forward-digits all port 1/1</pre>

```

Router A

dial-peer voice 1 pots
 destination-pattern 8200

 port 1/1
 !
dial-peer voice 1000 vofr
 destination-pattern 8...
 session-target s0 1

```

## Number Expansion

In most corporate environments, the telephone network is configured so that you can reach a destination by dialing only a portion (an extension number) of the full E.164 telephone number. You can define an extension number as the destination pattern for a dial peer. The router can be configured to recognize the extension number and expand it into its full E.164 dialed number when the **num-exp** global configuration command is used with the **destination-pattern** dial-peer voice configuration command.

Number expansion is a globally applied rule that enables you to define a set of digits for the router to prepend to the beginning of a dialed string before passing it to the remote telephony device. Automatically prepending digits in the dial peer configuration reduces the number of digits that a user must dial to reach a remote location. Number expansion is similar to using a prefix, except that number expansion is applied globally to all dial peers.

Using a simple telephony-based example, suppose that user A works in a company where employees extensions are reached by dialing the last four digits of the full E.164 telephone number. The E.164 telephone number is 555-2123; user A's extension number is 2123. Suppose that every employee on user A's floor has a telephone number that begins with the same first four digits: 5552. You could define each dial peer's destination pattern using each extension number, and then use number expansion to prepend the first four digits onto the extension. In this example, the router could be configured as follows:

```

num-exp 2... 5552...
dial peer voice 1 pots
 destination pattern 2123

```

Number expansion can also be used to replace a dialed number with another number, as in the case of call forwarding. Suppose that for some reason, user A needs to have all of his telephone calls forwarded to another number, 555-6611. In this example, you would configure the router as follows:

```

num-exp 2123 5556611
dial peer voice 1 pots
 destination pattern 2123

```

In this example, every time the device receives a call for extension 2123, the dialed digits will be replaced with 555-6611 and the call will be forwarded to that telephone.

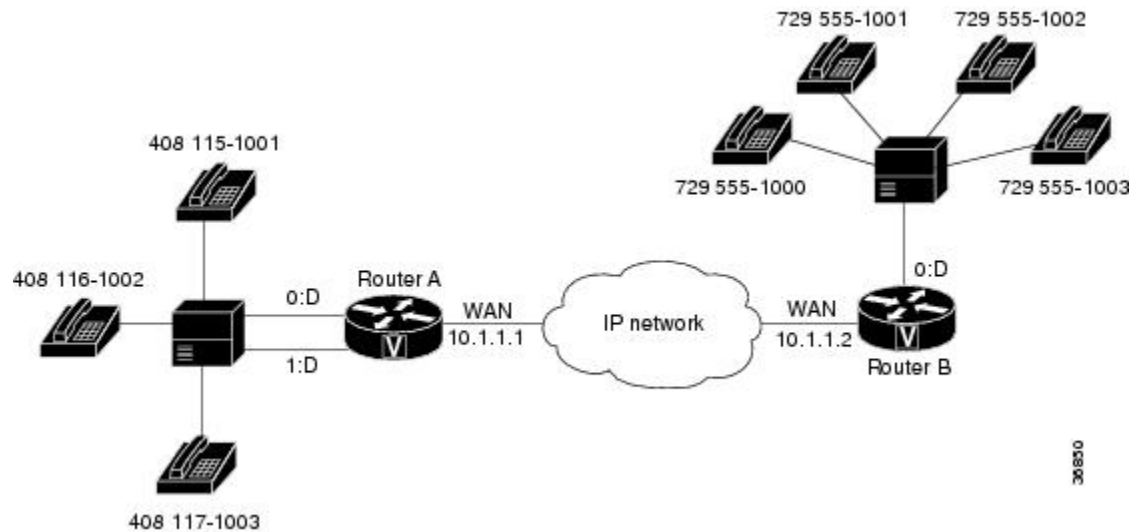
Before you configure the **num-exp** command, it is helpful to map individual telephone extensions to their full E.164 dialed numbers. This task can be done easily by creating a number expansion table.

### Creating a Number Expansion Table

The figure below shows a network for a small company that wants to use VoIP to integrate its telephony network with its existing IP network. The destination patterns (or expanded telephone numbers) associated with Router A are 408 115-xxxx, 408 116-xxxx, and 408 117-xxxx, where xxxx identifies the individual dial

peers by extension. The destination pattern (or expanded telephone number) associated with Router B is 729 555-xxxx.

**Figure 10: VoIP Example for Number Expansion**



The table below shows the number expansion table for this scenario. The information included in this example must be configured on both Router A and Router B.

**Table 5: Sample Number Expansion Table**

Extension	Destination Pattern	Num-Exp Command Entry
5....	408115....	num-exp 5.... 408115....
6....	408116....	num-exp 6.... 408116....
7....	408117....	num-exp 7.... 408117....
1...	729555....	num-exp 1... 729555....

The period (.) character represents wildcards (such as extension numbers) in a telephone number.

### Configuring Number Expansion

The purpose of this task is to expand an extension number into its full telephone number.

#### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **num-exp** *extension-number* *expanded-number*

#### DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enters privileged EXEC mode.

	Command or Action	Purpose
	<b>Example:</b> Router> enable	<ul style="list-style-type: none"> <li>Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> Router# configure terminal	Enters global configuration mode.
<b>Step 3</b>	<b>num-exp extension-number expanded-number</b> <b>Example:</b> Router(config)# num-exp 2123 5556611	Configures number expansion globally for all dial peers. The arguments are as follows: <ul style="list-style-type: none"> <li><i>extension-number</i> --Specifies the extension number to expand into the full telephone number that is specified by the <i>expanded-number</i> argument.</li> <li><i>expanded-number</i> --Specifies the full telephone number or destination pattern to which the extension number is expanded.</li> </ul>

## Translation Rules

Digit translation rules are used to manipulate the calling number (ANI) or called number (DNIS) digits for a voice call, or to change the numbering type of a call. Translation rules are used to convert a telephone number into a different number before the call is matched to an inbound dial peer or before the call is forwarded by the outbound dial peer. For example, within your company you may dial a 5-digit extension to reach an employee at another site. If the call is routed through the PSTN to reach the other site, the originating gateway must use translation rules to convert the 5-digit extension into the 10-digit format that is recognized by the central office switch.

Translation rules are defined by using the **translation-rule** command. After you define a set of translation rules, you can apply the rules to all inbound VoIP calls, to all inbound calls that terminate at a specific voice port, and to individual inbound or outbound call legs according to the dial peer.

The following example shows a dial peer that is configured to use translation-rule set 1, which contains ten translation rules. The first rule defined is rule 0, in which 910 is the pattern that must be matched and replaced, and 0 is the pattern that is substituted for 910.

```
translation-rule 1
 rule 0 ^910 0
 rule 1 ^911 1
 rule 2 ^912 2
 rule 3 ^913 3
 rule 4 ^914 4
 rule 5 ^915 5
 rule 6 ^916 6
 rule 7 ^917 7
 rule 8 ^918 8
 rule 9 ^919 9
!
!
dial-peer voice 2 voip
 destination-pattern 91.....
```



```
translate-outgoing called 1
session target ras
```

The configuration results in the stripping of the leading digits 91 from any called number that begins with 91 before the number is forwarded by the outbound VoIP dial peer. Use the caret (^) symbol to specify that the matched digits must occur at the start of a dial string.



**Note** Wildcard symbols such as the period (.), asterisk (\*), percent sign (%), plus sign (+), and question mark (?) are not valid in translation rules. The router ignores these symbols when converting a number if they are used in a translation rule.

Translation rules can also be used to change the numbering type for a call. For example, some gateways may tag any number with more than 11 digits as an international number, even when the user must dial a 9 to reach an outside line. The following example shows a translation rule that converts any called number that starts with 91, and that is tagged as an international number, into a national number without the 9 before sending it to the PSTN:

```
translation-rule 20
 rule 1 91 1 international national
 !
 !
dial-peer voice 10 pots
 destination-pattern 91.....
 translate-outgoing called 20
 port 1:D
 !
```



**Note** Using digit translation rules with the **num-exp** or **prefix** command is not recommended unless it is the only way to minimize confusion.

To create digit translation rules, perform the tasks in the following sections:

- [CreatingDigitTranslationRules, on page 33](#) (required)

To apply digit translation rules to VoIP calls, perform one or more of the following procedures:

- [Applying Translation Rules to Inbound POTS Calls, on page 35](#) (optional)
- [Applying Translation Rules to Inbound VoIP Calls, on page 36](#) (optional)
- [Applying Translation Rules to Outbound Call Legs, on page 37](#) (optional)

## CreatingDigitTranslationRules

The purpose of this task is to enter translation-rule configuration mode and specify a set of translation rules.

To create additional individual translation rules to include in the translation-rule set, repeat Step 4 .



**Note** Applying translation rules to more than one call leg in an end-to-end call is not recommended.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **translation-rule** *name-tag*
4. **rule** *name-tag input-matched-pattern substituted-pattern [match-type substituted-type]*

## DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> <pre>Router&gt; enable</pre>	Enters privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> <pre>Router# configure terminal</pre>	Enters global configuration mode.
<b>Step 3</b>	<b>translation-rule</b> <i>name-tag</i> <b>Example:</b> <pre>Router(config)# translation-rule 1</pre>	Defines a digit translation-rule set and enters translation-rule configuration mode. All subsequent commands that you enter in this mode before you exit will apply to this translation-rule set. <ul style="list-style-type: none"> <li>• The <i>name-tag</i> argument specifies a unique number that identifies the set of translation rules. Valid entries are from 1 to 2147483647.</li> </ul>
<b>Step 4</b>	<b>rule</b> <i>name-tag input-matched-pattern substituted-pattern [match-type substituted-type]</i> <b>Example:</b> <pre>Router(config-translate) # rule 0 ^910 0</pre>	Defines an individual translation rule. This command can be entered up to 11 times to add an individual translation rule to the translation rule set defined in Step 1. The arguments are as follows: <ul style="list-style-type: none"> <li>• <i>name-tag</i> --Specifies a unique number that identifies this individual translation rule. Valid entries are from 0 to 10.</li> <li>• <i>input-matched-pattern</i> --Specifies the digit string that must be matched, and then replaced with the <i>substituted-pattern</i> value.</li> <li>• <i>substituted-pattern</i> --Specifies the digit string that replaces the <i>input-matched-pattern</i> value.</li> <li>• <i>match-type</i> --(Optional) Specifies the numbering type that you want to replace with the numbering type defined in the <i>substituted-type</i> value. Enter <i>any</i> for the <i>match-type</i> if you want to match on any numbering type.</li> </ul>

	Command or Action	Purpose
		<ul style="list-style-type: none"> <li>• Otherwise, enter one of the following keywords for each of these arguments:               <ul style="list-style-type: none"> <li>• <b>abbreviated</b></li> <li>• <b>international</b></li> <li>• <b>national</b></li> <li>• <b>network</b></li> <li>• <b>reserved</b></li> <li>• <b>subscriber</b></li> <li>• <b>unknown</b></li> </ul> </li> </ul>

### Applying Translation Rules to Inbound POTS Calls

The purpose of this task is to apply a translation rule set to all inbound POTS calls that terminate on the same voice port.



**Note** When this method is used, the digit translation rules are executed before the inbound POTS dial peer is matched.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice-port** *location*
4. **translate** {called | calling} *name-tag*

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> <pre>Router&gt; enable</pre>	Enters privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> <pre>Router# configure terminal</pre>	Enters global configuration mode.
<b>Step 3</b>	<b>voice-port</b> <i>location</i> <b>Example:</b> <pre>Router(config)# voice-port 1/0:1</pre>	Specifies the voice port through which the call enters the router.  The <b>voice-port</b> command syntax is platform-specific. For more information about the syntax of this command, refer to the <i>Voice Port Configuration Guide</i> .

	Command or Action	Purpose
<b>Step 4</b>	<b>translate</b> {called   calling} <i>name-tag</i> <b>Example:</b> <pre>Router(config-voiceport)# translate called 4</pre>	Specifies the translation rule set to apply to the called number or calling number. The keywords and argument are as follows: <ul style="list-style-type: none"> <li>• <b>called</b> --Applies the translation rule to the called party number.</li> <li>• <b>calling</b> --Applies the translation rule to the calling party number.</li> <li>• <i>name-tag</i> --Specifies the reference number of the translation rule. Valid entries are 1 through 2147483647.</li> </ul>

### Applying Translation Rules to Inbound VoIP Calls

The purpose of this task is to apply a translation rule set to all inbound VoIP calls that originate at an H.323 gateway.



**Note** When using this method, the digit translation rules are executed before the inbound VoIP dial peer is matched.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voip-incoming translation-rule** {called | calling} *name-tag*

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> <pre>Router&gt; enable</pre>	Enters privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> <pre>Router# configure terminal</pre>	Enters global configuration mode.
<b>Step 3</b>	<b>voip-incoming translation-rule</b> {called   calling} <i>name-tag</i> <b>Example:</b> <pre>Router (config) # voip-incoming translation-rule called 5</pre>	Specifies the translation rule set to apply to all inbound VoIP call legs that originate from an H.323 gateway. The keywords and argument are as follows: <ul style="list-style-type: none"> <li>• <b>called</b> --Applies the translation rule to the called party number.</li> </ul>

	Command or Action	Purpose
		<ul style="list-style-type: none"> <li>• <b>calling</b> --Applies the translation rule to the calling party number.</li> <li>• <b>name-tag</b> --Specifies the reference number of the translation rule. Valid entries are 1 through 2147483647.</li> </ul>

### Applying Translation Rules to Outbound Call Legs

The purpose of this task is to apply a translation rule set to an outbound VoIP or POTS call leg.



**Note** Translation rules that are configured in a dial peer using the **translate-outgoing** command are not applied to inbound call legs. When two-stage dialing is used, the translation rules that are configured in the voice port using the **translate** command are applied twice: after the inbound dial peer is matched, and again after the digits are collected.



**Note** If the **prefix** command is also configured in the dial peer, the **translate-outgoing** command is executed first.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dial-peer voice** *number* **pots** | **voip** | **vofr** | **voatm**
4. **translate-outgoing** {**called** | **calling**} *name-tag*

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> <pre>Router&gt; enable</pre>	Enters privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> <pre>Router# configure terminal</pre>	Enters global configuration mode.
<b>Step 3</b>	<b>dial-peer voice</b> <i>number</i> <b>pots</b>   <b>voip</b>   <b>vofr</b>   <b>voatm</b> <b>Example:</b> <pre>Router(config)# dial-peer voice 345 pots</pre>	Enters dial-peer voice configuration mode and defines a local dial peer. <ul style="list-style-type: none"> <li>• The <i>number</i> argument identifies the dial peer. Valid entries are from 1 to 2147483647.</li> </ul>

	Command or Action	Purpose
<b>Step 4</b>	<b>translate-outgoing</b> {called   calling} <i>name-tag</i> <b>Example:</b> <pre>Router(config-dialpeer)# translate-outgoing called 6</pre>	Specifies the translation rule set to apply to the calling number or called number. The keywords and argument are as follows: <ul style="list-style-type: none"> <li>• <b>called</b> --Applies the translation rule to the called party number.</li> <li>• <b>calling</b> --Applies the translation rule to the calling party number.</li> <li>• <b>name-tag</b> --Specifies the reference number of the translation rule. Valid entries are 1 through 2147483647.</li> </ul>

## Data Dial Peers

In addition to standard voice-network and POTS dial peers, a newer type of dial peer has been introduced to service modem calls over POTS lines with automatic dial peer matching and priority assignment. These new dial peers are called data dial peers.

Traditionally, if a modem call came over a POTS line connected to a voice-network gateway, a procession of matching criteria was required to determine the nature of the incoming call. Only after it was determined that an incoming call was not a voice call could it then be assumed that the transmission was, in fact, a data-based modem call.

Now, however, you have the ability to specify particular dial peers as data dial peers and even assign them priority in relation to other dial peers in the system.

## Configuring Data Dial Peers

The purpose of this task is to configure a POTS dial peer to be a data dial peer.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dial-peer data** *tag* **pots**
4. **incoming called-number** *string*

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> <pre>Router&gt; enable</pre>	Enters privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>

	Command or Action	Purpose
Step 2	<b>configure terminal</b> <b>Example:</b> <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 3	<b>dial-peer data tag pots</b> <b>Example:</b> <pre>Router(config)# dial-peer data 2001 pots</pre>	Specifies a dial peer for data calls and enters dial-peer voice configuration mode. The keyword and argument are as follows: <ul style="list-style-type: none"> <li>• <b>tag</b> --Specifies the dial peer identifier. The valid range is from 1 to 2147483647.</li> </ul> <p><b>Note</b> You cannot have a data dial peer and a voice dial peer that are assigned to the same <i>tag</i> number. The <i>tag</i> must be unique for all dial peers.</p> <ul style="list-style-type: none"> <li>• <b>pots</b> --Specifies the dial peer as POTS.</li> </ul>
Step 4	<b>incoming called-number string</b> <b>Example:</b> <pre>Router(dial-peer)# incoming called-number 4085551212</pre>	Specifies the incoming called number that is associated with the data dial peer. <ul style="list-style-type: none"> <li>• The <i>string</i> argument specifies the number.</li> </ul>

## Configuring a Search for Dial Peers by Type

The purpose of this task is to configure a search for dial peers by type.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dial-peer search type {data voice | voice data | none}**

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>enable</b> <b>Example:</b> <pre>Router&gt; enable</pre>	Enters privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
Step 2	<b>configure terminal</b> <b>Example:</b> <pre>Router# configure terminal</pre>	Enters global configuration mode.

	Command or Action	Purpose
<b>Step 3</b>	<b>dial-peer search type</b> {data voice   voice data   none} <b>Example:</b> <pre>Router(config)# dial-peer search type data voice</pre>	Specifies the dial-peer search functionality. The keywords are as follows: <ul style="list-style-type: none"> <li>• <b>data</b> --Searches for data dial peers.</li> <li>• <b>voice</b> --Searches for voice dial peers.</li> <li>• <b>none</b> --Searches for all dial peers with the same preference based on the input order.</li> </ul> <p><b>Note</b> The default is <b>data</b> and <b>voice</b>.</p>

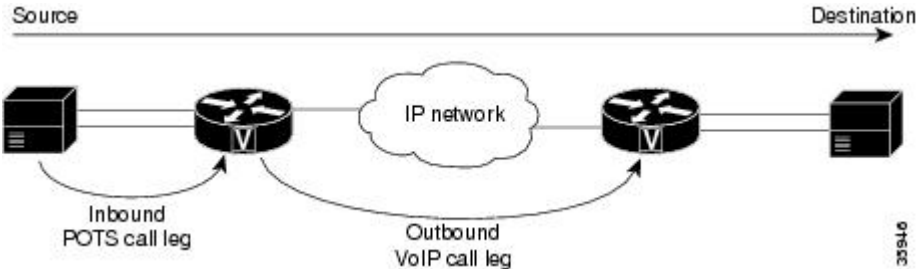
# Inbound and Outbound Dial Peers

Dial peers are used for both inbound and outbound call legs. It is important to remember that these terms are defined from the perspective of the router. An inbound call leg originates when an incoming call comes *to* the router. An outbound call leg originates when an outgoing call is placed *from* the router. The first figure below illustrates call legs from the perspective of the originating router; the second figure below illustrates call legs from the perspective of the terminating router.

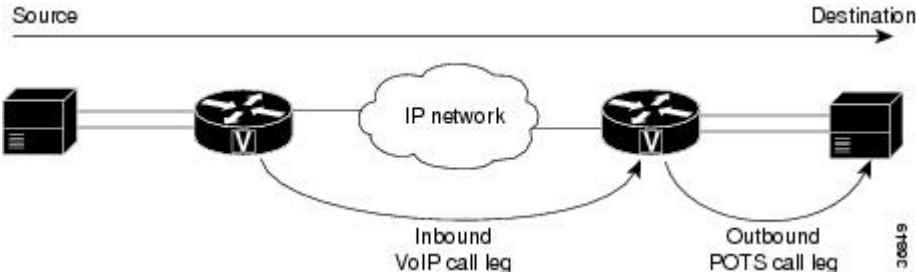


**Note** The figures below apply to voice calls that are being sent across the packet network. If the originating and terminating POTS interfaces share the same router or if the call requires hairpinning, then two POTS call legs are sufficient.

**Figure 11: Call Legs from the Perspective of the Originating Router**



**Figure 12: Call Legs from the Perspective of the Terminating Router**





For inbound calls from a POTS interface that are destined for the packet network, the router matches a POTS dial peer for the inbound call leg and a voice-network dial peer, such as VoIP or VoFR, for the outbound leg. For inbound calls from the packet network, the router matches a POTS dial peer to terminate the call and a voice-network dial peer to apply features such as codec or QoS.

For inbound POTS call legs going to outbound voice-network dial peers, the router forwards all digits that it collects. On outbound POTS call legs, the router strips off explicitly matching digits and forwards any excess digits out the designated port.

The following examples show basic configurations for POTS and VoIP dial peers:

```
dial-peer voice 1 pots
 destination-pattern 555....
 port 1/0:1
dial-peer voice 2 voip
 destination-pattern 555....
 session target ipv4:192.168.1.1
```

The router selects a dial peer for a call leg by matching the string that is defined by using the **answer-address**, **destination-pattern**, or **incoming called-number** command in the dial peer configuration.

## Matching Inbound Dial Peers

To match inbound call legs to dial peers, the router uses three information elements in the call setup message and four configurable dial peer attributes. The three call setup elements are:

- Called number or dialed number identification service (DNIS)--A set of numbers representing the destination, which is derived from the ISDN setup message or channel associated signaling (CAS) DNIS.
- Calling number or automatic number identification (ANI)--A set of numbers representing the origin, which is derived from the ISDN setup message or CAS ANI.
- Voice port--The voice port carrying the call.

The five configurable dial peer attributes are:

- Incoming called number--A string representing the called number or DNIS. It is configured by using the **incoming called-number** dial-peer voice configuration command in POTS or multimedia mail over IP (MMoIP) dial peers.
- Answer address--A string representing the calling number or ANI. It is configured by using the **answer-address** dial-peer voice configuration command in POTS or VoIP dial peers and is used only for inbound calls from the IP network.
- Destination pattern--A string representing the calling number or ANI. It is configured by using the **destination-pattern** dial-peer voice configuration command in POTS or voice-network dial peers.
- Application--A string representing the predefined application that you wish to enable on the dial peer. It is configured by using the **application** dial-peer voice configuration command on inbound POTS dial peers.
- Port--The voice port through which calls to this dial peer are placed.

The router selects an inbound dial peer by matching the information elements in the setup message with the dial peer attributes. The router attempts to match these items in the following order:

1. Called number with the **incoming called-number** command

2. Calling number with the **answer-address** command
3. Calling number with the **destination-pattern** command
4. Incoming voice port with the configured voice port

The router must match only one of these conditions. It is not necessary for all the attributes to be configured in the dial peer or that every attribute match the call setup information; only one condition must be met for the router to select a dial peer. The router stops searching as soon as one dial peer is matched and the call is routed according to the configured dial peer attributes. Even if there are other dial peers that would match, only the first match is used.




---

**Note** For a dial peer to be matched, its administrative state must be up. The dial peer administrative state is up by default when it is configured with at least one of these commands: **incoming called-number**, **answer-address**, or **destination-pattern**. If the **destination-pattern** command is used, the voice port or session target must also be configured.

---

## Variable-Length Matching

When matching dial peers, the router defaults to variable-length matching, which means that as long as the left-justified digits in the dial string match the configured pattern in the dial peer, any digits beyond the configured pattern are ignored for the purposes of matching. For example, dial string 5551212 would match both of the following dial peers:

```
dial-peer voice 1 voip
 destination-pattern 555
 session target ipv4:10.10.1.1
dial-peer voice 2 voip
 destination-pattern 5551212
 session target ipv4:10.10.1.2
```

To disable variable-length matching for a dial peer, add the dollar sign (\$) to the end of the destination pattern, as shown:

```
dial-peer voice 1 voip
 destination-pattern 555$
 session target ipv4:10.10.1.1
```

The \$ character in the configuration prevents this dial peer from being matched for dial string 5551212 because the extra digits beyond 555 are considered in the matching.

With two-stage dialing, the router collects the dialed string digit by digit. It attempts to match a dial peer after each digit is received. As soon as it finds a match, it immediately routes the call. For example, given the following configurations, the router would immediately match dial string 5551212 to dial peer 1.

```
dial-peer voice 1 voip
 destination-pattern 555
 session target ipv4:10.10.1.1
dial-peer voice 2 voip
 destination-pattern 5551212
 session target ipv4:10.10.1.2
```

If the router is performing two-stage dialing and you want to make sure that the full dial string is collected before a dial peer is matched, you can use the timeout T-indicator as in variable-length dial plans. For example,

after the router waits until the full dial string is collected, dial string 5551212 would match both of the following dial peers:

```
dial-peer voice 1 voip
 destination-pattern 555T
 session target ipv4:10.10.1.1
dial-peer voice 2 voip
 destination-pattern 5551212T
 session target ipv4:10.10.1.2
```

How the router selects a dial peer also depends on whether the dial peer is being matched for the inbound or outbound call leg.

## Configuring the incoming called-number Command

When a Cisco router is handling both modem and voice calls, it needs to identify the service type of the call—that is, whether the incoming call to the router is a modem or a voice call. When the router handles only modem calls, the service type identification is handled through modem pools. Modem pools associate calls with modem resources based on the called number (DNIS). In a mixed environment, where the router receives both modem and voice calls, you need to identify the service type of a call by using the **incoming called-number** command.

If the **incoming called-number** command is not configured, the router attempts to resolve whether an incoming call is a modem or voice call on the basis of the interface over which the call comes. If the call comes in over an interface associated with a modem pool, the call is assumed to be a modem call; if a call comes in over a voice port associated with a POTS dial peer, the call is assumed to be a voice call.

The purpose of this task is to identify the service type of a call as voice.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dial-peer voice** *number* **pots | voip | vofr| voatm**
4. **incoming called-number** *number*

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>enable</b> <b>Example:</b> Router> enable	Enters privileged EXEC mode. • Enter your password if prompted.
Step 2	<b>configure terminal</b> <b>Example:</b> Router# configure terminal	Enters global configuration mode.
Step 3	<b>dial-peer voice</b> <i>number</i> <b>pots   voip   vofr  voatm</b> <b>Example:</b>	Enters dial-peer voice configuration mode and defines a local dial peer.

	Command or Action	Purpose
	Router(config)# dial-peer voice 345 pots	<ul style="list-style-type: none"> <li>The <i>number</i> argument identifies the dial peer. Valid entries are from 1 to 2147483647.</li> </ul>
<b>Step 4</b>	<b>incoming called-number</b> <i>number</i> <b>Example:</b> Router(config-dialpeer)# incoming called-number 5551212	Defines the telephone number that identifies voice calls associated with this dial peer.

## answer-address Command

The purpose of this task is to specify the answer address for this dial peer.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dial-peer voice** *number* {pots | voip | vofr| voatm}
4. **answer-address** [+]*string* [T]

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> Router> enable	Enters privileged EXEC mode. <ul style="list-style-type: none"> <li>Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> Router# configure terminal	Enters global configuration mode.
<b>Step 3</b>	<b>dial-peer voice</b> <i>number</i> {pots   voip   vofr  voatm} <b>Example:</b> Router(config)# dial-peer voice 123 pots	Enters dial-peer voice configuration mode and defines a local dial peer. <ul style="list-style-type: none"> <li>The <i>number</i> argument identifies the dial peer. Valid entries are 1–2147483647.</li> </ul>
<b>Step 4</b>	<b>answer-address</b> [+] <i>string</i> [T] <b>Example:</b> Router(config-dialpeer)# answer-address 55534..	Defines the phone number that identifies voice calls associated with this dial peer. The keywords and argument are as follows: <ul style="list-style-type: none"> <li>+ --(Optional) Character indicating an E.164 standard number.</li> <li><i>string</i> --A series of digits specifying the E.164 or private dial plan phone number. Valid entries are as follows:</li> </ul>

	Command or Action	Purpose
		<ul style="list-style-type: none"> <li>• Digits 0 through 9, letters A through F, pound sign (#), and asterisk (*), which represent specific digits that can be entered.</li> <li>• Comma (,), which inserts a pause between digits.</li> <li>• Period (.), which matches any entered digit.</li> <li>• <b>T --</b>(Optional) Control character indicating that the <b>answer-address</b> value is a variable-length dial string.</li> </ul>

## Configuring the destination-pattern Command

See the [Configuring Destination Patterns, on page 22](#) for information on configuring the destination pattern for a dial peer.

## Configuring the port Command

See the [Assigning Voice Ports, on page 19](#) for information on associating a voice port with a dial peer.

## Matching Outbound Dial Peers

The method a router uses to select an outbound dial peer depends on whether ISDN DID is configured in the inbound POTS dial peer. If DID is not configured in the inbound POTS dial peer, the router collects the incoming dialed string digit by digit. As soon as one dial peer is matched, the router immediately places the call using the configured attributes in the matching dial peer.

If DID is configured in the inbound POTS dial peer, the router uses the full incoming dial string to match the destination pattern in the outbound dial peer. With DID, the setup message contains all the digits necessary to route the call; no additional digit collection is required. If more than one dial peer matches the dial string, all of the matching dial peers are used to form a rotary group. The router attempts to place the outbound call leg using all of the dial peers in the rotary group until one is successful.

## Using Default Routes

Default routes reduce the number of dial peers that must be configured when calls that are not terminated by other dial peers are sent to a central router, usually for forwarding to a PBX. A default route is a dial peer that automatically matches any call that is not terminated by other dial peers. For example, in the following configuration, the destination pattern 8... is a voice default route because all voice calls with a dialed string that starts with 8 followed by at least three additional digits will either match on 8208 or end with 8..., which is the last-resort voice route used by the router if no other dial peer is matched.

```
dial-peer voice 8 pots
 destination-pattern 8208
 port 1/1
!
dial-peer voice 1000 pots
 destination-pattern 8...
 port 1/1
```

A default route could also be defined by using a single wildcard character with the timeout T-indicator in the destination pattern, as shown in the following example:

```
dial-peer voice 1000 voip
 destination-pattern .T
 session-target ipv4:10.10.1.2
```

You should be careful, however, when using the T-indicator for default routes. Remember, when matching dial peers for outbound call legs, the router places the call as soon as it finds the first matching dial peer. The router could match on this dial peer immediately even if there were another dial peer with a more explicit match and a more desirable route.



---

**Note** The timeout T-indicator is appropriate only for two-stage dialing. If the router is configured for one-stage dialing, which means that DID is configured in the inbound POTS dial peer, then the timeout T-indicator is unnecessary.

---

## Additional Features

### One Stage and Two Stage Dialing

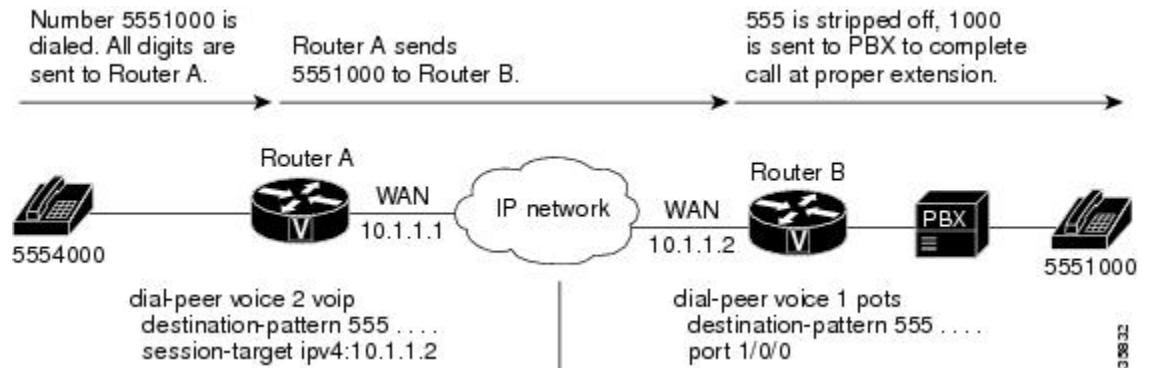
With two-stage dialing, when a voice call enters the network, the originating router collects dialed digits until it can match an outbound dial peer. As soon as the router matches a dial peer, it immediately places the call and forwards the associated dial string. No additional dialed digits are collected. The digits and wildcards that are defined in the destination pattern determine how many digits the originating router collects before matching the dial peer. Any digits dialed after the first dial peer is matched are dropped.

For example, if the dialed string is "1234599" and the originating router matches a dial peer with a destination pattern of 123.., then the digits "99" are not collected. The call is placed immediately after the digit "5" is dialed, and the dial string "12345" is forwarded to the next call leg.

On the terminating router, the left-justified digits that explicitly match the terminating POTS dial peer are stripped off. Any trailing wildcard digits are considered excess digits. The terminating router forwards these excess digits to the telephony interface. For example, if the dial string "1234599" is matched on a terminating router to a destination pattern of "123..," the digits "4599" are excess digits and are forwarded to the telephony interface.

The figure below illustrates how the originating router collects a dial string and the terminating router forwards the digits to the telephony device.

Figure 13: Collecting and Forwarding Dialed Digits



The examples in the table below demonstrate how the originating router collects dialed digits for a given destination pattern in the outbound voice-network dial peer.

Table 6: Digit Collection Based on Destination Pattern

Dialed Digits	Destination Pattern	Dial String Collected <sup>4</sup>
5551234	5.....	5551234
5551234	555....	5551234
5551234	555	555
555123499	555....	5551234

<sup>4</sup> These examples apply only to two-stage dialing, in which the router collects the dialed string digit by digit. If DID is enabled in the inbound POTS dial peer, the router performs one-stage dialing, which means that the full dialed string is used regardless of the destination pattern that is matched.

## Direct Inward Dialing

Unless otherwise configured, when a voice call comes into the router, the router presents a dial tone to the caller and collects digits until it can identify an outbound dial peer. This process is called *two-stage dialing*. After the outbound dial peer is identified, the router forwards the call through to the destination as configured in the dial peer.

The DID feature in dial peers enables the router to use the called number (DNIS) to directly match an outbound dial peer when receiving an inbound call from a POTS interface. When DID is configured on the inbound POTS dial peer, the called number (DNIS) is automatically used to match the destination pattern for the outbound call leg.



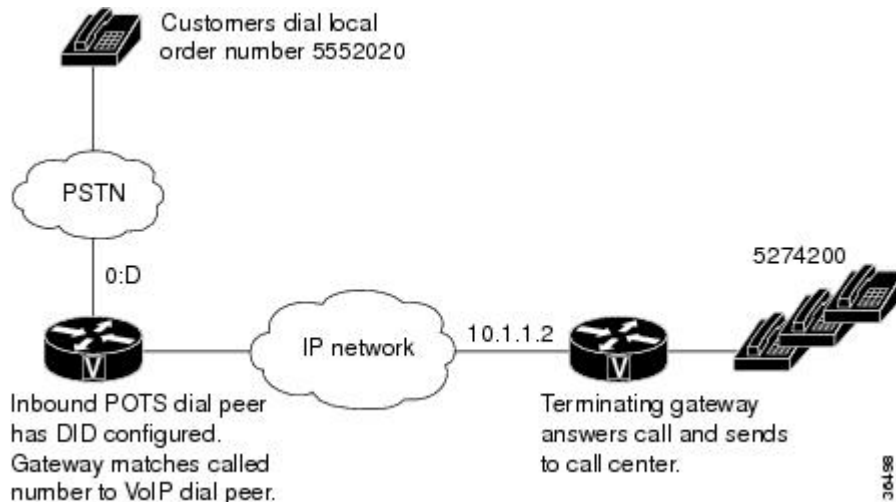
**Note** DID for POTS dial peers, as described here, is for ISDN connections only. It is not the same as analog DID for Cisco routers, which supports analog DID trunk service. For more information about analog DID see Analog Direct Inward Dialing .

You may prefer that the router use the called number (DNIS) to find a dial peer for the outbound call leg--for example, if the switch connecting the call to the router has already collected all the dialed digits. DID enables

the router to match the called number to a dial peer and then directly place the outbound call. With DID, the router does not present a dial tone to the caller and does not collect digits; it forwards the call directly to the configured destination. This is called *one-stage dialing*.

The figure below shows a call scenario using DID.

**Figure 14: VoIP Call Using DID**



In the figure above, the POTS dial peer that matches the incoming called-number has DID configured:

```
dial-peer voice 100 pots
  incoming called-number 5552020
  direct-inward-dial
  port 0:D
```

The **direct-inward-dial** command in the POTS dial peer tells the gateway to look for a destination pattern in a dial peer that matches the DNIS. For example, if the dialed number is 5552020, the gateway matches the following VoIP dial peer for the outbound call leg:

```
dial-peer voice 101 voip
  destination-pattern 5552020
  session target ipv4:10.1.1.2
```

The call is made across the IP network to 10.1.1.2, and a match is found in that terminating gateway:

```
dial-peer voice 555 pots
  destination-pattern 5552020
  port 0:D
  prefix 5274200
```

This dial peer matches on the dialed number and changes that number to 5274200 with the **prefix** command. The result is that the user dials a number, gets connected, and never knows that the number reached is different from the number dialed.

## Configuring Direct Inward Dialing

The purpose of this task is to configure a POTS dial peer for DID.



## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dial-peer voice *number* pots**
4. **direct-inward-dial**

## DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>enable</b> <b>Example:</b> Router> enable	Enters privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
Step 2	<b>configure terminal</b> <b>Example:</b> Router# configure terminal	Enters global configuration mode.
Step 3	<b>dial-peer voice <i>number</i> pots</b> <b>Example:</b> Router(config)# dial-peer voice 234 pots	Enters dial-peer voice configuration mode and defines a local dial peer that will connect to the POTS network. <ul style="list-style-type: none"> <li>• The <i>number</i> argument identifies the dial peer. Valid entries are from 1 to 2147483647.</li> </ul>
Step 4	<b>direct-inward-dial</b> <b>Example:</b> Router(config-dialpeer)# direct-inward-dial	Specifies DID for this POTS dial peer.

## What to do next



**Note** DID is configured for inbound POTS dial peers only.

## Hunt Groups

The router supports the concept of hunt groups, sometimes called *rotary groups*, in which multiple dial peers are configured with the same destination pattern. Because the destination of each POTS dial peer is a single voice port to a telephony interface, hunt groups help ensure that calls get through even when a specific voice port is busy. If the router is configured to hunt, it can forward a call to another voice port when one voice port is busy.

For example, in the following configuration for Router A, four POTS dial peers are configured with different destination patterns. Because each dial peer has a different destination pattern, no backup is available if the voice port mapped to a particular dial peer is busy with another call.

With a hunt group, if a voice port is busy, the router hunts for another voice port until it finds one that is available. In the following example for Router B, each dial peer is configured using the same destination pattern of 3000, forming a dial pool to that destination pattern.

Router A (Without Hunt Groups)	Router B (With Hunt Groups and Preferences)
<pre>dial-peer voice 1 pots  destination-pattern 3001  port 1/1 ! dial-peer voice 2 pots  destination-pattern 3002  port 1/2 ! dial-peer voice 3 pots  destination-pattern 3003  port 1/3 ! dial-peer voice 4 pots  destination-pattern 3004  port 1/4</pre>	<pre>dial-peer voice 1 pots  destination pattern 3000  port 1/1  preference 0 ! dial-peer voice 2 pots  destination pattern 3000  port 1/2  preference 1 ! dial-peer voice 3 pots  destination pattern 3000  port 1/3  preference 2 ! dial-peer voice 4 pots  destination pattern 3000  port 1/4  preference 3</pre>

To give specific dial peers in the pool a preference over other dial peers, you can configure the preference order for each dial peer by using the **preference** command. The router attempts to place a call to the dial peer with the highest preference. The configuration example given for Router B shows that all dial peers have the same destination pattern, but different preference orders.

The lower the preference number, the higher the priority. The highest priority is given to the dial peer with preference order 0. If the same preference is defined in multiple dial peers with the same destination pattern, a dial peer is selected randomly.

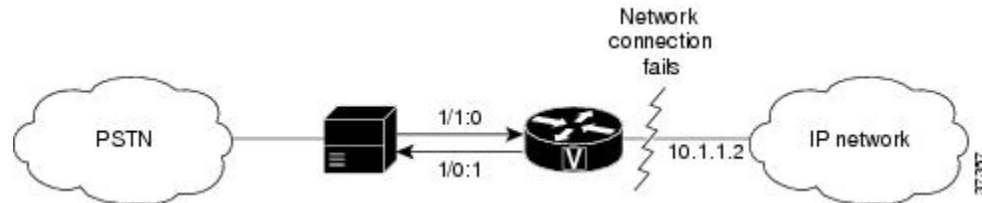
By default, dial peers in a hunt group are selected according to the following criteria, in the order listed:

1. Longest match in phone number--Destination pattern that matches the greatest number of dialed digits. For example, if one dial peer is configured with a dial string of 345.... and a second dial peer is configured with 3456789, the router would first select 3456789 because it has the longest explicit match of the two dial peers.
2. Explicit preference--Priority configured by using the **preference** dial peer command.
3. Random selection--All destination patterns weighted equally.

You can change this default selection order or choose different methods for hunting dial peers by using the **dial-peer hunt** global configuration command. An additional selection criterion is "least recent use," which selects the destination pattern that has waited the longest since being selected.

You can mix POTS and voice-network dial peers when creating hunt groups. Mixing dial peer types can be useful if you want incoming calls to be sent over the packet network, except that if network connectivity fails, you want to reroute the calls back through the PBX to the PSTN. This type of configuration is sometimes referred to as *hairpinning* . Hairpinning is illustrated in the figure below.

Figure 15: Voice Call Using Hairpinning



The following configuration shows an example of sending calls to the PSTN if the IP network fails:

```
dial-peer voice 101 voip
 destination-pattern 472....
 session target ipv4:192.168.100.1
 preference 0
!
dial-peer voice 102 pots
 destination-pattern 472....
 prefix 472
 port 1/0:1
 preference 1
```

You cannot use the same preference numbers for POTS and voice-network dial peers within a hunt group. You can set a separate preference order for each dial peer type, but the preference order does not work on both at the same time. For example, you can configure preference order 0, 1, and 2 for POTS dial peers, and you can configure preference order 0, 1, and 2 for the voice-network dial peers, but the two preference orders are separate. The system resolves preference orders among POTS dial peers first.

## Configuring Dial Peer Hunting Options

Dial peer hunting is enabled by default. The purpose of this task is to disable dial peer hunting on an individual dial peer.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dial-peer voice** *number* {pots | voip | vofr | voatm}
4. **huntstop**

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>enable</b> <b>Example:</b> Router> enable	Enters privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
Step 2	<b>configure terminal</b> <b>Example:</b> Router# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
<b>Step 3</b>	<b>dial-peer voice</b> <i>number</i> {pots   voip   vofr   voatm} <b>Example:</b> Router(config)# dial-peer voice 345 pots	Enters dial-peer voice configuration mode and defines a local dial peer. <ul style="list-style-type: none"> <li>The <i>number</i> argument identifies the dial peer. Valid entries are from 1 to 2147483647.</li> </ul>
<b>Step 4</b>	<b>huntstop</b> <b>Example:</b> Router(config-dialpeer)# huntstop	(Optional) Disables dial-peer hunting on the dial peer. Once you enter this command, no further hunting is allowed if a call fails on the selected dial peer.

## Configuring Dial Peer Hunting Options

Use the **no huntstop** command to enable dial peer hunting if it has been disabled.

The purpose of this task is to configure dial peer hunting options for all dial peers.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dial-peer hunt** *hunt-order-number*
4. **voice hunt** {user-busy | invalid-number | unassigned-number}

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> Router> enable	Enters privileged EXEC mode. <ul style="list-style-type: none"> <li>Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> Router# configure terminal	Enters global configuration mode.
<b>Step 3</b>	<b>dial-peer hunt</b> <i>hunt-order-number</i> <b>Example:</b> Router(config)# dial-peer hunt 2	(Optional) Specifies the hunt selection order for dial peers in a hunt group. Valid entries are 0 through 7. The default is 0. The allowable values are as follows: <ul style="list-style-type: none"> <li>0--Specifies longest match in phone number, explicit preference, random selection.</li> <li>1--Specifies longest match in phone number, explicit preference, least recent use.</li> <li>2--Specifies explicit preference, longest match in phone number, random selection.</li> </ul>

	Command or Action	Purpose
		<ul style="list-style-type: none"> <li>• 3--Specifies explicit preference, longest match in phone number, least recent use.</li> <li>• 4--Specifies least recent use, longest match in phone number, explicit preference.</li> <li>• 5--Specifies least recent use, explicit preference, longest match in phone number.</li> <li>• 6--Specifies random selection.</li> <li>• 7--Specifies least recent use.</li> </ul>
Step 4	<p><b>voice hunt</b> {<b>user-busy</b>   <b>invalid-number</b>   <b>unassigned-number</b>}</p> <p><b>Example:</b></p> <pre>Router(config)# voice hunt user-busy</pre>	<p>(Optional) Defines how the originating or tandem router handles rotary dial peer hunting if it receives a disconnect cause code from the terminating router. The keywords are as follows:</p> <ul style="list-style-type: none"> <li>• <b>user-busy</b> --Instructs the router to continue dial peer hunting if it receives a user-busy disconnect cause code from a destination router.</li> <li>• <b>invalid-number</b> --Instructs the router to stop dial peer hunting if it receives a an invalid-number disconnect cause code from a destination router.</li> <li>• <b>unassigned-number</b> --Instructs the router to stop dial peer hunting if it receives an unassigned-number disconnect cause code from a destination router.</li> </ul>

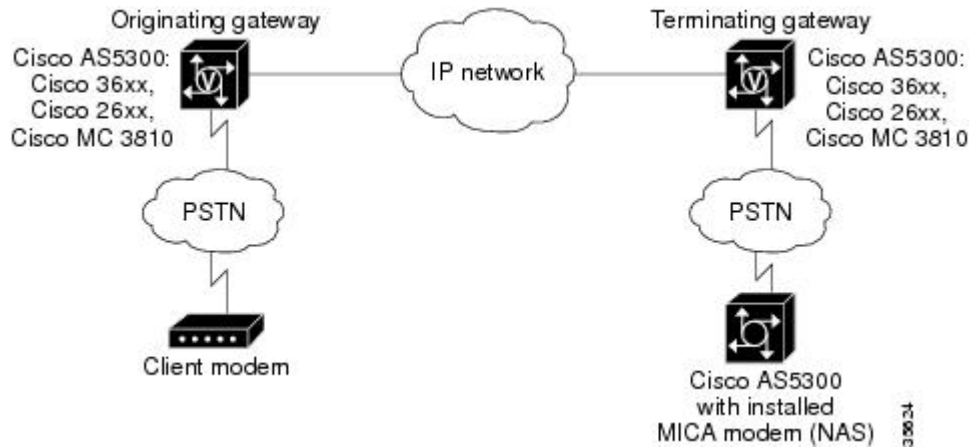
## Modem Pass Through

Like T.38 Fax Relay and Modem Relay, Modem Pass Through functionality can be enabled and configured on a per-dial peer basis. Modem Pass Through behavior enables you to take advantage of features such as the following:

- Repressing bandwidth- and resource-consuming functions like compression, echo cancellation, high-pass filtering, and voice activity detection (VAD).
- Automatically sending redundant packets to minimize the possibility of packet loss.
- Employing automatic static jitter buffers to protect against clock skew.
- Identifying signals that are for modem calls versus voice or fax calls.

The figure below illustrates a network featuring Modem Pass Through capability.

Figure 16: Modem Pass Through Connection Example



When a call over the network is identified as a modem call, both the originating and terminating voice gateway routers automatically "roll over" to using the G.711 codec for the duration of the modem call. Once the modem call has ceased, the digital signal processors (DSPs) in both the originating and terminating voice gateways revert to default operation, enabling fax and voice calls to be placed and received using those DSPs. The version of the G.711 codec you use (either a-law or u-law) is determined by the type of network on which your voice gateways are operating and the configuration you specify using the **modem passthrough** command in dial-peer voice configuration mode.

## Configuring Modem Pass Through Capability for Individual Dial Peers

The purpose of this task is to configure Modem Pass Through capability for individual dial peers.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dial-peer voice** *number* {pots | voip | vofr | voatm}
4. **modem passthrough** {system | nse [payload-type *number*] codec {g711ulaw | g711alaw}[redundancy]}

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>enable</b> <b>Example:</b> Router> enable	Enters privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
Step 2	<b>configure terminal</b> <b>Example:</b> Router# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	<p><b>dial-peer voice</b> <i>number</i> {pots   voip   vofr   voatm}</p> <p><b>Example:</b></p> <pre>Router(config)# dial-peer voice 123 pots</pre>	<p>Enters dial-peer voice configuration mode and defines a local dial peer.</p> <ul style="list-style-type: none"> <li>The <i>number</i> argument identifies the dial peer. Valid entries are from 1 to 2147483647.</li> </ul>
Step 4	<p><b>modem passthrough</b> {system   nse [payload-type <i>number</i>] codec {g711ulaw   g711alaw}[redundancy]}</p> <p><b>Example:</b></p> <pre>Router (config-dialpeer)# modem passthrough codec g711ulaw</pre>	<p>Configures the Modem Pass Through feature for a specific dial peer. The keywords and argument are as follows:</p> <ul style="list-style-type: none"> <li><b>system</b> --Defaults to the global configuration.</li> </ul> <p><b>Note</b> When the <b>system</b> keyword is used, the <b>nse</b>, <b>payload-type</b>, <b>codec</b>, and <b>redundancy</b> keywords are not valid. Instead, the values from the global configuration are used.</p> <ul style="list-style-type: none"> <li><b>nse</b> --Named signaling event.</li> <li><b>payload-type</b> --(Optional) NSE payload type. The <i>number</i> argument specifies the value of the payload type. Valid range is from 96 to 119, inclusive. The default value is 100.</li> </ul> <p><b>Note</b> When the payload type is 100, and you use the <b>show running-config</b> command, the <b>payload-type</b> parameter does not appear in the output.</p> <ul style="list-style-type: none"> <li><b>codec</b> --Voice compression for speech or audio signals. Codec selections for upspeed. The upspeed method is the method used to dynamically change the codec type and speed to meet network conditions. This means that you might move to a faster codec when you have both voice and data calls, and then slow down when there is only voice traffic.</li> <li><b>g711ulaw</b> --Codec G.711 u-law 64000 bps for T1.</li> <li><b>g711alaw</b> --Codec G.711 a-law 64000 bps for E1.</li> </ul> <p><b>Note</b> Be sure to use the same codec type for both the originating gateway and the terminating gateway. The g711ulaw codec is required for T1, and the g711alaw codec is required for E1.</p> <ul style="list-style-type: none"> <li><b>redundancy</b> --(Optional) Enables packet redundancy (RFC 2198) for modem traffic.</li> <li>The default behavior for Modem Pass Through behavior in dial-peer voice configuration mode is <b>modem passthrough system</b>.</li> </ul>

## Dual Tone Multifrequency Relay

Dual tone multifrequency (DTMF) tones are generated when a button on a touch-tone phone is pressed. When the tone is generated, it is compressed, transported to the other party, and then decompressed. If a low-bandwidth codec, such as G.729 or G.723, is used without a DTMF relay method, the tone may be distorted during compression and decompression.

DTMF relay sends DTMF tones out of band, or separately from the voice stream. Cisco gateways currently support the following methods of DTMF relay:

- Using a Cisco-proprietary Real-Time Transport Protocol (RTP). DTMF tones are sent in the same RTP channel as voice data. However, the DTMF tones are encoded differently from the voice samples and are identified by a different RTP payload type code. Use of this method accurately transports DTMF tones, but because it is proprietary, it requires the use of Cisco gateways at both the originating and terminating endpoints of the H.323 call.
- Using either the H.245 signal or H.245 alphanumeric method. These methods separate DTMF digits from the voice stream and send them through the H.245 signaling channel instead of through the RTP channel. The tones are transported in H.245 User Input Indication messages. The H.245 signaling channel is a reliable channel, so the packets that transport the DTMF tones are guaranteed to be delivered. However, because of the overhead of using a reliable protocol, and depending on network congestion conditions, the DTMF tones may be slightly delayed. All H.323 version 2-compliant systems are required to support the "h245-alphanumeric" method: support of the "h245-signal" method is optional.
- Using Named Telephone Events (NTEs). Using NTE to relay DTMF tones provides a standardized means of transporting DTMF tones in RTP packets according to section 3 of *RFC 2833, RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals*, developed by the IETF Audio/Video Transport (AVT) working group. RFC 2833 defines formats of NTE RTP packets used to transport DTMF digits, hookflash, and other telephony events between two peer endpoints. With the NTE method, the endpoints perform per-call negotiation of the DTMF relay method. They also negotiate to determine the payload type value for the NTE RTP packets.

Cisco H.323 gateways advertise capabilities using the H.245 capabilities messages. By default, they advertise that they can receive all DTMF relay modes. If the capabilities of the remote gateway do not match, the Cisco H.323 gateway sends DTMF tones as in-band voice. Configuring DTMF relay on the Cisco H.323 gateway sets preferences for how the gateway handles DTMF transmission. If multiple methods are configured, the priority is as follows:

- Cisco RTP
- RTP NTE
- H.245 signal
- H.245 alphanumeric

In addition to support for NTE, DTMF Relay provides support for asymmetrical payload types. Payload types can differ between local and remote endpoints. Therefore, the Cisco gateway can transmit one payload type value and receive a different payload type value.

## Configuring DTMF Relay and Payload Type

The purpose of this task is to configure DTMF Relay capability and the respective payload type for individual dial peers.



## SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `dial-peer voice number {voip | vofr}`
4. `dtmf-relay [cisco-rtp][h245-alphanumeric] [h245-signal]`
5. `rtp payload-type nte number`

## DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p><b>Example:</b></p> <pre>Router&gt; enable</pre>	<p>Enters privileged EXEC mode.</p> <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
Step 2	<p><code>configure terminal</code></p> <p><b>Example:</b></p> <pre>Router# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>dial-peer voice number {voip   vofr}</code></p> <p><b>Example:</b></p> <pre>Router(config)# dial-peer voice 123 voip</pre>	<p>Enters dial-peer voice configuration mode and defines a remote VoIP dial peer. The keywords and arguments are as follows:</p> <ul style="list-style-type: none"> <li>• The <i>number</i> argument is one or more digits that identify the dial peer. Valid entries are from 1 to 2147483647.</li> <li>• The <b>voip</b> keyword indicates a VoIP peer that uses voice encapsulation on the IP network.</li> </ul>
Step 4	<p><code>dtmf-relay [cisco-rtp][h245-alphanumeric] [h245-signal]</code></p> <p><b>Example:</b></p> <pre>Router(config-dialpeer)# dtmf-relay cisco-rtp</pre>	<p>Allows DTMF relay using Cisco proprietary RTP packets. DTMF tones are encoded in the Cisco proprietary format and transported in the same RTP channel as the voice.</p>
Step 5	<p><code>rtp payload-type nte number</code></p> <p><b>Example:</b></p> <pre>Router(config-dialpeer)# rtp payload-type nte 100</pre>	<p>Identifies the payload type of an RTP packet.</p> <ul style="list-style-type: none"> <li>• The <b>nte</b> keyword identifies the payload type as an NTE. Number values are 96 through 127. The default value is 101.</li> <li>• The following numbers have preassigned values: <ul style="list-style-type: none"> <li>• 96</li> <li>• 97</li> <li>• 100</li> <li>• 121 to 123</li> <li>• 125 to 127</li> </ul> </li> </ul>

	Command or Action	Purpose
		<ul style="list-style-type: none"> <li>If you use these values for the NTE payload type, the command fails. You must first reassign the value in use to a different unassigned number, for example:</li> </ul> <pre>rtp payload-type nse 105 rtp payload-type nte 100</pre>

## Connection PLAR

You can configure dial peers on your Cisco voice gateway router to take advantage of one-way and two-way private-line automatic ringdown (PLAR) functionality. By using the **connection plar** command you can enhance your voice network to offer a number of useful features including the following:

- Providing an off-premises extension (OPX) from a PBX, thus simulating direct connections between FXS port users on a voice gateway router and the PBX.
- Providing dial-tone from a remote PBX in order to offer toll-bypass functionality. Instead of relying on the gateway routers in your voice network to provide dial-tone, you can employ PLAR behavior to enable remote sites to behave as though they have a direct connection to a PBX.
- Connection PLAR behavior eliminates the need for user dialing, because both of the endpoints for the VoIP call are statically configured.

In addition to the features described, connection PLAR behavior does not dedicate bandwidth to a call unless one or the other of the privately associated endpoints goes off-hook.



### Note

The Connection PLAR and Connection Trunk feature behavior is configured on a per-voice-port basis. Therefore, you cannot employ the same voice port for both Connection PLAR or trunk mode and collect-dialed-digits mode.

## Configuring Connection PLAR

The purpose of this task is to enable Connection PLAR on the specified dial peer.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice-port** *port-number*
4. **connection plar** *number*
5. **exit**
6. **dial-peer voice** *number* {**voip** | **pots**}
7. **port** *port-number*

## DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>enable</b> <b>Example:</b>  Router> enable	Enters privileged EXEC mode.  • Enter your password if prompted.
Step 2	<b>configure terminal</b> <b>Example:</b>  Router# configure terminal	Enters global configuration mode.
Step 3	<b>voice-port</b> <i>port-number</i> <b>Example:</b>  Router(config)# voice-port 1/0/0	Enters voice-port configuration mode for the voice port specified.
Step 4	<b>connection plar</b> <i>number</i> <b>Example:</b>  Router(config-voiceport)# connection plar 51234	Configures the specified voice port to use the Connection PLAR feature to automatically dial the specified digits.
Step 5	<b>exit</b> <b>Example:</b>  Router(config-voiceport)# exit	Returns to global configuration mode.
Step 6	<b>dial-peer voice</b> <i>number</i> { <b>voip</b>   <b>pots</b> } <b>Example:</b>  Router(config)# dial-peer voice 123 voip	Enters dial-peer voice configuration mode to configure a VoIP or POTS dial peer.
Step 7	<b>port</b> <i>port-number</i> <b>Example:</b>  Router(config-dial-peer)# port 1/0/0	Configures the dial peer from the previous step to contact the VoIP network using the specified voice port.

## Connection PLAR Design Considerations

Take the following items into consideration when planning to configure Connection PLAR behavior on your VoIP network:

- Because Connection PLAR is a switched VoIP call (similar to a switched virtual circuit), calls are set up and torn down as needed--bandwidth is taken up only when a call is initiated.
- Connection PLAR will operate between any types of signaling endpoints--E&M, FXO, and FXS--and between any combination of analog and digital interfaces.

- Connection PLAR does not collect digits from the connected telephony device, so you can configure Connection PLAR without any subsequent changes to your dial plan.
- Connection PLAR can be enabled on one or both of the statically configured endpoints, thus allowing you the ability to use one-way or two-way Connection PLAR.




---

**Note** Because automatically forwarded digits can become distorted over the compressed VoIP audio path when low bit rate codecs like G.729 and G.723.1 are used, you can use the **dtmf relay** command to transport dual tone multifrequency (DTMF) tones out-of-band (separate from the compressed VoIP audio path) to avoid this potential problem. For more information regarding DTMF relay configuration, refer to [Dual Tone Multifrequency Relay, on page 56](#).

---

## Connection Trunk

In addition to configuring Connection PLAR, you can configure your dial peers to employ the Connection Trunk feature using the **connection trunk** command. Connection Trunk functionality offers some of the same advantages as Connection PLAR, such as eliminating the need for user dialing because both of the endpoints for the VoIP call are statically configured. In addition, Connection Trunk behavior offers you the ability to pass supplemental call signaling capability like hookflash and point-to-point Hoot-n-Holler between endpoints on the trunk.

However, unlike Connection PLAR, Connection Trunk endpoints are always active (or off-hook) as far as the voice network is concerned, so bandwidth is always being allocated to a trunk that you have configured.




---

**Note** The Connection PLAR and Connection Trunk feature behavior is configured on a per-voice-port basis. Therefore, you cannot employ the same voice port for both Connection PLAR or trunk mode and collect-dialed-digits mode.

---

## Configuring Connection Trunk

The purpose of this task is to enable Connection Trunk behavior on the specified dial peer.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice-port** *port-number*
4. **connection trunk** *number*
5. **exit**
6. **dial-peer voice** *number* {**voip** | **pots**}
7. **port** *port-number*

## DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>enable</b> <b>Example:</b> <pre>Router&gt; enable</pre>	Enters privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
Step 2	<b>configure terminal</b> <b>Example:</b> <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 3	<b>voice-port</b> <i>port-number</i> <b>Example:</b> <pre>Router(config)# voice-port 1/0/0</pre>	Enters voice-port configuration mode for the voice port specified.
Step 4	<b>connection trunk</b> <i>number</i> <b>Example:</b> <pre>Router(config-voiceport)# connection trunk 51234</pre>	Configures the specified voice port to use the Connection Trunk feature to automatically dial the specified digits.
Step 5	<b>exit</b> <b>Example:</b> <pre>Router(config-voiceport)# exit</pre>	Returns to global configuration mode.
Step 6	<b>dial-peer voice</b> <i>number</i> { <b>voip</b>   <b>pots</b> } <b>Example:</b> <pre>Router(config)# dial-peer voice 123 voip</pre>	Enters dial-peer configuration mode to configure a VoIP or POTS dial peer.
Step 7	<b>port</b> <i>port-number</i> <b>Example:</b> <pre>Router(config-dial-peer)# port 1/0/0</pre>	Configures the dial peer from the previous step to contact the VoIP network using the specified voice port.

## What to do next



**Note** Because automatically forwarded digits can become distorted over the compressed VoIP audio path when low bit rate codecs like G.729 and G.723.1 are used, you can use the **dtmf relay** command to transport DTMF tones out-of-band (separate from the compressed VoIP audio path) to avoid this potential problem. For more information regarding DTMF relay configuration, refer to [Dual Tone Multifrequency Relay, on page 56](#).

## Class of Restrictions

The Class of Restrictions (COR) feature provides the ability to deny certain call attempts based on the incoming and outgoing class of restrictions provisioned on the dial peers. This functionality provides flexibility in network design, allows users to block calls (for example, to 900 numbers), and applies different restrictions to call attempts from different originators.

COR is used to specify which incoming dial peer can use which outgoing dial peer to make a call. Each dial peer can be provisioned with an incoming and an outgoing COR list. The incoming COR list indicates the capability of the dial peer to initiate certain classes of calls. The outgoing COR list indicates the capability required for an incoming dial peer to deliver a call via this outgoing dial peer. If the capabilities of the incoming dial peer are not the same or a superset of the capabilities required by the outgoing dial peer, the call cannot be completed using this outgoing dial peer.

A typical application of COR is to define a COR name for the number that an outgoing dial peer serves, then define a list that contains only that COR name, and assign that list as **corlist outgoing** for this outgoing dial peer. For example, dial peer with destination pattern 5T can have a **corlist outgoing** that contains COR 5x, as shown in the following configuration.

The next step, in the typical application, is to determine how many call permission groups are needed, and define a COR list for each group. For example, group A is allowed to call 5x and 6x, and group B is allowed to call 5x, 6x, and 1900x. Then, for each incoming dial peer, we can assign a group for it, which defines what number an incoming dial peer can call. Assigning a group means assigning a **corlist incoming** to this incoming dial peer.

```
dial-peer cor custom
  name 5x
  name 6x
  name 1900x
!
dial-peer cor list listA
  member 5x
  member 6x
!
dial-peer cor list listB
  member 5x
  member 6x
  member 1900x
!
dial-peer cor list list5x
  member 5x
!
dial-peer cor list list6x
  member 6x
!
dial-peer cor list list1900x
  member 1900x
! outgoing dialpeer 100, 200, 300
dial-peer voice 100 pots
  destination-pattern 5T
  corlist outgoing list5x
dial-peer voice 200 pots
  destination-pattern 6T
  corlist outgoing list6x
dial-peer voice 300 pots
  destination-pattern 1900T
  corlist outgoing list1900x
!
! incoming dialpeer 400, 500
```

```
dial-peer voice 400 pots
  answer-address 525...
  corlist incoming listA
dial-peer voice 500 pots
  answer-address 526
  corlist incoming listB
```



**Note** To configure classes of restrictions for dial peers, use the following commands beginning in global configuration mode:

## SUMMARY STEPS

1. Router(config)# **dial-peer cor custom**
2. Router(config-dp-cor)# **name** *class-name*
3. Router(config-dp-cor)# **exit**
4. Router(config)# **dial-peer cor list** *list-name*
5. Router(config-dp-corlist)# **member** *class-name*
6. Router(config-dp-corlist)# **exit**
7. Router(config)# **dial-peer voice** *number* {**pots** | **voip**}
8. Router(config-dial-peer)# **corlist incoming** *cor-list-name*
9. Router(config-dial-peer)# **corlist outgoing** *cor-list-name*

## DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	Router(config)# <b>dial-peer cor custom</b>	Enters COR configuration mode to specify that named class of restrictions apply to dial peers.
<b>Step 2</b>	Router(config-dp-cor)# <b>name</b> <i>class-name</i>	Specifies a name for a custom class of restrictions.  <b>Note</b> Enter the <b>name</b> (dial peer cor custom) command for additional class names, as needed. You can define a maximum of 64 COR names. These class names are used to define the COR lists configured in Step 4 and Step 5.
<b>Step 3</b>	Router(config-dp-cor)# <b>exit</b>	Exits COR configuration mode.
<b>Step 4</b>	Router(config)# <b>dial-peer cor list</b> <i>list-name</i>	Defines a COR list name.
<b>Step 5</b>	Router(config-dp-corlist)# <b>member</b> <i>class-name</i>	Adds a COR class to this list of restrictions. The member is a class named in Step 2.  <b>Note</b> Enter the <b>dial-peer cor list</b> command and <b>member</b> (dial peer cor list) command (Step 4 and Step 5) to define another list and its membership, as needed.
<b>Step 6</b>	Router(config-dp-corlist)# <b>exit</b>	Exits COR-list configuration mode.

	Command or Action	Purpose
<b>Step 7</b>	Router(config)# <b>dial-peer voice</b> <i>number</i> { <b>pots</b>   <b>voip</b> }	Enters dial-peer configuration mode and defines a dial peer.
<b>Step 8</b>	Router(config-dial-peer)# <b>corlist incoming</b> <i>cor-list-name</i>	Specifies the COR list to be used when this is the incoming dial peer.
<b>Step 9</b>	Router(config-dial-peer)# <b>corlist outgoing</b> <i>cor-list-name</i>	Specifies the COR list to be used when this is the outgoing dial peer.  <b>Note</b> Repeat Step 7 through Step 9 for additional dial peers, as needed.

## Verifying Classes of Restrictions

To check the validity of your classes of restrictions configuration, perform the following tasks:

- Enter the **show dial-peer voice** command to learn whether the COR list fields are set as desired on a dial peer:

```
Router# show dial-peer voice 210
VoiceEncapPeer210
  information type = voice,
  tag = 210, destination-pattern = `221',
  answer-address = `', preference=0,
  numbering Type = `unknown'
  group = 210, Admin state is up, Operation state is up,
  incoming called-number = `221', connections/maximum = 4/unlimited,
  DTMF Relay = disabled,
  Modem = system passthrough ,
  huntstop = disabled,
  application associated:
  permission :both
  incoming COR list:maximum capability
  outgoing COR list:minimum requirement
  type = pots, prefix = `221',
  forward-digits default
  session-target = `', voice-port = `1/0/8:D',
  direct-inward-dial = enabled,
  digit_strip = enabled,
```

- Enter the **show dial-peer cor** command to display the COR names and lists you defined:

```
Router# show dial-peer cor

Class of Restriction
  name:900block
  name:800_call
  name:Catchall
COR list <list1>
  member:900block
  member:800_call
COR list <list2>
  member:900block
COR list <list3>
  member:900block
  member:800_call
  member:Catchall
```



## Configuring an iLBC Codec

This section includes the following tasks:

### Configuring an iLBC Codec on a Dial Peer

The internet Low Bit-rate Codec (iLBC) is intended for packet-based communication. Perform the following steps to configure the iLBC codec on a dial peer.

#### Before you begin

iLBC is supported on the following:

- Cisco AS5350XM and Cisco AS5400XM Universal Gateways with Voice Feature Cards (VFCs)
- IP-to-IP gateways with no transcoding and conferencing.

#### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dial-peer voice** *tag* **voip**
4. **rtp payload-type** **cisco-codec-ilbc** [*number*]
5. **codec ilbc** [*mode frame\_size*[*bytes payload\_size*]]
6. **exit**

#### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>enable</b> <b>Example:</b> <pre>Router&gt; enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
Step 2	<b>configure terminal</b> <b>Example:</b> <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 3	<b>dial-peer voice</b> <i>tag</i> <b>voip</b> <b>Example:</b> <pre>Router(config)# dial-peer voice 10 voip</pre>	Enters dial-peer configuration mode for the VoIP dial peer designated by <i>tag</i> .
Step 4	<b>rtp payload-type</b> <b>cisco-codec-ilbc</b> [ <i>number</i> ] <b>Example:</b> <pre>Router(config-dial-peer)# rtp payload-type cisco-codec-ilbc 100</pre>	Identifies the payload type of a Real-Time Transport Protocol (RTP) packet. Keyword and argument are as follows: <ul style="list-style-type: none"> <li>• <b>cisco-codec-ilbc</b> [<i>number</i>] --Payload type is for internet Low Bit Rate Codec (iLBC). Range: 96 to 127. Default: 116.</li> </ul>

	Command or Action	Purpose
		<p><b>Note</b> Do not use the following numbers because they have preassigned values: 96, 97, 100, 117, 121 to 123, and 125 to 127. If you use these values, the command will fail. You must first reassign the value in use to a different unassigned number, for example:</p> <pre>rtp payload-type nse 105 rtp payload-type cisco-codec-ilbc 100</pre>
<b>Step 5</b>	<p><b>codec ilbc</b> [<i>mode frame_size</i>[<i>bytes payload_size</i>]]</p> <p><b>Example:</b></p> <pre>Router(config-dial-peer)# codec ilbc mode 30 bytes 200</pre>	<p>Specifies the voice coder rate of speech for a dial peer. Keywords and arguments are as follows:</p> <ul style="list-style-type: none"> <li>• <b>mode</b> <i>frame_size</i> --The iLBC operating frame mode that will be encapsulated in each packet. Valid entries are 20 (20ms frames for 15.2kbps bit rate) or 30 (30ms frames for 13.33 kbps bit rate). Default is 20.</li> <li>• <b>bytes</b> <i>payload_size</i> --Number of bytes in an RTP packet. For mode 20, valid values are 38 (default), 76, 114, 152, 190, and 228. For mode 30, valid values are 50(default), 100, 150, and 200.</li> </ul>
<b>Step 6</b>	<p><b>exit</b></p> <p><b>Example:</b></p> <pre>Router(config-dial-peer)# exit</pre>	Exits the current mode.

### Troubleshooting Tips

You can use the following commands to check iLBC status:

- **show voice call summary**
- **show voice call status**
- **show voice dsmp stream**
- **show call active voice**
- **show call history voice**
- **show voice dsp** and its extensions
- **show dial-peer voice**
- **show voice dsp channel operational-status**

### What to Do Next

To set the codec preference, see the "Configuring Multiple Codecs" section in the *Cisco IOS H.323 Configuration Guide* .

## Configuring an iLBC Codec in the Voice Class

When using multiple codecs, you must create a voice class in which you define a selection order for codecs; then, you can apply the voice class to VoIP dial peers. The **voice class codec** global configuration command allows you to define the voice class that contains the codec selection order. Then, use the **voice-class codec** dial-peer configuration command to apply the class to individual dial peers.

To configure an iLBC codec in the voice class for multiple-codec selection order, perform the following steps.

Follow these procedures to create a voice class. For the complete dial-peer configuration procedure, see the [Common Practices, on page 19](#).

You can configure more than one voice class codec list for your network. Configure the codec lists and apply them to one or more dial peers based on which codecs (and the order) you want supported for the dial peers. Define a selection order if you want more than one codec supported for a given dial peer.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice class codec** *tag*
4. **codec preference** *value ilbc [mode frame\_size] [bytes payload\_size]*
5. **exit**
6. **dial-peer voice** *tag voip*
7. **voice-class codec** *tag*
8. **exit**

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>enable</b> <b>Example:</b>  Router> enable	Enters privileged EXEC mode. Enter your password if prompted.
Step 2	<b>configure terminal</b> <b>Example:</b>  Router# configure terminal	Enters global configuration mode.
Step 3	<b>voice class codec</b> <i>tag</i> <b>Example:</b>  Router(config)# voice class codec 99	Enters voice-class configuration mode and assigns an identification tag number for a codec voice class. The argument is as follows: <ul style="list-style-type: none"> <li>• <i>tag</i> --Unique identifier on the router. Range: 1 to 10000.</li> </ul>
Step 4	<b>codec preference</b> <i>value ilbc [mode frame_size] [bytes payload_size]</i> <b>Example:</b>	Specifies a list of preferred codecs to use on a dial peer. Keywords and arguments are as follows: <ul style="list-style-type: none"> <li>• <i>value</i> --Order of preference, with 1 being the most preferred and 14 being the least preferred.</li> </ul>

	Command or Action	Purpose
	<pre>Router(config-voice-class)# codec preference 1 ilbc 30 200</pre>	<ul style="list-style-type: none"> <li>• <b>mode</b> <i>frame_size</i> --The iLBC operating frame mode that will be encapsulated in each packet. Valid entries are 20 (20ms frames for 15.2kbps bit rate) or 30 (30ms frames for 13.33 kbps bit rate). Default is 20.</li> <li>• <b>bytes</b> <i>payload_size</i> --Number of bytes in an RTP packet. For mode 20, valid values are 38 (default), 76, 114, 152, 190, and 228. For mode 30, valid values are 50(default), 100, 150, and 200.</li> </ul>
<b>Step 5</b>	<p><b>exit</b></p> <p><b>Example:</b></p> <pre>Router(config-voice-class)# exit</pre>	Exits the current mode.
<b>Step 6</b>	<p><b>dial-peer voice</b> <i>tag</i> <b>voip</b></p> <p><b>Example:</b></p> <pre>Router(config)# dial-peer voice 16 voip</pre>	Enters dial-peer configuration mode for the specified VoIP dial peer.
<b>Step 7</b>	<p><b>voice-class codec</b> <i>tag</i></p> <p><b>Example:</b></p> <pre>Router(config-dial-peer)# voice-class codec 99</pre>	<p>Assigns a previously configured codec selection preference list (the codec voice class that you defined in step 3) to the specified VoIP dial peer.</p> <p><b>Note</b> The <b>voice-class codec</b> command in dial-peer configuration mode contains a hyphen. The <b>voice class</b> command in global configuration mode does not contain a hyphen.</p>
<b>Step 8</b>	<p><b>exit</b></p> <p><b>Example:</b></p> <pre>Router(config-dial-peer)# exit</pre>	Exits the current mode.

## Where to Go for Dial Peer Troubleshooting Information

Dial peer troubleshooting consists of a vast array of techniques applicable to a variety of dial peer configuration practices. This document does not cover the techniques necessary to effectively troubleshoot potential pitfalls relating to dial peer configuration. Instead, refer to the TAC Web Site at <http://www.cisco.com/public/support/tac/technologies.shtml>.



## CHAPTER 5

# PSTN Fallback

The PSTN Fallback feature monitors congestion in the IP network and redirects calls to the Public Switched Telephone Network (PSTN) or rejects calls on the basis of network congestion. This feature can also use the ICMP ping mechanism to detect loss of network connectivity and then reroute calls. The fallback subsystem has a network traffic cache that maintains the Calculated Planning Impairment Factor (ICPIF) or delay/loss values for various destinations. Performance is improved because each new call to a well-known destination does not have to wait on a probe to be admitted and the value is usually cached from a previous call.

ICPIF calculates an impairment factor for every piece of equipment along the voice path and then adds them up to get the total impairment value. Refer to International Telecommunication Union (ITU) standard G.113 for more information. The ITU assigns a value to the types of impairment, such as noise, delay, and echo.

### Feature Information

Not all commands may be available in your Cisco IOS software release. For release information about a specific command, see the command reference documentation.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

- [Information About PSTN Fallback, on page 69](#)
- [Restrictions for PSTN Fallback, on page 71](#)
- [How to Configure PSTN Fallback, on page 71](#)
- [How to Verify and Monitor the PSTN Fallback Feature, on page 84](#)
- [What To Do Next, on page 84](#)

## Information About PSTN Fallback

The VoIP Alternate Path Fallback SNMP Trap feature adds a Simple Network Management Protocol (SNMP) trap generation capability. This feature is built on top of the fallback subsystem to provide an SNMP notification trap when the fallback subsystem redirects or rejects a call because a network condition has failed to meet the configured threshold. The SNMP trap provides VoIP management status MIB information without flooding management systems with unnecessary messages about call status by triggering only when a call has been redirected to the public switched telephone network (PSTN) or the alternative IP port. A call can be rejected because of a network problem such as loss of WAN connection, delay, packet loss, or jitter. This feature

supports only VoIP signaling protocol with H.323 in this release. This feature has to be configured on the originating gateway and the terminating gateway.

The **call fallback map** command option provides a target network summary/consolidation mode. For example, if there are four individual voice gateway routers connected together on a remote LAN via a separate LAN-to-WAN access router, the map option allows a single probe to be sent to the single remote WAN access router (instead of having to maintain separate probes for each of the four voice gateway routers' IP addresses). Because the remote access and voice gateway routers are connected together on the same remote LAN, the probes to the access router returns similar results to probes to the individual voice gateway routers.

## Service Assurance Agent

Service Assurance Agent (SAA) is a network congestion analysis mechanism that provides delay, jitter, and packet loss information for the configured IP addresses. SAA is based on a client/server protocol defined on the User Datagram Protocol (UDP). UDP is a connectionless transport layer protocol in the IP protocol stack. UDP is a simple protocol that exchanges datagrams without acknowledgments or guaranteed delivery, requiring that error processing and retransmission be handled by other protocols. The SAA probe packets go out on randomly selected ports from the top end of the audio UDP port range.

The information that the SAA probes gather is used to calculate the ICPIF or delay/loss values that are stored in a fallback cache, where they remain until the cache ages out or overflows. Until an entry ages out, probes are sent periodically for that particular destination. This time interval is user configurable.

With this feature enhancement, you can also configure codes that indicate the cause of the network rejection; for example, packets that are lost or that take too long to be transmitted. A default cause code of 49 displays the message **qos-unavail**, which means Quality of Service is unavailable.



### Note

The Cisco SAA functionality in Cisco IOS software was formerly known as Response Time Reporter (RTR). In the [How to Configure PSTN Fallback, on page 71](#) section, note that the command-line interface still uses the keyword **rtr** for configuring RTR probes, which are now actually the SAA probes.

## Application of PSTN Fallback

The PSTN Fallback feature and enhancement provide the following benefits:

- Automatically re-routes calls when the data network is congested at the time of the call setup.
- Enables the service provider to give a reasonable guarantee about the quality of the conversation to its Voice over IP (VoIP) users at the time of call admission.
- Provides delay, jitter, and packet loss information for the configured IP addresses.
- Caches call values from previous calls. New calls do not have to wait for probe results before they are admitted.
- Enables a user-configurable cause code display that indicates the type of call rejection.

## Restrictions for PSTN Fallback

The PSTN Fallback feature has the following restrictions:

- When detecting network congestion, the PSTN fallback feature does nothing to the existing call. It affects only subsequent calls.
- Only a single ICPIF/delay-loss value is allowed per system.
- A small additional call setup delay can be expected for the first call to a new IP destination.



### Caution

Configuring **call fallback active** in a gateway creates an SAA jitter probe against other (target) gateways to which the calls are sent. In order for the call fallback active to work properly, the target gateways must have the **rtr responder** command (in Cisco IOS releases prior to 12.3(14)T) or the **ip sla monitor responder** command (in Cisco IOS Release 12.3(14)T or later) in their configurations. If one of these commands is not included in the configuration of each target gateway, calls to the target gateway will fail.

## How to Configure PSTN Fallback

### Configuring Call Fallback to Use MD5 Authentication for SAA Probes

To configure call fallback to use MD5 authentication for SAA probes, use the following commands.

#### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **call fallback active**
4. **call fallback key-chain** *name-of-chain*

#### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>enable</b> <b>Example:</b> Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
Step 2	<b>configure terminal</b> <b>Example:</b> Router# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
<b>Step 3</b>	<b>call fallback active</b> <b>Example:</b> <pre>Router(config)# call fallback active</pre>	Enables the PSTN fallback feature to alternate dial peers in case of network congestion.
<b>Step 4</b>	<b>call fallback key-chain <i>name-of-chain</i></b> <b>Example:</b> <pre>Router(config)# call fallback key-chain sample</pre>	Specifies the use of message digest algorithm 5 (MD5) authentication for sending and receiving Service Assurance Agents (SAA) probes.

## Configuring Destination Monitoring without Fallback to Alternate Dial Peers

To configure destination monitoring without fallback to alternate dial peers, use the following commands.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **call fallback monitor**

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> <pre>Router&gt; enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> <pre>Router# configure terminal</pre>	Enters global configuration mode.
<b>Step 3</b>	<b>call fallback monitor</b> <b>Example:</b> <pre>Router(config)# call fallback monitor</pre>	Enables the monitoring of destinations without fallback to alternate dial peers.

## Configuring Call Fallback Cache Parameters

To configure the call fallback cache parameters, use the following commands.

### SUMMARY STEPS

1. **enable**



2. **configure terminal**
3. **call fallback cache-size** *number*
4. **call fallback cache-timeout** *seconds*
5. **clear call fallback cache** [*ip-address*]

## DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> <pre>Router&gt; enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> <pre>Router# configure terminal</pre>	Enters global configuration mode.
<b>Step 3</b>	<b>call fallback cache-size</b> <i>number</i> <b>Example:</b> <pre>Router(config)# call fallback cache-size 5</pre>	Specifies the call fallback cache size.
<b>Step 4</b>	<b>call fallback cache-timeout</b> <i>seconds</i> <b>Example:</b> <pre>Router(config)# call fallback cache-timeout 300</pre>	Specifies the time after which the cache entry is purged, in seconds. Default: 600.
<b>Step 5</b>	<b>clear call fallback cache</b> [ <i>ip-address</i> ] <b>Example:</b> <pre>Router(config)# clear call fallback cache 10.1.1.1</pre>	Clears the current ICPIF estimates for all IP addresses or a specific IP address in the cache.

## Configuring Call Fallback Jitter-Probe Parameters

To configure call fallback jitter-probe parameters, use the following commands.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **call fallback jitter-probe num-packets** *number-of-packets*
4. **call fallback jitter-probe precedence** *precedence*
5. **call fallback jitter-probe priority-queue**

## DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> <pre>Router&gt; enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> <pre>Router# configure terminal</pre>	Enters global configuration mode.
<b>Step 3</b>	<b>call fallback jitter-probe num-packets</b> <i>number-of-packets</i> <b>Example:</b> <pre>Router(config)# call fallback jitter-probe num-packets 10</pre>	Specifies the number of packets for jitter. Default: 15.
<b>Step 4</b>	<b>call fallback jitter-probe precedence</b> <i>precedence</i> <b>Example:</b> <pre>or</pre> <b>Example:</b> <pre>call fallback jitter-probe dscp dscp-number</pre> <b>Example:</b> <pre>Router(config)# call fallback jitter-probe precedence 2</pre> <b>Example:</b> <pre>or</pre> <b>Example:</b> <pre>Router(config)# call fallback jitter-probe dscp 2</pre>	Specifies the treatment of the jitter-probe transmission. Default: 2. Specifies the differentiated services code point (dscp) packet of the jitter-probe transmission. <b>Note</b> The <b>call fallback jitter-probe precedence</b> command is mutually exclusive with the <b>call fallback jitter-probe dscp</b> command. Only one of these command can be enabled on the router. Usually, the <b>call fallback jitter-probe precedence</b> command is enabled. When the <b>call fallback jitter-probe dscp</b> command is configured, the precedence value is replaced by the DSCP value. To disable DSCP and restore the default jitter probe precedence value, use the <b>no call fallback jitter-probe dscp</b> command.
<b>Step 5</b>	<b>call fallback jitter-probe priority-queue</b> <b>Example:</b> <pre>Router(config)# call fallback jitter-probe priority-queue</pre>	Assigns a priority to the queue for jitter probes.

## Configuring Call Fallback Probe-Timeout and Weight Parameters

To configure call fallback probe-timeout and weight parameters, use the following commands.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **call fallback probe-timeout** *seconds*
4. **call fallback instantaneous-value-weight** *percent*

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>enable</b> <b>Example:</b> <pre>Router&gt; enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
Step 2	<b>configure terminal</b> <b>Example:</b> <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 3	<b>call fallback probe-timeout</b> <i>seconds</i> <b>Example:</b> <pre>Router(config)# call fallback probe-timeout 20</pre>	Sets the timeout for an SAA probe, in seconds. Default: 30.
Step 4	<b>call fallback instantaneous-value-weight</b> <i>percent</i> <b>Example:</b> <pre>Router(config)# call fallback instantaneous-value-weight 50</pre>	Configures the call fallback subsystem to take an average from the last two probes registered in the cache for call requests: <ul style="list-style-type: none"> <li>• <i>percent</i> --Instantaneous value weight, expressed as a percentage. Range: 0 to 100. Default: 66.</li> </ul>

## Configuring Call Fallback Threshold Parameters

To configure call fallback threshold parameters, use the following commands.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **call fallback threshold delay** *delay-value* **loss** *loss-value*

## DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> <pre>Router&gt; enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> <pre>Router# configure terminal</pre>	Enters global configuration mode.
<b>Step 3</b>	<b>call fallback threshold delay</b> <i>delay-value</i> <b>loss</b> <i>loss-value</i> <b>Example:</b> or <b>Example:</b> <pre>call fallback threshold icpif</pre> <i>threshold-value</i> <b>Example:</b> <pre>Router(config)# call fallback threshold delay 100</pre> loss 150 <b>Example:</b> or <b>Example:</b> <pre>Router(config)# call fallback threshold icpif 100</pre>	Specifies fallback threshold to use packet delay and loss values. No defaults.  <b>Note</b> The amount of delay set by the <b>call fallback threshold delay loss</b> command should not be more than half the amount of the time-to-wait value set by the <b>call fallback wait-timeout</b> command; otherwise the threshold delay will not work correctly. Because the default value of the <b>call fallback wait-timeout</b> command is set to 300 milliseconds, you can configure a delay of up to 150 milliseconds for the <b>call fallback threshold delay loss</b> command. If you want to configure a higher threshold, the time-to-wait delay has to be increased from its default (300 milliseconds) using the <b>call fallback wait-timeout</b> command.  Specifies fallback threshold to use the Calculated Planning Impairment Factor (ICPIF) threshold for network traffic.

## Configuring Call Fallback Wait-Timeout

To configure the call fallback wait-timeout parameters, use the following commands:

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **call fallback wait-timeout** *milliseconds*

## DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>enable</b> <b>Example:</b> <pre>Router&gt; enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
Step 2	<b>configure terminal</b> <b>Example:</b> <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 3	<b>call fallback wait-timeout</b> <i>milliseconds</i> <b>Example:</b> <pre>Router(config)# call fallback wait-timeout 200</pre>	Configures the waiting timeout interval for a response to a probe in milliseconds. Default: 300 milliseconds. <p><b>Note</b> The time-to-wait period set by the <b>call fallback wait-timeout</b> command should always be greater than or equal to twice the amount of the threshold delay time set by the <b>call fallback threshold delay loss</b> command; otherwise the probe will fail. The delay configured by the <b>call fallback threshold delay loss</b> command corresponds to a one-way delay, whereas the time-to-wait period configured by the <b>call fallback wait-timeout</b> command corresponds to a round-trip delay. The threshold delay time should be set at half the value of the time-to-wait value.</p>

## Configuring VoIP Alternate Path Fallback SNMP Trap

To configure the SNMP trap parameters, use the following commands:

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **call fallback active**
4. **snmp-server enable traps voice fallback**

## DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>enable</b> <b>Example:</b> <pre>Router&gt; enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>

	Command or Action	Purpose
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b>  Router# configure terminal	Enters global configuration mode.
<b>Step 3</b>	<b>call fallback active</b> <b>Example:</b>  Router(config)# call fallback active	Enables the PSTN fallback feature to alternate dial peers in case of network congestion.
<b>Step 4</b>	<b>snmp-server enable traps voice fallback</b> <b>Example:</b>  Router(config)# snmp-server enable traps voice fallback	Configures the SNMP trap parameters.

## What to Do Next

Configure the **rtr responder** command on the terminating voice gateway. If the **rtr responder** is enabled on the terminating gateway, the terminating gateway responds to the probe request when the originating gateway sends an Response Time Report (RTR) probe to the terminating gateway to check the network conditions.

## Configuring Call Fallback Map Parameters

To configure call fallback map parameters, use the following commands.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. Do one of the following:
  - **call fallback map** *map* **target** *ip-address* **address-list** *ip-address1 ip-address2 ... ip-address7*
  - 
  - **call fallback map** *map* **target** *ip-address* **subnet** *ip-network netmask*

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b>  Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b>	Enters global configuration mode.

	Command or Action	Purpose
	Router# configure terminal	
<b>Step 3</b>	<p>Do one of the following:</p> <ul style="list-style-type: none"> <li>• <b>call fallback map</b> <i>map</i> <b>target</b> <i>ip-address address-list ip-address1 ip-address2 ... ip-address7</i></li> <li>•</li> <li>• <b>call fallback map</b> <i>map</i> <b>target</b> <i>ip-address subnet ip-network netmask</i></li> </ul>	<p>Specifies the call fallback router to keep a cache table (by IP addresses) of distances for several destination peers sitting behind the router.</p> <ul style="list-style-type: none"> <li>• <i>map</i> --Fallback map. Range is from 1 to 16. There is no default.</li> <li>• <b>target</b> <i>ip-address</i> --Target IP address.</li> <li>• <i>ip-address1 ip-address2 ... ip-address7</i> --Lists the IP addresses that are kept in the cache table. The maximum number of IP addresses is seven.</li> </ul> <p>Specifies the call fallback router to keep a cache table (by subnet addresses) of distances for several destination peers sitting behind the router.</p>

## Configuring ICMP Pings to Monitor IP Destinations

This capability to monitor ICMP pings is enabled to monitor the IP destinations in a VoIP network, which may not support RTR. This monitoring is referred to as ICMP ping. Based on the RTR or ICMP ping, results change the operational state of the dial-peer. The configurations described in this section also provide support for monitoring the following session targets configured under a VoIP dial-peer:

- DNS
- IP version 4
- SIP-server
- enum

To configure call-fallback monitor probes to ping IP destinations, complete one of the following tasks:

### Dial Peer Configuration of the call fallback icmp-ping and monitor probe Commands

To configure dial-peer parameters to use ICMP pings to monitor IP destinations, complete this task. This configuration applies only to VoIP dial peers.

#### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dial-peer voice** *tag* **voip**
4. **call fallback** [**icmp-ping** | **rtr**]
5. **monitor probe** {**icmp-ping** | **rtr**} [*ip address*]

## DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> <pre>Router&gt; enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> <pre>Router# configure terminal</pre>	Enters global configuration mode.
<b>Step 3</b>	<b>dial-peer voice tag voip</b> <b>Example:</b> <pre>Router(config)# dial-peer voice 10 voip</pre>	Enters dial peer configuration mode, specifies the method of voice encapsulation, and defines a particular dial peer: <i>tag</i> --Digits that define a particular dial peer. Range is from 1 to 2147483647.
<b>Step 4</b>	<b>call fallback [icmp-ping   rtr]</b> <b>Example:</b> <pre>Router(config-dial-peer)# call fallback icmp-ping</pre>	Configures dial-peer parameters for pings to IP destinations: <ul style="list-style-type: none"> <li>• <b>icmp-ping</b> --Uses ICMP pings to monitor the IP destinations.</li> <li>• <b>rtr</b> --Uses RTR probes to monitor the session target and update the status of the dial peer. RTR probes are the default.</li> </ul> <p><b>Note</b> If this <b>call fallback icmp-ping</b> command is not entered, the <b>call fallback active</b> command in global configuration is used for measurements. If this <b>call fallback icmp-ping</b> command is entered, these values override the global configuration. One of these two commands must be in effect before the <b>monitor probe icmp-ping</b> command can be used. If neither of <b>call fallback</b> commands is in effect, the <b>monitor probe icmp-ping</b> command will not work properly.</p>
<b>Step 5</b>	<b>monitor probe {icmp-ping   rtr} [ip address]</b> <b>Example:</b> <pre>Router(config-dial-peer)# monitor probe icmp-ping</pre>	Enables dial-peer status changes based on the result of the probe: <ul style="list-style-type: none"> <li>• <b>icmp-ping</b> --Uses ICMP ping as the method for the probe.</li> <li>• <b>rtr</b> --Uses RTR as the method for the probe.</li> <li>• <i>ip address</i> --IP address of the destination to be probed. If no IP address is specified, the IP address is read from the session target.</li> </ul>



## Global Configuration of the call fallback icmp-ping Command

To configure global parameters to use ICMP pings to monitor IP destinations, complete this task.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **call fallback active [icmp-ping | rtr]**
4. **call fallback icmp-ping [count number] [codec type] | size bytes] interval seconds [loss number] [timeout milliseconds]**

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>enable</b> <b>Example:</b> <pre>Router&gt; enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
Step 2	<b>configure terminal</b> <b>Example:</b> <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 3	<b>call fallback active [icmp-ping   rtr]</b> <b>Example:</b> <pre>Router(config)# call fallback active icmp-ping</pre>	Configures global parameters for pings to IP destinations: <ul style="list-style-type: none"> <li>• <b>icmp-ping</b> --Uses ICMP pings to monitor the IP destinations.</li> <li>• <b>rtr</b> --Uses RTR probes to monitor the IP destinations. RTR probes are the default.</li> </ul> <p><b>Note</b> The <b>call fallback active icmp-ping</b> command must be entered before the <b>call fallback icmp-ping</b> command can be used. If you do not enter this command first, the <b>call fallback icmp ping</b> command will not work properly.</p>
Step 4	<b>call fallback icmp-ping [count number] [codec type]   size bytes] interval seconds [loss number] [timeout milliseconds]</b> <b>Example:</b> <pre>Router(config)# call fallback icmp ping codec g729 interval 10 loss 10</pre>	Configures the parameters for ICMP pings: <ul style="list-style-type: none"> <li>• <b>count</b> --Number of ping packets to be sent to the destination IP address. Default is 5.</li> <li>• <b>codec</b> --Codec type for deciding the ping packet size.</li> <li>• <b>type</b> --Acceptable codec types are <b>g711a</b>, <b>g711u</b>, <b>g729</b>, and <b>g729b</b>.</li> <li>• <b>size</b> --Size (in bytes) of the ping packet. Default is 32.</li> </ul>

	Command or Action	Purpose
		<ul style="list-style-type: none"> <li>• <b>interval</b> --Time (in seconds) between ping packet sets. Default is 5. This value should be more than the <b>timeout</b> value.</li> <li>• <b>loss</b> --Threshold packet loss, expressed as a percentage. Default is 20.</li> <li>• <b>timeout</b> --Timeout (in milliseconds) for the echo packets. Default is 500.</li> </ul>

## Voice Port Configuration of the busyout monitor probe icmp-ping Command

To configure voice-port parameters to use ICMP pings to monitor IP destinations, complete this task.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice-port slot / port**
4. **busyout monitor probe icmp-ping ip address [codec type | size bytes][loss percent]**

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> Router# configure terminal	Enters global configuration mode.
<b>Step 3</b>	<b>voice-port slot / port</b> <b>Example:</b> Router(config)# voice-port 1/0	Enters voice-port configuration mode and identifies the slot and port where the configuration parameters take effect. <b>Note</b> The syntax for this command varies by platform. For more information, see the Cisco IOS Voice Command Reference.
<b>Step 4</b>	<b>busyout monitor probe icmp-ping ip address [codec type   size bytes][loss percent]</b> <b>Example:</b> Router(config-voiceport)# busyout monitor probe 10.1.1.1 g711u loss 10 delay 2000	Specifies the parameters for ICMP pings for monitoring under voice-port configuration: <ul style="list-style-type: none"> <li>• <b>ip address</b> --IP address of the destination to which the ping is sent.</li> <li>• <b>codec</b> --(Optional) Codec type for deciding the ping packet size.</li> </ul>

	Command or Action	Purpose
		<ul style="list-style-type: none"> <li>• <i>type</i> --Acceptable codec types are <b>g711a</b>, <b>g711u</b>, <b>g729</b>, and <b>g729b</b>.</li> <li>• <i>size</i> --(Optional) Size (in bytes) of the ping packet. Default is 32.</li> <li>• <i>loss</i> --(Optional) Threshold packet loss, expressed as a percentage. Default is 20.</li> </ul>

## Voice Class Configuration of the busyout monitor probe icmp-ping Command

To configure voice-class parameters to use ICMP pings to monitor IP destinations, complete this task.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice class busyout tag**
4. **busyout monitor probe icmp-ping ip address [codec type | size bytes][loss percent]**

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> <pre>Router&gt; enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> <pre>Router# configure terminal</pre>	Enters global configuration mode.
<b>Step 3</b>	<b>voice class busyout tag</b> <b>Example:</b> <pre>Router(config)# voice class busyout 10</pre>	Creates a voice class for local voice busyout functions: <i>tag</i> --Unique identification number assigned to one voice class. Range is 1 to 10000.
<b>Step 4</b>	<b>busyout monitor probe icmp-ping ip address [codec type   size bytes][loss percent]</b> <b>Example:</b> <pre>Router(config-class)# busyout monitor probe icmp-ping 10.1.1.1 codec g729b size 32</pre>	Configures the parameters for ICMP pings for monitoring under voice-port: <ul style="list-style-type: none"> <li>• <i>ip address</i> --IP address of the destination to which the ping is sent.</li> <li>• <i>codec</i> --(Optional) Codec type for deciding the ping packet size.</li> <li>• <i>type</i> --Acceptable codec types are <b>g711a</b>, <b>g711u</b>, <b>g729</b>, and <b>g729b</b>.</li> </ul>

	Command or Action	Purpose
		<ul style="list-style-type: none"> <li>• <b>size</b> --(Optional) Size (in bytes) of the ping packet. Default is 32.</li> <li>• <b>loss</b> --(Optional) Threshold packet loss, expressed as a percentage. Default is 20.</li> </ul>

## How to Verify and Monitor the PSTN Fallback Feature

### Verifying PSTN Fallback Configuration

The **show** commands in this section can be used to display statistics and configuration parameters to verify the operation of the PSTN Callback feature:

- **show running-config** --Displays the contents of the currently running configuration file to see if the new feature is configured.
- **show call history voice** --Displays the call history table for voice calls and verify call fallback, call delay, and call loss parameters.
- **show call fallback cache** --Displays the current Calculated Planning Impairment Factor (ICPIF) estimates for all IP addresses in the call fallback cache.
- **show call fallback config** --Displays the current configuration.
- **show call fallback stats** --Displays the call fallback statistics.

### Monitoring and Maintaining PSTN Fallback

Use the following commands to monitor and maintain the PSTN Fallback feature:

- **clear call fallback cache** --Clears the current ICPIF estimates for all IP addresses in the cache.
- **clear call fallback stats** --Clears the call fallback statistics.
- **debug call fallback detail** --Displays details of VoIP call fallback.
- **debug call fallback probes** --Displays details of voice fallback probes.
- **test call fallback probe** *ip-address* --Tests a probe to a particular IP address and displays the ICPIF SAA values.
- **debug snmp packets** --Displays information about every Simple Network Management Protocol (SNMP) packet sent or received by the router.

## What To Do Next

The [Configuring ICMP Pings to Monitor IP Destinations, on page 79](#) describes the mechanism whereby a dial-peer becomes temporarily disabled because of poor SAA/RTR probe results (for example, ICPIF, jitter,

or loss), or because of failure of the ICMP ping test. When this occurs, the normal alternate dial-peer selection process (hunting) is triggered to search for an alternate dial-peer that represents an alternate route.

The global configuration **voice hunt** command controls whether hunting (continue to look or "hunt" for an alternate dial-peer match) occurs, based on the specific cause code that describes why the initial dial-peer path failed. Hunting is usually appropriate if the cause code indicates network congestion, but usually inappropriate if the failure cause code indicates that the called user is actually busy. Even if an alternate path is taken to reach the called user, and if the user is actually busy, the user will be busy regardless of which path is used.

For more information about the **voice hunt** command, see the Cisco IOS Voice Command Reference.





## CHAPTER 6

# Dial Peer Configuration Examples

This appendix contains a series of configuration examples featuring the minimum required components and critical Cisco IOS command lines extracted from voice gateway configuration files necessary to complete an endpoint-to-endpoint call. Each example is designed to focus on a specific combination of components or configuration concept essential to voice over IP (VoIP) communication. This appendix covers the following topics:

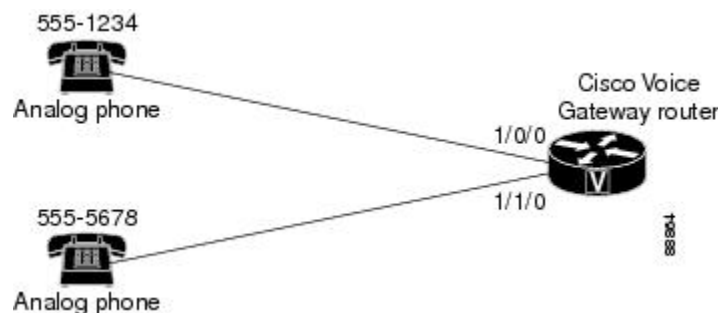
- [Two Analog Phones, on page 87](#)
- [Two Fax Machines, on page 89](#)
- [An Analog Phone and an IP Phone Connected over an IP Network, on page 90](#)
- [Two IP Phones Connected via a Voice over Frame Relay Network, on page 91](#)
- [Using Digit Manipulation to Overcome the Obstacle of an IP Network Failure, on page 93](#)

## Two Analog Phones

The simplest and most ubiquitous implementation of dial peer configuration involves connecting two standard analog telephones over an IP network. The following two examples illustrate the minimum required configurations necessary to connect two analog phones, where they are attached to the same voice gateway router and where each phone is attached to its own voice gateway router via FXS ports installed in the voice gateway routers in question.

## Both Connected to the Same Voice Gateway Router

*Figure 17: Two analog phones connected to the same voice gateway router*



### Voice Gateway Router Configuration File

```
voice-port 1/0/0
!
voice-port 1/0/1
!
!
dial-peer voice 1 pots
 destination-pattern 5551234
 port 1/0/0
!
dial-peer voice 2 pots
 destination-pattern 555678
 port 1/0/1
```

## Each Connected to Their Own Voice Gateway Routers Using the G.711 Codec

Figure 18: Two analog phones each connected to their own voice gateway router



### Voice Gateway Router 1 Configuration File

```
voice-port 1/0/0
!
dial-peer voice 1 pots
 destination-pattern 5551234
 port 1/0/0
!
dial-peer voice 10 voip
 destination-pattern 555678
 session target ipv4:10.5.6.7
 codec g711ulaw
```

### Voice Gateway Router 2 Configuration File

```
voice-port 1/0/0
!
dial-peer voice 2 pots
 destination-pattern 555678
 port 1/0/0
!
dial-peer voice 20 voip
 destination-pattern 5551234
 session target ipv4:10.2.3.4
 codec g711ulaw
```



## Each Connected to Their Own Voice Gateway Routers Using the G.729r8 Codec

### Voice Gateway Router 1 Configuration File

```
voice class codec 1
  codec preference 1 g729r8
  codec preference 2 g711ulaw
!
voice-port 1/0/0
!
voice-port 1/0/1
!
!
dial-peer voice 1 pots
  destination-pattern 5551234
  port 1/0/0
!
dial-peer voice 2 voip
  destination-pattern 5555678
  voice-class codec 1
  session target ipv4:10.5.6.7
```

### Voice Gateway Router 2 Configuration File

```
voice class codec 1
  codec preference 1 g729r8
  codec preference 2 g711ulaw
!
voice-port 1/0/0
!
voice-port 1/0/1
!
!
dial-peer voice 1 pots
  destination-pattern 5555678
  port 1/0/0
!
dial-peer voice 2 voip
  destination-pattern 5551234
  voice-class codec 1
  session target ipv4:10.2.3.4
```

## Two Fax Machines

Once the connection between two analog phones over the IP network can be set up, you can then alter the configuration slightly to enable fax communication over the IP network. The figure below illustrates the configuration files necessary to establish T.38 Fax Relay functionality over the IP network.

**Figure 19: Two fax machines connected via T.38 Fax Relay**



**Voice Gateway Router 1 Configuration File**

```

interface FastEthernet0/0
 ip address 10.21.9.4 255.255.255.0
 !
voice-port 1/0/0
 !
voice-port 1/0/1
 !
dial-peer voice 1 pots
 destination-pattern 5551234
 port 1/0/0
 !
dial-peer voice 2 voip
 destination-pattern 5555678
 session target ipv4:10.5.6.7
 codec g711ulaw
 fax protocol t38 ls-redundancy 0 hs-redundancy 0 fallback cisco
 fax rate voice

```

**Voice Gateway Router 2 Configuration File**

```

interface FastEthernet0/0
 ip address 10.21.7.61 255.255.255.0
 !
voice-port 1/0/0
 !
voice-port 1/0/1
 !
dial-peer voice 1 pots
 destination-pattern 5555678
 port 1/0/0
 !
dial-peer voice 2 voip
 destination-pattern 5551234
 voice-class codec 1
 session target ipv4:10.2.3.4
 codec g711ulaw
 fax protocol t38 ls-redundancy 0 hs-redundancy 0
 fax rate voice

```

# An Analog Phone and an IP Phone Connected over an IP Network

Once you are able to establish the connection of two analog phones over an IP network, you can then expand the scope of configuration coverage to include an analog phone and an IP phone connected over the IP network. The configuration for each of the voice gateway routers is essentially the same as if you were connecting two analog phones; you will need to ensure that you have allowed for a Cisco CallManager server connection to the appropriate Cisco voice gateway router to accommodate the introduction of the IP phone.

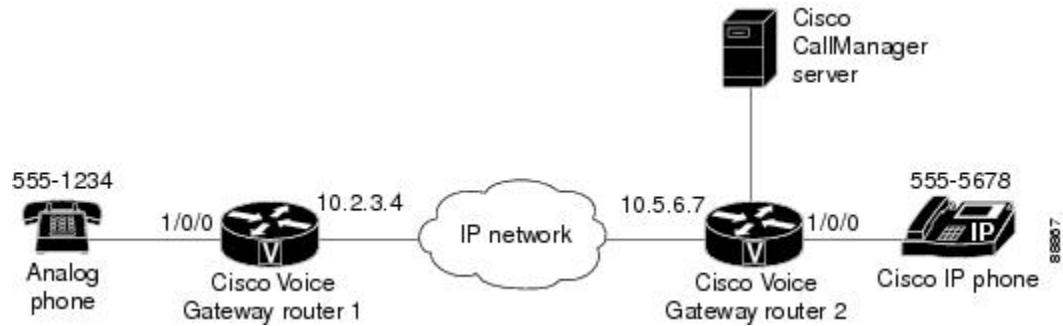



---

**Note** A CallManager server has been used in this configuration to manage the Cisco IP phone. However, this document does not address Cisco CallManager configuration. For more information on Cisco CallManager setup and configuration, refer to the Cisco CallManager documentation available on Cisco.com.

---

**Figure 20: An analog phone and an IP phone each connected to their own voice gateway router**



### Voice Gateway Router 1 Configuration File

```
voice-port 1/0/0
!
dial-peer voice 1 pots
 destination-pattern 5551234
 port 1/0/0
!
dial-peer voice 2 voip
 destination-pattern 5555678
 session target ipv4:10.5.6.7
```

### Voice Gateway Router 2 Configuration File

```
voice-port 1/0/0
!
dial-peer voice 1 pots
 destination-pattern 5555678
 port 1/0/0
!
dial-peer voice 2 voip
 destination-pattern 5551234
 session target ipv4:10.2.3.4
```

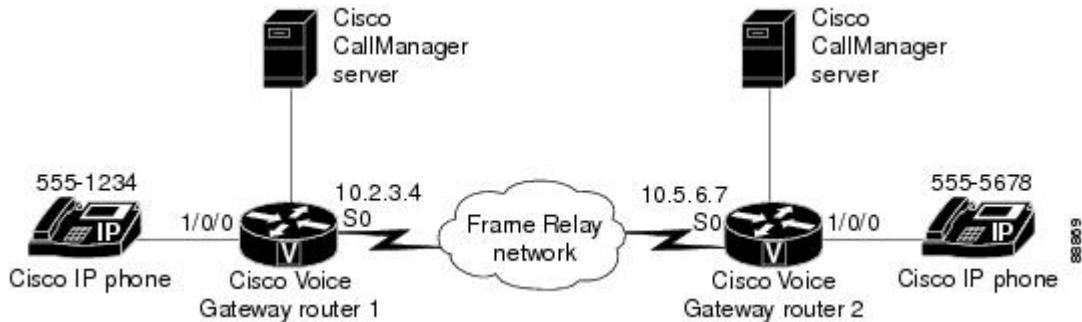
## Two IP Phones Connected via a Voice over Frame Relay Network

The examples thus far in this appendix have described connecting endpoints over an IP network based primarily on Ethernet connections. However, you may find that you must configure a Frame Relay WAN to effectively serve the voice communications demands for your system. The figure below and the subsequent Cisco voice gateway router configuration examples illustrate the dial peer configuration and Frame Relay Cisco IOS commands necessary to enable Frame Relay communication across your IP network.



**Note** Although it is an essential portion of the configuration of this system, and the following configuration files contain the Cisco IOS commands necessary to establish Frame Relay connectivity, Frame Relay configuration is not addressed in this document. For more information on Frame Relay configuration, refer to the appropriate documentation available on Cisco.com.

**Figure 21: Two IP phones connected over a Frame Relay network**



### Voice Gateway Router 1 Configuration File

The configuration for this voice gateway features a Cisco 3620 running Cisco IOS Release 12.2(11)T6.

```
interface Serial0/0
 ip address 10.2.1.1 255.0.0.0
 encapsulation frame-relay
 !
 voice-port 1/0/0
 !
 voice-port 1/0/1
 !
 voice-port 1/1/0
 !
 voice-port 1/1/1
 !
 dial-peer cor custom
 !
 dial-peer voice 1 pots
 destination-pattern 5551234
 port 1/0/0
 !
 dial-peer voice 1000 voip
 preference 1
 destination-pattern 5555678
 session target ipv4:10.5.6.7
```

### Voice Gateway Router 2 Configuration File

The configuration for this voice gateway features a Cisco 3620 running Cisco IOS Release 12.2(15)T.

```
interface Serial0/0
 ip address 10.2.3.4 255.0.0.0
 encapsulation frame-relay
 clockrate 2000000
 no fair-queue
```

```

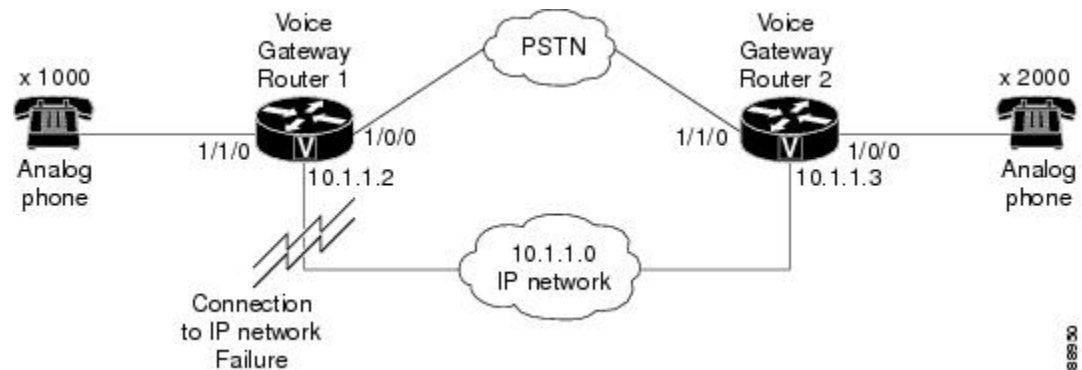
!
voice-port 1/0/0
!
voice-port 1/0/1
!
voice-port 1/1/0
!
voice-port 1/1/1
!
dial-peer cor custom
!
dial-peer voice 1 pots
 destination-pattern 5555678
 port 1/0/0
!
dial-peer voice 2000 voip
 preference 1
 destination-pattern 5551234
 session target ipv4:10.2.3.4

```

## Using Digit Manipulation to Overcome the Obstacle of an IP Network Failure

The figure below and the subsequent Cisco voice gateway router configuration examples illustrate the dial peer configuration necessary to automatically route an outgoing voice call over the PSTN in the event of a temporary IP network outage. An advantage to this method of setting up and connecting the call over the PSTN (while still originating the transmission from a voice gateway router) is more commonly known as "toll bypass."

**Figure 22: Using the PSTN in the event of an IP network failure**



### Voice Gateway Router 1 Configuration File

The configuration for this voice gateway features a Cisco 3620 running Cisco IOS Release 12.2(15)T.

```

translation-rule 21
 Rule 1 ^2 5552
!
translation-rule 11
 Rule 1 ^5551 1
!
interface FastEthernet0/0

```

```

ip address 10.1.1.2 255.255.255.0
duplex auto
speed auto
!
voice-port 1/0/0
  translate called 11
!
voice-port 1/0/1
!
voice-port 1/1/0
!
voice-port 1/1/1
!
dial-peer cor custom
!
dial-peer voice 1 pots
  destination-pattern 1000
  port 1/1/0
!
dial-peer voice 2000 voip
  preference 1
  destination-pattern 2000
  session target ipv4:10.1.1.3
!
dial-peer voice 20 pots
  preference 2
  destination-pattern 2000
  translate-outgoing called 21
  port 1/0/0
  forward-digits all

```

### Voice Gateway Router 2 Configuration File

The configuration for this voice gateway features a Cisco 3620 running Cisco IOS Release 12.2(11)T6.

```

translation-rule 11
  Rule 1 ^1 5551
!
translation-rule 21
  Rule 1 ^5552 2
!
interface Ethernet0/0
  ip address 10.1.1.3 255.255.255.0
  full-duplex
!
voice-port 1/0/0
!
voice-port 1/0/1
!
voice-port 1/1/0
  translate called 21
!
voice-port 1/1/1
!
dial-peer cor custom
!
dial-peer voice 1 pots
  destination-pattern 2000
  port 1/0/0
!
dial-peer voice 1000 voip
  preference 1
  destination-pattern 1000

```

```
    session target ipv4:10.1.1.2
  !
dial-peer voice 10 pots
  preference 2
  destination-pattern 1000
  translate-outgoing called 11
  port 1/1/0
  forward-digits all
```







## CHAPTER 7

# Matching Inbound Dial Peers by URI

The Matching Inbound Dial Peers by URI feature allows you to configure the selection of inbound dial peers by matching parts of the URI sent by a remote (neighboring) SIP entity. The match can be done on different parts of the URI like hostname, IP address, DNS name. This feature can be used to configure configuration policies, enforce specific call-treatment, security, and routing policies on each SIP trunk by originating SIP entity.

In a scenario where multiple SIP hops are involved in a call, there would be multiple via headers involved, and the topmost via header of an incoming SIP invite represents the last hop that forwarded the SIP request, and the bottom-most via header would represent the originator of the SIP request. This feature supports matching by the last hop that forwarded the request (neighboring SIP entity), which is the topmost via header.



**Note** For incoming dial-peer match based on URI, if there are multiple dial-peer matches, then the longest matching dial-peer is chosen (similar to multiple dial-peer match based on incoming called number). However for URI pattern match, there is no match length and hence this is the least preferred.

- [Configuring an Inbound Dial Peer to Match on URI, on page 97](#)
- [Examples for Configuring an Inbound Dial Peer to Match on a URI, on page 99](#)

## Configuring an Inbound Dial Peer to Match on URI

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice class uri** *voice-class-uri-tag*
4. Specify a URI field for the voice class:
  - **host** *hostname-pattern*
  - **host ipv4:** *ipv4-address*
  - **host ipv6:** *ipv6-address*
  - **host dns:** *dns-address*
  - **pattern** *uri-pattern*
  - **user-id** *username-pattern*

5. `exit`
6. `dial-peer voice tag voip`
7. `session protocol sipv2`
8. `incoming uri { from | request | to | via } voice-class-uri-tag`
9. `end`

## DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<p><code>enable</code></p> <p><b>Example:</b></p> <pre>Device&gt; enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<p><code>configure terminal</code></p> <p><b>Example:</b></p> <pre>Device&gt; configure terminal</pre>	<p>Enters global configuration mode.</p>
<b>Step 3</b>	<p><code>voice class uri voice-class-uri-tag</code></p> <p><b>Example:</b></p> <pre>Device(config)# voice class uri 200</pre>	<p>Creates a voice class for matching SIP dial peers and enters voice URI class configuration mode.</p>
<b>Step 4</b>	<p>Specify a URI field for the voice class:</p> <ul style="list-style-type: none"> <li>• <code>host hostname-pattern</code></li> <li>• <code>host ipv4: ipv4-address</code></li> <li>• <code>host ipv6: ipv6-address</code></li> <li>• <code>host dns: dns-address</code></li> <li>• <code>pattern uri-pattern</code></li> <li>• <code>user-id username-pattern</code></li> </ul> <p><b>Example:</b></p> <pre>Device(config-voice-uri-class)# host server1</pre> <p><b>Example:</b></p> <pre>Device(config-voice-uri-class)# host ipv4:10.0.0.0</pre> <p><b>Example:</b></p> <pre>Device(config-voice-uri-class)# host dns:xxx.yyy.com</pre>	<ul style="list-style-type: none"> <li>• You can specify up to ten instances of the <b>host ipv4:</b>, <b>host ipv6:</b>, and <b>host dns:</b> commands.</li> <li>• You can specify only one instance of the <b>host hostname-pattern</b> commands.</li> <li>• Length of <i>uri-pattern</i>, <i>username-pattern</i>, and <i>hostname-pattern</i> should be less than 32.</li> <li>• <i>username-pattern</i> is matched against the username field of the URI.</li> <li>• <i>hostname-pattern</i> is matched against the host field of the URI.</li> <li>• <i>uri-pattern</i> is matched against the entire URI.</li> <li>• Only one instance of the <b>pattern</b> and <b>host</b> commands are possible.</li> </ul> <p><b>Note</b> Patterns are case-sensitive.</p>
<b>Step 5</b>	<p><code>exit</code></p> <p><b>Example:</b></p> <pre>Device(config-voice-uri-class)# exit</pre>	<p>Enters global configuration mode.</p>

	Command or Action	Purpose
Step 6	<b>dial-peer voice tag voip</b> <b>Example:</b> Device(config)# dial-peer voice 6000 voip	Enters dial peer voice configuration mode.
Step 7	<b>session protocol sipv2</b> <b>Example:</b> Device(config-dial-peer)# session protocol sipv2	Configures SIP as the session protocol type.
Step 8	<b>incoming uri { from   request   to   via } voice-class-uri-tag</b> <b>Example:</b> Device(config-dial-peer)# incoming uri via 200	Configures the voice class with an inbound dial peer, so that it matches against configured URI fields.
Step 9	<b>end</b> <b>Example:</b> Device(config-dial-peer)# end	Exits dial peer voice configuration mode and enters privileged EXEC mode.

## Examples for Configuring an Inbound Dial Peer to Match on a URI

### Matching Against IPv4 Address and VIA

CUBE is configured to use incoming dial-peer 101 for incoming SIP calls from remote SIP endpoint having an IP address of 10.10.10.1

```
voice class uri 201 sip
host ipv4:10.10.10.1
```

```
dial-peer voice 101 voip
session protocol sipv2
incoming uri via 201
```

Incoming INVITE that can be matched against this dial peer.

```
INVITE sip:123@1.2.3.4:5060 SIP/2.0
Via: SIP/2.0/TCP 10.10.10.1:5093;branch=z9hG4bK-17716-1-0
Via: SIP/2.0/TCP 10.10.14.20:5093;branch=z9hG4bK-28280-1-0
```

### Matching Against DNS Name and VIA

CUBE is configured to use incoming dial-peer 102 for incoming SIP calls from sample.com or an IP address that represents one of the resolved IP address of sample.com.

```
voice class uri 202 sip
host dns:sample.com
```

```
dial-peer voice 101 voip
  session protocol sipv2
  incoming uri via 202
```

Incoming INVITE that can be matched against this dial peer.

```
INVITE sip:123@1.2.3.4:5060 SIP/2.0
Via: SIP/2.0/TCP sample.com;branch=z9hG4bK-17716-1-0

INVITE sip:123@1.2.3.4:5060 SIP/2.0
Via: SIP/2.0/TCP 10.10.10.25:5093;branch=z9hG4bK-17716-1-0
```

10.10.10.25 is a resolved IP address of sample.com.

### Matching Against Multiple Attributes and VIA

CUBE is configured to use incoming dial-peer 103 for incoming SIP calls from xxx.yyy.com, abc.def.com and IP addresses 10.10.10.10, 10.9.10.11 and 10.10.10.10.

```
voice class uri 203 sip
  host dns:xxx.yyy.com
  host dns:abc.def.com
  host ipv4:10.10.10.10
  host ipv4:10.9.10.11
  host ipv4:10.10.10.10
```

```
dial-peer voice 103 voip
  session protocol sipv2
  incoming uri via 203
```

Incoming INVITE that can be matched against this dial peer.

```
INVITE sip:123@1.2.3.4:5060 SIP/2.0
Via: SIP/2.0/TCP 10.10.10.10:5093;branch=z9hG4bK-17716-1-0
Via: SIP/2.0/TCP 10.10.14.20:5093;branch=z9hG4bK-28280-1-0
```

10.10.10.25 is a resolved IP address of sample.com.



## CHAPTER 8

# Multiple Pattern Support on a Voice Dial Peer

The Multiple Pattern Support on a Voice Dial Peer feature enables you to configure multiple patterns on a VoIP dial peer using an E.164 pattern map. A dial peer can be configured to match multiple patterns to an incoming calling or called number or an outgoing destination number.

- [Feature Information for Multiple Pattern Support on a Voice Dial Peer, on page 101](#)
- [Restrictions for Multiple Pattern Support on a Voice Dial Peer, on page 102](#)
- [Information About Multiple Pattern Support on a Voice Dial Peer, on page 102](#)
- [Configuring Multiple Pattern Support on a Voice Dial Peer, on page 102](#)
- [Verifying Multiple Pattern Support on a Voice Dial Peer, on page 104](#)
- [Configuration Examples for Multiple Pattern Support on a Voice Dial Peer, on page 106](#)

## Feature Information for Multiple Pattern Support on a Voice Dial Peer

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [www.cisco.com/go/cfn](http://www.cisco.com/go/cfn). An account on Cisco.com is not required.

**Table 7: Feature Information for Multiple Pattern Support on a Voice Dial Peer**

Feature Name	Releases	Feature Information
Configuring Multiple Pattern Support on a Voice Dial Peer (Inbound Calls)	Cisco IOS 15.4 (1)T Cisco IOS XE 3.11S	This feature was extended for inbound VoIP dial peers for incoming calling and called numbers.  The following commands were introduced or modified: <b>incoming called e164-pattern-map</b> , <b>incoming calling e164-pattern-map</b>

Feature Name	Releases	Feature Information
Configuring Multiple Pattern Support on a Voice Dial Peer (Outbound Calls)	Cisco IOS 15.2(4)M Cisco IOS XE 3.7S	<p>This feature allows you to add more than one E.164 destination pattern inside a pattern map and configure that pattern map for one or more VoIP dial peers.</p> <p>This feature is supported for outbound peers only.</p> <p>The following commands were introduced or modified: <b>destination e164-pattern-map</b>, <b>e164</b>, <b>show voice class e164-pattern-map</b>, <b>url</b>, <b>voice class e164-pattern-map load</b>, <b>voice class e164-pattern-map</b>.</p>

## Restrictions for Multiple Pattern Support on a Voice Dial Peer

- This feature is supported only on a VoIP dial peer.
- Duplicate patterns cannot be added to a pattern map.

## Information About Multiple Pattern Support on a Voice Dial Peer

Matching an incoming or outgoing call using a pattern defined in a VoIP dial peer is an existing feature on the Cisco Unified Border Element (Enterprise) and Session Initiation Protocol (SIP) Gateway. You can now support multiple patterns on a VoIP dial peer using an E.164 pattern map. You can create a E.164 pattern map and then link it to one or more VoIP dial peers.

When a pattern is the only source to enable a dial peer, a valid E.164 pattern map enables the linked dial peers, whereas an invalid E.164 pattern map disables the linked dial peers. Additionally, whenever an E.164 pattern map is created or reloaded, one or more dial peers linked with an E.164 pattern map is enabled or disabled based on the validation of a pattern map.

You can match a pattern map to an incoming calling or called number or an outgoing destination number.

When a dial peer has multiple patterns, the pattern with the longest prefix is considered as the matching criteria.

## Configuring Multiple Pattern Support on a Voice Dial Peer

### SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **voice class e164-pattern-map** *pattern-map-id*
4. Do one of the following:
  - **e164** *pattern-map-tag*
  - **url** *url*
5. (Optional) **description** *string*
6. **exit**
7. **dial-peer voice** *dial-peer-id* **voip**
8. {**destination** | **incoming called** | **incoming calling**} **e164-pattern-map** *pattern-map-group-id*
9. **end**
10. (Optional) **voice class e164-pattern-map load** *pattern-map-group-id*
11. **show dial-peer voice** [**summary** | *dial-peer-id*]

## DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> Device> <b>enable</b>	Enters privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> Device# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 3</b>	<b>voice class e164-pattern-map</b> <i>pattern-map-id</i> <b>Example:</b> Device (config)# <b>voice class e164-pattern-map 1111</b>	Creates a pattern map for configuring one or multiple E.164 patterns on a dial peer and enters voice class configuration mode.
<b>Step 4</b>	Do one of the following: <ul style="list-style-type: none"> <li>• <b>e164</b> <i>pattern-map-tag</i></li> <li>• <b>url</b> <i>url</i></li> </ul> <b>Example:</b> Using URL text file: Device (voice-class)# <b>url</b> <b>http://http-host/config-files/pattern-map.cfg</b> Directly specifying match patterns: Device (voice-class)# <b>e164 5557123</b>	Configure one or more E.164 telephone number prefix match patterns for the pattern map. <ul style="list-style-type: none"> <li>• Repeat this step for each pattern if you are using the <b>e164</b> command.</li> <li>• You can specify a file URL containing the patterns for this dial peer using the <b>url url</b> command. You must then load the E.164 telephone prefixes using Step 10. The file can be internal (on the device) or external.</li> </ul>

	Command or Action	Purpose
<b>Step 5</b>	(Optional) <b>description</b> <i>string</i> <b>Example:</b>  Device (voice-class) # <b>description</b> It has 1 entry	Provides a description for the pattern map.
<b>Step 6</b>	<b>exit</b> <b>Example:</b>  Device (voice-class) # <b>exit</b>	Exits voice class configuration mode and enters global configuration mode.
<b>Step 7</b>	<b>dial-peer voice</b> <i>dial-peer-id</i> <b>voip</b> <b>Example:</b>  Device (config) # <b>dial-peer voice</b> 2222 <b>voip</b>	Defines a VoIP dial peer and enters dial peer configuration mode.
<b>Step 8</b>	{ <b>destination</b>   <b>incoming called</b>   <b>incoming calling</b> } <b>e164-pattern-map</b> <i>pattern-map-group-id</i> <b>Example:</b>  Device (config-dial-peer) # <b>incoming calling</b> <b>e164-pattern-map</b> 1111	Links a pattern-map group with a dial peer. <ul style="list-style-type: none"> <li>• Use the <b>destination</b> keyword for outbound dial peers.</li> <li>• Use the <b>incoming called</b> or <b>incoming calling</b> keywords for inbound dial peers using called or calling numbers.</li> </ul>
<b>Step 9</b>	<b>end</b> <b>Example:</b>  Device (config-dial-peer) # <b>end</b>	Exits dial peer configuration mode and enters privileged EXEC mode.
<b>Step 10</b>	(Optional) <b>voice class e164-pattern-map load</b> <i>pattern-map-group-id</i> <b>Example:</b>  Device# <b>voice class e164-pattern-map load</b> 1111	Loads the specified pattern map with E.164 match patterns from a text file configured in the pattern map. <ul style="list-style-type: none"> <li>• This step is required only if patterns have been defined for the specified pattern map using a file URL in Step 4.</li> </ul>
<b>Step 11</b>	<b>show dial-peer voice</b> [ <b>summary</b>   <i>dial-peer-id</i> ] <b>Example:</b>  Device# <b>show dial-peer voice</b> 1111	Displays the status of a pattern map when the pattern map is associated with a dial peer.

## Verifying Multiple Pattern Support on a Voice Dial Peer

### SUMMARY STEPS

1. **show voice class e164-pattern-map** [**summary** | *pattern-map-id*]
2. **show dial-peer voice** [**summary** | *dial-peer-id*]



### 3. show dialplan incall {sip | h323} {calling | called} e164-pattern

#### DETAILED STEPS

##### Step 1 show voice class e164-pattern-map [summary | pattern-map-id]

Displays the status and contents of a specified pattern map or a status summary of all pattern maps.

##### Example:

```
Device# show voice class e164-pattern-map 200
```

```
e164-pattern-map 200
-----
  It has 1 entries
  It is not populated from a file.
  Map is valid.
```

```
E164 pattern
-----
200
```

##### Step 2 show dial-peer voice [summary | dial-peer-id]

Displays the status of pattern maps associated with all or a specified dial peer.

##### Example:

```
Device# show dial-peer voice | include e164-pattern-map
```

```
incoming calling e164-pattern-map tag = `200' status = valid,
destination e164-pattern-map tag = 3000 status = valid,
```

```
Device# show dial-peer voice 2222 | include e164-pattern-map
```

```
incoming calling e164-pattern-map tag = `200' status = valid,
```

##### Step 3 show dialplan incall {sip | h323} {calling | called} e164-pattern

Displays inbound dial peer details and associated pattern maps based on an incoming calling or called number.

##### Example:

```
Device# show dialplan incall voip calling 23456
```

```
VoiceOverIpPeer1234567
  peer type = voice, system default peer = FALSE, information type = voice,
  description = `',
  tag = 1234567, destination-pattern = `',
  destination e164-pattern-map tag = 200 status = valid,
  destination dpg tag = 200 status = valid,
  voice reg type = 0, corresponding tag = 0,
  allow watch = FALSE
  answer-address = `', preference=0,
  incoming calling e164-pattern-map tag = `200' status = valid,
  CLID Restriction = None
```

# Configuration Examples for Multiple Pattern Support on a Voice Dial Peer

## Example: Configuring Multiple Patterns for Outbound Dial Peers Using a File URL

```
Device# voice class e164-pattern-map 1111
Device(voice-class)# url http://http-host/config-files/pattern-map.cfg
Device(voice-class)# description For Outbound Dial Peer
Device(voice-class)# exit
Device(config)# dial-peer voice 2222 voip
Device(voice-dial-peer)# destination e164-pattern-map 1111
Device(voice-dial-peer)# exit
Device(config)# voice class e164-pattern-map load 1111
Device(config)# end
```

## Example: Configuring Multiple Patterns for Outbound Dial Peers by Specifying Each E164 Pattern

```
Device# voice class e164-pattern-map 1112
Device(voice-class)# e164 5557456
Device(voice-class)# e164 5557455
Device(voice-class)# e164 5557454
Device(voice-class)# e164 5557453
Device(voice-class)# e164 5557452
Device(voice-class)# description For Outbound Dial Peer
Device(voice-class)# exit
Device(config)# dial-peer voice 2222 voip
Device(voice-dial-peer)# destination e164-pattern-map 1112
Device(voice-dial-peer)# end
!
```

## Example: Configuring Multiple Patterns for Inbound Dial Peer

```
Device# voice class e164-pattern-map 1113
Device(voice-class)# url http://http-host/config-files/pattern-map.cfg
Device(voice-class)# description For Inbound Dial Peer
Device(voice-class)# exit
Device(config)# dial-peer voice 2222 voip
Device(voice-dial-peer)# incoming calling e164-pattern-map 1113
Device(voice-dial-peer)# exit
Device(config)# voice class e164-pattern-map load 1113
Device(config)# end
```



## CHAPTER 9

# Outbound Dial-Peer Group as an Inbound Dial-Peer Destination

This feature can group multiple outbound dial peers into a dial-peer group and configure this dial-peer group as the destination of an inbound dial peer.

- [Feature Information for Outbound Dial-Peer Group as an Inbound Dial-Peer Destination, on page 107](#)
- [Restrictions, on page 108](#)
- [Information About Outbound Dial-Peer Group as an Inbound Dial-Peer Destination, on page 108](#)
- [Configuring Outbound Dial-Peer Group as an Inbound Dial-Peer Destination, on page 109](#)
- [Verifying Outbound Dial-Peer Groups as an Inbound Dial-Peer Destination, on page 111](#)
- [Troubleshooting Tips, on page 112](#)
- [Configuration Examples for Outbound Dial Peer Group as an Inbound Dial-Peer Destination, on page 113](#)

## Feature Information for Outbound Dial-Peer Group as an Inbound Dial-Peer Destination

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [www.cisco.com/go/cfn](http://www.cisco.com/go/cfn). An account on Cisco.com is not required.

**Table 8: Feature Information for Outbound Dial-Peer Group as an Inbound Dial-Peer Destination**

Feature Name	Releases	Feature Information
Support for POTS dial-peer	Cisco IOS 15.5(1)T Cisco IOS XE 3.14S	An outgoing POTS dial peer can be part of a dial-peer group. An inbound POTS dial peer can have a dial-peer group as the destination.

Feature Name	Releases	Feature Information
Outbound Dial-Peer Group as an Inbound Dial-Peer Destination	Cisco IOS 15.4(1)T Cisco IOS XE 3.11S	This feature groups multiple outbound dial-peers into a dial-peer group and configures this dial-peer group as a destination of an inbound dial peer.  The following commands were introduced or modified: <b>voice class dpg</b> , <b>description</b> , <b>dial-peer preference</b> , <b>destination dpg</b> , <b>show voice class dpg</b> .

## Restrictions

- If a dial-peer group is in the shutdown state, regular dial-peer search occurs.
- If all dial peers in an active dial-peer group are unavailable, call is disconnected.
- The number of matched digits is zero.
- The **destination-pattern** command is required on the outbound dial peer even though matching is not done based on this command.
- The outgoing call setup is deferred until inter-digit timer expires or a terminator is entered.

### For POTS dial peers:

- Two-stage dialing is not supported.
- Overlapping dialing is not supported.
- TCL and VXML routing changes are not supported.
- Digit-stripping is not supported.

## Information About Outbound Dial-Peer Group as an Inbound Dial-Peer Destination

You can group up to 20 outbound (H.323, SIP or POTS) dial peers into a dial-peer group and configure this dial-peer group as the destination of an inbound dial peer. Once an incoming call is matched by an inbound dial peer with an active destination dial-peer group, dial peers from this group are used to route the incoming call. No other outbound dial-peer provisioning to select outbound dial peers is used.

A preference can be defined for each dial peer in a dial-peer group. This preference is used to decide the order of selection of dial peers from the group for the setup of an outgoing call.

You can also specify various dial-peer hunt mechanism using the existing **dial-peer hunt** command.

# Configuring Outbound Dial-Peer Group as an Inbound Dial-Peer Destination

Perform this task to configure a dial-peer group with multiple outbound peers and an inbound dial peer referencing this dial-peer group as a destination.

## Before you begin

- Configure SIP, H.323 or POTS outbound dial peers to be associated with a dial-peer group.
- For an outbound POTS dial peer, ensure that **destination-pattern .T** and **no digit-strip** are configured to avoid unexpected dialed digit strip.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dial-peer voice** *outbound-dial-peer-id* [**voip** | **pots**]
4. **destination-pattern** *pattern*
5. **no digit-strip** for POTS dial peers.
6. **exit**
7. (Optional) **dial-peer hunt** *hunt-order-number*
8. **voice class dpg** *dial-peer-group-id*
9. **dial-peer** *outbound-dial-peer-id* [**preference** *preference-order*]
10. (Optional) **description** *string*
11. **exit**
12. **dial-peer voice** *inbound-dial-peer-id* [**voip** | **pots**]
13. **destination dpg** *dial-peer-group-id*
14. **end**

## DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>enable</b> <b>Example:</b> Device> <b>enable</b>	Enters privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
Step 2	<b>configure terminal</b> <b>Example:</b> Device# <b>configure terminal</b>	Enters global configuration mode.
Step 3	<b>dial-peer voice</b> <i>outbound-dial-peer-id</i> [ <b>voip</b>   <b>pots</b> ] <b>Example:</b>	Defines a dial peer and enters dial peer configuration mode.

	Command or Action	Purpose
	For VoIP dial peer:  Device (config) # <b>dial-peer voice 123 voip</b>  <b>Example:</b> For POTS dial peer:  Device (config) # <b>dial-peer voice 345 pots</b>	
<b>Step 4</b>	<b>destination-pattern</b> <i>pattern</i>  <b>Example:</b> For VoIP Dial Peers Device (config-dial-peer) # <b>destination-pattern 1004</b>  <b>Example:</b> For POTS Dial Peers Device (config-dial-peer) # <b>destination-pattern .T</b>	Configures a destination pattern. This step is required even though the value is not used for dial-peer matching.
<b>Step 5</b>	<b>no digit-strip</b> for POTS dial peers.  <b>Example:</b>  Device (config-dial-peer) # <b>no digit-strip</b>	Disable unexpected dialed digit strip.
<b>Step 6</b>	<b>exit</b>  <b>Example:</b>  Device (config-dial-peer) # <b>exit</b>	Exits to global configuration mode.
<b>Step 7</b>	(Optional) <b>dial-peer hunt</b> <i>hunt-order-number</i>  <b>Example:</b>  Device (config) # <b>dial-peer hunt 0</b>	Specifies a hunt selection mechanism for dial peers. <ul style="list-style-type: none"> <li>The default mechanism is random selection.</li> </ul>
<b>Step 8</b>	<b>voice class dpg</b> <i>dial-peer-group-id</i>  <b>Example:</b>  Device (config) # <b>voice class dpg 181</b>	Creates a dial-peer group for grouping multiple outbound dial peers and enters voice class configuration mode. <ul style="list-style-type: none"> <li>You can use the <b>shutdown</b> command to resume regular outbound dial-peer provisioning in dial-peers with this dial-peer group as destination.</li> </ul>
<b>Step 9</b>	<b>dial-peer</b> <i>outbound-dial-peer-id</i> [ <b>preference</b> <i>preference-order</i> ]  <b>Example:</b>  Device (config-class) # <b>dial-peer 123 preference 1</b>	Associates a configured outbound dial peer with this dial-peer group and configures a preference value. <ul style="list-style-type: none"> <li>Repeat this step for all outbound dial-peers that need to be added to this dial-peer group.</li> </ul>

	Command or Action	Purpose
		<ul style="list-style-type: none"> <li>If preference is not specified, the order of selection is random or as specified by the <b>dial-peer hunt</b> command.</li> </ul>
<b>Step 10</b>	(Optional) <b>description</b> <i>string</i> <b>Example:</b> <pre>Device(config-class)# description Boston Destination</pre>	Provides a description for the dial-peer group.
<b>Step 11</b>	<b>exit</b> <b>Example:</b> <pre>Device(config-class)# exit</pre>	Exits voice class configuration mode and enters global configuration mode.
<b>Step 12</b>	<b>dial-peer voice</b> <i>inbound-dial-peer-id</i> [ <b>voip</b>   <b>pots</b> ] <b>Example:</b> For VoIP dial peer: <pre>Device(config)# dial-peer voice 789 voip</pre> <b>Example:</b> For POTS dial peer: <pre>Device(config)# dial-peer voice 678 pots</pre>	Defines a dial peer and enters dial peer configuration mode.
<b>Step 13</b>	<b>destination dpg</b> <i>dial-peer-group-id</i> <b>Example:</b> <pre>Device(config-dial-peer)# destination dpg 181</pre>	Specifies a dial peer group from which an outbound dial peer can be chosen.
<b>Step 14</b>	<b>end</b> <b>Example:</b> <pre>Device(config-dial-peer)# end</pre>	Exits dial peer configuration mode and enters privileged EXEC mode.

## Verifying Outbound Dial-Peer Groups as an Inbound Dial-Peer Destination

### SUMMARY STEPS

1. **show voice class dpg** *dial-peer-group-id*
2. **show dial-peer voice** *inbound-dial-peer-id*

**DETAILED STEPS****Step 1** `show voice class dpg dial-peer-group-id`

Displays the configuration of an outbound dial-peer group.

**Example:**

```
Device# show voice class dpg 200

Voice class dpg: 200      AdminStatus: Up
Description: Boston Destination
Total dial-peer entries: 4

Peer Tag      Pref
-----      -
1001          1
1002          2
1004          0
1003          1
-----
```

**Step 2** `show dial-peer voice inbound-dial-peer-id`

Displays the referencing of destination dial-peer group from an inbound dial peer.

**Example:**

```
Device# show dial-peer voice 100 | include destination dpg

destination dpg tag = 200 status = valid,
```

# Troubleshooting Tips

**SUMMARY STEPS**

1. Enter the following:
  - `debug voip dialpeer inout`
  - `debug voip ccapi inout`

**DETAILED STEPS**

Enter the following:

- `debug voip dialpeer inout`
- `debug voip ccapi inout`

Displays the configuration of an outbound dial-peer group.

**Example:**



```
*Jul 19 10:15:53.310 IST: //-1/ED647BD1B0F9/DPM/dpMatchCore:
  Dial String=4001, Expanded String=4001, Calling Number=
  Timeout=TRUE, Is Incoming=TRUE, Peer Info Type=DIALPEER_INFO_SPEECH
*Jul 19 10:15:53.310 IST: //-1/xxxxxxxxxxxx/DPM/vepm_match_pattern_map:
  DEPM 1000 use caching dialstring 4001 status 0
*Jul 19 10:15:53.310 IST: //-1/ED647BD1B0F9/DPM/MatchNextPeer:
```

Incoming dial peer is first matched:

```
Result=Success(0); Incoming Dial-peer=600 Is Matched
*Jul 19 10:15:53.310 IST: //-1/ED647BD1B0F9/DPM/dpMatchPeertype:exit@6602
*Jul 19 10:15:53.310 IST: //-1/ED647BD1B0F9/DPM/dpAssociateIncomingPeerCore:
  Result=Success(0) after DP_MATCH_INCOMING_DNIS; Incoming Dial-peer=600
*Jul 19 10:15:53.310 IST: //-1/ED647BD1B0F9/DPM/dpMatchSafModulePlugin:
  dialstring=NULL, saf_enabled=0, saf_dndb_lookup=0, dp_result=0
*Jul 19 10:15:53.310 IST: //-1/ED647BD1B0F9/DPM/dpAssociateIncomingPeerSPI:exit@7181
*Jul 19 10:15:53.311 IST: //-1/ED647BD1B0F9/DPM/dpMatchPeersCore:
  Calling Number=, Called Number=4001, Peer Info Type=DIALPEER_INFO_SPEECH
```

The dial-peer group associated with a dial peer is selected:

```
*Jul 19 10:15:53.311 IST: //-1/ED647BD1B0F9/DPM/dpMatchPeersCore:
  Outbound Destination DPG Group Request; Destination DPG=1
*Jul 19 10:15:53.311 IST: //-1/ED647BD1B0F9/DPM/dpMatchDestDPGGroup:
  Result=0
*Jul 19 10:15:53.311 IST: //-1/ED647BD1B0F9/DPM/dpMatchPeersCore:
  Result=SUCCESS(0) after DestDPGGroup
*Jul 19 10:15:53.311 IST: //-1/ED647BD1B0F9/DPM/dpMatchSafModulePlugin:
  dialstring=4001, saf_enabled=0, saf_dndb_lookup=1, dp_result=0
```

List of active Dial-peers configured within the DPG, sorted by preference:

```
*Jul 19 10:15:53.311 IST: //-1/ED647BD1B0F9/DPM/dpMatchPeersMoreArg:
  Result=SUCCESS(0)
List of Matched Outgoing Dial-peer(s):
  1: Dial-peer Tag=1004
  2: Dial-peer Tag=1001
  3: Dial-peer Tag=1003
  4: Dial-peer Tag=1002
```

## Configuration Examples for Outbound Dial Peer Group as an Inbound Dial-Peer Destination

```
Device> enable
Device# configure terminal
! Configuring outbound dial peers that are to be grouped.
Device(config)# dial-peer voice 1001 voip
Device(config-dial-peer)# destination-pattern 1001
Device(config-dial-peer)# session protocol sipv2
Device(config-dial-peer)# session target ipv4:10.1.1.1
```

```

Device(config-dial-peer)# exit

Device(config)# dial-peer voice 1002 voip
Device(config-dial-peer)# destination-pattern 1002
Device(config-dial-peer)# session protocol sipv2
Device(config-dial-peer)# session target ipv4:10.1.1.2
Device(config-dial-peer)# exit

Device(config)# dial-peer voice 1003 voip
Device(config-dial-peer)# destination-pattern 1003
Device(config-dial-peer)# session protocol sipv2
Device(config-dial-peer)# session target ipv4:10.1.1.3
Device(config-dial-peer)# exit

Device(config)# dial-peer voice 1004 pots
Device(config-dial-peer)# destination-pattern 5...
Device(config-dial-peer)# no digit-strip
Device(config-dial-peer)# direct-inward-dial
Device(config-dial-peer)# port 1/0/0:23
Device(config-dial-peer)# forward-digits all
Device(config-dial-peer)# exit

!Grouping outbound dial peers and configuring preferences if needed.
Device(config)# voice class dpg 200
Device(config-class)# dial-peer 1001 preference 1
Device(config-class)# dial-peer 1002 preference 2
Device(config-class)# dial-peer 1003 preference 3
Device(config-class)# dial-peer 1004 preference 4
Device(config-class)# description Boston Destination
Device(config-class)# exit

!Associating outbound dial peer group with an inbound dial peer group.
Device(config)# dial-peer voice 100 voip
Device(config-dial-peer)# incoming called-number 13411
Device(config-dial-peer)# destination dpg 200
Device(config-dial-peer)# end

!Associating outbound dial peer group with an inbound POTS dial peer group.
Device(config)# dial-peer voice 600 pots
Device(config-dial-peer)# incoming called-number 4T
Device(config-dial-peer)# destination dpg 200
Device(config-dial-peer)# end

```

### Verifying Outbound Dial-Peer Group Configuration

```

Device# show voice class dpg 200

Voice class dpg: 200      AdminStatus: Up
Description: Boston Destination
Total dial-peer entries: 4

Peer Tag      Pref
-----
1001          1
1002          2
1004          0
1003          1

```

-----

### Verifying Inbound Dial-Peer Referencing Outbound Dial-Peer Group

```
Device# show dial-peer voice 100 | include destination dpg
          destination dpg tag = 200 status = valid,
Device# show dial-peer voice 600 | include destination dpg
          destination dpg tag = 200 status = valid,
```





## CHAPTER 10

# Inbound Leg Headers for Outbound Dial-Peer Matching

---

The Inbound Leg Headers for Outbound Dial-Peer Matching feature allows you to match and provision an outbound dial peer for an outbound call leg using the headers from an inbound call leg. The following headers of an incoming call leg can be used for outbound dial-peer matching:

- VIA (SIP Header)
- FROM (SIP Header)
- TO (SIP Header)
- DIVERSION (SIP Header)
- REFERRED BY (SIP Header)
- Called Number
- Calling Number
- Carrier ID
  
- [Feature Information for Inbound Leg Headers for Outbound Dial-Peer Matching, on page 117](#)
- [Prerequisites for Inbound Leg Headers for Outbound Dial-Peer Matching, on page 118](#)
- [Restrictions for Inbound Leg Headers for Outbound Dial-Peer Matching, on page 118](#)
- [Information About Inbound Leg Headers for Outbound Dial-Peer Matching, on page 119](#)
- [Configuring Inbound Leg Headers for Outbound Dial-Peer Matching, on page 119](#)
- [Verifying Inbound Leg Headers for Outbound Dial-Peer Matching, on page 122](#)
- [Configuration Example: Inbound Leg Headers for Outbound Dial-Peer Matching, on page 124](#)

## Feature Information for Inbound Leg Headers for Outbound Dial-Peer Matching

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [www.cisco.com/go/cfn](http://www.cisco.com/go/cfn). An account on Cisco.com is not required.

*Table 9: Feature Information for Inbound Leg Headers for Outbound Dial-Peer Matching*

Feature Name	Releases	Feature Information
Inbound Leg Headers for Outbound Dial-Peer Matching	15.4(2)T, Cisco IOS XE Release 3.12S	<p>The Inbound Leg Headers for Outbound Dial-Peer Matching feature allows you to match and provision an outbound call leg using the headers of an inbound call leg.</p> <p>The following commands were introduced by this feature:  <b>destination provision-policy</b>, <b>destination uri-via</b>, <b>destination uri-to</b>, <b>destination uri-from</b>, <b>destination uri-diversion</b>, <b>destination uri-referred-by</b>, <b>show voice class dial-peer provision-policy</b></p> <p>The following commands were modified.  <b>show command incall</b>, <b>show dialplan dialpeer</b>.</p>

## Prerequisites for Inbound Leg Headers for Outbound Dial-Peer Matching

- CUBE or Voice Gateway must be configured.

## Restrictions for Inbound Leg Headers for Outbound Dial-Peer Matching

- The existing **header-passing** command supports modification of SIP headers of INVITE message by the Tool Command Language (TCL) application. If the above SIP headers are modified by the TCL application, they cannot be used for outbound dial-peer provisioning.
- If multiple SIP via headers and diversion headers are found in an incoming INVITE or REFER message, only the top-most via header and top-most diversion header of an incoming INVITE or REFER message are used for outbound dial-peer provisioning.
- When an incoming call is matched to an inbound dial peer with an associated provision profile without rules, outbound dial-peer provisioning is disabled and the incoming call is disconnected by CUBE or voice gateway with cause code "unassigned number (1)".

# Information About Inbound Leg Headers for Outbound Dial-Peer Matching

This feature allows you to match headers of an inbound call leg and provision an outbound dial peer for an outbound call leg. The following SIP headers of an incoming call leg can be used for outbound dial-peer matching

- VIA (SIP Header)
- FROM (SIP Header)
- TO (SIP Header)
- DIVERSION (SIP Header)
- REFERRED BY (SIP Header)
- Called Number
- Calling Number
- Carrier ID

The above headers are retrieved from an incoming INVITE or REFER message and used for outbound dial-peer provisioning.

SIP headers of an INVITE message are saved to an associated call leg. For example, an INVITE message is received for a new call leg A. Then, SIP headers are saved to call leg A itself for outbound dial-peer lookup.

On the other hand, SIP headers of a REFER message are saved to the peer call leg of the associated call leg. For example, call leg A and call leg B are connected in CUBE. The party at Call Leg B makes a blind transfer to the party at Call Leg C. A REFER message is received in CUBE for call leg B (transferor). But, SIP headers of the REFER message are saved under call leg A (transferee) for an outbound dial-peer lookup for Party C.

## Configuring Inbound Leg Headers for Outbound Dial-Peer Matching

### Before you begin

Necessary pattern maps have been configured.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice class dial-peer provision-policy tag**
4. (Optional) **description string**
5. **preference preference-order first-attribute second-attribute**
6. **exit**
7. **dial-peer voice inbound-dial-peer-tag voip**
8. **destination provision-policy tag**
9. **exit**
10. **dial-peer voice outbound-dial-peer-tag voip**

11. Configure a match command for an outbound dial peer according to the provision policy rule attribute configured.
12. **end**

## DETAILED STEPS

	Command or Action	Purpose																				
<b>Step 1</b>	<b>enable</b> <b>Example:</b> Device> enable	Enters privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>																				
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> Device# configure terminal	Enters global configuration mode.																				
<b>Step 3</b>	<b>voice class dial-peer provision-policy tag</b> <b>Example:</b> Device(config)# voice class dial-peer provision-policy 200	Creates a provision policy profile in which a set of attributes for dial-peer matching can be defined. <ul style="list-style-type: none"> <li>• You can use the <b>shutdown</b> command to deactivate the provision policy and allow normal outbound dial-peer provisioning.</li> </ul>																				
<b>Step 4</b>	(Optional) <b>description string</b> <b>Example:</b> Device(voice-class)# description match both calling and called	Provides a description for the provision policy profile.																				
<b>Step 5</b>	<b>preference preference-order first-attribute second-attribute</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>First Attribute</th> <th>Second Attribute</th> </tr> </thead> <tbody> <tr> <td>diversion</td> <td>from, referred-by, to, uri, via</td> </tr> <tr> <td>from</td> <td>diversion, referred-by, to, uri, via</td> </tr> <tr> <td>referred-by</td> <td>diversion, from, to, uri, via</td> </tr> <tr> <td>to</td> <td>diversion, referred-by, from, uri, via</td> </tr> <tr> <td>uri</td> <td>diversion, referred-by, to, from, via, carrier-id</td> </tr> <tr> <td>via</td> <td>diversion, referred-by, to, uri, from</td> </tr> <tr> <td>called</td> <td>calling, carrier-id</td> </tr> <tr> <td>calling</td> <td>called</td> </tr> <tr> <td>carrier-id</td> <td>called, uri</td> </tr> </tbody> </table> <b>Example:</b>	First Attribute	Second Attribute	diversion	from, referred-by, to, uri, via	from	diversion, referred-by, to, uri, via	referred-by	diversion, from, to, uri, via	to	diversion, referred-by, from, uri, via	uri	diversion, referred-by, to, from, via, carrier-id	via	diversion, referred-by, to, uri, from	called	calling, carrier-id	calling	called	carrier-id	called, uri	Configures a provision policy rule. <ul style="list-style-type: none"> <li>• You can configure up to two rules. This means up to four attributes can be configured for matching outbound dial peers.</li> <li>• If rules are not configured, outbound dial-peer provisioning is disabled, and an incoming call matched to an inbound dial peer associated with this profile is disconnected by CUBE or voice gateway with cause code "unassigned number (1)".</li> </ul>
First Attribute	Second Attribute																					
diversion	from, referred-by, to, uri, via																					
from	diversion, referred-by, to, uri, via																					
referred-by	diversion, from, to, uri, via																					
to	diversion, referred-by, from, uri, via																					
uri	diversion, referred-by, to, from, via, carrier-id																					
via	diversion, referred-by, to, uri, from																					
called	calling, carrier-id																					
calling	called																					
carrier-id	called, uri																					



	Command or Action	Purpose																				
	Device(voice-class)# preference 2 calling called																					
<b>Step 6</b>	<b>exit</b> <b>Example:</b> Device(voice-class)# exit	Exits voice class configuration mode and enters global configuration mode.																				
<b>Step 7</b>	<b>dial-peer voice inbound-dial-peer-tag voip</b>	Enters dial peer configuration mode for an inbound dial peer.																				
<b>Step 8</b>	<b>destination provision-policy tag</b> <b>Example:</b> Device(config)# dial-peer voice 100 voip Device(config-dial-peer)# destination provision-policy 200 Device(config)# exit	Associates a provision policy profile with an inbound dial peer.																				
<b>Step 9</b>	<b>exit</b>	Exits dial peer configuration mode.																				
<b>Step 10</b>	<b>dial-peer voice outbound-dial-peer-tag voip</b>	Enters dial peer configuration mode for an outbound dial peer.																				
<b>Step 11</b>	Configure a match command for an outbound dial peer according to the provision policy rule attribute configured. <table border="1" data-bbox="284 1008 893 1816"> <thead> <tr> <th>Provision Policy Rule Attribute</th> <th>Dial-peer Match command</th> </tr> </thead> <tbody> <tr> <td>called</td> <td><b>destination-pattern pattern</b> <b>destination e164-pattern-map pattern-map-class-id</b></td> </tr> <tr> <td>calling</td> <td><b>destination calling e164-pattern-map pattern-map-class-id</b></td> </tr> <tr> <td>carrier-id</td> <td><b>carrier-id target</b></td> </tr> <tr> <td>uri</td> <td><b>destination uri uri-class-tag</b></td> </tr> <tr> <td>via</td> <td><b>destination uri-via uri-class-tag</b></td> </tr> <tr> <td>to</td> <td><b>destination uri-to uri-class-tag</b></td> </tr> <tr> <td>from</td> <td><b>destination uri-from uri-class-tag</b></td> </tr> <tr> <td>diversion</td> <td><b>destination uri-diversion uri-class-tag</b></td> </tr> <tr> <td>referred-by</td> <td><b>destination uri-referred-by uri-class-tag</b></td> </tr> </tbody> </table>	Provision Policy Rule Attribute	Dial-peer Match command	called	<b>destination-pattern pattern</b> <b>destination e164-pattern-map pattern-map-class-id</b>	calling	<b>destination calling e164-pattern-map pattern-map-class-id</b>	carrier-id	<b>carrier-id target</b>	uri	<b>destination uri uri-class-tag</b>	via	<b>destination uri-via uri-class-tag</b>	to	<b>destination uri-to uri-class-tag</b>	from	<b>destination uri-from uri-class-tag</b>	diversion	<b>destination uri-diversion uri-class-tag</b>	referred-by	<b>destination uri-referred-by uri-class-tag</b>	Configure a match command based on any of the four attributes defined in the provision policy rule.
Provision Policy Rule Attribute	Dial-peer Match command																					
called	<b>destination-pattern pattern</b> <b>destination e164-pattern-map pattern-map-class-id</b>																					
calling	<b>destination calling e164-pattern-map pattern-map-class-id</b>																					
carrier-id	<b>carrier-id target</b>																					
uri	<b>destination uri uri-class-tag</b>																					
via	<b>destination uri-via uri-class-tag</b>																					
to	<b>destination uri-to uri-class-tag</b>																					
from	<b>destination uri-from uri-class-tag</b>																					
diversion	<b>destination uri-diversion uri-class-tag</b>																					
referred-by	<b>destination uri-referred-by uri-class-tag</b>																					

	Command or Action	Purpose
	<b>Example:</b> Device(config)# dial-peer voice 300 voip Device(config-dial-peer)# destination uri-from 200 Device(config)# exit	
<b>Step 12</b>	<b>end</b>  <b>Example:</b>  Device(config-dial-peer)# end	Exits dial peer configuration mode and enters privileged EXEC mode.

## Verifying Inbound Leg Headers for Outbound Dial-Peer Matching

### SUMMARY STEPS

1. `show dialplan incall {sip | h323} {calling | called} e164-pattern | include voice`
2. `show dialplan dialpeer inbound-dial-peer-id number e164-pattern [timeout] | include Voice`
3. `show voice class dial-peer provision-policy`

### DETAILED STEPS

#### Step 1 `show dialplan incall {sip | h323} {calling | called} e164-pattern | include voice`

Displays inbound dial peers based on an incoming calling or called number. Once you have the dial peer number, you can use it to search for the complete dial-peer details in the running-config.

##### Example:

```
Device# show dialplan incall sip calling 3333 | include Voice
```

```
VoiceOverIpPeer1
```

```
Device# show dialplan incall sip calling 4444 | include Voice
```

```
VoiceOverIpPeer1
```

```
Device# show running-config | section dial-peer voice 1 voip
```

```
dial-peer voice 1 voip
 destination dpg 10000
 incoming calling e164-pattern-map 100
 dtmf-relay rtp-nte
 codec g711ulaw
```

```
Device# show dialplan incall sip called 6000 timeout | include Voice
```

```
VoiceOverIpPeer100
```

```
Device# show running-config | section dial-peer voice 100 voip
```

```
dial-peer voice 100 voip
 incoming called e164-pattern-map 1
 incoming calling e164-pattern-map 1
```

```
dtmf-relay rtp-nte
codec g711ulaw
```

```
Device# show dialplan incall voip calling 23456
```

```
VoiceOverIpPeer1234567
  peer type = voice, system default peer = FALSE, information type = voice,
  description = `',
  tag = 1234567, destination-pattern = `',
  destination e164-pattern-map tag = 200 status = valid,
  destination dpg tag = 200 status = valid,
  voice reg type = 0, corresponding tag = 0,
  allow watch = FALSE
  answer-address = `', preference=0,
  incoming calling e164-pattern-map tag = `200' status = valid,
  CLID Restriction = None
```

## Step 2 **show dialplan dialpeer inbound-dial-peer-id number e164-pattern [timeout] | include Voice**

Displays a list of outbound dial peers based on a specified inbound dial peer. This command line will be helpful find a list of outbound dial peer of a destination dial-peer group.

### Example:

```
Device# show dialplan dialpeer 1 number 23457 timeout | include Voice
```

```
VoiceOverIpPeer100013
VoiceOverIpPeer100012
```

### Example:

```
voice class dial-peer provision-policy 2000
  preference 2 diversion to
  !
  ...
  !
dial-peer voice 32555 voip
  session protocol sipv2
  session target ipv4:1.5.14.9
  destination uri-diversion 1
  destination uri-to test2
  !
dial-peer voice 32991 voip
  destination provision-policy 2000
  incoming called-number 1234
  !
```

```
Device# show dialplan dialpeer 32991 number 2234 timeout
```

```
Macro Exp.: 2234
Enter Diversion header:sip:1234@cisco.com
Enter To header:sip:2234@10.0.0.0
VoiceOverIpPeer32134
  peer type = voice, system default peer = FALSE, information type = voice,
  description = `',
```

## Step 3 **show voice class dial-peer provision-policy**

Displays a list of configured provision policies and associated rules.

### Example:

```
Device# show voice class dial-peer provision-policy
```

```
Voice class dial-peer provision-policy: 100 AdminStatus: Up
```

```

Description: match only called

Pref  Policy Rule
----  -
1     called

Voice class dial-peer provision-policy: 101  AdminStatus: Up
Description: match both calling and called

Pref  Policy Rule
----  -
1     called calling

Voice class dial-peer provision-policy: 102  AdminStatus: Up
Description: match calling first; if no match then match called

Pref  Policy Rule
----  -
1     calling
2     called

Voice class dial-peer provision-policy: 200  AdminStatus: Up
Description: match referred-by and via uri; if no match then match request- uri

Pref  Policy Rule
----  -
1     referred-by via
2     uri

voice class dial-peer provision-policy: 300  AdminStatus: Up
Description: match only request-uri

Pref  Policy Rule
----  -
1     uri

Voice class dial-peer provision-policy: 400  AdminStatus: Up
Description: match only request uri; if no match then match called

Pref  Policy Rule
----  -
1     uri
2     called

```

---

## Configuration Example: Inbound Leg Headers for Outbound Dial-Peer Matching

### Example: Configuring Inbound Called or Calling Numbers Used for Outbound Dial-Peer Matching

```

Device> enable
Device# configure terminal

Device(config)# voice class dial-peer provision-policy 200
Device(voice-class)# description match both calling and called
Device(voice-class)# preference 2 calling called

```

```
Device(voice-class)# exit

Device(config)# voice class e164-pattern-map 300
Device(voice-class)# description patterns
Device(voice-class)# e164 5557123
Device(voice-class)# e164 5558123
Device(voice-class)# e164 5559123
Device(voice-class)# exit

!Associating the Provision Policy with an Inbound Dial Peer
Device(config)# dial-peer voice 100 voip
Device(config-dial-peer)# destination provision-policy 200
Device(config-dial-peer)# end

!Associates a Pattern Map with an Outbound Dial Peer.
! The called number in the SIP headers of the inbound leg is matched to select the below
outbound dial peer.
Device(config)# dial-peer voice 200 voip
Device(config-dial-peer)# destination e164-pattern-map 300
Device(config-dial-peer)# end
```

### Example: Configuring Inbound SIP Headers for Outbound Dial-Peer Matching

```
Device> enable
Device# configure terminal

Device(config)# voice class dial-peer provision-policy 200
Device(voice-class)# description match both calling and called
Device(voice-class)# preference 2 via from
Device(voice-class)# exit

!Associating the Provision Policy with an Inbound Dial Peer
Device(config)# dial-peer voice 100 voip
Device(config-dial-peer)# destination provision-policy 200
Device(config-dial-peer)# end

Device(config)# voice class uri 200 sip
Device(config-voice-uri-clas)# pattern 25054..

!Associates a Provision Policy with an Outbound Dial Peer.
The FROM SIP headers of the inbound leg is matched to select the below outbound dial peer.
Device(config)# dial-peer voice 200 voip
Device(config-dial-peer)# destination uri-from 200
Device(config-dial-peer)# end
```





# CHAPTER 11

## Server Groups in Outbound Dial Peers

This feature configures a server group (group of server addresses) that can be referenced from an outbound dial peer.

- [Feature Information for Configuring Server Groups in Outbound Dial Peers, on page 127](#)
- [Information About Server Groups in Outbound Dial Peers, on page 128](#)
- [How to Configure Server Groups in Outbound Dial Peers, on page 129](#)
- [Configuration Examples for Server Groups in Outbound Dial Peers, on page 133](#)

## Feature Information for Configuring Server Groups in Outbound Dial Peers

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [www.cisco.com/go/cfn](http://www.cisco.com/go/cfn). An account on Cisco.com is not required.

**Table 10: Feature Information for Configuring Server Groups in Outbound Dial Peers**

Feature Name	Releases	Feature Information
Server Groups in Outbound Dial Peers	Cisco IOS XE Release 3.11S 15.4(1)T	This feature configures server groups (groups of IPv4 and IPv6 addresses) which can be referenced from an outbound SIP dial peer.  The following command is introduced under: <b>voice class server-group, description, ipv4 port preference, ipv6 port preference, hunt-scheme, show voice class server-group, shutdown (Server Group).</b>

Feature Name	Releases	Feature Information
Hunt Stop for Server Groups	Cisco IOS XE Bengaluru 17.4.1a	<p>This feature allows you to configure hunt-stop based on (configurable) response codes in the Server Group.</p> <p>The following command is introduced under voice class server-group. <b>huntstop rule-tag resp-code from_resp_code to_resp_code</b></p>

## Information About Server Groups in Outbound Dial Peers

Server groups allow you to create simpler configurations by specifying a list of destination SIP servers for a single dial peer. When a call matches a dial peer that is configured with a server group, the destination is selected from the list of candidates based on a configured policy. If it is not possible to complete that call, the next candidate is selected. Alternatively, you can also choose to stop hunting through the group if a specified response code is received. If the call cannot be placed to any of the servers in the group, or hunting is stopped, call processing continues to the next preferred dial-peer.

You can configure server groups for SIP dial peers to include up to five IPv4 and IPv6 target server addresses listed in strict order of preference, or with equal weight for round robin or random selection.

If a server-group is in the shutdown mode, all dial-peers using this destination are out of service.




---

**Note** You can use Server Groups only with SIP dial-peers.

---




---

**Caution** Huntstop cannot be used with the following cause codes 401, 407, 415, 417, 422, 480, 485, 486, and 488.

1. If you attempt to configure one of the listed cause codes specifically, the following CLI error message appears.

Example, huntstop 1 resp-code 401

```
Error: The specified response code cannot be used with Huntstop.
```

2. If you attempt to configure range of codes that includes one of those codes that are listed, the command is accepted with the following warning message.

Example, huntstop 1 resp-code 420–430

```
Warning: Range includes code(s) that will not stop hunting.
```

---



# How to Configure Server Groups in Outbound Dial Peers

## Configuring Server Groups in Outbound Dial Peers

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice class server-group** *server-group-id*
4. **{ipv4 | ipv6}** *address* [**port** *port*] [**preference** *preference-order*]
5. (Optional) **hunt-scheme** **round-robin**
6. (Optional) **description** *string*
7. (Optional) **huntstop** *rule-tag* **resp-code** *from\_resp\_code* to *to\_resp\_code*
8. **dial-peer voice** *dial-peer-id* **voip**
9. **session protocol** **sipv2**
10. **destination-pattern** **[+]** *string* [**T**]
11. **session server-group** *server-group-id*
12. **end**
13. **show voice class server-group** *server-group-id*

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> Device> <b>enable</b>	Enters privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> Device# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 3</b>	<b>voice class server-group</b> <i>server-group-id</i> <b>Example:</b> Device(config)# <b>voice class server-group</b> 171	Configures a voice class server group and enters voice class configuration mode. <ul style="list-style-type: none"> <li>• You can use the <b>shutdown</b> command to make the server group inactive.</li> </ul>
<b>Step 4</b>	<b>{ipv4   ipv6}</b> <i>address</i> [ <b>port</b> <i>port</i> ] [ <b>preference</b> <i>preference-order</i> ] <b>Example:</b> Device(config-class)# <b>ipv4</b> 10.1.1.1 <b>preference</b> 3	Configures a server IP address as a part of this server group along with an optional port number and preference order. <ul style="list-style-type: none"> <li>• Repeat this step to add up to five servers to the server group.</li> </ul>

	Command or Action	Purpose
		<ul style="list-style-type: none"> <li>The servers are not selected by the preference value if round robin is configured in the next step.</li> <li>Default and highest value of preference is zero.</li> </ul>
<b>Step 5</b>	(Optional) <b>hunt-scheme round-robin</b> <b>Example:</b> <pre>Device(config-class)# hunt-scheme round-robin</pre>	Defines a hunt method for the order of selection of target server IP addresses (from the IP addresses configured for this server group) for the setting up of outgoing calls. <ul style="list-style-type: none"> <li>If a hunt scheme is not defined, an available IP address of highest preference value is selected. If neither a round-robin hunt scheme nor a preference value is configured, the selection of servers is random.</li> </ul>
<b>Step 6</b>	(Optional) <b>description string</b> <b>Example:</b> <pre>Device(config-class)# description It has 3 entries</pre>	Provides a description for the server group.
<b>Step 7</b>	(Optional) <b>huntstop rule-tag resp-code from_resp_code to_resp_code</b> <b>Example:</b> You can configure hunting in 2 ways, providing the following cause codes - <ol style="list-style-type: none"> <li><b>Range</b>  <pre>Device(config-class)# huntstop 1 resp-code 400 to 410</pre> </li> <li><b>Standalone</b>  <pre>Device(config-class)# huntstop 2 resp-code 414</pre> </li> </ol>	Stops hunting for servers in the Server Group based on configurable response codes. <ul style="list-style-type: none"> <li>Huntstop rule identifier tags range: 1-1000.</li> <li>Configurable SIP error response codes range: 400 to 599. The range must be in between 400 and 599 and must be entered in minimum to maximum order (Example: 450 to 460). You can enter multiple ranges using additional instances of this command. Response codes do not need to be ordered between instances (Example: huntstop 1 500 to 510 and huntstop 2 400 to 450).</li> </ul> The following error message appears if there is an invalid input for SIP Response codes. <pre>Error: Invalid SIP response code range configured for hunt-stop.</pre> <ul style="list-style-type: none"> <li>Overlapping of SIP error response codes configuration is not permitted. For example, huntstop 1 resp-code 400 to 510.</li> </ul> The following error message appears if you try to configure the overlapping SIP error response codes. <pre>Error: Overlap of response codes.</pre> When one of the cause codes is in the configured range, the following warning message appears. Example, huntstop 1 resp-code 400 to 405 <pre>Warning: Range includes codes that will not stop hunting.</pre>

	Command or Action	Purpose
Step 8	<b>dial-peer voice</b> <i>dial-peer-id</i> <b>voip</b> <b>Example:</b> Device(config)# <b>dial-peer voice 123 voip</b>	Defines a VoIP dial peer and enters dial peer configuration mode.
Step 9	<b>session protocol sipv2</b> <b>Example:</b> Device(config-dial-peer)# <b>session protocol sipv2</b>	Specifies SIP version 2 as the session protocol for calls between local and remote routers using the packet network.
Step 10	<b>destination-pattern</b> <b>[+] string [T]</b> <b>Example:</b> Device(config-dial-peer)# <b>destination-pattern +5550179</b>	Specifies either the prefix or the full E.164 telephone number to be used for a dial peer.
Step 11	<b>session server-group</b> <i>server-group-id</i> <b>Example:</b> Device(config-dial-peer)# <b>session server-group 171</b>	Configures the specified server group as the destination of the dial peer. <ul style="list-style-type: none"> <li>• This command is available for SIP dial peers only.</li> <li>• If the specified server group is in shutdown mode, the dial peer is not selected to route outgoing calls.</li> </ul>
Step 12	<b>end</b> <b>Example:</b> Device(config-dial-peer)# <b>end</b>	Exits dial peer configuration mode and enters privileged EXEC mode.
Step 13	<b>show voice class server-group</b> <i>server-group-id</i> <b>Example:</b> Device# <b>show voice class server-group 171</b>	Displays information about the voice class server group.

## Verifying Server Groups in Outbound Dial Peers

### SUMMARY STEPS

1. **show voice class server-group** [*server-group-id*]
2. **show running-config** |section **server-group**

### DETAILED STEPS

**Step 1** **show voice class server-group** [*server-group-id*]

The following example displays the configurations for all configured server groups or a specified server group.

**Example:**

```

Device# show voice class server-group 1

Voice class server-group: 1
AdminStatus: Up           OperStatus: Up
Hunt-Scheme: preference   Last returned server:
Description: It has 3 entries
Total server entries: 3

  Pref  Type  IP Address          IP Port
  ----  ----  -
  1     ipv4  10.1.1.1
  2     ipv4  10.1.1.2
  3     ipv4  10.1.1.3
-----

Total Huntstop tags: 2
Tag ID From Response code      To Response code
-----
  1     404
  2     410
-----

```

The following example displays the configurations for dial peers that are associated with server groups.

**Example:**

```

Device# show voice class server-group dialpeer 1

Voice class server-group: 1   AdminStatus: Up
Hunt-Scheme: preference
Total Remote Targets: 3

  Pref  Type  IP Address          IP Port
  ----  ----  -
  1     ipv4  10.1.1.1
  2     ipv4  10.1.1.2
  3     ipv4  10.1.1.3

```

**Step 2** **show running-config |section server-group**

The following example displays the running configuration for server groups.

**Example:**

```

Device#show running-config | section server-group
voice class server-group 1
ipv4 10.1.1.1 preference 1
ipv4 10.1.1.2 preference 2
ipv4 10.1.1.3 preference 3
description It has 3 entries
huntstop 1 resp-code 404 to 404
huntstop 2 resp-code 410 to 599
voice class server-group 2
ipv4 10.1.1.1
ipv4 10.1.1.2
ipv4 10.1.1.3
description It has 3 entries
hunt-scheme round-robin
huntstop 1 resp-code 401 to 599

```

# Configuration Examples for Server Groups in Outbound Dial Peers

## Server Groups in Outbound Dial Peers (Preference-Based Selection)

```

! Configuring the Server Group
Device(config)# voice class server-group 1
Device(config-class)# ipv4 10.1.1.1 preference 1
Device(config-class)# ipv4 10.1.1.2 preference 2
Device(config-class)# ipv4 10.1.1.3 preference 3
Device(config-class)# description It has 3 entries
Device (config-class)# huntstop 1 resp-code 404
Device(config-class)# huntstop 2 resp-code 410 to 599
Device(config-class)# exit

! Configuring an outbound SIP dial peer.
Device(config)# dial-peer voice 1 voip
!Associate a destination pattern
Device(config-dial-peer)# destination-pattern 3001
Device(config-dial-peer)# session protocol sipv2
!Associate a server group with the dial peer
Device(config-dial-peer)# session server-group 1
Device(config-dial-peer)# end

! Displays the configurations made for the outbound dial peer 181 associated with a server
group
Device# show voice class server-group dialpeer 1

Voice class server-group: 1      AdminStatus: Up
  Hunt-Scheme: preference
  Total Remote Targets: 3

  Pref   Type   IP Address                IP Port
  ----   ---   -
  1      ipv4   10.1.1.1                  -----
  2      ipv4   10.1.1.2
  3      ipv4   10.1.1.3

! Displays the configurations made for the server group.

Device# show voice class server-group 1

Voice class server-group: 1
  AdminStatus: Up              OperStatus: Up
  Hunt-Scheme: preference      Last returned server:
  Description: It has 3 entries
  Total server entries: 3

  Pref   Type   IP Address                IP Port
  ----   ---   -
  1      ipv4   10.1.1.1                  -----
  2      ipv4   10.1.1.2
  3      ipv4   10.1.1.3
-----

```

```

Total Huntstop tags: 2
Tag ID From Response code      To Response code
-----
 1      404                      404
 2      410                      599
-----

```

### Server Groups in Outbound Dial Peers (Round-Robin-Based Selection)

```

! Configuring the Server Group
Device(config)# voice class server-group 2
Device(config-class)# ipv4 10.1.1.1
Device(config-class)# ipv4 10.1.1.2
Device(config-class)# ipv4 10.1.1.3
Device(config-class)# hunt-scheme round-robin
Device(config-class)# huntstop 1 resp-code 401 to 599
Device(config-class)# description It has 3 entries
Device(config-class)# exit

! Configuring an outbound SIP dial peer.
Device(config)# dial-peer voice 2 voip
! Associate a destination pattern
Device(config-dial-peer)# destination-pattern 3001
Device(config-dial-peer)# session protocol sipv2
! Associate a server group with the dial peer
Device(config-dial-peer)# session server-group 2
Device(config-dial-peer)# end

! Displays the configurations made for the outbound dial peer 181 associated with a server
group
Device# show voice class server-group dialpeer 2

Voice class server-group: 2      AdminStatus: Up
Hunt-Scheme: round-robin
Total Remote Targets: 3

Pref   Type   IP Address                IP Port
----   ----   -
0      ipv4   10.1.1.3
0      ipv4   10.1.1.1
0      ipv4   10.1.1.2

! Displays the configurations made for the server group.
Device# show voice class server-group 2

Voice class server-group: 2
AdminStatus: Up                OperStatus: Up
Hunt-Scheme: round-robin      Last returned server: 10.1.1.2
Description: It has 3 entries
Total server entries: 3

Pref   Type   IP Address                IP Port
----   ----   -
0      ipv4   10.1.1.1
0      ipv4   10.1.1.2
0      ipv4   10.1.1.3

-----

Total Huntstop tags: 1

```

Tag ID	From Response code	To Response code
1	401	599

