Cisco Unified SIP SRST on Cisco 4000 Series Integrated Services Router

This chapter describes the support for Unified SIP SRST on the Cisco 4000 Series Integrated Services platform.

Note Unified SRST 12.6 on Cisco IOS XE Gibraltar 16.11.1a Release is not a recommended release version for call flows that include Multicast Music On Hold.

Overview

This chapter describes Unified SRST functionality on Cisco 4000 Series Integrated Services Routers for SIP phones. Unified SIP SRST provides backup to Unified Communications Manager when the IP connectivity to Unified Communications Manager is down.

Cisco Unified SIP SRST supports the following during a WAN outage:

- Basic Registration of SIP phones.
- Basic call support on SIP phones.
- Basic supplementary services such as Call Transfer, MOH, and Conference
- SIP phone to SIP phone
- SIP phone to PSTN / router voice-port
- SIP phone to Skinny Client Control Protocol (SCCP) phone
- SIP phone to WAN VoIP using SIP

Platform and Memory Support

From Unified SRST Release 10.0 (Cisco IOS XE Release 3.10S), Unified SIP SRST is supported on the Cisco 4000 Series Integrated Services platform. As part of the Cisco IOS XE Release 3.10S Release, support was introduced on the Cisco 4451-X Integrated Services Router. From Unified SRST Release 10.5 (Cisco IOS XE Release 3.13S), SIP SRST is supported on all Cisco 4000 Series Integrated Services Routers.

The following Cisco 4000 Series Integrated Services Router platforms are supported:

- Cisco ISR 4321 Integrated Services Routers
- Cisco ISR 4331 Integrated Services Routers
- Cisco ISR 4351 Integrated Services Routers
- Cisco ISR 4431 Integrated Services Routers
- Cisco ISR 4451 Integrated Services Routers

For more information on Platform and Memory Support, see Compatibility Information.

Cisco IOS Software Releases that Support Unified SRST

For information on the Unified SRST Release and the corresponding IOS Software, see Unified CME, Unified SRST, and Cisco IOS Software Version Compatibility Matrix for related compatibility information.

To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Install Cisco IOS XE Software

To verify that the recommended software is installed on the Cisco router and if necessary, download and install a Cisco IOS XE image, perform the following steps.

Before you begin

The Cisco router is installed including sufficient memory, all Cisco voice services hardware, and other optional hardware.

SUMMARY STEPS

1. Identify which Cisco IOS XE software release is installed on router. Log in to the router and use the **show version EXEC** command.
2. Compare the Cisco IOS XE release installed on the Cisco router to the information in the Cisco Unified CME, Unified SRST, and Cisco IOS Software Version Compatibility Matrix to determine whether the Cisco IOS release supports the recommended Unified SRST.
3. If necessary, download and extract the recommended Cisco IOS XE image to flash memory in the router.
4. To reload the Unified SRST router with the new software after replacing or upgrading the Cisco IOS XE release, use the **reload** privileged EXEC command.

DETAILED STEPS

	Command or Action	Purpose

Step 1	Identify which Cisco IOS XE software release is installed on router. Log in to the router and use the **show version EXEC** command.
	Example:
	`Router> show version`
	`Cisco IOS XE Software, Version`
	`BLD_POLARIS_DEV_LATEST_20200621_053200`
	`Cisco IOS Software [Amsterdam], ISR Software`
	`(X86_64_LINUX_IOSD-UNIVERSALK9-M),`
	`Version 17.3.1`
	`[S2C-build-polaris_dev-116144-/nobackup/mcpre/BLD-BLD_POLARIS_DEV_LATEST_20200621_0532`
	`00 259]`
	`Copyright (c) 1986-2020 by Cisco Systems, Inc.`
	`Compiled Sun 21-Jun-20 07:03 by mcpre`	

Step 2	Compare the Cisco IOS XE release installed on the Cisco router to the information in the Cisco Unified CME, Unified SRST, and Cisco IOS Software Version Compatibility Matrix to determine whether the Cisco IOS release supports the recommended Unified SRST.	
Step 3	If necessary, download and extract the recommended Cisco IOS XE image to flash memory in the router.	To find software installation information, access information located at www.cisco.com > Support > Products & Downloads > Networking Software > {Choose release} > Configuration Guides / System Management / Configuration fundamentals.

Step 4	To reload the Unified SRST router with the new software after replacing or upgrading the Cisco IOS XE release, use the **reload** privileged EXEC command.
	Example:
	`Router# reload`
	`System configuration has been modified. Save?`
	`[yes/no]: yes`	

	Command or Action	Purpose

	`Building configuration...`
	`[OK]`
	`Proceed with reload? [confirm]`
	`Jun 24 00:45:13.827: %PMAN-5-EXITACTION: R0/0: pvp:`
	` Process manager is exiting:`
	`process exit with reload chassis code`
	`Initializing Hardware ...`
	`Checking for PCIe device presence...done`
	`System integrity status: 0x610`
	`Rom image verified correctly`
	`System Bootstrap, Version 16.12(2r), RELEASE SOFTWARE`
	`Copyright (c) 1994-2019 by cisco Systems, Inc.`
	`Current image running: Boot ROM0`
	`Last reset cause: LocalSoft`
	`ISR4331/K9 platform with 4194304 Kbytes of main memory`
	`........`
	`Located isr4300-universalk9.BLD_POLARIS_DEV_LATEST_20200621_053200.SSA.bin`
	`Router>`	

Feature Support

The following features are supported for Unified SIP SRST on Cisco 4000 Series Integrated Services Platform:

- Auto-answer (If enabled on Unified Communications Manager)
- Alert/Semi-Consult/Attended/Consult Transfer
- Ad-hoc Software Conference
- Hold or Resume
- Headset Answer
- Caller ID Display
- Call Forward to Voice Hunt Group
- Call Transfer to a Voice Hunt Group
- Voicemail
- Message Waiting Indicator (MWI)
- Do Not Disturb (DND)
- DTMF
- Feature Button or Programmable Line Key (PLK) - If enabled on Unified Communications Manager
- Key Expansion Module (KEM - Supported only on the 8851/8851NR/8861 phones)
- Bulk Registration Support
- Enabling or Disabling KPML

- Alias Feature
- Call Forward (All, Busy, No Answer, Mailbox)
- Call Forward All Softkey on Phone
- Unicast MOH
- Audio codecs (G.722, G.711, G.729, iLBC)
- Translation Profile
- Conference Blocking
- Transfer Blocking
- COR
- Voice Class Codec
- SNMP/MIB (Supported only to get mode and number of registered phones)
- Speed Dial (If enabled on Unified Communications Manager)
- Call Waiting (If enabled on Unified Communications Manager)
- Forced Authorization Code
- Redial
- Speakerphone (Dialing, Answering)
- System Message
- After Hours
- SSH to Phone
- Span to PC (except Cisco IP Phone 8831)
- Web Access to Phone
- Voice Hunt Group (Support for Parallel, Sequential, Peer, and Longest-idle hunt groups). Basic features such as Call, Hold or Resume are only supported.)

Restrictions of Unified SRST on Cisco 4000 Series Integrated Services Routers

- Multicast MOH for SIP is not supported on the Cisco 4000 Series Integrated Services Routers.
- Transcoding is not supported on the Unified SRST.
- Voice VRF is not supported for SCCP SRST on Cisco Integrated Services Router Generation 2 (ISR G2).
- Shared lines and Mixed shared lines are not supported on the Unified SRST (supported on the Unified E-SRST).
- Privacy (on hold) is not supported on the Unified SRST (supported on the Unified E-SRST).
- SNMP/MIB support is restricted to fetching information on mode and number of registered phones.

- The CLI command **max-redirect** is not supported for SIP on Unified SRST.
- Unified SRST supports only the basic voice hunt group features. To configure advanced voice hunt group features, you must deploy the Cisco Unified Enhanced Survivable Remote Site Telephony.
- Video Calling is not supported on Unified SIP SRST.

Unified IP Phone Support

Unified SIP SRST on Cisco 4000 Series Integrated Services Platform is supported on all the SIP phones, including Cisco IP Phone 7800 Series and Cisco IP Phone 8800 Series.

Cisco Jabber with Unified SRST

Unified SRST 12.8 (Cisco IOS XE Amsterdam 17.2.1r) and later releases support the following Cisco Jabber clients:

- Cisco Jabber for Windows,12.9
- Cisco Jabber for Mac, 12.9

Cisco Unified Communications Manager Compatibility

For more information on Unified Communications Manager compatibility, see Cisco Unified Communications Manager Compatibility Matrix.

Installing Cisco Unified Communications Manager

When installing Cisco Unified Communications Manager, consider the following:

- See the installation instructions for your version in the Cisco Unified Communications Manager Install and Upgrade Guides.
- Integrate Cisco Unified SRST with Cisco Unified Communications Manager. Integration is performed from Cisco Unified Communications Manager. See the Integrating Cisco Unified SIP SRST with Cisco Unified Communications Manager section.

Integrating Cisco Unified SIP SRST with Cisco Unified Communications Manager

The procedure for integrating Unified SRST with Cisco Unified Communications Manager is as follows:

For Cisco Communications Manager integration with Unified SIP SRST, you must create an SRST reference and apply it to a device pool. An SRST reference is the IP address of the Cisco Unified SRST Router.

SUMMARY STEPS

1. Create an SRST reference.

2. Apply the SRST reference or the default gateway to one or more device pools.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Create an SRST reference.	
Step 2	Apply the SRST reference or the default gateway to one or more device pools.	

Supported PSTN Trunk Connectivity

Unified SRST is supported with SIP trunks. Also, Unified SIP SRST supports the following trunk types:

- FXO/FXS
- Basic Rate ISDN
- Primary Rate ISDN (T1 or E1)

Language Support

For information on language support, see Localization Matrix.

Switch Support

Unified SRST supports all PRI and BRI switches including the following:

- basic-1tr6
- basic-5ess
- basic-dms100
- basic-net3
- basic-ni
- basic-ntt NTT switch type for Japan
- basic-ts013
- primary-4ess Lucent 4ESS switch type for the United States
- primary-5ess Lucent 5ESS switch type for the United States
- primary-dms100 Northern Telecom DMS-100 switch type for the United States
- primary-net5 NET5 switch type for the United Kingdom, Europe, Asia, and Australia
- primary-ni National ISDN switch type for the United States
- primary-ntt NTT switch type for Japan
- primary-qsig QSIG switch type

primary-ts014 TS014 switch type for Australia (obsolete)

Interface Support for Unified SRST

Unified SRST routers have multiple interfaces that are used for signaling and data packet transfers. The two types of interfaces available on a Cisco router include the physical interface and the virtual interface. The type of physical interfaces available on a router depends on its interface processors or port adapters. Virtual interfaces are software-based interfaces that you create in the memory of the networking device using Cisco IOS commands. To configure a virtual interface for connectivity, you can use the Loopback Interface for Unified SRST.

The following interfaces are supported on Unified SRST:

- Gigabit Ethernet Interface (IEEE 802.3z) (**interface gigabitethernet**)
- Loopback Interface (**interface loopback**)
- Fast Ethernet Interface (**interface fastethernet**)

Simple Network Management Protocol (SNMP) Support for Unified SRST

Unified SRST supports Simple Network Management Protocol (SNMP) Management Information Base (MIBs) for monitoring the product status. Unified SRST Release 12.6 and later versions is SNMP Version 3 (SNMPv3) compliant. The following is the main SNMP MIB supported by Unified SRST:

- CISCO-SRST-MIB

For information on configuration of SNMP version 3 on Unified SRST router, see SNMP Configuration Guide.

Licensing

This section provides information on licensing of Cisco Unified Survivable Remote Site Telephony (Unified SRST).

Cisco Smart Licensing for Unified SRST

Cisco Smart Licensing is a software licensing model that provides visibility of ownership and usage through the Cisco Smart Software Manager (CSSM) portal. CSSM is a central license repository that manages licenses across all Cisco products that you own, including Unified SRST. Devices send license usage to CSSM either directly or use an on-premises satellite. Your Smart Account Administrator controls your access to CSSM. Use your Cisco credentials to access the CSSM portal using http://software.cisco.com.

Smart Licensing applies to all platform technology (UCK9, Security) and Unified SRST feature licenses that the router uses. Unified SRST requires one license entitlement (SRST_EP) for each configured SIP or SCCP phone.

CSSM shows license usage across all devices that are registered to a virtual account. A Virtual Account License Inventory displays the quantity of licenses that are purchased, those licenses in use, and a balance. An **Insufficient Licenses** alert is displayed if the license balance is below 0.

For example, consider a smart account in CSSM with 50 SRST_EP licenses. If you have a single registered Unified SRST router with 20 phones configured, the CSSM licenses page shows **Purchased** as 50, **In Use** as 20 and **Balance** as 30.

For more information on Smart Software Manager, see the Cisco Smart Software Manager User Guide.

Note The SRST_EP license count reflects the total phone count for both the ephones and voice register pools that are configured in the Unified SRST irrespective of whether the phones are registered or not. To avoid unnecessary reporting while Unified SRST is being configured, license usage is reported three minutes after the last configuration change.

Note Unified SRST Smart Licenses also provide RTU entitlement for routers that are not configured for Smart Licensing.

Smart License Operation

Cisco IOS XE Everest 16.5.1 Release to Cisco IOS XE Fuji 16.9.1 Release

Cisco 4000 Series Integrated Services Routers support Smart Licensing as an alternative to Cisco Software RTU Licensing. Use the **license smart enable** command to enable Smart Licensing. To disable Smart Licensing, use the **no** form of the command and re-accept the EULA using the **license accept end user agreement** command.

Cisco IOS XE Gibraltar 16.10.1 Release Onwards

The Cisco RTU Licensing and the CLI **license smart enable** command are deprecated. Smart Licensing is mandatory from this release.

Cisco IOS XE Everest 16.5.1 Release to Cisco IOS XE Amsterdam 17.3.1a Release

Routers configured to use Smart Licensing offer a 90-day evaluation period, during which you can use all the features without registering to CSSM. A Unified SRST device is associated with CSSM using a registration token. You can obtain the registration token from the virtual CSSM account or from an on-premises satellite. Once registered, the evaluation period pauses and you can use the balance license later. You cannot renew the evaluation period on its expiry.

Warning Unified SRST shuts down when the router is unregistered and allowed to pass in to the Evaluation Expired state.

To register the Unified SRST router with CSSM, use **license smart register idtoken** command. For information on registering the device with CSSM, see Software Activation Configuration Guide.

Upon successful registration, the device sends an authorization request to CSSM for the licenses in use. For each license type requested, if the Smart Account has sufficient licenses, CSSM responds with **Authorized** . If the Smart Account does not have sufficient licenses, CSSM responds with **Out of Compliance** .

Post successful authorization of the request, licenses are bound to the requesting device until the next authorization request submission. An authorization request is sent every 30 days or when there is any change in license consumption, to maintain the registration with CSSM. The authorization expires if you do not update the license request for the router within 90 days. The certificate issued to identify the router at the time of registration is valid for one year and renewed every six months. The router displays the License authorization as follows:

```
Router# show license summary
Smart Licensing is ENABLED
Registration:
Status: REGISTERED
Smart Account: ABC
Virtual Account: XYZ
Export-Controlled Functionality: Not Allowed
Last Renewal Attempt: None
Next Renewal Attempt: Jun 07 12:08:10 2017 UTC
License Authorization:
Status: AUTHORIZED
Last Communication Attempt: SUCCESS
Next Communication Attempt: Apr 13 07:11:48 2017 UTC
License Usage:
License                  Entitlement tag            Count Status
-----------------------------------------------------------------------------
ISR_4351_UnifiedCommun.. (ISR_4351_UnifiedCommun..) 1     AUTHORIZED
SRST v12 Endpoint Li...  (SRST_EP)                  4     AUTHORIZED
```

Cisco IOS XE Gibraltar 16.12.1 Release to Cisco IOS XE Amsterdam 17.3.1a Release

Specific License Reservation (SLR) is supported on Cisco 4000 Series Integrated Services Routers. SLR allows reservation and utilization of Cisco Smart Licenses without communicating the license information to CSSM. To reserve specific licenses for a device, generate request code from the device. Enter the request code in CSSM along with the required licenses and their quantity, and generate authorization code. Enter the authorization code on the device to map the license to the Unique Device identifier (UDI).

Note If upgrading to IOS XE Amsterdam 17.3.1a with a license reservation in place, update the reservation to include version 14, rather than version 12 SRST licenses. The reservation may be updated before or after the software upgrade.

Cisco IOS XE Amsterdam 17.3.2 and Cisco IOS XE Bengaluru 17.4.1a Release Onwards

This release introduces a new paradigm for tracking license usage across your business. In earlier releases, license authorization was forward looking, binding licenses to a device until the next authorization request. Actual license usage during the proceeding reporting period is now sent to CSSM, allowing you to plan ongoing license requirements based on historical usage data. Initial device registration is no longer required to use most platform functionality and the evaluation period is deprecated.

License usage reports are submitted periodically according to a minimum reporting policy set for your account. Typically, this period could be once per year. However, you can generate reports more frequently if the use of licensed features varies over time. CSSM acknowledges each Resource Utilization Monitoring (RUM) report to ensure that the usage is recorded reliably. If the router does not receive an acknowledgment within the minimum reporting period, call processing is disabled. Call processing is resumed when a valid acknowledgment is received.

Reports can be submitted to CSSM directly or through a satellite. Cisco Smart Licensing Utility (CSLU) applications can also receive usage reports, providing you with more flexibility in managing your license usage. Also, when a device is not able to communicate directly with a licensing server, a signed usage report can be generated and manually uploaded to CSSM. The acknowledgment that is generated by CSSM must be uploaded to the device within the license reporting policy period to ensure continued use.

As license reporting is now based on historical usage, the registration process that is used previously has been replaced with a trust association that also defines the reporting policy set in your account. Establishing trust with CSSM or Cisco Smart Software Manager Satellite uses an identity token similar to earlier registrations. Use the **license smart trust idtoken** *token* command to establish the trust relationship within the initial reporting period set for the device. The CLI **license smart register** command is deprecated from this release.

Current license usage for Cisco Unified SRST is displayed using the **show license summary** command:

Warning When using any of the following releases, Unified SRST shuts down if the router does not receive a report acknowledgment from CSSM before the acknowledgment deadline set by the account policy: 17.3.2, 17.3.3, 17.3.4a, 17.6.1a, or any 17.4 or 17.5 release. Unified SRST does not shut down in this way with later releases.

Note Smart License Reservation (SLR) for SRST licenses is not compatible with IOS XE Amsterdam 17.3.2 and later releases. Even if a reservation is in place when upgrading to one of these releases, license use reporting will still be required in accordance with the device policy.

```
Router#show license summary
License Usage:
License                    Entitlement tag              Count   Status
-----------------------------------------------------------------------------
appxk9..................(ISR_4400_Application) ......1...... IN USE
uck9................... (ISR_4400_UnifiedCommun..)...1.......IN USE
securityk9............. (ISR_4400_Security)......... 1.......IN USE
SRST_E_EP.............. (SRST_E_EP)..................2.......IN USE
SRST_EP.................(SRST_EP)...................18...... IN USE
```

Configure SIP Registrar Functionality for SIP Phones on Unified SRST

Session Initiation Protocol (SIP) registrar functionality in Cisco IOS software is an essential part of Cisco Unified SIP Survivable Remote Site Telephony (SRST). According to RFC 3261, a SIP registrar is a server that accepts Register requests.

Unified SIP SRST provides backup to Cisco Unified Communications Manager. The registrar functionality is configured on the Unified SRST gateway so as to assist fallback of endpoints to Unified SRST from Unified Communications Manager.

These services are used by a SIP IP phone if there is a WAN connection outage, and the SIP phone is unable to communicate with its primary SIP call control (IP-PBX). The Unified SIP SRST device also provides PSTN gateway access for placing and receiving PSTN calls.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice service voip**
4. **allow-connections sip to sip**
5. **sip**
6. **registrar server** [**expires** [**max** *sec*] [**min** *sec*]]
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**voice service voip** **Example:** `Router(config)# voice service voip`	Enters voice service configuration mode.
Step 4	**allow-connections sip to sip** **Example:** `Router(config-voi-srv)# allow-connections sip to sip`	Allows connections from SIP to SIP endpoints.
Step 5	**sip** **Example:** `Router(config-voi-srv)# sip`	Enters SIP configuration mode.
Step 6	**registrar server** [**expires** [**max** *sec*] [**min** *sec*]] **Example:** `Router(config)# call-manager-fallback`	Enables SIP registrar functionality. The keywords and arguments are defined as follows: • **expires** : (Optional) Sets the active time for an incoming registration. • **max** *sec* : (Optional) Maximum expiration time for a registration, in seconds. The range is from 600 to 86400. The default is 3600. **Note** Ensure that the registration expiration timeout is set to a value smaller than the TCP connection aging timeout to avoid disconnection from the TCP.

	Command or Action	Purpose
		• **min** *sec* : (Optional) Minimum expiration time for a registration, in seconds. The range is from 60 to 3600. The default is 60.
Step 7	**end** **Example:** `Router(conf-serv-sip)# end`	Returns to privileged EXEC mode.

Configure Backup Registrar Service to SIP Phones

Backup registrar service to SIP IP phones can be provided by configuring a voice register pool on SIP gateways. The voice register pool configuration provides registration permission control and can be used to configure some dial-peer attributes that are applied to the dynamically created VoIP dial peers when SIP phone registrations match the pool. The following call types are supported:

- SIP IP phone to or from:
- Local PSTN
- Local analog FXS phones
- Local SIP IP phone

The commands in the configuration provide registration permission control and set up a basic voice register pool. The pool gives users control over which registrations are accepted by a Cisco Unified SIP SRST device and which can be rejected. Registrations that match this pool create VoIP SIP dial peers with the dial-peer attributes set to these configurations. Although only the **id** command is mandatory, this configuration example shows basic functionality.

Restrictions

- The **id** command identifies the individual SIP IP phone or sets of SIP IP phones that are to be configured. Thus, the**id**command configured in Step 5 is required and must be configured before any other voice register pool commands. For Unified SRST, It is recommended to configure **id ip/nework/device-id-name** and avoid using **id mac**.

Note To monitor SIP proxies, the **call fallback active** command must be configured, as described in Step 3.

Note The command **proxy** described in Step 7 is an optional configuration.

Note It is recommended that **id mac** command is not configured for Unified SRST, as the phones falling back from Unified Communications Manager to Unified SRST do not mostly fall back on the same network.

Before you begin

The SIP registrar must be configured before a voice register pool is set up.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **call fallback active**
4. **voice register pool** *tag*
5. **id** [{**network** *address* **mask** *mask* |**ip** *address* **mask** *mask* |**mac** *address* }] [**device-id-name** *devicename*]
6. **preference** *preference-order*
7. **proxy** *ip-address* [**preference** *value*] [**monitor probe** {**icmp-ping** | **rtr**} [*alternate-ip-address*]]
8. **voice-class codec** *tag*
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**call fallback active** **Example:** `Router(config)# call fallback active`	(Optional) Enables a call request to fall back to alternate dial peers if there is network congestion. • This command is used if you want to monitor the proxy dial peer and fallback to the next preferred dial peer. For full information on the **call fallback active** command, see PSTN Fallback Feature.
Step 4	**voice register pool** *tag* **Example:** `Router(config)# voice register pool 12`	Enters voice register pool configuration mode for SIP phones. Use this command to control which registrations are accepted or rejected by a Cisco Unified SIP SRST device.
Step 5	**id** [{**network** *address* **mask** *mask* |**ip** *address* **mask** *mask* |**mac** *address* }] [**device-id-name** *devicename*] **Example:** `Router(config-register-pool)# id network 172.16.0.0 mask 255.255.0.0`	Explicitly identifies a locally available individual or set of SIP IP phones. The keywords and arguments are defined as follows: • **network** *address* **mask** *mask* : The **network** *address* mask *mask* keyword/argument combination is used to accept SIP Register messages for the indicated phone numbers from any IP phone within the indicated IP subnet.

	Command or Action	Purpose
		• **ip** *address* **mask** *mask*: The **ip** *address* mask *mask* keyword/argument combination is used to identify an individual phone. • **mac** *address*: MAC address of a particular Cisco Unified IP Phone. • **device-id-name** *devicename*: Defines the device name to be used to download the phone's configuration file.
Step 6	**preference** *preference-order* **Example:** `Router(config-register-pool)# preference 2`	Sets the preference order for the VoIP dial peers to be created. Range is from 0 to 10. Default is 0, which is the highest preference. The preference must be greater (lower priority) than the preference configured with the **preference** keyword in the **proxy** command.
Step 7	**proxy** *ip-address* [**preference** *value*] [**monitor probe** {**icmp-ping** | **rtr**} [*alternate-ip-address*]] **Example:** `Router(config-register-pool)# proxy 10.2.161.187 preference 1`	(Optional) Autogenerates additional VoIP dial peers to reach the main SIP proxy whenever a Cisco Unified SIP IP Phone registers with a Cisco Unified SIP SRST gateway. The keywords and arguments are defined as follows: • *ip-address* : IP address of the SIP proxy. • **preference** *value* : (Optional) Defines the preference of the proxy dial peers that are created. The preference must be less (higher priority) than the preference configured with the **preference** *value* command. Range is from 0 to 10. The highest preference is 0. There is no default. • **monitor probe** : (Optional) Enables monitoring of proxy dial peers. • **icmp-ping**: Enables monitoring of proxy dial peers using ICMP ping. **Note** The dial peer on which the probe is configured will be excluded from call routing only for outbound calls. Inbound calls can arrive through this dial peer. • **rtr**: Enables monitoring of proxy dial peers using RTR probes. • *alternate-ip-address* : (Optional) Enables monitoring of alternate IP addresses other than the proxy address. For example, to monitor a gateway front end to a SIP proxy.

	Command or Action	Purpose
Step 8	**voice-class codec** *tag* **Example:** `Router(config-register-pool)# voice-class codec 15`	Sets the voice class codec parameters. The *tag* argument is a codec group number between 1 and 10000.
Step 9	**end** **Example:** `Router(config-register-pool)# end`	Returns to privileged EXEC mode.

Configure Backup Registrar Service to SIP Phones (Using Optional Commands)

The prior configurations set up a basic voice register pool. The configuration in this procedure adds optional attributes to increase functionality. As part of this configuration, you can support:

- Translation Profile—Applies the translation profile to a specific directory number or to all directory numbers on a SIP phone.
- Alias—Allows Cisco Unified SIP IP Phones to handle inbound PSTN calls to phone numbers that are unavailable when the main SIP call control (IP-PBX) is not available.
- Class of restriction (COR)—COR specifies which incoming dial peers can use which outgoing dial peers to make a call. Each dial peer can be provisioned with an incoming and outgoing COR list.

Before you begin

Before configuring the **alias** command, translation rules must be set using the translation-profile outgoing (**voice register pool**) command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice register pool***tag*
4. **translation-profile outgoing** *profile-tag*
5. **alias** *tag pattern* **to** *target* [**preference** *value*]
6. **cor** {**incoming** | **outgoing**} *cor-list-name* {*cor-list-number starting-number* [- *ending-number*] | **default** }
7. **incoming called-number** *[number]*
8. **number** *tag number-pattern* {**preference***value*} [**huntstop**]
9. **dtmf-relay** [**cisco-rtp**] [**rtp-nte**] [**sip-notify**]
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: `Router> enable`	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**voice register pool***tag* **Example:** `Router(config)# voice register pool 12`	Enters voice register pool configuration mode. Use this command to control which registrations are accepted or rejected by a Cisco Unified SIP SRST device.

Step 4	**translation-profile outgoing** *profile-tag*
	Example:
	`Router(config-register-pool)#`
	`voice translation-rule 1`
	`rule 1 /1000/ /1006/`
	`!`
	`!`
	`voice translation-profile 1`
	`translate called 1`
	`!`
	`voice register pool xxx`
	`translation-profile outgoing 1`
		Use this command to apply the translation profile to a specific directory number or to all directory numbers on a SIP phone.
		• *Profile-tag* : Translation profile name to handle translation to outgoing calls.

Step 5	**alias** *tag pattern* **to** *target* [**preference** *value*] **Example:** `Router(config-register-pool)# alias 1 94... to 91011 preference 8`	Allows Cisco Unified SIP IP Phones to handle inbound PSTN calls to phone numbers that are unavailable when the main proxy is not available. The keywords and arguments are defined as follows: • *tag* : Number from 1 to 5 and the distinguishing factor when there are multiple alias commands. • *pattern*: The prefix number; matches the incoming phone number and may include wildcards. • **to** : Connects the tag number pattern to the alternate number. • *target*: The target number; an alternate phone number to route incoming calls to match the number pattern. • **preference** *value* : (Optional) Assigns a dial-peer preference value to the alias. The *value* argument is the value of the associated dial peer, and the range is from 1 to 10. There is no default.
Step 6	**cor** {**incoming** | **outgoing**} *cor-list-name* {*cor-list-number starting-number* [- *ending-number*] | **default** } **Example:** `Router(config-register-pool)# cor incoming call91 1 91011`	Configures a class of restriction (COR) on the VoIP dial peers associated with directory numbers. COR specifies which incoming dial peers can use which outgoing dial peers to make a call. Each dial peer can be provisioned with an incoming and outgoing COR list. The keywords and arguments are defined as follows:

	Command or Action	Purpose
		• **incoming** : COR list to be used by incoming dial peers. • **outgoing** : COR list to be used by outgoing dial peers. • *cor-list-name*: COR list name. • *cor-list-number*: COR list identifier. The maximum number of COR lists that can be created is four, comprised of incoming or outgoing dial peers. • *starting-n*umber: Start of a directory number range, if an ending number is included. Can also be a standalone number. • (Optional) Indicator that a full range is configured. • *ending-number*: (Optional) End of a directory number range. • **default** : Instructs the router to use an existing default COR list.
Step 7	**incoming called-number** *[number]* **Example:** `Router(config-register-pool)# incoming called-number 308`	Applies incoming called parameters to dynamically created dial peers. The number argument is optional and indicates a sequence of digits that represent a phone number prefix.
Step 8	**number** *tag number-pattern* {**preference***value*} [**huntstop**] **Example:** `Router(config-register-pool)# number 1 50.. preference 2`	Indicates the E.164 phone numbers that the registrar permits to handle the Register message from the Cisco Unified SIP IP Phone. The keywords and arguments are defined as follows: • *tag* : Number from 1 to 10 and the distinguishing factor when there are multiple number commands. • *number-pattern*: Phone numbers (including wildcards and patterns) that are permitted by the registrar to handle the Register message from the SIP IP phone. • **preference** *value* : (Optional) Defines the number list preference order. • **huntstop**: (Optional) Stops hunting if the dial peer is busy.
Step 9	**dtmf-relay** [**cisco-rtp**] [**rtp-nte**] [**sip-notify**] **Example:** `Router(config-register-pool)# dtmf-relay rtp-nte`	Specifies how a SIP gateway relays dual tone multifrequency (DTMF) tones between telephony interfaces and an IP network. The keywords are defined as follows: • **cisco-rtp** : (Optional) Forwards DTMF tones by using Real-Time Transport Protocol (RTP) with a Cisco proprietary payload type.

	Command or Action	Purpose
		• **rtp-nte** : (Optional) Forwards DTMF tones by using RTP with the Named Telephone Event (NTE) payload type. • **sip-notify** : (Optional) Forwards DTMF tones using SIP NOTIFY messages.
Step 10	**end** **Example:** `Router(config-register-pool)# end`	Returns to privileged EXEC mode.

Verify SIP Registrar Configuration

To help you troubleshoot a SIP registrar and voice register pool, perform the following steps.

SUMMARY STEPS

1. **debug voice register errors**
2. **debug voice register events**
3. **show sip-ua status registrar**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**debug voice register errors**
	Example:
	`Router# debug voice register errors`
	`*Apr 22 11:52:54.523 PDT: VOICE_REG_POOL: Contact doesn't match any pools`
	`*Apr 22 11:52:54.539 PDT: VOICE_REG_POOL: Register request for (33015) from (10.2.152.39)`
	`*Apr 22 11:52:54.539 PDT: VOICE_REG_POOL: Contact doesn't match any pools.`
	`*Apr 22 11:52:54.559 PDT: VOICE_REG_POOL: Register request for (33017) from (10.2.152.39)`
	`*Apr 22 11:53:04.559 PDT: VOICE_REG_POOL: Maximum registration threshold for pool(3) hit`
		Use this command to debug errors that happen during registration.
		If there are no voice register pools configured for a particular registration request, the message “Contact doesn’t match any pools” is displayed.

Step 2	**debug voice register events**
	Example:
	`Router# debug voice register events`
	`Apr 22 10:50:21.731 PDT: VOICE_REG_POOL: Contact matches pool 1`
	`Apr 22 10:50:21.731 PDT: VOICE_REG_POOL: key(91011) contact(192.168.0.2) add to contact table`
	`Apr 22 10:50:21.731 PDT: VOICE_REG_POOL: key(91011) exists in contact table`
	`Apr 22 10:50:21.731 PDT: VOICE_REG_POOL: contact(192.168.0.2) exists in contact table, ref updated`
	`Apr 22 10:50:21.731 PDT: VOICE_REG_POOL: Created`
		Using the **debug voice register events** command should suffice to display registration activity. Registration activity includes matching of pools, registration creation, and automatic creation of dial peers. For more details and error conditions, you can use the **debug voice register errors** command.
		The phone number 91011 registered successfully, and *type 1* is reported, which means there is a pre-existing VoIP dial peer.

	Command or Action	Purpose

	`dial-peer entry of type 1`
	`Apr 22 10:50:21.731 PDT: VOICE_REG_POOL:`
	`Registration successful for 91011, registration id`
	`is 257`	

Step 3	**show sip-ua status registrar**
	
	Example:
	
	`Router# show sip-ua status registrar`
	`Line    destination expires(sec) contact`
	`======= =========== ============ =======`
	`91021   192.168.0.3 227           192.168.0.3`
	`91011   192.168.0.2 176           192.168.0.2`
	`95021   10.2.161.50 419           10.2.161.50`
	`95012   10.2.161.50 419           10.2.161.50`
	`95011   10.2.161.50 420           10.2.161.50`
	`95500   10.2.161.50 420           10.2.161.50`
	`94011   10.2.161.40 128           10.2.161.40`
	`94500   10.2.161.40 129           10.2.161.40`
		Use this command to display all the SIP endpoints currently registered with the contact address.

Verify Proxy Dial-Peer Configuration

To use the **icmp-ping** keyword with the **proxy** command to assist in troubleshooting proxy dial peers, perform the following steps.

SUMMARY STEPS

1. **configure terminal**
2. **voice register pool** *tag*
3. **proxy** *ip-address* [**preference** *value*] [**monitor probe** {**icmp-ping** | **rtr**} [*alternate-ip-address*]]
4. **end**
5. **show voice register dial-peers**
6. **show dial-peer voice**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Use this command to enter global configuration mode.
Step 2	**voice register pool** *tag* **Example:** `Router(config)# voice register pool 1`	Use this command to enter voice register pool configuration mode.

Step 3	**proxy** *ip-address* [**preference** *value*] [**monitor probe** {**icmp-ping** | **rtr**} [*alternate-ip-address*]]
	
	Example:
	
	`Router(config-register-pool)# proxy 10.2.161.187`
	`preference 1 monitor probe icmp-ping`
		Set the **proxy** command to monitor with **icmp-ping**.

	Command or Action	Purpose
Step 4	**end** **Example:** `Router(config-register-pool)# end`	Returns to privileged EXEC mode.

Step 5	**show voice register dial-peers**
	Example:
	`Router# show voice register dial-peers`
	`dial-peer voice 40035 voip`
	`preference 5`
	`destination-pattern 91011`
	`session target ipv4:192.168.0.2`
	`session protocol sipv2`
	`voice-class codec 1`
	`dial-peer voice 40036 voip`
	`preference 1`
	`destination-pattern 91011`
	`session target ipv4:10.2.161.187`
	`session protocol sipv2`
	`voice-class codec 1`
	`monitor probe icmp-ping 10.2.161.187`
		Use this command to verify dial-peer configurations, and notice that icmp-ping monitoring is set.

Step 6	**show dial-peer voice**
	Example:
	`Router# show dial-peer voice`
	`VoiceOverIpPeer40036`
	`peer type = voice, information type = voice,`
	``description = `',``
	``tag = 40036, destination-pattern = `91011',``
	``answer-address = `', preference=1,``
	`CLID Restriction = None`
	``CLID Network Number = `'``
	`CLID Second Number sent`
	``source carrier-id = `', target carrier-id = `',``
	``source trunk-group-label = `', target``
	``trunk-group-label = `',``
	``numbering Type = `unknown'``
	`group = 40036, Admin state is up, Operation state`
	` is`
	`up,`
	``incoming called-number = `', connections/maximum``
	`=`
	`0/unlimited,`
	`! Default output for incoming called-number command`
	`DTMF Relay = disabled,`
	`modem transport = system,`
	`huntstop = disabled,`
	`in bound application associated: 'DEFAULT'`
	`out bound application associated: ''`
	`dnis-map =`
	`permission :both`
	`incoming COR list:maximum capability`
	`! Default output for cor command`
	`outgoing COR list:minimum requirement`
	`! Default output for cor command`
	`Translation profile (Incoming):`
	`Translation profile (Outgoing):`
	`incoming call blocking:`
		Use the **show dial-peer voice** command on dial peer 40036, and notice the monitor probe status.
		Note Also highlighted is the output of the **cor** and **incoming called-number** commands.

	Command or Action	Purpose
	translation-profile = `' disconnect-cause = `no-service' advertise 0x40 capacity_update_timer 25 addrFamily 4 oldAddrFamily 4 type = voip, session-target = `ipv4:10.2.161.187', technology prefix: settle-call = disabled ip media DSCP = ef, ip signaling DSCP = af31, ip video rsvp-none DSCP = af41,ip video rsvp-pass DSCP = af41 ip video rsvp-fail DSCP = af41, UDP checksum = disabled, session-protocol = sipv2, session-transport = system, req-qos = best-effort, acc-qos = best-effort, req-qos video = best-effort, acc-qos video = best-effort, req-qos audio def bandwidth = 64, req-qos audio max bandwidth = 0, req-qos video def bandwidth = 384, req-qos video max bandwidth = 0, RTP dynamic payload type values: NTE = 101 Cisco: NSE=100, fax=96, fax-ack=97, dtmf=121, fax-relay=122 S=123, ClearChan=125, PCM switch over u-law=0,A-law=8 RTP comfort noise payload type = 19 fax rate = voice, payload size = 20 bytes fax protocol = system fax-relay ecm enable fax NSF = 0xAD0051 (default) codec = g729r8, payload size = 20 bytes, Media Setting = flow-through (global) Expect factor = 0, Icpif = 20, Playout Mode is set to adaptive, Initial 60 ms, Max 300 ms Playout-delay Minimum mode is set to default, value 40 ms Fax nominal 300 ms Max Redirects = 1, signaling-type = cas, VAD = enabled, Poor QOV Trap = disabled, Source Interface = NONE voice class sip url = system, voice class sip rel1xx = system, monitor probe method: icmp-ping ip address: 10.2.161.187, Monitored destination reachable voice class perm tag = `' Time elapsed since last clearing of voice call statistics never Connect Time = 0, Charged Units = 0, Successful Calls = 0, Failed Calls = 0, Incomplete Calls = 0 Accepted Calls = 0, Refused Calls = 0, Last Disconnect Cause is "", Last Disconnect Text is "", Last Setup Time = 0.	

Unified SRST, Unified E-SRST, and Unified Secure SRST Password Policy

From Unified SRST 12.6 Release (Cisco IOS XE Gibraltar 16.11.1a) onwards, all configurations on Unified SRST, Unified E-SRST, and Unified Secure SRST must meet the password policy.

General Password Policy Guidelines:

- Passwords must have a minimum of 6 alphanumeric characters, and a maximum of 15 alphanumeric characters.
- Passwords must not contain symbols or special characters.
- Passwords must contain at least one numeral, one uppercase alphabet, and one lowercase alphabet.

If the password is not configured as per the policy, the Unified SRST router displays an error message:

```
Error: The password you have entered is incorrect.
Your password must contain:
1. A minimum of 6 and a maximum of 15 alphanumeric characters, excluding symbols and
special characters.
2. A minimum of one numeral, one uppercase alphabet, and one lowercase alphabet.
```

The Unified CME password policy is applicable for Unified SRST configurations on Cisco IOS XE 16.11.1a and later. Unified SRST password policy is not applicable in the following scenarios:

- Upgrade from an older IOS version to Cisco IOS XE 16.11.1a
- Downgrade from Cisco IOS XE 16.11.1a to an older version

Guidelines for Password Configuration and Encryption

Configure the passwords relevant to Unified SRST, Unified E-SRST, and Unified Secure SRST using the CLI commands as follows:

- **call-manager-fallback** configuration mode
- **xml user** *username* **password [0|6]***password privilege-level*

Note The 0 in the parameter **[0|6]** mentioned in the CLI command represents plain, unencrypted text and 6 represents level 6 password encryption.

- Apart from the parameter configurations ([0|6]) at the command level, configure the Unified SRST router to support encryption.
- Configure the CLI command **encrypt password** under **call-manager-fallback** configuration mode to support type 6 encryption on the Unified SRST router.
- Also, it is mandatory to configure **key config-key password-encrypt***[key]***password encryption** *aes* to support encryption on the Unified SRST router.

- If the key used to encrypt the password is replaced with a new key (replace key or re-key), then the password is re-encrypted with the new key.
- You must adhere to SRST Password Policy for both type 0 and type 6 parameters that you configure on Unified SRST.
- Configure **no encrypt password** for type 0 password on the Unified SRST router. A type 0 password is displayed as unencrypted plain text.
- If you are performing a downgrade from Unified SRST 12.6 to an earlier version, then you must execute the CLI command **no encrypt password**. If the CLI command **no encrypt password** is configured, the password is presented as plain text.

The following is a sample configuration on Unified SRST router to support password encryption:

```
Router(config)#key config-key password-encrypt <cisco123>
Router(config)#password encryption aes
Router(config)#call-manager-fallback
Router(config-cm-fallback)encrypt password
```

Deprecation of CLI commands

From Unified SRST Release 12.6 onwards, the following CLI commands that are configured under **call-manager-fallback** configuration mode are deprecated to enhance product security:

- **log password***password-string*
- **xmltest**
- **xmlschema***schema-url*
- **xmlthread** *number*

Removal of Passwords and Keys from Logs

From Unified SRST Release 12.6 onwards, passwords and sRTP keys are not printed to logs to enhance security of Unified SRST. The information about keys is available only in the show commands from Unified SRST 12.6 release onwards. The CLI command **show ephone offhook** for SCCP and **show sip-ua calls** for SIP are enhanced to display the keys that are in use per media stream, along with the sRTP Ciphers.

The following is a sample output for the show command, **show sip-ua calls**. The lines that are added to the show command output as part of the Unified SRST 12.6 enhancement are the local crypto key and the remote crypto key:

```
SIP UAC CALL INFO
Number of SIP User Agent Client(UAC) calls: 0
SIP UAS CALL INFO
Call 1
SIP Call ID : 007278df-12e00376-6ed02377-6ffbaca9@8.55.0.195
State of the call : STATE_ACTIVE (7)
Substate of the call : SUBSTATE_NONE (0)
Calling Number : 1001
Called Number : 6901%23
Called URI : sip:6901%23@8.39.25.11;user=phone
Bit Flags : 0x10C0401C 0x10000100 0x4
CC Call ID : 196
```

```
Local UUID : 61488a9100105000a000007278df12e0
Remote UUID : c4b7f9475629538096ef61699b96746f
Source IP Address (Sig ): 8.39.25.11
Destn SIP Req Addr:Port : [8.55.0.195]:52704
Destn SIP Resp Addr:Port: [8.55.0.195]:52704
Destination Name : 8.55.0.195
Number of Media Streams : 1
Number of Active Streams: 1
RTP Fork Object : 0x0
Media Mode : flow-through
Media Stream 1
State of the stream : STREAM_ACTIVE
Stream Call ID : 196
Stream Type : voice+dtmf (1)
Stream Media Addr Type : 1
Negotiated Codec : g711ulaw (160 bytes)
Codec Payload Type : 0
Negotiated Dtmf-relay : rtp-nte
Dtmf-relay Payload Type : 101
QoS ID : -1
Local QoS Strength : BestEffort
Negotiated QoS Strength : BestEffort
Negotiated QoS Direction : None
Local QoS Status : None
Media Source IP Addr:Port: [8.39.25.11]:8080
Media Dest IP Addr:Port : [8.55.0.195]:23022
Local Crypto Suite : AEAD_AES_256_GCM
Remote Crypto Suite : AEAD_AES_256_GCM (
AEAD_AES_256_GCM
AEAD_AES_128_GCM
AES_CM_128_HMAC_SHA1_80
AES_CM_128_HMAC_SHA1_32 )
Local Crypto Key : 3taqc13ClF6BBpvd65WTMPrad/i0uyQ6iNouh+jYHxbf48d4TFmsOGyh4Vs=
Remote Crypto Key : 2/TNTV+Rc1Nh/wbGj0MGwIsLrJ4l+N2jKWGczolEnf7sgsA0Q9AEIz0a4eg=
Mid-Call Re-Assocation Count: 0
SRTP-RTP Re-Assocation DSP Query Count: 0
```

The following is a sample output for the show command, **show ephone offhook** . The lines that are added to the show command output as part of the Unified SRST 12.6 enhancement are local key and remote key.

```
ephone-1[0] Mac:549A.EBB5.8000 TCP socket:[1] activeLine:1 whisperLine:0 REGISTERED in
SCCP
ver 21/17 max_streams=1 + Authentication + Encryption with TLS connection
mediaActive:1 whisper_mediaActive:0 startMedia:1 offhook:1 ringing:0 reset:0 reset_sent:0
paging 0 debug:0 caps:8
IP:8.44.22.63 * 17872 SCCP Gateway (AN) keepalive 28 max_line 1 available_line 1
port 0/0/0
button 1: cw:1 ccw:(0 0)
dn 1 number 6901 CM Fallback CH1 CONNECTED CH2 IDLE
Preferred Codec: g711ulaw
Lpcor Type: none Active Secure Call on DN 1 chan 1 :6901 8.44.22.63 18116
to 8.39.25.11 8066 via 8.39.0.1
G711Ulaw64k 160 bytes no vad
SRTP cipher: AES_CM_128_HMAC_SHA1_32
local key: 0OPV0yxvcnRLPMzHfmYbwgHfdxcuS1uPbp5j/Tjk
remote key: e8DQl3Kvk7LjZlipaCoMg9TMreBmiPsFmNiVHwIA
Tx Pkts 0 bytes 0 Rx Pkts 0 bytes 0 Lost 0
Jitter 0 Latency 0 callingDn -1 calledDn -1
```

Toll Fraud Prevention for SIP Line Side on Unified SRST

Unified SRST Release 12.6 enhances the existing Toll Fraud Prevention feature by enforcing security on the SIP line side of Unified SRST. The feature enhancement secures the Unified SRST system against potential toll fraud exploitation by unauthorized users from the SIP line side.

Note Unified SRST 8.1 to 12.5 Releases restricts toll fraud prevention only to securing calls over the SIP trunk. For more information about Toll Fraud Prevention over a SIP trunk, see Configuring a Trusted IP Address List for Toll-Fraud Prevention.

Some of the key features of Toll Fraud Prevention on Unified SRST for secure calls over SIP lines are:

- Authenticates all the SIP line messages that are triggered from the endpoints to Unified SRST.
- If the IP address of the endpoint is not part of the IP address trusted list, the call is rejected by Unified SRST.
- Unified SRST authenticates both IPv4 an IPv6 addresses as part of the toll fraud prevention mechanism.

Prerequisites for Configuring Toll Fraud Prevention for SIP Line Side

- Unified SRST 12.6 or a later version.
- Cisco IOS XE Gibraltar Release 16.11.1a or later.

Configuration Recommendations for Toll Fraud Prevention on Unified SRST

Unified SRST 12.6 enforces security and toll fraud prevention for SIP line side on Unified SRST. The **ip address trusted authentication** configuration blocks unauthorized calls from the line side. Hence, the toll fraud prevention feature secures Unified SRST 12.6 and later from unauthorized users on the line side.

The IP addresses of SRST endpoints are available before registration with Unified SRST, as they are configured (under **voice register pool**) for fallback from Unified CM. Hence, it is not mandatory that the endpoints are registered to Unified SRST for configuring toll fraud prevention.

The IP trust list for Unified SRST is populated based on the IP address information available under **voice register pool** configuration mode. You can find the IP address of the SIP endpoints on Unified SRST under the following commands in voice register pool configuration mode:

- **id ip** (For example, **id ip***192.168.0.0*)
- **id network** (For example, **id network***192.168.25.0***mask***255.255.255.0*)

Sometimes, IP addresses of endpoints are not available to Unified SRST before registration. Consider a scenario where **id device-id** is the CLI command configured under voice register pool configuration mode to define the device name. Then, the IP address of the device or endpoint is available to Unified SRST only during registration.

The following are the configurations of Toll Fraud Prevention in Unified SRST, 12.6:

- The CLI command **ip address trusted authentication** is enabled by default in Unified SRST. The command **ip address trusted authentication** ensures that security is enabled on the Unified SRST system.
- You can manually configure your Unified SRST endpoints as trusted by entering the IP address or subnet of the trusted phone under the**iptrust-list** configuration mode, as follows:

```
Router#config t
Router(config)#voice service voip
Router(conf-voi-serv)#ip address trusted list
Router(cfg-iptrust-list)#ipv4 192.168.10.0 /16
OR
Router(cfg-iptrust-list)#ipv4 192.168.12.0 255.255.255.0
```

- You can verify the manually added IP address of the Unified SRST endpoint, as follows:

```
Router#show running-config | section voice service voip
voice service voip
ip address trusted list
ipv4 192.168.10.1
ipv4 192.168.10.2 255.255.0.0
ipv4 192.168.10.3 255.255.0.0
ipv4 192.168.10.4 255.255.255.0
```

- The CLI command **ip address trusted list** under **voice service voip** configuration mode supports manual configuration of trusted IP addresses.
- The CLI command **show ip address trusted check** provides information on whether a particular IP address is trusted or not.
- The CLI command **silent-discard untrustedsip** in configuration mode silently discards SIP requests from untrusted sources. This command is enabled by default on Unified SRST.
- The **show ip address trusted list** CLI command displays a list of trusted IP addresses. The trusted IP addresses are displayed under the following lists:
- Dial Peer (only applicable for trunk side): Provides details on the IP address of the trunk that is configured under the dial-peer configuration mode.
- Configured IP Address Trusted List: Provides details on the manually configured IP addresses that are trusted.
- Dynamic IP Address Trusted List: Provides details on the IP address of all the phones that are configured for fallback from Unified CM. This list is introduced in Unified CME 12.6 Release.
- Server Group: Provides details on the IP address of the phones that are configured under server-groups configuration mode.

```
Router>enable
Router#show ip address trusted list
IP Address Trusted Authentication
Administration State: UP
Operation State: UP
IP Address Trusted Call Block Cause: call-reject (21)
VoIP Dial-peer IPv4 and IPv6 Session Targets:
Peer Tag Oper State Session Target
-------- ---------- --------------
4        UP         ipv4:10.65.125.155
Configured IP Address Trusted List:
ipv4 192.168.20.1
```

```
ipv4 192.168.20.2 255.255.0.0
ipv4 192.168.20.3 255.255.0.0
ipv4 192.168.20.4 255.255.255.0
Dynamic IP Address Trusted List:
IP Address                                    Subnet Mask     Count Reason
--------------------------------------------- --------------- ----- ----------------
ipv4:8.55.0.0                                 255.255.0.0         1 Pool Configured
ipv4:192.168.0.1                              255.255.0.0         2 Pool Configured
ipv6:2001:420:54FF:13::312:0                  119                 1 Pool Configured
ipv4:8.55.22.15                                                   1 Phone Registered
```

Note The column Count in Dynamic IP Address Trusted List displays the number of directory numbers (DNs) sharing the same IP address. For example, ipv4 192.168.0.1 with count 2 represents two DNs sharing the IP address 192.168.0.1.

Note The output of **show ip address trusted list** command displays the entry in column **Type** as ‘Phone Registered’ if **id device-id** is configured.

Upgrade Considerations

When you upgrade to Unified SRST 12.6 version, you need not perform extra configurations for supporting toll fraud prevention. All the endpoints that are manually configured or auto-registered on Unified SRST are added to the Unified SRST IP Address Trust List. You can view the list of trusted IP addresses under the output of the CLI command **show ip address trusted list**.

Configure Toll Fraud Prevention

Configure IP Address Trusted Authentication for Incoming VoIP Calls

Before you begin

- Unified SRST 8.1 or a later version for secure trunk calls.
- Unified SRST 12.6 or a later version for secure line and trunk calls.
- The CLI command **silent-discard untrusted** needs to be configured for the feature to work

Restrictions

For an incoming VoIP call, IP trusted authentication must be invoked when the IP address trusted authentication is in “UP” operational state.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice service voip**

4. **ip address trusted authenticate**
5. **ip-address trusted call-block cause**
6. **end**
7. **show ip address trusted list**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**voice service voip** **Example:** `Router(config)# voice service voip`	Enters voice service voip configuration mode.
Step 4	**ip address trusted authenticate** **Example:** `Router(conf-voi-serv)# ip address trusted authenticate`	Enables IP address authentication on incoming H.323 or SIP trunk calls for toll fraud prevention support. IP address trusted list authenticate is enabled by default. Use the **no ip address trusted list authenticate** command to disable the IP address trusted list authentication.
Step 5	**ip-address trusted call-block cause** **Example:** `Router(conf-voi-serv)#ip address trusted call-block cause call-reject`	Issues a cause-code when the incoming call is rejected to the IP address trusted authentication. This command is enabled by default. **Note** If the IP address trusted authentication fails, a call-reject (21) cause-code is issued to disconnect the incoming VoIP call.
Step 6	**end** **Example:** `Router()# end`	Returns to privileged EXEC mode.

Step 7	**show ip address trusted list**
	Example:
	`Router# #show ip address trusted list`
	`IP Address Trusted Authentication`
	`Administration State: UP`
	`Operation State: UP`
	`IP Address Trusted Call Block Cause:`
	`call-reject (21)`
		Verifies a list of valid IP addresses.

Example

```
Router>enable
Router#show ip address trusted list
IP Address Trusted Authentication
Administration State: UP
Operation State: UP
IP Address Trusted Call Block Cause: call-reject (21)
VoIP Dial-peer IPv4 and IPv6 Session Targets:
Peer Tag Oper State Session Target
-------- ---------- -------------
Configured IP Address Trusted List:
ipv4 192.168.20.1
ipv4 192.168.20.2 255.255.0.0
ipv4 192.168.20.3 255.255.0.0
ipv4 192.168.20.4 255.255.255.0
Dynamic IP Address Trusted List:
IP Address                                   Subnet Mask     Count Type
-------------------------------------------- --------------- ----- ----------------
ipv4:8.55.0.0                                255.255.0.0         1 Pool Configured
ipv4:192.168.0.1                             255.255.0.0         1 Pool Configured
```

Add Valid IP Addresses For Incoming VoIP Calls

Before you begin

Cisco Unified CME 8.1 or a later version.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice service voip**
4. **ip address trusted list**
5. **ipv4 ipv4 address network mask** { <*ipv4 address*>[<*network mask*>] }
6. **end**
7. **show ip address trusted list**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**voice service voip** **Example:** `Router(config)# voice service voip`	Enters voice service voip configuration mode.

Step 4	**ip address trusted list**
	Example:
	`Router(conf-voi-serv)# ip address trusted list`
	`Router(cfg-iptrust-list)#`
		Enters ip address trusted list mode and allows to manually add additional valid IP addresses.

Step 5	**ipv4 ipv4 address network mask** { <*ipv4 address*>[<*network mask*>] }
	Example:
	`Router(cfg-iptrust-list)#ipv4 172.19.245.1`
	`Router(cfg-iptrust-list)#ipv4 172.19.243.1`
		Allows you to add up to 100 IPv4 addresses in ip address trusted list. Duplicate IP addresses are not allowed in the ip address trusted list.
		• *network mask* — allows to define a subnet IP address.

Step 6	**end** **Example:** `Router(config-register-pool)# end`	Returns to privileged EXEC mode.
Step 7	**show ip address trusted list** **Example:** `Router# show shared-line`	Displays a list of valid IP addresses for incoming H.323 or SIP trunk calls.

Example

The following example shows three IP addresses configured as trusted IP addresses:

```
Router#show ip address trusted list
IP Address Trusted Authentication
Administration State: UP
Operation State: UP
IP Address Trusted Call Block Cause: call-reject (21)
Configured IP Address Trusted List:
ipv4 192.168.20.1
ipv4 192.168.20.2 255.255.0.0
ipv4 192.168.20.3 255.255.0.0
ipv4 192.168.20.4 255.255.255.0
```

Troubleshooting Tips for Toll Fraud Prevention

For troubleshooting toll fraud mechanism supported on Unified SRST, you can enable the CLI commands **debug voip iptrust debug** and **debug voip iptrust detail**, as follows:

```
Router#debug voip iptrust
voip iptrust debugging is on
Router#debug voip iptrust detail
voip iptrust detail debugging is on
```

VRF Support for Unified SRST

Virtual Routing and Forwarding (VRF) for Unified SRST divides a physical router into multiple logical routers. Each of these logical routers has its own set of interfaces and routing and forwarding tables. VRF support allows you to bind the Unified SRST feature to a specific VRF. Previously with the Cisco 4000 Series Integrated Services Routers, Unified SRST was always associated with the global or default routing instance.

From Unified SRST Release 12.8 (Cisco IOS XE 17.2.1r), support is introduced for VRF functionality on Cisco 4000 Series Integrated Services Router. Before Unified SRST Release 12.8 (Cisco IOS XE 17.2.1r), support for VRF was available only on Cisco Integrated Services Router Generation 2 platform.

From Unified SRST Release 12.8, the following support is available for VRF:

- VRF for line side on Cisco 4000 Series Integrated Services Routers– Introduced in Unified SRST 12.8
- VRF support for Unified SRST 12.8 and later releases is compatible with SIP trunks that are configured to use a VRF. However, you can configure different VRFs for the trunk and Unified SRST.

Information About VRF Support

Typically, service providers use a VRF between Provider Edge (PE) and Customer Edge (CE) routers to provide VPN support for customers. VRF is also used to segment data and voice traffic for improved traffic management. VRF can be configured on an interface to process incoming packets according to the assigned VRF.

By configuring VRF-awareness on voice gateways, you can specify a VRF for the voice traffic that is generated from within the gateway. Voice VRF is added to the VoIP service provider interface (SPI) of the gateway to send and receive signaling and media packets in the configured VRF. The SPI can send and receive signaling and media packets only in the configured VRF.

Note We recommend that you configure **voice vrf** for Unified SRST. For more information, see Design Recommendations for VRF.

Design Recommendations for VRF

- SIP endpoints supported by Unified SRST, including Cisco IP Phone 7800 Series, Cisco IP Phone 8800 Series, and Cisco Jabber support VRF for Unified SRST.
- VRF support is offered for both secure and nonsecure deployments of Unified SRST.
- Configuring SRST to use a VRF is compatible with both SIP and TDM trunk configurations.
- If Global Bind and **voice vrf** are configured on the Unified SRST, then preference is given to the Global Bind.
- We recommend that
    - For SRST line side, configure VRF using **voice vrf** command.

  - For SIP trunk side, configure VRF using **bind** command configured under **voice class tenant** configuration mode and attach the tenant to the required SIP trunk dial-peer.

- VRF Preference Order—The following is the binding preference order for call processing on the trunk side and line side for SRST:

Preference Order	Bind	Configuration
1	Dial-peer Bind	**bind** command is configured under **dial-peer** configuration mode **Note** This configuration is only for trunk side.
2	Tenant Bind	**bind** command is configured under **voice class tenant** configuration mode **Note** This configuration is only for trunk side.
3	Global Bind	**bind** command is configured under **sip** in **voice service voip** configuration mode. **Note** This configuration is both for trunk side and Unified SRST line side.
4	Voice VRF	**voice vrf** command configuration **Note** This configuration is both for trunk side and Unified SRST line side.

Configuration Examples for VRF

The following is a sample configuration for **voice vrf** in Unified SRST line side:

```
vrf definition vrf1
rd 100:101
!
address-family ipv4
exit-address-family

voice vrf vrf1
interface GigabitEthernet0/0/0
    vrf forwarding vrf1
    ip address 8.44.22.77 255.255.0.0
ip route vrf vrf1 8.0.0.0 255.0.0.0 8.44.0.1
```

The following is a sample configuration of Global bind (**voice service voip**). In this case, both Unified SRST line side and SIP trunks without an explicit binding use the same VRF configuration.

```
voice service voip
 no ip address trusted authenticate
 media statistics
 media bulk-stats
 media disable-detailed-stats
 allow-connections sip to sip
 no supplementary-service sip moved-temporarily
 no supplementary-service sip refer
 supplementary-service media-renegotiate
```

```
 fax protocol t38 version 0 ls-redundancy 0 hs-redundancy 0 fallback none
 sip
  bind all source-interface GigabitEthernet 0/0/0
  session transport tcp
  min-se 90
  session refresh
  registrar server expires max 120 min 60
!
```

Configure Virtual Routing and Forwarding (VRF) for Unified SRST

Before you begin

- Unified SRST 12.8 or a later version.
- For design recommendations, see Design Recommendations for VRF.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vrf definition vrf-name**
4. **rd route-distinguisher**
5. **address-family ipv4**
6. **exit-address-family**
7. **voice vrf vrf-name**
8. **interface interface-name**
9. **vrf forwarding customer-vrf-name**
10. **ip address <ip address> <network mask>**
11. **ip route vrf vrf-name <ip address> <networkmask> <ip address>**
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**vrf definition vrf-name** **Example:** `Router(config)# vrf definition vrf1`	Creates a VRF with the specified name. In the example, VRF name is vrf1. **Note** Space is not allowed in VRF name.

	Command or Action	Purpose
Step 4	**rd route-distinguisher** **Example:** `Router (config)# rd 100:101`	Creates a VRF table by specifying a route distinguisher.Enter either an AS number and an arbitrary number (xxx:y) or an IP address and arbitrary number (A.B.C.D:y).
Step 5	**address-family ipv4** **Example:** `Router(config)# address-family ipv4`	Configures IPv4 or IPv6 address-family sessions for a VRF configuration in Unified SRST.
Step 6	**exit-address-family** **Example:** `Router(config)# exit-address-family`	Leaves address-family configuration mode without removing the address family configuration.
Step 7	**voice vrf vrf-name** **Example:** `Router(config)# voice vrf vrf1`	Configures a voice VRF in global configuration mode.
Step 8	**interface interface-name** **Example:** `Router(config)# interface GigabitEthernet0/0/0`	Enters the interface configuration mode.
Step 9	**vrf forwarding customer-vrf-name** **Example:** `Router(config-if)# vrf forwarding vrf1`	Associates the customer VRF instance with the tunnel. Packets exiting the tunnel are forwarded to this VRF (inner IP packet routing).
Step 10	**ip address <ip address> <network mask>** **Example:** `Router(config-if)# ip address 8.44.22.77 255.255.0.0`	IP address is assigned to the interface.
Step 11	**ip route vrf vrf-name <ip address> <networkmask> <ip address>** **Example:** `Router(config-if)# ip route vrf vrf1 8.0.0.0 255.0.0.0 8.44.0.1`	(Optional) Generates IP routing information associated with a VRF. **Note** Required only if you need to add static routes.
Step 12	**end** **Example:** `Router(config-if)# end`	Exits to privileged EXEC mode.

IPv6 Support for Unified SRST SIP IP Phones

Internet Protocol version 6 (IPv6) is the latest version of the Internet Protocol (IP). IPv6 uses packets to exchange data, voice, and video traffic over digital networks. Also, IPv6 increases the number of network address bits from 32 bits in IPv4 to 128 bits. From Unified SRST Release 12.0 onwards, Unified SRST supports IPv6 protocols for SIP IP phones.

IPv6 support in Unified SRST allows the network to behave transparently in a dual-stack (IPv4 and IPv6) environment and provides additional IP address space to SIP IP phones that are connected to the network. If you do not have a dual-stack configuration, configure the CLI command **call service stop** under **voice service voip** configuration mode before changing to dual-stack mode. For an example of switching to dual-stack mode, see Examples for Configuring IPv6 Pools for SIP IP Phones.

The Cisco IP Phone 7800 Series and 8800 Series are supported on IPv6 for Unified SRST.

For more information on configuring SIP IP phones for IPv6 source address, see Configure IPv6 Pools for SIP IP Phones.

For an example of configuring IPv6 Support on Unified SRST, see Examples for Configuring IPv6 Pools for SIP IP Phones.

For more details about IPv6 deployment, see IPv6 Deployment Guide for Cisco Collaboration Systems Release 12.0.

Feature Support for IPv6 in Unified SRST SIP IP Phones

The following basic features are supported for a IPv6 WAN down scenario:

- Basic SIP Line (IPv4 or IPv6) to SIP Line calls (IPv4 or IPv6) when Unified SRST is in dual-stack **no anat** mode.

The following supplementary services are supported as part of IPv6 in Unified SRST IP Phones:

- Hold/Resume
- Call Forward
- Call Transfer
- Three-way Conference (with BIB conferencing only)
- Line to T1/E1 Trunk and Trunk to Line with Supplementary Service Features
- Fax to and from PSTN (IPv4 ATA to ISDN T1/E1) for both T.38 Fax Relay and Fax Passthrough

Restrictions

The following are the known restrictions for IPv6 support on Unified SRST:

- SIP Trunks are not supported on Unified SRST for IPv6 deployment. PSTN calls are supported only through T1/E1 trunks.
- SCCP IP Phones are not supported in a deployment of IPv6 for Unified SRST.
- SIP Phones can be either in IPv4 only or IPv6 only mode (**no anat**).
- Trancoding and Transrating are not supported.
- H.323 trunks are not supported.
- Secure SIP lines or trunks are not supported.

- IPv6 on Unified SRST is not supported on the Cisco IOS platform. The support is restricted to Cisco IOS XE platform with Cisco IOS Release 16.6.1 or later versions.

Configure IPv6 Pools for SIP IP Phones

Before you begin

- Unified SRST 12.0 or a later version.
- IPv6 option only appears if protocol mode is dual-stack configured under sip-ua configuration mode or IPv6.
- Cisco Unified SRST License must be configured for the gateway to function as a Unified SRST gateway to support IPv6 functionality. For more information on licenses, see Licensing.
- Cisco Unified Communications Manager (Unified Communications Manager) is provisioned with the IPv6 address of Unified SRST. For information on configuration of Unified SRST on Unified Communications Manager, see the section Survivable Remote Site Telephony Configuration in Cisco Unified Communications Manager Administration Guide.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 unicast-routing**
4. **voice service voip**
5. **sip**
6. **no ant**
7. **call service stop**
8. **exit**
9. **exit**
10. **sip-ua**
11. **protocol mode {ipv4 | ipv6 | dual-stack [preference {ipv4 | ipv6}]}**
12. **exit**
13. **voice service {voip}**
14. **sip**
15. **no call service stop**
16. **exit**
17. **voice register global**
18. **default mode**
19. **max-dn** *max-directory-numbers*
20. **max-pool** *max-voice-register-pools*
21. **exit**
22. **voice register pool***pool-tag*
23. **id { network** *address* **mask** *mask* | **ip address mask** *mask* | **mac** *address* }
24. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 unicast-routing** **Example:** `Router(config)# ipv6 unicast-routing`	Enables the forwarding of IPv6 unicast datagrams.
Step 4	**voice service voip** **Example:** `Router (config)# voice service voip`	Enters voice-service configuration mode to specify a voice encapsulation type. **voip** —Specifies Voice over IP (VoIP) parameters.
Step 5	**sip** **Example:** `Router(config-voi-serv)# sip`	Enters SIP configuration mode.
Step 6	**no ant** **Example:** `Router(config-serv-sip)# no anat`	Disables Alternative Network Address Types (ANAT) on a SIP trunk.
Step 7	**call service stop** **Example:** `Router(config-serv-sip)# call service stop`	Shuts down SIP call service.
Step 8	**exit** **Example:** `Router(config-serv-sip)# exit`	Exits SIP configuration mode.
Step 9	**exit** **Example:** `Router(config-voi-serv)# exit`	Exits voice service voip configuration mode.
Step 10	**sip-ua** **Example:** `Router(config)# sip-ua`	Enters SIP user-agent configuration mode.
Step 11	**protocol mode {ipv4 | ipv6 | dual-stack [preference {ipv4 | ipv6}]}** **Example:**	Allows phones to interact with phones on IPv6 voice gateways. You can configure phones for IPv4 addresses, IPv6 addresses, or for a dual-stack mode.

	Command or Action	Purpose
	`Router(config-sip-ua)# protocol mode dual-stack preference ipv6`	• ipv4—Allows you to set the protocol mode as an IPv4 address. • ipv6—Allows you to set the protocol mode as an IPv6 address. • dual-stack—Allows you to set the protocol mode for both IPv4 and IPv6 addresses. • preference—Allows you to choose a preferred IP address family if protocol mode is dual-stack.
Step 12	**exit** **Example:** `Router(config-sip-ua)# exit`	Exits SIP configuration mode.
Step 13	**voice service {voip}** **Example:** `Router (config)# voice service voip`	Enters voice-service configuration mode to specify a voice encapsulation type. **voip**—Specifies Voice over IP (VoIP) parameters.
Step 14	**sip** **Example:** `Router(config-voi-serv)# sip`	Enters SIP configuration mode.
Step 15	**no call service stop** **Example:** `Router(config-serv-sip)# call service stop`	Activates SIP call service.
Step 16	**exit** **Example:** `Router(config-serv-sip)# exit`	Exits SIP configuration mode.
Step 17	**voice register global** **Example:** `Router(config)# voice register global`	Enters voice register global configuration mode to set parameters for all supported SIP phones in Unified SRST.
Step 18	**default mode** **Example:** `Router(config-register-global)# default mode`	Enables mode for provisioning SIP phones in Unified SRST. The default mode is Unified SRST itself.
Step 19	**max-dn** *max-directory-numbers* **Example:** `Router(config-register-global)# max-dn 50`	Limits number of directory numbers to be supported by this router. Maximum number is platform and version-specific. Type ? for value.
Step 20	**max-pool** *max-voice-register-pools* **Example:**	Sets maximum number of SIP phones to be supported by the Unified SRST router.

	Command or Action	Purpose
	`Router(config-register-global)# max-pool 40`	
Step 21	**exit** **Example:** `Router(config-register-global)# exit`	Exits voice register global configuration mode.
Step 22	**voice register pool***pool-tag* **Example:** `Router(config)# voice register pool 1`	Enters voice register pool configuration mode to set phone-specific parameters for a SIP phone.

Step 23	**id { network** *address* **mask** *mask* | **ip address mask** *mask* | **mac** *address* }
	Example:
	`Router(config-register-pool)# id network 2001:420:54FF:13::901:0/117`
	`Router(config-register-pool)# id network 10.64.88.0 mask 255.255.255.0`
		Explicitly identifies a locally available individual SIP phone to support a degree of authentication.

Step 24	**end** **Example:** `Router(config)# end`	Exits to privileged EXEC mode.

Configure Unified SRST on Cisco 4000 Series Integrated Services Platform

For Unified SRST Release 10.5 and later, Unified SRST is supported on Cisco 4000 Series Integrated Services Routers. A Unified SRST system supports SIP phones with standard-based RFC 3261 feature support locally and across SIP WAN networks. With Cisco Unified SIP SRST, SIP phones can place calls across SIP networks with similar features, as SCCP phones do. For example, most SCCP phone features such as caller ID, speed dial, and redial are supported on SIP networks, that give users the opportunity to choose SCCP or SIP.

Before you begin

- Cisco IOS XE Denali 16.3.1 or a later release.
- Cisco IP Phones 7800 Series or 8800 Series.
- An appropriate feature license to support Unified SIP SRST on the router.
- You need to configure **voice register global** in your router.
- You need to ensure that your router is in **default mode** (for Unified SRST).

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **voice service voip**
4. **allow-connections** *from-type to to-type*
5. **no supplementary-service sip moved-temporarily**
6. **no supplementary-service sip refer**
7. **supplementary-service media-renegotiate**
8. **sip**
9. **registrar server [expires[max** *sec* **][min** *sec* **]]**
10. **exit**
11. **exit**
12. **voice register global**
13. **default mode**
14. **max-dn** *max-directory-numbers*
15. **max-pool** *max-voice-register-pools*
16. **exit**
17. **voice register pool** *pool-tag*
18. **id [network** *address* **mask** *mask* | **ip** *address* **mask** *mask*]
19. **dtmf-relay rtp-nte**
20. **no vad**
21. **codec** *codec-type [bytes]*
22. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**voice service voip** **Example:** `Router(config)# voice service voip`	Enters voice-service configuration mode and specifies voice-over-IP encapsulation. Enters voice register global configuration mode to set global parameters for all supported Cisco SIP IP phones in a Cisco Unified SIP SRST environment.
Step 4	**allow-connections** *from-type to to-type* **Example:** `Router(config-voi-serv)# allow-connections sip to sip`	Allows connections between specific types of endpoints in a VoIP network.
Step 5	**no supplementary-service sip moved-temporarily** **Example:**	Disables supplementary service for call forwarding.

	Command or Action	Purpose
	`Router(config-voi-serv)# no supplementary-service sip moved-temporarily`	
Step 6	**no supplementary-service sip refer** **Example:** `Router(config-voi-serv)# no supplementary-service sip refer`	Prevents the router from forwarding a REFER message to the destination for call transfers.
Step 7	**supplementary-service media-renegotiate** **Example:** `Router(config-voi-serv)# supplementary-service media-renegotiate`	Enables mid-call media renegotiation for supplementary services.
Step 8	**sip** **Example:** `Router(config-voi-serv)# sip`	Enters SIP configuration mode. Required only if you perform the following step for enabling the SIP registrar function.
Step 9	**registrar server [expires[max** *sec* **][min** *sec* **]]** **Example:** `Router(config-serv-sip)# registrar server expires max 120 min 60`	Enables SIP registrar functionality in Unified SRST. • **expires** : (Optional) Sets the active time for an incoming registration. • **max** *sec* : (Optional) Maximum time for a registration to expire, in seconds. Range: 600 to 86400. Default: 3600. Recommended value: 600. • **min** *sec* : (Optional) Minimum expiration time for a registration, in seconds. The range is from 60 to 3600. The default is 60.
Step 10	**exit** **Example:** `Router(config-serv-sip)# exit`	Exits SIP configuration mode.
Step 11	**exit** **Example:** `Router(config-voi-serv)# exit`	Exits voice-service configuration mode.
Step 12	**voice register global** **Example:** `Router(config)# voice register global`	Enters voice register global configuration mode to set parameters for all supported SIP phones in Unified SRST.
Step 13	**default mode** **Example:** `Router(config-register-global)# default mode`	Enables mode for provisioning SIP phones in Unified SRST. The default mode is Unified SRST itself.
Step 14	**max-dn** *max-directory-numbers* **Example:**	Limits number of directory numbers to be supported by this router.

	Command or Action	Purpose
	`Router(config-register-global)# max-dn 50`	Maximum number is platform and version-specific. Type ? for value.
Step 15	**max-pool** *max-voice-register-pools* **Example:** `Router(config-register-global)# max-pool 40`	Sets maximum number of SIP phones to be supported by the Unified SRST router. Maximum number is platform and version-specific. Type ? for value.
Step 16	**exit** **Example:** `Router(config-register-global)# exit`	Exits voice register global configuration mode.
Step 17	**voice register pool** *pool-tag* **Example:** `Router(config)# voice register pool 1`	Enters voice register pool configuration mode to set phone-specific parameters for a SIP phone.
Step 18	**id [network** *address* **mask** *mask* | **ip** *address* **mask** *mask*] **Example:** `Router(config)# voice service voip`	Enters voice service voip configuration mode.
Step 19	**dtmf-relay rtp-nte** **Example:** `Router(config-register-pool)# dtmf-relay rtp-nte`	Forwards DTMF tones by using Real-Time Transport Protocol (RTP) with the Named Telephone Event (NTE) payload type and enables DTMF relay using the RFC 2833 standard method.
Step 20	**no vad** **Example:** `Router(config-register-pool)# no vad`	Disables voice activity detection (VAD) on the VoIP dial peer. VAD is enabled by default. Because there is no comfort noise during periods of silence, the call may seem to be disconnected. You may prefer to set no vad on the SIP phone pool.
Step 21	**codec** *codec-type [bytes]* **Example:** `Router(config-register-pool)# codec g729r8`	Specifies the codec supported by a single SIP phone or a VoIP dial peer in a Cisco Unified SIP SRST environment. The *codec - type* argument specifies the preferred codec and can be one of the following: • **g711alaw**: G.711 a–law 64,000 bps. • **g711ulaw**: G.711 mu–law 64,000 bps. • **g729r8**: G.729 8000 bps (default). The *bytes* argument is optional and specifies the number of bytes in the voice payload of each frame
Step 22	**end** **Example:**	Returns to privileged EXEC mode.

	Command or Action	Purpose
	`Router()# end`	

Configure Voice Hunt Groups on Unified SRST

To redirect calls for a specific number (pilot number) to a defined group of directory numbers on Cisco Unified SCCP and SIP IP phones, perform the following steps.

Voice Hunt Group on Unified SRST is supported for Parallel, Sequential, Peer, and Longest-idle hunt groups. Only the basic call features such as Call, Hold or Resume are supported for Unified SRST on Cisco 4000 Series Integrated Services Routers. For support of advanced features such as Auto Logout, Members Logout, and supplementary call features, you need to configure Unified E-SRST. For more information on Voice Hunt Group support on Unified E-SRST, see Unified E-SRST with Support for Voice Hunt Group.

For a list of restrictions of Unified SRST on Cisco 4000 Series Integrated Services Routers, see Restrictions of Unified SRST on Cisco 4000 Series Integrated Services Routers, page 33

Before you begin

- Cisco IOS XE Denali 16.3.1 or later versions.
- Shared Lines are not supported on Unified SRST.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice hunt-group** *hunt-tag* [**longest-idle** | **parallel** | **peer** | **sequential**]
4. **pilot** *number* [**secondary** *number*]
5. **list** *number*
6. **final** *number*
7. **preference** *preference-order* [**secondary***secondary-order*]
8. **hops** *number*
9. **timeout** *seconds*
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**voice hunt-group** *hunt-tag* [**longest-idle** | **parallel** | **peer** | **sequential**] **Example:** `Router(config)# voice hunt-group 1 longest-idle`	Enters voice hunt-group configuration mode to define a hunt group. • *hung-tag* —Unique sequence number of the hunt group to be configured. Range is 1 to100. • **longest idle** —Hunt group in which calls go to the directory number that has been idle for the longest time. • **parallel** —Hunt group in which calls simultaneously ring multiple phones. • **peer** —Hunt group in which the first directory number is selected round-robin from the list. • **sequential** —Hunt group in which directory numbers ring in the order in which they are listed, left to right. • To change the hunt-group type, remove the existing hunt group first by using the **no** form of the command; then, recreate the group.
Step 4	**pilot** *number* [**secondary** *number*] **Example:** `Router(config-voice-hunt-group)# pilot number 8100`	Defines the phone number that callers dial to reach a voice hunt group. • *number*—String of up to 16 characters that represents an E.164 phone number. • Number string may contain alphabetic characters when the number is to be dialed only by the Unified SRST router, as with an intercom number, and not from phone keypads. • **secondary** *number*—(Optional) Keyword and argument combination defines the number that follows as an additional pilot number for the voice hunt group. • Secondary numbers can contain wildcards. A wildcard is a period (.), which matches any entered digit.
Step 5	**list** *number* **Example:** `Router(config-voice-hunt-group)# list 8000, 8010, 8020, 8030`	Creates a list of extensions that are members of a voice hunt group. To remove a list from a router configuration, use the **no** form of this command. • *number*—List of extensions to be added as members to the voice hunt group. Separate the extensions with commas. • Add or delete all extensions in a hunt-group list at one time. You cannot add or delete a single number in an existing list.

	Command or Action	Purpose
		• There must be from 2 to 10 extensions in the hunt-group list, and each number must be a primary or secondary number. • Any number in the list cannot be a pilot number of a parallel hunt group.
Step 6	**final** *number* **Example:** `Router(config-voice-hunt-group)# final 8888`	Defines the last extension in a voice hunt group. • If a final number in one hunt group is configured as a pilot number of another hunt group, the pilot number of the first hunt group cannot be configured as a final number in any other hunt group.
Step 7	**preference** *preference-order* [**secondary***secondary-order*] **Example:** `Router(config-voice-hunt-group)# preference 6`	Sets the preference order for the directory number associated with a voice hunt-group pilot number. **Note** We recommend that the parallel hunt-group pilot number be unique in the system. Parallel hunt groups may not work if there are more than one partial or exact dial-peer match. For example, if the pilot number is “8000” and there is another dial peer that matches “8…”. If multiple matches cannot be avoided, give parallel hunt groups the highest priority to run by assigning a lower preference to the other dial peers. Note that 8 is the lowest preference value. By default, dial peers created by parallel hunt groups have a preference of 0. • *preference-order*—Range is 0 to 8, where 0 is the highest preference and 8 is the lowest preference. Default is 0. • **secondary** *secondary-order*—(Optional) Keyword and argument combination is used to set the preference order for the secondary pilot number. Range is 1 to 8, where 0 is the highest preference and 8 is the lowest preference. Default is 7.
Step 8	**hops** *number* **Example:** `Router(config-voice-hunt-group)# hops 2`	For configuring a peer or longest-idle voice hunt group only. Defines the number of times that a call can hop to the next number in a peer or longest-idle voice hunt group before the call proceeds to the final number. • *number*—Number of hops. Range is 2 to 10, and the value must be less than or equal to the number of extensions specified by the **list** command.

	Command or Action	Purpose
		• Default is the same number as there are destinations defined under the **list** command.
Step 9	**timeout** *seconds* **Example:** `Router(config-voice-hunt-group)# timeout 100`	Defines the number of seconds after which a call that is not answered is redirected to the next directory number in a voice hunt-group list. Default is 180 seconds.
Step 10	**end** **Example:** `Router(config-voice-hunt-group)# end`	Exits to privileged EXEC mode.

Configure Feature Support on Unified SIP SRST

This section provides configuration information for some of the features supported on Unified SIP SRST.

Configure SIP-to-SIP Call Forwarding

SIP-to-SIP call forwarding (call routing) is available. Call forwarding is provided either by the phone or by using a back-to-back user agent (B2BUA), which allows call forwarding on any dial peer. Calls into a SIP device may be forwarded to other SIP or SCCP devices (including Cisco Unity, third-party voice-mail systems, or an auto attendant or IVR system such as IPCC and IPCC Express). In addition, SCCP IP phones may be forwarded to SIP phones.

Cisco Unity or other voice messaging systems connected by a SIP trunk or SIP user agent are able to pass a message-waiting indicator (MWI) when a message is left. The SIP phone then displays the MWI when indicated by the voice messaging system.

Note SIP-to-H.323 call forwarding is not supported.

To configure SIP-to-SIP call forwarding, you must first allow connections between specific types of endpoints in a Cisco IP-to-IP gateway. The **allow-connections** command grants this capability. Once the SIP-to-SIP connections are allowed, you can configure call forwarding under an individual SIP phone pool. Any of the following commands can be used to configure call forwarding, according to your needs:

Under the **voice register pool**

- **call-forward b2bua all** *directory-number*
- **call-forward b2bua busy** *directory-number*
- **call-forward b2bua mailbox** *directory-number*
- **call-forward b2bua noan** *directory-number* [**timeout** *seconds*]

In a typical Cisco Unified SIP SRST setup, the **call-forward b2bua mailbox** command is not used; however, it is likely to be used in a Cisco Unified SIP Communications Manager Express (CME) environment. Detailed procedures for configuring the **call-forward b2bua mailbox** command are found in the Cisco Unified Communications Manager (CallManager) documentation on Cisco.com.

The command **call-forward b2bua all** needs to point towards the trunk.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice register pool** *tag*
4. **call-forward b2bua all** *directory- number*
5. **call-forward b2bua busy** *directory- number*
6. **call-forward b2bua mailbox** *directory- number*
7. **call-forward b2bua noan** *directory- number* **timeout** *seconds*
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**voice register pool** *tag* **Example:** `Router(config)# voice register pool 15`	Enters voice register pool configuration mode. • Use this command to control which phone registrations are accepted or rejected by a Cisco Unified SIP SRST device.
Step 4	**call-forward b2bua all** *directory- number* **Example:** `Router(config-register-pool)# call-forward b2bua all 5005`	Enables call forwarding for a SIP back-to-back user agent (B2BUA) so that all incoming calls are forwarded to another non-SIP station extension (that is, SIP trunk, H.323 trunk, SCCP device or analog/digital trunk). • *directory-number* : Phone number to which calls are forwarded. Represents a fully qualified E.164 number. Maximum length of the phone number is 32.
Step 5	**call-forward b2bua busy** *directory- number* **Example:** `Router(config-register-pool)# call-forward b2bua busy 5006`	Enables call forwarding for a SIP B2BUA so that incoming calls to a busy extension are forwarded to another extension. • *directory-number* : Phone number to which calls are forwarded. Represents a fully qualified E.164 number. Maximum length of the phone number is 32.

	Command or Action	Purpose

Step 6	**call-forward b2bua mailbox** *directory- number*
	Example:
	`Example:`
	`Router(config-register-pool)# call-forward b2bua mailbox 5007`
		Controls the specific voice-mail box selected in a voice-mail system at the end of a call forwarding exchange.
		• *directory-number* : Phone number to which calls are forwarded when the forwarded destination is busy or does not answer. Represents a fully qualified E.164 number. Maximum length of the phone number is 32.

Step 7	**call-forward b2bua noan** *directory- number* **timeout** *seconds* **Example:** `Router(config-register-pool)# call-forward b2bua noan 5010 timeout 10`	Enables call forwarding for a SIP B2BUA so that incoming calls to an extension that does not answer after a configured amount of time are forwarded to another extension. This command is used if a phone is registered with a Cisco Unified SIP SRST router, but the phone is not reachable because there is no IP connectivity (there is no response to Invite requests). • *directory-number* : Phone number to which calls are forwarded. Represents a fully qualified E.164 number. Maximum length of the phone number is 32. • **timeout** *seconds*: Duration, in seconds, that a call can ring with no answer before the call is forwarded to another extension. Range is 3 to 60000. The default value is 20.
Step 8	**end** **Example:** `Router(config-register-pool)# end`	Returns to privileged EXEC mode.

Configure Call Blocking Based on Time of Day, Day of Week, or Date

This section applies to both SCCP and SIP SRST. Call blocking prevents the unauthorized use of phones and is implemented by matching a pattern of up to 32 digits during a specified time of day, day of week, or date. Cisco Unified SIP SRST provides SIP endpoints the same time-based call blocking mechanism that is currently provided for SCCP phones. The call blocking feature supports all incoming calls, including incoming SIP and analog FXS calls.

Note Pin-based exemptions and the "Login" toll-bar override are not supported in Cisco Unified SIP SRST.

The commands used for SIP phone call blocking are the same commands that are used for SCCP phones on your Cisco Unified SRST system. The Cisco SRST session application accesses the current after-hours configuration under call-manager-fallback mode and applies it to calls originated by Cisco SIP phones that are registered to the Cisco SRST router. The commands used in call-manager-fallback mode that set block criteria (time/date/block pattern) are the following:

- **after-hours block pattern** *pattern-tag pattern* [**7-24**]

- **after-hours day** *day start-time stop-time*
- **after-hours date** *month date start-time stop-time*

When a user attempts to place a call to digits that match a pattern that has been specified for call blocking during a time period that has been defined for call blocking, the call is immediately terminated and the caller hears a fast busy.

In SRST (call-manager-fallback configuration mode), there is no phone- or pin-based exemption to after-hours call blocking. However, in Cisco Unified SIP SRST (voice register pool mode), individual IP phones can be exempted from all call blocking using the **after-hours exempt** command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **call-manager-fallback**
4. **after-hours block pattern** *tag pattern* [**7-24**]
5. **after-hours day** *day start-time stop-time*
6. **after-hours date** *month date start-time stop-time*
7. **exit**
8. **voice register pool** *tag*
9. **after-hour exempt**
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**call-manager-fallback** **Example:** `Router(config)# call-manager-fallback`	Enters call-manager-fallback configuration mode.
Step 4	**after-hours block pattern** *tag pattern* [**7-24**] **Example:** `Router(config-cm-fallback)# after-hours block pattern 1 91900`	Defines a pattern of outgoing digits to be blocked. Up to 32 patterns can be defined, using individual commands. • If the **7-24** keyword is specified, the pattern is always blocked, 7 days a week, 24 hours a day. • If the **7-24** keyword is not specified, the pattern is blocked during the days and dates that are defined using the **after-hours day** and **after-hours date** commands.

	Command or Action	Purpose
Step 5	**after-hours day** *day start-time stop-time* **Example:** `Router(config-cm-fallback)# after-hours day mon 19:00 07:00`	Defines a recurring time period based on the day of the week during which calls are blocked to outgoing dial patterns that are defined using the **after-hours block pattern** command. • *day* : Day of the week abbreviation. The following are valid day abbreviations: **sun**, **mon**, **tue**, **wed**,**thu**, **fri**, **sat**. • *start-time stop-time* : Beginning and ending times for call blocking, in an HH:MM format using a 24-hour clock. If the stop time is a smaller value than the start time, the stop time occurs on the day following the start time. For example, "mon 19:00 07:00" means "from Monday at 7 p.m. until Tuesday at 7 a.m." The value 24:00 is not valid. If 00:00 is entered as a stop time, it is changed to 23:59. If 00:00 is entered for both start time and stop time, calls are blocked for the entire 24-hour period on the specified date.
Step 6	**after-hours date** *month date start-time stop-time* **Example:** `Router(config-cm-fallback)# after-hours date jan 1 00:00 00:00`	Defines a recurring time period based on month and date during which calls are blocked to outgoing dial patterns that are defined using the **after-hours block pattern** command. • *month* : Month abbreviation. The following are valid month abbreviations: **jan**, **feb**, **mar**, **apr**, **may**,**jun**, **jul**, **aug**, **sep**, **oct**, **nov**,**dec**. • *date* : Date of the month. Range is from 1 to 31. • *start-time stop-time* : Beginning and ending times for call blocking, in an HH:MM format using a 24-hour clock. The stop time must be larger than the start time. The value 24:00 is not valid. If 00:00 is entered as a stop time, it is changed to 23:59. If 00:00 is entered for both start time and stop time, calls are blocked for the entire 24-hour period on the specified date.
Step 7	**exit** **Example:** `Router(config-cm-fallback)# exit`	Exits call-manager-fallback configuration mode.
Step 8	**voice register pool** *tag* **Example:** `Router(config)# voice register pool 12`	Enters voice register pool configuration mode. • Use this command to control which registrations are accepted or rejected by a Cisco Unified SIP SRST device.

	Command or Action	Purpose
Step 9	**after-hour exempt** **Example:** `Router(config-register-pool)# after-hour exempt`	Specifies that for a particular voice register pool, none of its outgoing calls are blocked although call blocking is enabled.
Step 10	**end** **Example:** `Router(config-register-pool)# end`	Returns to privileged EXEC mode.

Verification

To verify the feature's configuration, enter one of the following commands:

- **show voice register dial-peer** : Displays all the dial peers created dynamically by phones that have registered. This command also displays configurations for after hours blocking and call forwarding.
- **show voice register pool** : Displays information about a specific pool.
- **debug ccsip message** : Debugs basic B2BUA calls.

For more information about these commands, see Cisco Unified SRST and Cisco Unified SIP SRST Command Reference (All Versions).

SIP Call Hold and Resume

Unified SRST supports the ability for SIP phones to place calls on hold and to resume from calls placed on hold. This also includes support for a consultative hold where A calls B, B places A on hold, B calls C, and B disconnects from C and then resumes with A. Support for call hold is signaled by SIP phones using "re-INVITE c=0.0.0.0" and also by the receive-only mechanism.

No configuration is necessary.

Configure Music On Hold for Unified SRST

Unified SRST supports the ability for SIP phones to play music for calls placed on hold. The following is the recommended configuration for Music On Hold (MOH) on a SIP Phone that falls back to Unified SRST.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no telephony-service**
4. **call-manager-fallback**
5. **moh enable-g711 "bootflash:** *filename"*
6. **moh enable-g729 "bootflash:** *filename"*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**no telephony-service** **Example:** `Router# no telephony-service`	Removes all the configurations for IP phones configured under the telephony-service configuration mode.
Step 4	**call-manager-fallback** **Example:** `Router(config)# call-manager-fallback`	Enters call-manager-fallback configuration mode.
Step 5	**moh enable-g711 "bootflash:** *filename"* **Example:** `Router(config-cm-fallback)# moh enable-g711 "bootflash:music-on-hold.au"`	Generates an audio stream from a router flash file that supports G.711 codec for Music On Hold (MOH) in Unified. SRST.
Step 6	**moh enable-g729 "bootflash:** *filename"* **Example:** `Router(config-cm-fallback)# moh g729 "flash:SampleAudioSource.g729.wav"`	Generates an audio stream from a router flash file that supports G.729 codec for MOH in Unified SRST.
Step 7	**end** **Example:** `Router(config-cm-fallback)# end`	Returns to privileged EXEC mode.

Enabling KPML for SIP Phones

Perform the following steps to enable KPML digit collection on a SIP phone.

Restrictions

A dial plan assigned to a phone has priority over KPML.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice register pool** *pool-tag*
4. **digit collect kpml**
5. **end**

6. **show voice register dial-peers**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**voice register pool** *pool-tag* **Example:** `Router(config)# voice register pool 4`	Enters voice register pool configuration mode to set phone-specific parameters for a SIP phone. • *pool-tag*: Unique sequence number of the SIP phone to be configured. Range is version and platform-dependent; type **?** to display range. You can modify the upper limit for this argument with the **max-pool** command.
Step 4	**digit collect kpml** **Example:** `Router(config-register-pool)# digit collect kpml`	Enables KPML digit collection for the SIP phone. **Note** This command is enabled by default for supported phones in Cisco Unified CME and Cisco Unified SRST.
Step 5	**end** **Example:** `Router(config-register-pool)# end`	Exits to privileged EXEC mode.
Step 6	**show voice register dial-peers** **Example:** `Router# show voice register dial-peer`	Displays details of all dynamically created VoIP dial peers associated with the Cisco Unified CME SIP register including the defined digit collection method.

Disabling SIP Supplementary Services for Call Forward and Call Transfer

Perform the following steps to disable REFER messages for call transfers and redirect responses for call forwarding from being sent to the destination by Unified SRST. You can disable these supplementary features if the destination gateway does not support them.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice service voip** OR **dial-peer voice** *tag* **voip**
4. **no supplementary-service sip** {**moved-temporarily** |**refer**}

5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

Step 3	**voice service voip** OR **dial-peer voice** *tag* **voip**
	Example:
	`Router(config)# voice service voip`
	`or`
	`Router(config)# dial-peer voice 99 voip`
		Enters voice-service configuration mode to set global parameters for VoIP features.
		or
		Enters dial peer configuration mode to set parameters for a specific dial peer.

Step 4	**no supplementary-service sip** {**moved-temporarily** |**refer**}
	Example:
	`Router(conf-voi-serv)# no supplementary-service sip refer`
	`or`
	`Router(config-dial-peer)# no supplementary-service sip refer`
		Disables SIP call forwarding or call transfer supplementary services globally or for a dial peer.
		• **moved-temporarily**: SIP redirect response for call forwarding.
		• **refer**: SIP REFER message for call transfers.
		• Sending REFER and redirect messages to the destination is the default behavior.
		Note This command is supported for calls between SIP phones and calls between SCCP phones. It is not supported for a mixture of SCCP and SIP endpoints.

Step 5	**end**
	Example:
	`Router(config-voi-serv)# end`
	`OR`
	`Router(config-dial-peer)# end`
		Exits to privileged EXEC mode.

Configuring idle Prompt Status for SIP Phones

Perform the following steps to customize the message that displays on SIP phones after the phones failover to Cisco Unified SRST.

Note: You do not need to create new configuration files with the **create profile** command and restart the phones after changing the idle status message in Cisco Unified SRST. Modifying the status message takes effect immediately in Cisco Unified SRST.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice register global**
4. **system message** *string*
5. **end**
6. **show voice register global**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**voice register global** **Example:** `Router(config)# voice register global`	Enters voice register global configuration mode to set global parameters for all supported SIP phones in a Cisco Unified CME environment.
Step 4	**system message** *string* **Example:** `Router(config-register-global)# system message fallback active`	Defines a status message that displays on SIP phones registered to Cisco Unified SRST. • *string*: Up to 32 alphanumeric characters. Default is “CM Fallback Service Operating.”
Step 5	**end** **Example:** `Router(config-register-global)# end`	Exits to privileged EXEC mode.
Step 6	**show voice register global** **Example:** `Router# show voice register global`	Displays all global configuration parameters associated with SIP phones.

Examples

The following are sample configurations for supporting SIP SRST on Cisco 4000 Series Integrated Services Router.

Example for Configuring Unified SIP SRST on Cisco 4000 Series Integrated Services Routers

The following example shows how to configure Unified SIP SRST on Cisco 4000 Series Integrated Services Routers.

```
!
voice service voip
allow-connections sip to sip
no supplementary-service sip moved-temporarily
no supplementary-service sip refer
supplementary-service media-renegotiate
sip
registrar server expires max 120 min 60
!
!
voice register global
default mode
max-dn 40
max-pool 40
!
voice register pool 1
id network 8.55.0.0 mask 255.255.0.0
dtmf-relay rtp-nte
codec g711ulaw
no vad
!
!
```

Example for Configuring Voice Hunt Groups in Unified SIP SRST

The following example shows how to configure longest-idle hunt group 20 with pilot number 4701, final number 5000, and 6 numbers in the list. After a call is redirected six times (makes 6 hops), it is redirected to the final number 5000.

```
Router(config)# voice hunt-group 20 longest-idle
Router(config-voice-hunt-group)# pilot 4701
Router(config-voice-hunt-group)# list 4001, 4002, 4023, 4028, 4045, 4062
Router(config-voice-hunt-group)# final 5000
Router(config-voice-hunt-group)# hops 6
Router(config-voice-hunt-group)# timeout 20
Router(config-voice-hunt-group)# exit
```

Examples for Configuring IPv6 Pools for SIP IP Phones

The following example provides configuration of IPv6 pools for SIP IP Phones:

```
ipv6 unicast-routing
voice service voip
sip
```

```
no anat
call service stop
exit
exit
sip-ua
protocol mode dual-stack
exit
voice service voip
sip
no call service stop
exit
voice register global
default mode
max-dn 50
max-pool 40
exit
voice register pool 1
id network 2001:420:54FF:13::901:0/117
end
```

The following example provides interface configuration for IPv6 supported on Unified SRST:

```
configure terminal
interface GigabitEthernet0/0/1
ip address 10.64.86.229 255.255.255.0
negotiation auto
ipv6 address 2001:420:54FF:13::312:82/119
ipv6 enable
```

The following example provides IP route configuration for IPv6 supported on Unified SRST:

```
ipv6 route 2001:420:54FF:13::312:0/119 2001:420:54FF:13::312:1
ipv6 route 2001:420:54FF:13::901:0/119 2001:420:54FF:13::312:1
```

The following example displays output when SIP call service is shut down with the call service stop CLI command:

```
Router# show sip service
SIP service is shut
under 'voice service voip', 'sip' submode
```

The following example displays output when SIP call service is active with the no call service stop CLI command:

```
Router# show sip-ua service
SIP Service is up
under 'voice service voip', 'sip' submode
```

Example for Configuring Call Blocking Based on Time of Day, Day of Week, or Date

The following example defines several patterns of digits for which outgoing calls are blocked. Patterns 1 and 2, which block calls to external numbers that begin with 1 and 011, are blocked on Monday through Friday before 7 a.m. and after 7 p.m. Pattern 3 blocks calls to 900 numbers 7 days a week, 24 hours a day.

```
call-manager-fallback
after-hours block pattern 1 91
after-hours block pattern 2 9011
after-hours block pattern 3 91900 7-24
after-hours day mon 19:00 07:00
after-hours day tue 19:00 07:00
after-hours day wed 19:00 07:00
```

```
after-hours day thu 19:00 07:00
after-hours day fri 19:00 07:00
```

The following example exempts a Cisco SIP phone pool from the configured blocking criteria:

```
voice register pool 1
after-hour exempt
```

Example for Configuring Music On Hold for Unified SIP SRST

The following example shows how to configure Music On Hold (MOH) for Unified SIP SRST on Cisco 4000 Series Integrated Services Routers.

```
enable
configure terminal
no telephony-service
call-manager-fallback
moh enable-g711 "flash:music-on-hold.au"
moh g729 "flash:SampleAudioSource.g729.wav"
```

Example for Configuring SIP-to-SIP Call Forwarding on Unified SRST

The following is a sample configuration for SIP-to-SIP Call Forwarding on Unified SRST.

```
enable
configure terminal
voice register pool 15
call-forward b2bua busy 5006
call-forward b2bua mailbox 5007
call-forward b2bua noan 5010 timeout 8
```

Example for Configuring idle Prompt Status for SIP Phones

The following is a sample configuration for idle prompt status for SIP phones on Unified SRST.

```
enable
configure terminal
voice register global
system message fallback active
end
show voice register global
```

Example for Disabling SIP Supplementary Services for Call Forward and Call Transfer

The following is a sample configuration for disabling SIP supplementary services for call forward and call transfer on Unified SRST.

```
enable
configure terminal
voice service voip
no supplementary-service sip {moved-temporarily | refer}
end
```