Survivability for Hosted and Cloud Services

The Survivability for Hosted and Cloud Services on the CUBE is used to:

- Monitor the WAN status periodically from the CUBE.

- Route calls and handle line-side subscriptions locally when the WAN link is down.

- Synchronize the registrations with the server when the WAN link is up.

Information About Survivability for Hosted and Cloud Services

Advantages of Using CUBE Survivability Feature

The survivability feature on CUBE addresses the following issues by providing local fallback or registration synchronization:

1. When a WAN link or registrar server comes up, it waits until each SIP phone sends the REGISTER message to the server, so that outside phones can reach that phone.

2. If the phone register timer setting is too large, the outside phone waits that much time to reach that phone, after a link flap.

3. If the phone register timer setting is too small, it floods the WAN link.

4. When the WAN link or registrar server is down, you cannot make any local calls.

Local Fallback

- CUBE does not need to configure credentials, as the phones trigger registration. Although CUBE receives REGISTER messages for each phone every 5 minutes; for example, it throttles and sends REGISTER messages every 1 hour to the registrar server, avoiding high WAN bandwidth usage. This addresses the issues 1, 2, and 3.

- In normal operation when the WAN link or registrar server is up, the phone's primary server URL is the registrar server (E2E) registration.

- "OPTIONS ping" is used to monitor the registrar server link status. When the detected link is down, CUBE replies with a 500 message and when the phone receives this message, it sends the REGISTER message to CUBE, which is the secondary server (P2P registration). CUBE replies with a 200 OK message to P2P registration when the link is down. The dial-peer keeps the dynamic registrar session target and the local call does not fail. This addresses issue 4.

Registration Synchronization

- If you configure the phones to send REGISTER messages every 1 hour (to help alleviate the WAN link), the CUBE uses the credentials that were configured to respond to registrar server authentication challenge. This addresses issue 3.

- When the WAN link or registration server is down (detected by OPTIONS ping), the CUBE keeps the registration database of the SIP phones that were previously registered successfully, and it does not send REGISTER messages out; CUBE replies with a 200 OK message and dial-peer keeps the dynamic registrar session target. The local call does not fail, addressing issue 4.

- When the registrar link is up after a link flap, the CUBE sends REGISTER message for each phone that was earlier successfully registered to the registrar server. This is throttled to avoid bulk REGISTER messages flooding WAN link and the registrar. This addresses issues 1 and 2.

Registration Through Alias Mapping

The following illustration shows how a phone (with alias mapping) registers to the service provider through CUBE.

Figure 1: SIP Phone Registration
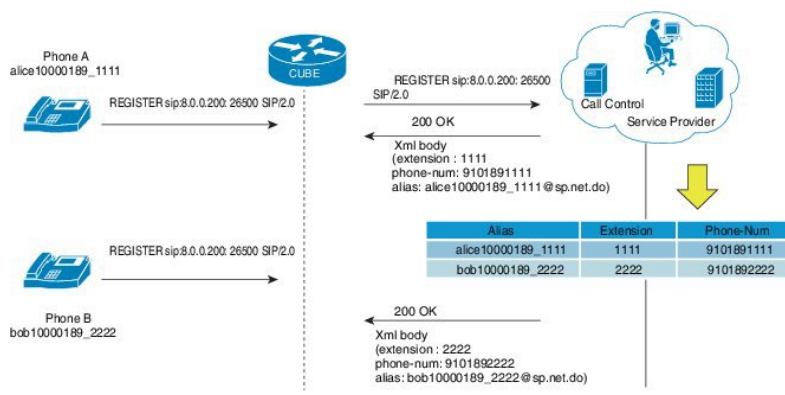
The addresses-of-record (AOR) sent in the REGISTER is an alias which is mapped to an extension and (or) phone number by the service provider. The service provider returns the mapping details in the 200 OK response sent to the REGISTER. CUBE has the ability to cache the alias mapping details in its call routing database. When a call is made from the phone, the Request-URI of the INVITE contains the dialed number (short extension or phone number).

If WAN is up, CUBE always routes the INVITE sent from the phone to the service provider without looking up at the alias mapping cache.

If WAN or the service provider is down, that is, in survivability mode, CUBE routes the INVITE locally by looking up at the alias mapping cache.

Alias Mapping—Supported Methods

1. When the service provider returns the mapping details in the 200 OK message of the REGISTER in the following predefined format:

| Alias	Extension	Phone
alice10000189_1111	1111	10000189

2. The short extension or phone number is embedded in the AOR of the REGISTER. For example, AOR is **alice10000189_1111** and the short extension is **1111**.

An inbound sip profile can be applied to the REGISTER which extracts the extension part from the AOR and adds an X-CISCO-EXTENSION header.

CUBE when WAN is UP

The following illustration provides an example as to how a typical phone makes a call to another local phone registered in the same server when WAN or the registrar server is up in a typical hosted deployment. The circled numbers in the image indicate the numerical order in which the sequence occurs.

Figure 2: WAN Link is UP - CUBE Deployment
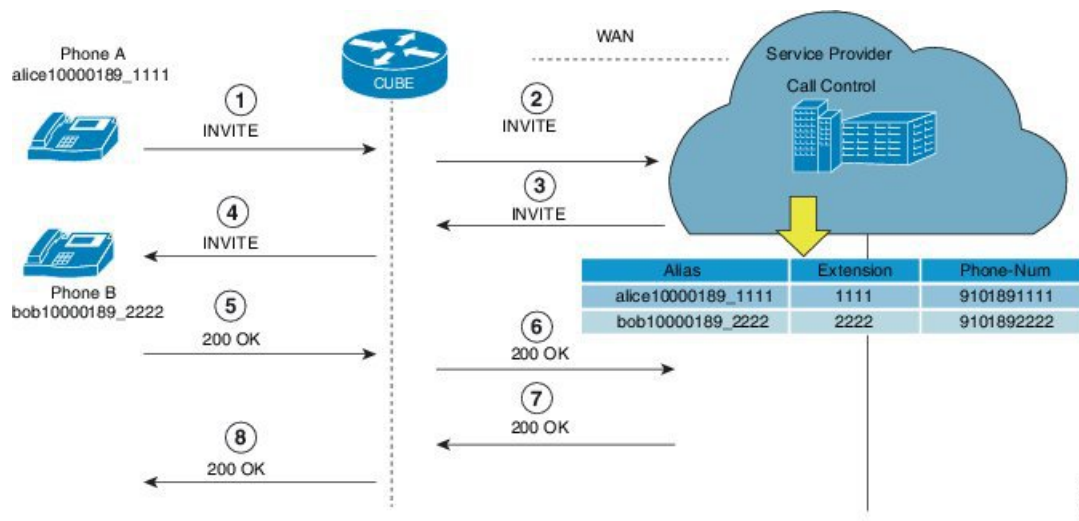
The call flow scenario is as follows: Phone A initiates a call to the Phone B registered to the same server.

1. Phone A sends an initial INVITE request to Phone B to participate in a call session through CUBE.

2. CUBE sends this INVITE to the service provider.

3. The service provider in turn sends the INVITE to CUBE. Since the WAN link is up, the service provider maps details of the user from the register server and provides details of the user, for example, alias of the user, short extension number, and phone number.

4. CUBE sends INVITE with all the above mentioned information to Phone B.

5. Phone B sends a 200 OK response to CUBE for the received INVITE.

6. CUBE sends a 200 OK answer to the service provider.

7. The service provider responds to CUBE with a 200 OK answer.

8. A final 200 OK response is sent to Phone A by CUBE and the call is established between Phone A and Phone B.

Example: Normal Mode (WAN is Up in P2P Mode)

```
CUBE# show sip-ua registration passthrough status

CallId    DirectoryNum   peer    mode   In-Exp     reg-I   Out-Exp   survival
=======   =========      =====   ====   ========   ====    ====      =========
21        NCPhone1006    1       p2p    135 /144   1       144       normal
===========================================================================
```

Example: Normal Mode (WAN is Up in E2E Mode)

```
CUBE# show sip-ua registration passthrough status

CallId    DirectoryNum   peer    mode    In-Exp      reg-I   Out-Exp   survival
=======   ===========    ====    =====   =====       =====   =======   ========
14574     NCPhone1006    301     e2e     117 /120    --      120       normal
===========================================================================
```

CUBE Survivability When WAN Is Down

In survivability mode, CUBE provides end-to end telephony services when access to the centralized servers is interrupted because of a WAN outage or other factors, like the server being down.

The following illustration shows how a call is established between two endpoints when WAN link is down during survivability by directly dialing into an extension.

Figure 3: CUBE Survivability When WAN Is Down
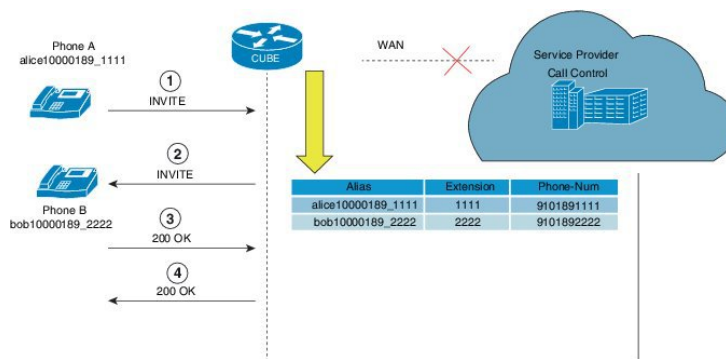
Earlier, when WAN was down, User A could only contact User B using either the alias or the user-id of User B, and not using their extensions or phone numbers.

Now, in the event the WAN link or registration server is down, when a local call is made, INVITE is sent to CUBE. CUBE maps the details of the user like the extension number and phone-number stored during registration. Local phones can now be reached on their short extensions or phone numbers by similar phones that are subscribed to the server through the same CUBE.

It is possible to register multiple contacts for a single AOR; however, if multiple contacts are registered for a single subscriber, the CUBE uses only the topmost registered contact to deliver the call to that subscriber. For this reason, multiple contacts are not supported.

A few phone models, such as, Cisco IP Phone 7800 Series with Multiplatform Firmware and Cisco IP Phone 8800 Series with Multiplatform Firmware, sends register request to primary registrar only and do not send secondary REGISTER request to the secondary registrar (CUBE) in E2E mode when primary registrar could not be reached. In such scenarios, phone service goes down after it receives 500 response from CUBE for REGISTER request toward primary registrar.

To avoid phones getting into such error condition, CUBE checks for the response from the primary registrar side. When CUBE receives request timeout on WAN side or responses other than 200, 4XX, and 3XX from primary registrar, survivability will be enabled.

To enable survivability on such phones, refer Configuring Survivability for Phones Sending Single Register Request, on page 9.

Survivability Support for Public Switched Telephone Network Access When WAN Is Down

If WAN link going down or registrar service unavailable, you can access the phones in the Public Switched Telephone Network (PSTN) through FXO or PRI cards that are configured on Cisco Unified Border Element.

✎

Note   Survivability support for Public Switched Telephone Network (PSTN) access is supported only for CUBE running on Cisco 4000 Series Integrated Services Router.

Figure 4: Survivability Support for PSTN Access When WAN Is Down
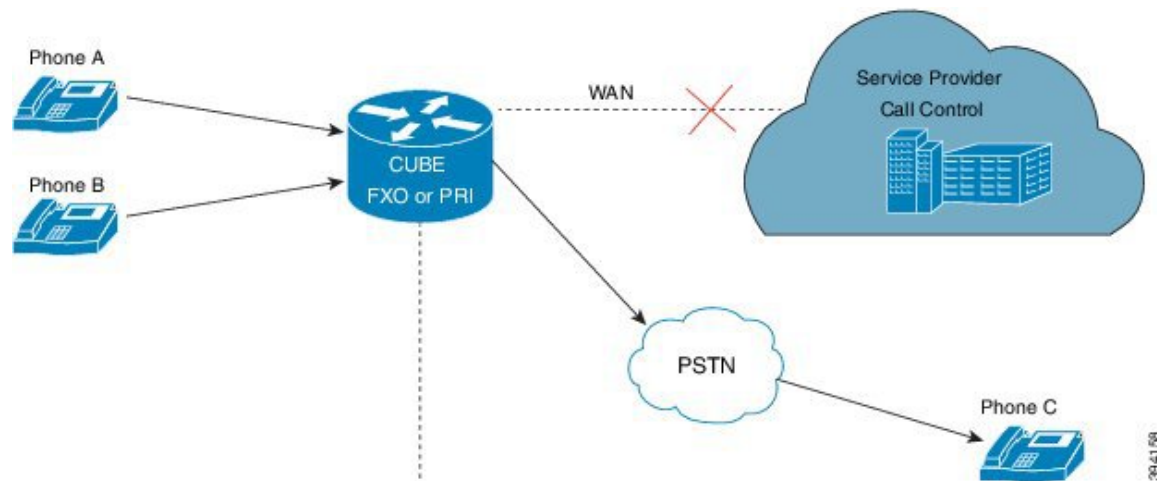
Example: Survivability Mode in P2P (regsync mode) when WAN is Down

```
CUBE# show sip-ua registration passthrough status

CallId   DirectoryNum    peer    mode     In-Exp     reg-I  Out-Exp   survival
=======  =========  ============  =====  =========  =====  =======  ========
38       NCPhone1008      1       p2p    3595 /3600   1      3600     regsync
==============================================================================
```

Example: Survivability Mode in E2E (local fallback mode) when WAN is Down

```
CUBE# show sip-ua registration passthrough status

CallId   DirectoryNum   peer    mode   In-Exp   reg-I   Out-Exp   survival
======   ============   ======  ====   ======   =====   =======   ========
70       NCPhone1006    1       e2e    35 /70   --      0         locfall
==========================================================================


CallId   DirectoryNum   peer    mode   In-Exp   reg-I   Out-Exp   survival
======   ============   ======  ====   ======   =====   =======   ========
513      NCPhone1008    1       e2e    40 /70   --      0         locfall
```

How to Configure Survivability for Hosted and Cloud Services

Configuring Local Fallback or Registration Synchronization Globally

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice service voip**
4. **sip**
5. **registration passthrough local-fallback** *tag*
6. **end**

DETAILED STEPS

Procedure

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Device# configure terminal	Enters global configuration mode.
Step 3	**voice service voip** **Example:** Device(config)# voice service voip	Enters voice service VoIP configuration mode.

	Command or Action	**Purpose**
Step 4	**sip** **Example:** Device(conf-voi-serv)# sip	Enters voice service SIP configuration mode.
Step 5	**registration passthrough local-fallback** *tag* **Example:** Device(conf-serv-sip)# registration passthrough local-fallback 10	Configures SIP registration passthrough for local fallback mode; this will locally respond to REGISTER in p2p mode when WAN is down. The *tag* is the WAN link or registrar server dial-peer tag. • To configure the registration sync mode, you can use the **registration passthrough reg-sync** *tag* command. Use the **static** keyword to set the phone URL to p2p registration.
Step 6	**end** **Example:** Device(conf-serv-sip)# end	Returns to privileged EXEC mode.

Configuring Local Fallback or Registration Synchronization at the Tenant Level

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice class tenant** *tag*
4. **registration passthrough local-fallback** *tag*
5. **exit**
6. **dial-peer voice** *tag* **voip**
7. **voice-class sip tenant** *tag*
8. **exit**

DETAILED STEPS

Procedure

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	**Purpose**
Step 2	**configure terminal** **Example:** Device# configure terminal	Enters global configuration mode.
Step 3	**voice class tenant** *tag* **Example:** Device(config)# voice class tenant 1	Enters voice class tenant configuration mode.
Step 4	**registration passthrough local-fallback** *tag* **Example:** Device(config-class)# registration passthrough local-fallback 10	Configures SIP registration passthrough for local fallback mode; this locally responds to REGISTER in p2p mode when WAN is down. The *tag* is the WAN link or registrar server dial-peer tag. • To configure the registration sync mode, you can use the **registration passthrough reg-sync** *tag* command. Use the **static** keyword to set the phone URL to p2p registration.
Step 5	**exit** **Example:** Device(config-class)# exit	Exits tenant configuration mode and returns to global configuration mode.
Step 6	**dial-peer voice** *tag* **voip** **Example:** Device(config)# dial-peer voice 444 voip	Enters dial peer voice configuration mode.
Step 7	**voice-class sip tenant** *tag* **Example:** Device(config-dial-peer)# voice-class sip tenant 1	Associates the dial-peer with the tenant.
Step 8	**exit** **Example:** Device(config-class)# exit	Exits dial-peer configuration mode and returns to global configuration mode.

Configuring Local Fallback or Registration Synchronization on a Dial Peer

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **dial-peer voice** *tag* **voip**
4. **voice-class sip registration passthrough local-fallback** *tag*
5. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Device# configure terminal	Enters global configuration mode.
Step 3	**dial-peer voice** *tag* **voip** **Example:** Device(config)# dial-peer voice 4 voip	Enters dial peer VoIP configuration mode.
Step 4	**voice-class sip registration passthrough local-fallback** *tag* **Example:** Device(config-dial-peer)# voice-class sip registration passthrough local-fallback 10	Configures SIP registration passthrough for local fallback mode; this will locally respond to REGISTER in p2p mode when WAN is down. The *tag* is the WAN link or registrar server dial-peer tag. • To configure the registration sync mode, you can use the **voice-class sip registration passthrough reg-sync** *tag* command.
Step 5	**end** **Example:** Device(conf-serv-sip)# end	Returns to privileged EXEC mode.

Configuring Survivability for Phones Sending Single Register Request

The following configuration enables CUBE to always check for the response from remote side. Request timeout on WAN side or response other than 200, 4XX, and 3XX received by CUBE from SBC enables the survivability.

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **voice service voip**
4. **sip**
5. **survivability single-register**
6. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Device# configure terminal	Enters global configuration mode.
Step 3	**voice service voip** **Example:** Device(config)# voice service voip	Enters voice service VoIP configuration mode.
Step 4	**sip** **Example:** Device(conf-voi-serv)# sip	Enters voice service SIP configuration mode.
Step 5	**survivability single-register** **Example:** Device(conf-serv-sip)# survivability single-register	Enables CUBE to always check for the response from the remote side. Request timeout on WAN side or response other than 200, 4XX, and 3XX received by CUBE from SBC enables the survivability.
Step 6	**end** **Example:** Device(conf-serv-sip)# end	Returns to privileged EXEC mode.

Configuring OPTIONS Ping

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dial-peer voice** *tag* **voip**
4. **voice-class sip options-keepalive up-interval** *value* **down-interval** *value*
5. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**enable**
	
	Example:
	
	  Device> enable
		Enables privileged EXEC mode.
		
		  • Enter your password if prompted.

Step 2	**configure terminal** **Example:** Device# configure terminal	Enters global configuration mode.
Step 3	**dial-peer voice** *tag* **voip** **Example:** Device(config)# dial-peer voice 3 voip	Enters dial peer configuration mode.
Step 4	**voice-class sip options-keepalive up-interval** *value* **down-interval** *value* **Example:** Device(config-dial-peer)# voice-class sip options-keepalive up-interval 120 down-interval 120	Configures OPTIONS keepalive timer interval for DOWN and UP endpoints.
Step 5	**end** **Example:** Device(config-dial-peer)# end	Returns to privileged EXEC mode.

Configuring Registration Timer

Perform the following task to configure the registration timer in the CUBE rather than on all SIP phones.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice service voip**
4. **sip**
5. **registrar server expires max** *value* **min** *value*
6. **end**

DETAILED STEPS

Procedure

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Device# configure terminal	Enters global configuration mode.
Step 3	**voice service voip** **Example:** Device(config)# voice service voip	Enters voice service VoIP configuration mode.
Step 4	**sip** **Example:** Device(conf-voi-serv)# sip	Enters voice service SIP configuration mode.
Step 5	**registrar server expires max** *value* **min** *value* **Example:** Device(conf-serv-sip)# registrar server expires max 300 min 200	Configures the maximum and minimum time (in seconds) for the registration expiry in CUBE. • If the phone sends expiry time as 600 seconds, then the CUBE will reply with 200 OK message and expiry time 300 seconds, and the phone will resend with expiry 300.
Step 6	**end** **Example:** Device(conf-serv-sip)# end	Returns to privileged EXEC mode.

Configuring the REGISTER Message Throttling in CUBE

Perform the following task to throttle the REGISTER message in CUBE.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice service voip**
4. **sip**
5. **registration passthrough rate-limit expires** *value* **local-fallback** *tag*
6. **end**

DETAILED STEPS

Procedure

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Device# configure terminal	Enters global configuration mode.
Step 3	**voice service voip** **Example:** Device(config)# voice service voip	Enters voice service VoIP configuration mode.
Step 4	**sip** **Example:** Device(conf-voi-serv)# sip	Enters voice service SIP configuration mode.
Step 5	**registration passthrough rate-limit expires** *value* **local-fallback** *tag* **Example:** Device(conf-serv-sip)# registration passthrough rate-limit expires 3600 local-fallback 3	Configures the SIP registration passthrough rate-limit expiry value for local-fallback (e2e). Although CUBE receives the REGISTER message every 5 minutes (300 seconds), it will send only one register message every one hour. • Under dial peer configuration mode, you can use the **voice-class sip registration passthrough rate-limit expires** *value* **reg-sync** *dial-peer-tag* command.

	Command or Action	Purpose
Step 6	**end** **Example:** `Device(conf-serv-sip)# end`	Returns to privileged EXEC mode.

Configuring the Class of Restrictions (COR) List

Class of Restrictions (COR) provides the ability to deny certain call attempts based on the incoming and outgoing class of restrictions that are provisioned on the dial peers.

COR specifies which incoming dial peer can use which outgoing dial peer to make a call. You can provision each dial peer with an incoming and an outgoing COR list. The incoming COR list indicates the capability of the dial peer to initiate certain classes of calls. The outgoing COR list indicates the capability that is required for an incoming dial peer to deliver a call through this outgoing dial peer.

Before you begin

You must configure COR Groups. For more information, see Dial Peer Configuration Guide.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dial-peer voice** *tag* **voip**
4. **corlist incoming** *dial-peer*
5. **corlist outgoing** *dial-peer*
6. **description** *string*
7. **destination-pattern** *number*
8. **session protocol sipv2**
9. **session target registrar**
10. **voice-class sip registration passthrough local-fallback** *tag*
11. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** Device# configure terminal	Enters global configuration mode.
Step 3	**dial-peer voice** *tag* **voip** **Example:** Device(config)# dial-peer voice 3 voip	Enters dial peer configuration mode.
Step 4	**corlist incoming** *dial-peer* **Example:** Device(config-dial-peer)# corlist incoming FromPhone	Specifes the COR to be applied on an incoming dial peer (for incoming calls).
Step 5	**corlist outgoing** *dial-peer* **Example:** Device(config-dial-peer)# corlist outgoing FromSP	Specifes the COR to be applied for outgoing dial peer (for outgoing calls).
Step 6	**description** *string* **Example:** Device(config-dial-peer)# description registration	Adds a description to a dial peer.
Step 7	**destination-pattern** *number* **Example:** Device(config-dial-peer)# destination-pattern 1111	Specifies either the prefix or the full E.164 phone number to be used for the dial peer.
Step 8	**session protocol sipv2** **Example:** Device(config-dial-peer)# session protocol sipv2	Specifies the session protocol for SIP calls between local and remote devices using the packet network.
Step 9	**session target registrar** **Example:** Device(config-dial-peer)# session target registrar	Specifies to route the call to the registrar endpoint for SIP dial peers.
Step 10	**voice-class sip registration passthrough local-fallback** *tag* **Example:**	Configures SIP registration passthrough for local fallback mode.

	Command or Action	Purpose
	`Device(config-dial-peer)# voice-class sip registration passthrough local-fallback 5`	
Step 11	**end** **Example:** `Device(config-dial-peer)# end`	Returns to privileged EXEC mode.

Verifying Survivability for Hosted and Cloud Services

The **show** commands can be entered in any order.

SUMMARY STEPS

1. **enable**
2. **show dial-peer voice summary**
3. **show sip-ua registration passthrough status**
4. **show sip-ua register status**
5. **show voip rtp connections**
6. **show call active voice compact**

DETAILED STEPS

Procedure

Step 1   **enable**

Enables privileged EXEC mode.

Example:

`Device> `**`enable`**

Step 2   **show dial-peer voice summary**

Displays the summary information for each voice dial peer.

Example:

`Device# `**`show dial-peer voice summary`**

```
dial-peer hunt 0
          AD                                     PRE PASS              OUT
TAG     TYPE  MIN  OPER PREFIX    DEST-PATTERN    FER THRU SESS-TARGET    STAT PORT    KEEPALIVE
1       voip  up   up             1111...         0   syst registrar
2       voip  up   down           1......         0   syst ipv4:10.104.45.253          busyout
1000    voip  down down           9900...         0   syst ipv4:9.0.0.174:30601
101     voip  down down           1......         0   syst ipv4:10.104.45.31
102     voip  down down           11.....         0   syst ipv4:10.104.45.253
300     voip  down down           .T              0   syst
```

```
400    voip  down down        11110...        0  syst registrar
```

Step 3   **show sip-ua registration passthrough status**

Displays information about the SIP user agent registration passthrough status. In the sample output shown below, the parameter In-Exp shows the remaining expiry time and the survival field parameters can be regsync, locfall, or normal.

Example:

```
Device# show sip-ua registration passthrough status

CallId       Line         peer         mode In-Exp        reg-I Out-Exp survival
============ ============ ============ ==== ============= ===== ======= ========
5300         1111008      1            e2e  1041  /1200  ----- 1200    normal *
5305         1111002      1            e2e  2847  /3000  ----- 3000    normal *
5311         1111020      1            e2e  1070  /1200  ----- 1200    normal *
================================================================================
```

Step 4   **show sip-ua register status**

Displays information about the SIP user agent register status.

Example:

```
Device# show sip-ua register status

Line       peer  expires(sec) reg survival P-Associ-URI
========   ===== ============ === ========
11123      23    59           yes regsync
```

Step 5   **show voip rtp connections**

Displays Real-Time Transport Protocol (RTP) named event packets.

Example:

```
Device# show voip rtp connections

VoIP RTP Port Usage Information:
Max Ports Available: 8091, Ports Reserved: 101, Ports in Use: 2
Port range not configured, Min: 16384, Max: 32767

                                                                      Ports
Ports       Ports
Media-Address Range                                                   Available
Reserved    In-use

Default Address-Range                                                 8091
101         2

VoIP RTP active connections :
No. CallId     dstCallId  LocalRTP RmtRTP LocalIP                     RemoteIP
1    5324       5325       16410    16464  9.40.1.168                  9.40.1.173
2    5325       5324       16412    16528  9.40.1.168                  9.40.1.174
Found 2 active RTP connections
```

Step 6   **show call active voice compact**

Displays the compact version of the call information for voice calls in progress.

Example:

```
Device# show call active voice compact

<callID>  A/O FAX T<sec> Codec        type       Peer Address      IP R<ip>:<udp>
Total call-legs: 2
      5324 ANS     T9    g711ulaw    VOIP        P1111008       9.40.1.173:16464
      5325 ORG     T9    g711ulaw    VOIP        P1111020       9.40.1.174:16528
```

Configuration Examples—Survivability for Hosted and Cloud Services

Example: Configuring Local Fallback Globally

In the following example, local fallback is configured at global level:

```
Device> enable
Device# configure terminal
Device(config)# voice service voip
Device(conf-voi-serv)# sip
Device(conf-serv-sip)# registration passthrough local-fallback 10
Device(config-serv-sip)# end
```

Example: Configuring Local Fallback at the Tenant Level

In the following example, local fallback is configured for tenant 1 and is applied for dial-peer 444:

```
Device>enable
Device#configure terminal
Device(config)#voice class tenant 1
Device(config-class)#registration passthrough local-fallback 10
Device(config-class)#exit
Device(config)#dial-peer voice 444 voip
Device(config-dial-peer)#voice-class sip tenant 1
Device(config-class)# exit
```

Example: Configuring Local Fallback on a Dial Peer

In the following example, local fallback is configured on dial-peer 2.

```
Device> enable
Device# configure terminal
Device(config)# dial-peer voice 2 voip
Device(config-dial-peer)# voice-class sip registration passthrough local-fallback 10
Device(config-dial-peer)# end
```

Example: Configuring Survivability for Phones Sending Single Register Request

In the following example, survivability is configured for phones sending single register request:

```
Device> enable
Device# configure terminal
Device(config)# voice service voip
Device(conf-voi-serv)# sip
Device(conf-serv-sip)# survivability single-register
Device(config-serv-sip)# end
```

Example: Configuring OPTIONS Ping

In the following example, OPTIONS Ping is configured on dial-peer 3:

```
Device> enable
Device# configure terminal
Device(config)# dial-peer voice 3 voip
Device(config-dial-peer)# voice-class sip options-keepalive up-interval 120 down-interval
120
Device(config-dial-peer)# end
```

Example: Configuring the Registration Timer

In the following example, registration timer is configured with a expiration value of minimum 200 and maximum 300 seconds.

```
Device> enable
Device# configure terminal
Device(config)# voice service voip
Device(conf-voi-serv)# sip
Device(conf-serv-sip)# registrar server expires max 300 min 200
Device(conf-serv-sip)# end
```

Example: Configuring REGISTER Message Throttling

In the following example, REGISTER message throttling is configured:

```
Device>enable
Device#configure terminal
Device(config)#voice service voip
Device(conf-voi-serv)#sip
Device(conf-serv-sip)#registration passthrough rate-limit expires 3600 local-fallback 3
Device(conf-serv-sip)#end
```

Example: Configuring the COR List

In the following example, "FromPhone" and "FromSP" COR groups are configured and applied to dial-peer 2:

```
Device>enable
Device# configure terminal
Device(config)#dial-peer cor list FromPhone
Device(config-dp-corlist)#member 911
Device(config-dp-corlist)#member 1800
Device(config)#dial-peer cor list FromSP
Device(config-dp-corlist)#member 911
Device(config-dp-corlist)#member 1800
Device(config-dp-corlist)#exit
Device(config)# dial-peer voice 2 voip
Device(config-dial-peer)# corlist incoming FromPhone
Device(config-dial-peer)# corlist outgoing FromSP
Device(config-dial-peer)# description registration
Device(config-dial-peer)# destination-pattern 1111
Device(config-dial-peer)# session protocol sipv2
Device(config-dial-peer)# session target registrar
Device(config-dial-peer)# voice-class sip registration passthrough local-fallback 5
Device(config-dial-peer)# end
```

Feature Information for Survivability for Hosted and Cloud Services

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

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to https://cfnng.cisco.com/. An account on Cisco.com is not required.

Table 1: Feature Information for Survivability for Hosted and Cloud Services

Feature Name	Releases	Feature Information
Survivability for Hosted and Cloud Services	Cisco IOS XE Fuji 16.9.1	Supports survivability for Hosted and Cloud Services.