Analog FXS port SIP Registration with CUCM
Introduction:

This document helps configure FXS ports as SIP endpoints registered to Cisco Unified Communications Manager (CUCM) in order to support supplementary services on SIP Endpoints.

Prerequisites:
Cisco recommends to have knowledge of these subjects

SIP protocol
Foreign Exchange Station (FXS) ports
Cisco Unified Communications Manager (CUCM)
Cisco Analog Voice Gateway (VG Series)

Requirements:

The FXS Ports for Supplementary Services feature is supported on Cisco VG450 Voice Gateway and Cisco 4461 ISR. The FXS ports for Supplementary Services supports CUCM version 12.5.1 SU1 or later with IOS XE 16.12.1 and above.

Supported Features:
The following supplementary services are supported.

Call Hold
Call Waiting
Call Transfer (unattended/attended)
Call Forward no Answer
Audio Message Waiting Indicator
Call Forwarding Unrestricted
Call Forwarding Busy
Directed Call Park
Directed Call Pickup
Call Pickup Group
Three way Conference
Configure:

To implement supplementary services for Foreign Exchange Station (FXS) ports the call control server (CUCM) should be able to subscribe to the hookflash or onhook events. This requires FXS ports to be registered to CUCM as SIP endpoints. The use of SIP over SCCP facilitates features such as SIP Header modification, endpoint based call routing and enables new features such as directed call retrieval.

Network Diagram:

![Network Diagram](image)

Configuration steps:
This section describes the configuration required for this feature to work:

**Configuring the Device Control Session Application**
DSAPP (Device control Session Application) is the application that drives these Hook Flash features. It can be configured globally or on a dial-peer basis.

**SUMMARY STEPS**
1. enable
2. configure terminal
3. application
4. global
5. service default dsapp
6. param dialpeer *number*
7. param callWaiting *string*
8. param callConference *string*
### 9. param callTransfer string

#### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong>&lt;br&gt;enable&lt;br&gt;Example: <code>Router&gt; enable</code></td>
<td>Enables privileged EXEC mode.&lt;br&gt;Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Step 2</strong>&lt;br&gt;configure terminal&lt;br&gt;Example: <code>Router# configure terminal</code></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Step 3</strong>&lt;br&gt;application global service default dsapp&lt;br&gt;Example:&lt;br&gt;<code>router(config)#application</code>&lt;br&gt;<code>router(config-app)#global</code>&lt;br&gt;<code>router(app-global)#service default dsapp</code></td>
<td><em>(Optional)</em> Enables the new hookflash functionality globally. Device Control Session Application (DSAPP) drives these hookflash features and it must be configured for new hookflash functionality for an application framework module in IOS. DSAPP can be configured globally or on a dial-peer basis.&lt;br&gt;Note: Only required if all calls on the gateway are to be controlled by DSAPP else it may lead to call failure.</td>
</tr>
<tr>
<td><strong>Step 4</strong>&lt;br&gt;param dialpeer number&lt;br&gt;Example:&lt;br&gt;<code>router(config)#application</code>&lt;br&gt;<code>router(config-app)#service dsapp</code>&lt;br&gt;<code>router(app-global)#param dialpeer 100</code></td>
<td>If multiple dial-peer matches are made for the destination-pattern, <code>dialpeer</code> 100 command is used.&lt;br&gt;Note: When you configure DSAPP on a dial-peer basis, specify a VOIP dial-peer for any outbound call. If all outbound calls that use the hookflash functionality are on the same server, it is recommended to use the <code>param dial-peer</code> command.&lt;br&gt;When multiple matches are possible on hookflash, enable <code>peer parameters callXXXX TRUE</code> for DSAPP to interpret hookflash to SIP supplementary service messages.</td>
</tr>
<tr>
<td><strong>Step 5</strong>&lt;br&gt;param callWaiting string&lt;br&gt;Example:&lt;br&gt;<code>router(config)#application</code>&lt;br&gt;<code>router(config-app)#service dsapp</code>&lt;br&gt;<code>router(app-global)#param dialpeer 100</code>&lt;br&gt;<code>router(app-global)#param callWaiting TRUE</code></td>
<td>Enables call waiting feature.</td>
</tr>
<tr>
<td><strong>Step 6</strong>&lt;br&gt;param callConference string</td>
<td>Enables call conference feature.</td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>router(config)#application</td>
<td></td>
</tr>
<tr>
<td>router(config-app)#service dsapp</td>
<td></td>
</tr>
<tr>
<td>router(app-global)#param dialpeer 100</td>
<td></td>
</tr>
<tr>
<td>router(app-global)#param callWaiting TRUE</td>
<td></td>
</tr>
<tr>
<td>router(app-global)#param callConference TRUE</td>
<td></td>
</tr>
</tbody>
</table>

**Step 7**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>param callTransfer string</td>
<td>Enables call transfer feature.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>router(config)#application</td>
<td></td>
</tr>
<tr>
<td>router(config-app)#service dsapp</td>
<td></td>
</tr>
<tr>
<td>router(app-global)#param dialpeer 100</td>
<td></td>
</tr>
<tr>
<td>router(app-global)#param callWaiting TRUE</td>
<td></td>
</tr>
<tr>
<td>router(app-global)#param callConference TRUE</td>
<td></td>
</tr>
<tr>
<td>router(app-global)#param callTransfer TRUE</td>
<td></td>
</tr>
</tbody>
</table>

**Configuring the Outbound VoIP Dial-peer**

Outbound dial-peer is configured like regular voip dial-peer for SIP. In addition to the parameters required, the following configurations are required:

- **service dsapp**— specifies this dial-peer will be controlled by dsapp application
- **session transport tcp** – specifies only TCP signaling is supported now
- **voice-class sip extension gw-ana** – used to interop with CUCM
- **voice-class sip bind control source-interface GigabitEthernet0/0/1** – need this interface’s mac as the base mac

```plaintext
dial-peer voice 714281111 voip
```
service dsapp
destination-pattern .+
session protocol sipv2
session transport tcp
session target ipv4:172.16.10.10
incoming called-number 7141116...
voice-class sip bind control source-interface GigabitEthernet0/0/0
codec g711ulaw
no shut

Note- G711 is the only codec supported for conference calls. It is recommended to add this command.

Configuring Pots Dial-peer

You can configure the pots dial-peer like a regular pots dial-peer for FXS. In addition to the parameters required, you have to configure the following command under pots dial-peer to interpret hook flash correctly and interop with CUCM:

- service dsapp—specifies this dial-peer to be controlled by DSAPP application.
- voice-class sip extension gw-ana—this parameter is used to interop with CUCM.

dial-peer voice 19993000 pots
service dsapp
destination-pattern 2124506300
voice-class sip extension gw-ana
port 3/0/0

Configuring Voice-card and SIP

When you configure the voice-card, all the traffic should go through the CUCM and the hairpin calls are not supported. You have to configure no local-bypass command for the voice-card that have FXS SIP endpoints.
For FXS SIP endpoints to register, configure the registrar IP address under the sip-ua mode and use the TCP as the transport type. UDP protocol is not supported.

voice service voip
sip
bind control source-interface GigabitEthernet0/0/0
session transport tcp
no shut
!
voice-card 3/0
no local-bypass
no watchdog
!
!
sip-ua
registrar ipv4:172.16.10.10 expires 3600 tcp
protocol mode dual-stack
!

**Note**- SIP service should be shut down before configuring the protocol mode. After configuring the protocol mode as dual-stack, SIP service should be reenabled.

**Enabling Device Control Session Application Line features**

To register to CUCM as a SIP endpoint, and to distinguish line feature from trunk, configure the `dsapp line` command.

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `dsapp line`
**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>enable</td>
<td>Enables privileged EXEC mode. Enter your password if prompted.</td>
</tr>
<tr>
<td></td>
<td>Example: Router&gt; enable</td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td>configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example: Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td>dsapp line</td>
<td>Specifies the format of each call feature.</td>
</tr>
<tr>
<td></td>
<td>Example: router(config)# dsapp line</td>
<td></td>
</tr>
<tr>
<td></td>
<td>router(config)#dsapp line</td>
<td></td>
</tr>
<tr>
<td></td>
<td>router(config)#</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Note</td>
<td>If you do not configure the <em>dsapp line</em> command, the gateway will act like a SIP trunk and the analog phones may not register as SIP endpoints. Also, you cannot configure the FAC. Ensure to configure the <em>dsapp line</em> command to enable the FXS for SIP supplementary services.</td>
</tr>
</tbody>
</table>

### Configuring Feature Access Code

The *dsapp line feature access-code* command invokes the feature to translate the Feature Access Code (FAC) to the format that the CUCM understands. If you do not configure this command, the whole FAC digits are sent to the CUCM and may not invoke features. You can also change the default FAC in the sub-mode.

Analog phones do not have soft keys. The required supplementary service features are invoked through FAC. By default, prefix of the FAC is ‘**’ and it can also be changed using the CLI command.

```cli
router(config)#dsapp line feature access-code
router(config-dsappline-fac)#prefix *#
router(config-dsappline-fac)#cancel-call-waiting **4
router(config-dsappline-fac)#exit
router# show dsapp line feature codes
dsapp line feature access-code
prefix *#
call forward all *#1
call forward cancel *#2
pickup local *#5
pickup group *#7
pickup direct *#6
```
cancel-call-waiting **4
last-redial *#3

If the **dsapp line feature access-code** is not configured, the voice gateway does not translate the FAC to the format that the CUCM understands. The whole FAC digits is sent to the CUCM.

After the FAC is disabled and re-enabled, all the FAC and prefix are rolled back to the default values.

```
router(config)#no dsapp line feature access-code
  Feature access-code disabled
router(config)#do show dsapp line feature codes

dsapline feature access-code disabled

router(config)#dsapp line feature access-code

router(config-dsappline-fac)#do show dsapp line feature codes

dsapp line feature access-code
  prefix **
call forward all **1
call forward cancel **2
pickup local **5
pickup group **7
pickup direct **6
cancel-call-waiting **9
last-redial **3
```

```
router(config-dsappline-fac)#do show run | b dsapp line
dsapp line
!
dsapp line feature access-code
```
Auto Configuration

To enable the auto-configuration, use the `ccm-manager sipana auto-config local` command. To get the XML configuration file, use the `ccm-manager config server` command to download the configuration file from the CUCM TFTP server. Configurations are needed on both CUCM and voice gateway. CUCM needs to be configured first, then those configurations can be pushed to the voice gateway.

```
ccm-manager sipana auto-config local GigabitEthernetx/y/z
```

```
ccm-manager config server 172.xx.0.0
```

**Note**- Auto-Config only adds the dialpeers for each endpoint configured on CUCM. All other required SIP CLI commands need to be added.

CUCM Configuration

1. Navigate to Device> Gateway>Add New>Gateway Type.

   ![Select the type of gateway you would like to add:](GigabitEthernetx/y/z)

   VG450 is used in this example.

   Select SIP as the protocol and click Next

   ![Select the type of gateway you would like to add:](SIP)

2. Enter the mac address of the interface used in sip bind control.
3. Configure the voice module and individual voice ports.

### Configured Slots, VICs and Endpoints

<table>
<thead>
<tr>
<th>Module in Slot 0</th>
<th>ISR-3NIM-MBRD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subunit 1</td>
<td>&lt; None &gt;</td>
</tr>
<tr>
<td>Subunit 2</td>
<td>NIM-2FXS-SIP</td>
</tr>
<tr>
<td>Subunit 3</td>
<td>&lt; None &gt;</td>
</tr>
<tr>
<td>Module in Slot 1</td>
<td>&lt; None &gt;</td>
</tr>
<tr>
<td>Module in Slot 2</td>
<td>&lt; None &gt;</td>
</tr>
<tr>
<td>Module in Slot 3</td>
<td>&lt; None &gt;</td>
</tr>
</tbody>
</table>
4 Add a directory Number > Save>Apply config.

### Directory Number Information

<table>
<thead>
<tr>
<th>Directory Number*</th>
<th>5104431020</th>
</tr>
</thead>
<tbody>
<tr>
<td>Route Partition</td>
<td>&lt; None &gt;</td>
</tr>
<tr>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>Alerting Name</td>
<td></td>
</tr>
<tr>
<td>ASCII Alerting Name</td>
<td></td>
</tr>
<tr>
<td>External Call Control Profile</td>
<td>&lt; None &gt;</td>
</tr>
<tr>
<td><strong>Allow Control of Device from CTI</strong></td>
<td></td>
</tr>
<tr>
<td>Associated Devices</td>
<td>AN90A5EF7611100</td>
</tr>
</tbody>
</table>
The Analog port would now show as registered

<table>
<thead>
<tr>
<th>Phone Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product Type: SIP Station</td>
</tr>
<tr>
<td>Device Protocol: SIP</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Real-time Device Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registration: Registered with Cisco Unified Communications Manager ccm237</td>
</tr>
<tr>
<td>IPv4 Address: 10.77.31.252</td>
</tr>
</tbody>
</table>

Limitations:

1. Shared line on SIP registered FXS port is not supported.
2. Line side SIP endpoints can be controlled by one CUCM.
3. CUCM failover for analog SIP phones is not supported.
4. Music on hold is not supported.
5. Third-party call agents are not supported.
6. Only non-secure calls are supported.

Verify:

Use the following commands to verify the DSAPP configuration:

- show dsapp line device summary
- show dsapp line feature codes
- show ccm-manager config-download

The show dsapp line device summary command shows whether the FXS ports are successfully registered to the CUCM as SIP endpoints.

```
router#show dsapp line device summary
Total Devices: 3
Port Device Registration Dev Directory Last Number Identifier Name State Type Number Dialed
---------- --------------- ------ ----------- ---------- -------- ------- ------- ---------
3/0/0 ANDD309DD761600 REGISTERED ALG 2124506300 Not Avail
3/0/1 ANDD309DD761601 REGISTERED ALG 2124506301 Not Avail
```
The `show dsapp line feature codes` command shows whether FAC is enabled and feature codes.

```
router#show dsapp line feature codes

dsapp line feature access-code
  prefix **
  call forward all **1
  call forward cancel **2
  pickup local **5
  pickup group **7
  pickup direct **6
  cancel-call-waiting **9
  last-redial **3
```

The `show ccm-manager config-download` command provides download status and history of the auto-configuration.

```
Art_Utah_73#show ccm-manager config-download

SIP Line Side Analog auto-configuration status

=====================================================================================================
Registered with Call Manager: Yes
Local interface: GigabitEthernet0/0/0 (2c5a.0fc8.8b70)
Current version-id: 1541004382-f60b9ac2-ce5b-439e-92e5-02b62e26d15c
Current config applied at: 16:47:40 UTC Oct 31 2018
Gateway downloads succeeded: 2
Gateway download attempts: 2
Last gateway download attempt: 16:47:40 UTC Oct 31 2018
Last successful gateway download: 16:47:40 UTC Oct 31 2018
Current TFTP server: 172.19.156.84
Gateway resets: 1
Managed endpoints: 3
Endpoint downloads succeeded: 6
Endpoint download attempts: 6
```
Last endpoint download attempt: 16:47:40 UTC Oct 31 2018
Last successful endpoint download: 16:47:40 UTC Oct 31 2018
Endpoint resets: 0
Endpoint restarts: 0

Configuration Error History:

**Troubleshoot:**

For registration issues capture CUCM SDL/SDI traces and run the following debugs on the gateway

d debug voip application session
d debug ccsip messages
d debug ccsip states
d debug ccsip error

For auto-configuration issues run the debug
d debug ccm-manager config-download all

**Related Information**

[Configuring the Cisco Fourth-Generation T1/E1 Voice and WAN Network Interface Module](#)