

Unified Border Element (CUBE) Session Initiation Protocol (SIP) Normalization with SIP Profiles Configuration Example

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Conventions](#)

[Configure](#)

[General Configuration Command Structure](#)

[Configuration Steps](#)

[Configuration Caveats](#)

[Configuration Examples](#)

[Add](#)

[Remove](#)

[Modify](#)

[Full Sample Configuration with SIP Profiles](#)

[Additional Configuration Notes](#)

[Supported SIP Messages](#)

[SIP Requests](#)

[SIP Responses](#)

[SIP Headers](#)

[SDP Headers](#)

[Verify](#)

[Troubleshoot](#)

[NetPro Discussion Forums - Featured Conversations](#)

[Related Information](#)

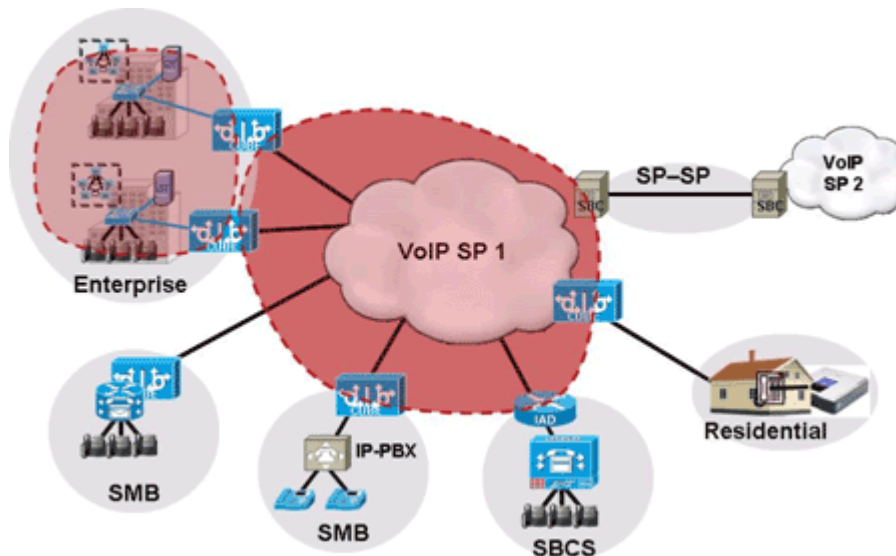
Introduction

Protocol translation and repair is a key Cisco Unified Border Element (CUBE) function. CUBE can be deployed between two devices that support the same VoIP protocol (SIP), but do not interwork because of differences in how the protocol is implemented or interpreted. CUBE can customize the SIP messaging on either side to what the devices in that segment of the network expect to see by normalizing the SIP messaging on the network border, or between two non-interoperable devices within the network.

Service providers may have policies for which SIP messaging fields should be present (or what constitutes valid values for the header fields) before a SIP call enters their network. Similarly, enterprises and small businesses may have policies for the information that can enter or exit their

networks for policy or security reasons from a service provider SIP trunk.

In order to customize SIP messaging in both directions, you can place CUBE with a SIP normalization configuration at the boundary of these networks as shown in this image:



In addition to network policy compliance, the CUBE SIP normalization capabilities can be used to resolve incompatibilities between SIP devices inside the enterprise network. These are the situations in which incompatibilities can arise:

- A device rejects an unknown header (value or parameter) instead of ignoring it
- A device sends incorrect data in a SIP message
- A device does not implement (or implements incorrectly) protocol procedures
- A device expects an optional header value or parameter, or an optional protocol procedure that can be implemented in multiple ways
- A device sends a value or parameter that must be changed or suppressed before it leaves or enters the network
- Variations in the SIP standards on how to achieve certain functions

The SIP profiles feature on CUBE provides a solution to these SIP normalization and customization issues.

Prerequisites

Requirements

Ensure that you meet these requirements before you attempt this configuration:

- Basic knowledge of how to configure and use Cisco IOS voice features (such as dial-peers)

- Basic knowledge of how to configure and use the Cisco Unified Border Element (CUBE)
- Intermediate knowledge of the SIP protocol and its messages, fields, and parameters

Components Used

The information in this document is based on these software and hardware versions:

- Cisco Unified Border Element release on a Cisco ISR, Cisco AS5400XM Access Gateway, Cisco AS5350XM Access Gateway, Cisco 7200 router, or a Cisco 7300 router that runs Cisco IOS release 12.4.15XZ or later
- Cisco TDM-to-SIP gateway on a Cisco ISR Cisco AS5400XM Access Gateway, or a Cisco AS5350XM Access Gateway that runs Cisco IOS release 12.4.15XZ or later

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Conventions

Refer to the [Cisco Technical Tips Conventions](#) for more information on document conventions.

Configure

The configuration of the SIP profiles feature allows you to add, modify, or remove any SIP or SDP header value in an outgoing SIP message on CUBE. A list of the exact messages and headers supported is provided in the [Supported SIP Messages](#) section of this document. SIP profiles can be configured either at the dial-peer level or the global level.

The syntax for message modification uses regular expression notation to *match* and *replace* fields in messages. Matched substrings can be used in *replace* patterns. When multiple regular expression rules apply to the same [method/response]:header combination, the second rule applies to the result string of the first rule.

SIP Profiles does not allow you to *remove* or *add* mandatory SIP headers. Only the *modify* option is available for mandatory headers. Mandatory SIP headers include To, From, Via, CSeq, Call-Id, and Max-Forwards. Mandatory SDP headers include v, o, s, t, c, and m.

The *ANY* special keyword is provided in the CLI to indicate that a rule must be applied to any message within the specified category. The rules configured for an *INVITE* message are applied only to the first INVITE in the protocol sequence for the call. A special *REINVITE* keyword is supported to define operation needed on subsequent INVITEs in a protocol sequence for the call.

SIP profiles can also be used to change a header name from the long form to the compact form; for example, *From* to *f*. This can be used as a way to reduce the length of a SIP message. By default Cisco IOS SIP never sends the compact form of the SIP messages although it receives either the long or the short form.

The SIP profiles feature affects only outgoing SIP messages. The rules are applied as the last step before the message leaves the CUBE router; that is, after destination dial-peer matching has taken place. Changes to the SIP messages are not remembered or acted on by the CUBE application.

The Content-length field is recalculated after the SIP Profiles rules are applied to the outgoing message.

Note: Use the [Command Lookup Tool](#) ([registered](#) customers only) to obtain more information on the commands used in this section.

General Configuration Command Structure

The general command that defines a rule to *add* a field to a SIP method/response is:

```
<request/response> <message code> <sip-header/sdp-header> <header-name> add <add-value>
```

The general command that defines a rule to *remove* a field to a SIP method/response is:

```
<request/response> <message code> <sip-header/sdp-header> <header-name> remove
```

The general command that defines a rule to *modify* a field to a SIP method/response is:

```
<request/response> <message code> <sip-header/sdp-header> <header-name> modify  
<match-pattern> <replace-pattern>
```

Configuration Steps

The first step is to define the rules. In order to define the rules, use the general command structure given in the previous section. For example:

```
voice class sip-profiles 100  
  request INVITE sip-header...  
  response 100 sip-header...  
  request INVITE sdp-header...
```

The second step is to apply the rules either to the global or dial-peer level of the CUBE configuration. In order to apply the rules globally to all calls traversing CUBE, use this command structure:

```
voice service voip  
  sip  
    sip-profiles 100
```

In order to apply the rules selectively to calls traversing only a particular outgoing dial-peer, use this command structure:

```
dial-peer voice 555 voip  
  voice-class sip-profiles 100
```

If rules are configured at both the global and dial-peer level, the dial-peer configuration takes precedence over global level configuration.

These notes apply generically to all Cisco IOS features that use regular expressions (which includes SIP profiles):

- When an *add-value*, *match-pattern* or *replace-pattern* contain white-space characters, the entire value must be included between double quotes. For example:

response 100 sip-header add “User-Agent: CISCO CUBE”

- When an *add-value*, *match-pattern* or *replace-pattern* contains double quotes (") and white space characters, a backslash (\) must prefix the inner quotes. For example, in order to add "CISCO" CUBE, use this command:

response 100 sip-header add “User-Agent: \”CISCO\” CUBE”

Configuration Caveats

- In order to provide the most flexibility, syntax checking is not performed on the SIP messages that result after the rules are applied. You must ensure through adequate testing that the changes you specify in the profile rules result in valid SIP protocol exchanges.
- The SIP message type cannot be changed with SIP profiles. That is, a *180 Ringing* response cannot be changed to a *183 Session Progress* response.
- Mandatory headers can only be modified; they cannot be added or removed. Mandatory SIP headers include To, From, Via, CSeq, Call-Id, and Max-Forwards. Mandatory SDP headers include v, o, s, t, c, and m.
- While regular expression variables can be used in *match* and *replace* substrings in order to store and reuse values, information cannot be extracted from one message and applied to another. For example, the calling number cannot be extracted from the INVITE and inserted into a subsequent REFER message. The rules specified apply to a single message at a time; only information in that message is manipulated.

Configuration Examples

This section provides examples of SIP profile rules to achieve specific changes to SIP messages.

Add

This section provides examples of how to add SIP and SDP headers to messages.

Example 1

Message: INVITE

Action: Add the b=AS:4000 SDP header to the video-media line

Rules:

```
voice class sip-profiles 100
  request INVITE sdp-header Video-Bandwidth-Info add "b=AS:4000"
```

Example 2

Message: 480 Temporarily Not Available

Action: Add the Retry-After SIP header

Rules:

```
voice class sip-profiles 100
  response 480 sip-header Retry-After add "Retry-After: 60"
```

Example 3

Message: INVITEs and REINVITEs

Action: Add the “user=phone” tag to the SIP URI header

Rules:

```
voice class sip-profiles 100
  request INVITE sip-header SIP-Req-URI modify "; SIP/2.0" ";user=phone SIP/2.0
  request REINVITE sip-header SIP-Req-URI modify "; SIP/2.0" ";user=phone SIP/2
```

Example 4

Message: 200 Response

Action: Add the User-Agent SIP header

Rules:

```
voice class sip-profiles 100
  response 200 sip-header User-Agent add "User-Agent: CiscoSystems-SIP-GW-UA"
```

Remove

This section provides examples on how to remove SIP and SDP headers to messages.

Example 5

Message: All requests and responses

Action: Remove the Cisco-Guid SIP header

Rules:

```
voice class sip-profiles 100
  request ANY sip-header Cisco-Guid remove
```

```
response ANY sip-header Cisco-Guid remove
```

Example 6

Message: BYE and CANCEL

Action: Remove the Reason SIP header

Rules:

```
voice class sip-profiles 100
  request BYE sip-header Reason remove
  request CANCEL sip-header Reason remove
```

Example 7

Message: 100 and 180 Responses

Action: Remove the Server SIP header

Rules:

```
voice class sip-profiles 100
  response 100 sip-header Server remove
  response 180 sip-header Server remove
```

Modify

This section provides examples on how to modify SIP and SDP headers in messages.

Example 8

Message: INVITE

Action: Modify the From: header to “gateway@gw-ip-address” format e.g. change 2222000020@9.13.24.7 to gateway@9.13.24.7

Rules:

```
voice class sip-profiles 100
  request INVITE sip-header From modify "<.*:)(.*@)" "\1gateway@"
```

Example 9

Message: INVITE

Action: replace “CiscoSystems-SIP-GW-UserAgent” with “-” in the o= line of the SDP header

Rules:

```
voice class sip-profiles 100
```

```
request INVITE sip-header Session-Owner modify "CiscoSystems-SIP-GW-UserAgent"
```

Example 10

Message: INVITE

Action: Convert “sip url” to “tel url” in the Req-URI, From and To headers e.g. from “sip:2222000020@9.13.24.6:5060” to “tel:2222000020”

Rules:

```
voice class sip-profiles 100
  request INVITE sip-header SIP-Req-URI modify "sip:(.*)@[^ ]+" "tel:\1"
  request INVITE sip-header From modify "<sip:(.*)@.*>" "<tel:\1>"
  request INVITE sip-header To modify "<sip:(.*)@.*>" "<tel:\1>"
```

Full Sample Configuration with SIP Profiles

This section provides a full configuration example for a CUBE router configured with SIP profiles.

CUBE
<pre>CUBE#show run Building configuration... Current configuration : 5888 bytes ! ! Last configuration change at 13:16:50 CDT Mon Feb 11 2008 ! NVRAM config last updated at 13:24:35 CDT Mon Feb 11 2008 ! version 12.4 service timestamps debug datetime msec localtime service timestamps log datetime msec localtime service password-encryption ! hostname CUBE ! boot-start-marker boot system flash:c2800nm-ipvoice_ivs-mz.124-18.2.2.PIA1p.bin boot-end-marker ! logging message-counter syslog logging buffered 2000000 no logging console no logging monitor enable lab 0 ! no aaa new-model memory-size iomem 10 clock timezone CDT -6 clock summer-time CDT recurring no network-clock-participate slot 1 ! voice-card 0 no dspfarm ! ip cef</pre>

```

!
voice service voip
media flow-around
allow-connections sip to sip
sip
sip-profiles 100
!
voice class sip-profiles 100
request INVITE sip-header SIP-Req-URI modify "; SIP/2.0" ";user=phone SIP/2.0"
request REINVITE sip-header SIP-Req-URI modify "; SIP/2.0" ";user=phone SIP/2.0"
!
interface GigabitEthernet0/0
ip address x.x.x.x 255.255.255.0
duplex full
speed 100
!
ip http server
!
control-plane
!
dial-peer voice 100 voip
destination-pattern .T
session protocol sipv2
session target ipv4:x.x.x.x
dtmf-relay rtp-nte
codec g711ulaw
no vad
!
sip-ua
!
line con 0
line aux 0
line vty 0 4
exec-timeout 90 0
!
scheduler allocate 20000 1000
end

```

Additional Configuration Notes

- The header length (which includes the header name) should not exceed 300 characters after modification with SIP profiles. The maximum header length for an *add* value is approximately 220 characters. The maximum SDP header length is 2048 characters. If any header length exceeds the maximum value after a rule is applied, that rule will be ignored, and the changes are not applied. If the total SDP length exceeds 2048 characters after modifications, all changes to the SDP are ignored and not applied.
- The SIP profiles feature cannot be used to drop an entire SIP message; it can only be used to manipulate (add, modify, or remove) the content in the message
- Regular expression *variables* can be used to extract and store parameters from an existing header, but the values stored in variables in one rule can be used **ONLY** in the replace-pattern of the same rule; it can NOT be used by any other rules, which means you cannot insert the values extracted by one rule into another rule.

Content specified between open parenthesis (() and closed parenthesis ()) in a match-pattern are stored in variables denoted by 1, 2, 3, ... 9 in the order that they are found. The stored values of these variables can then be inserted again in the replace-pattern by referencing the variables with \1, \2, ... \9 respectively.

For example, if we want to change:

```
Remote-Party-ID: "CUBE"  
<sip:2001@123.123.123.123>;privacy=off;screen=no to P-Asserted-Identity:  
"CUBE" <sip:2001@123.123.123.123>
```

This can be achieved by the following SIP Profile rule:

```
request INVITE sip-header Remote-Party-ID modify  
"Remote-Party-ID:(.*)".*" "P-Asserted-Identity:\1"
```

In this example, the sequence (.*>) matches "CUBE" <sip:2001@123.123.123.123>. This value is stored in variable 1, which is referenced in the replace-pattern with \1.

Supported SIP Messages

This section provides the CLI options of the SIP messages that can be customized with the CUBE SIP profiles feature.

SIP Requests

These SIP requests are supported:

```
router(config-class)#request ?  
ACK      sip ack  
ANY      any sip request  
BYE      sip bye  
CANCEL   sip cancel  
COMET    sip comet  
INFO     sip info  
INVITE    sip invite  
NOTIFY    sip notify  
OPTIONS   sip options  
PRACK     sip prack  
PUBLISH   sip publish  
REFER     sip refer  
REGISTER  sip register  
REINVITE  sip reinvite  
SUBSCRIBE sip subscribe  
UPDATE    sip info
```

SIP Responses

These SIP responses are supported:

```
router(config-class)#response ?  
100      Response code 100  
180      Response code 180
```

181	Response code	181
182	Response code	182
183	Response code	183
200	Response code	200
202	Response code	202
300	Response code	300
301	Response code	301
302	Response code	302
305	Response code	305
380	Response code	380
400	Response code	400
401	Response code	401
402	Response code	402
403	Response code	403
404	Response code	404
405	Response code	405
406	Response code	406
407	Response code	407
408	Response code	408
409	Response code	409
410	Response code	410
412	Response code	412
413	Response code	413
414	Response code	414
415	Response code	415
416	Response code	416
417	Response code	417
420	Response code	420
421	Response code	421
422	Response code	422
423	Response code	423
480	Response code	480
481	Response code	481
482	Response code	482
483	Response code	483
484	Response code	484
485	Response code	485
486	Response code	486
487	Response code	487
488	Response code	488
489	Response code	489
491	Response code	491
493	Response code	493
500	Response code	500
501	Response code	501
502	Response code	502
503	Response code	503
504	Response code	504
505	Response code	505
513	Response code	513
580	Response code	580
600	Response code	600
603	Response code	603
604	Response code	604
606	Response code	606
ANY	Any Response	

SIP Headers

These SIP headers are supported:

```
rtr(config-class)#request INVITE sip-header ?
Accept-Contact
Accept-Encoding
Accept-Header
Accept-Language
Accept-Resource-Priority
Alert-Info
Allow-Events
Allow-Header
Also
Authorization
CC-Diversion
CC-Redirect
CSeq
Call-ID
Call-Info
Cisco-Gcid
Cisco-Guid
Contact
Content-Disposition
Content-Encoding
Content-Id
Content-Length
Content-Type
Date
Diversion
Event
Expires
From
History-Info
Location
MIME-Version
Max-Forwards
Min-Expires
Min-SE
Orig-dial-plan
P-Asserted-Identity
P-Preferred-Identity
Privacy
Proxy-Authenticate
Proxy-Authorization
Proxy-Require
Rack
Reason
Record-Route
Refer-To
Referred-By
Reject-Contact
Remote-Party-ID
Replaces
Request-Disposition
Requested-By
Require
Resource-Priority
Retry-After
Route
Rseq
```

SIP-ETag
 SIP-If-Match
 SIP-Req-URI
 Server
 Session-Expires
 Session-Header
 Subscription-State
 Supported
 Term-dial-plan
 Timestamp
 To
 Unsupported
 User-Agent
 Via
 WWW-Authenticate
 Warning

SDP Headers

These SDP headers are supported:

```

rtr (config-class) #response 200 sdp-header ?
Attribute                               a=
Audio-Attribute                         a=
Audio-Bandwidth-Info                    b=
Audio-Connection-Info                  c=
Audio-Encryption-Key                   k=
Audio-Media                             m=audio
Audio-Session-Info                     i=
Bandwidth-Key                           b=
Connection-Info                        c=
Email-Address                           e=
Encrypt-Key                             k=
Phone-Number                           p=
Repeat-Times                            r=
Session-Info                           i=
Session-Name                           s=
Session-Owner                           o=
Time-Adjust-Key                         z=
Time-Header                             t=
Url-Descriptor                          u=
Version                                 v=
Video-Attribute                         a=
Video-Bandwidth-Info                   b=
Video-Connection-Info                  c=
Video-Encryption-Key                   k=
Video-Media                             m=video
Video-Session-Info                     i=

```

Verify

The INVITE messages with (and without) a SIP profiles configuration applied are shown in this section. Use this method to verify that the SIP profile rules in the configuration affect the correct and desired changes in SIP messages.

This example shows a sample configuration:

```
voice class sip-profiles 1
  request INVITE sdp-header Audio-Bandwidth-Info add "b=AS:1600"
  request ANY sip-header Cisco-Guid remove
  request INVITE sdp-header Session-Owner modify "CiscoSystems-SIP-GW-UserAgent"
```

This example shows a sample SIP INVITE message without the SIP profiles configuration applied (salient fields are highlighted in bold):

```
INVITE sip:2222000020@9.13.40.250:5060 SIP/2.0
Via: SIP/2.0/UDP 9.13.40.249:5060;branch=z9hG4bK1A203F
From: "sipp " <sip:1111000010@9.13.40.249>;tag=F11AE0-1D8D
To: <sip:2222000020@9.13.40.250>
Date: Mon, 29 Oct 2007 19:02:04 GMT
Call-ID: 4561B116-858811DC-804DEF2E-4CF2D71B@9.13.40.249
Cisco-Guid: 1163870326-2240287196-2152197934-1290983195
Content-Length: 290

v=0
o=CiscoSystemsSIP-GW-UserAgent 6906 8069 IN IP4 9.13.40.249
s=SIP Call
c=IN IP4 9.13.40.249
t=0 0
m=audio 17070 RTP/AVP 0
c=IN IP4 9.13.40.249
a=rtpmap:0 PCMU/8000
a=ptime:20
```

This example shows the same sample SIP INVITE message with the SIP profiles configuration applied (changed fields are highlighted in bold):

```
INVITE sip:2222000020@9.13.40.250:5060 SIP/2.0
Via: SIP/2.0/UDP 9.13.40.249:5060;branch=z9hG4bK1A203F
From: "sipp " <sip:1111000010@9.13.40.249>;tag=F11AE0-1D8D
To: <sip:2222000020@9.13.40.250>
Date: Mon, 29 Oct 2007 19:02:04 GMT
Call-ID: 4561B116-858811DC-804DEF2E-4CF2D71B@9.13.40.249
Content-Length: 279

v=0
o=- 6906 8069 IN IP4 9.13.40.249
s=SIP Call
c=IN IP4 9.13.40.249
t=0 0
m=audio 17070 RTP/AVP 0
c=IN IP4 9.13.40.249
a=rtpmap:0 PCMU/8000
a=ptime:20
b=AS:1600
```

Troubleshoot

debug ccsip all is a useful troubleshooting command for SIP profiles.

In this example, the text “voice class SIP Profiles” shows which SIP profile is applied. Here is a sample of the command output:

```

router#debug ccsip all
...
Oct 12 06:51:53.619: //-1/735085DC8F3D/SIP/Info/sipSPIGetShrlPeer:
Try match incoming dialpeer for Calling number:
: sippOct 12 06:51:53.619:
//-1/735085DC8F3D/SIP/Info/sipSPIGetCallConfig:
Peer tag 2 matched for incoming call
Oct 12 06:51:53.619: //-1/xxxxxxxxxxxx/SIP/Info/sipSPIGetCallConfig:
voice class SIP Profiles tag is set : 1
Oct 12 06:51:53.619: //-1/735085DC8F3D/SIP/Info/sipSPIGetCallConfig:
Not using Voice Class Codec
Oct 12 06:51:53.619: //-1/735085DC8F3D/SIP/Info/sipSPIGetCallConfig:
xcoder high-density disabled
Oct 12 06:51:53.619: //-1/735085DC8F3D/SIP/Info/sipSPIGetCallConfig:
Flow Mode set to FLOW_THROUGH
...

```

In this example, the text “sip_profiles” highlights the modifications performed by the SIP profiles configuration. Here is a sample of the command output:

```

router#debug ccsip all
...
Oct 12 06:51:53.647: //-1/xxxxxxxxxxxx/SIP/Info/
sip_profiles_application_change_sdp_line:
New SDP header is added : b=AS: 1600
Oct 12 06:51:53.647: //-1/xxxxxxxxxxxx/SIP/Info/
sip_profiles_update_content_length:
Content length header before modification :
Content-Length: 290
Oct 12 06:51:53.647: //-1/xxxxxxxxxxxx/SIP/Info/
sip_profiles_update_content_length:
Content length header after modification :
Content-Length: 279
...

```

NetPro Discussion Forums - Featured Conversations

Networking Professionals Connection is a forum for networking professionals to share questions, suggestions, and information about networking solutions, products, and technologies. The featured links are some of the most recent conversations available in this technology.

NetPro Discussion Forums - Featured Conversations for Voice
Service Providers: Voice over IP
AS5300 radius accounting - Oct 13, 2008 Asked about H.248 or Megacable - Oct 10, 2008 Need help on basic setup - Oct 9, 2008 MGCP alert script - Oct 9, 2008 GGSN 6.x or 7 on 6500 + MWAM - Oct 8, 2008
Voice & Video: Voice over IP
VIC2FXO w/ 3620 Router - Oct 13, 2008 Video-Conference MeetingPlace - Oct 13, 2008 DMS-DMP Local Video Playback - Oct 13, 2008 Get an error message when scheduling video conferencing from Meeting Place - Oct 10, 2008 Why does WAN speed degrad asymmetrically w distance? - Oct 9, 2008

Voice & Video: IP Telephony
assigning different ringtones to different phones - Oct 14, 2008 MGCP E1 PRI CM 6.1 - Oct 14, 2008 device profile settings for ip communicator - Oct 14, 2008 how do i activate nokia intellisync license - Oct 14, 2008 trouble with admin of air controller 526 - Oct 14, 2008
Voice & Video: IP Phone Services for End Users
Unable to call-forward-all to an external Number with CME - Oct 14, 2008 Do Phones Need Access to the Internet for Demo Apps in SDK - Oct 13, 2008 Support Line - Call Forward All through Hunt Pilot to cell phone - Oct 13, 2008 7921 audio problems - Oct 13, 2008 Interconnecting DME and CM - Oct 12, 2008
Voice & Video: Unified Communications
Unity 5.0 in failover - Oct 14, 2008 winmail.dat attachment in the email notifications from Unity - Oct 14, 2008 Unity MWI pending ON/ OFF in failover - Oct 14, 2008 Unity 5.0 with CUCM 5.1.3.4000-4 conflict - Oct 14, 2008 CUE and CallManager - Oct 14, 2008
Voice & Video: IP Phone Services for Developers
RisPort 401 Unauthorized - Oct 13, 2008 How to catch ALL call events with JTAPI - Oct 13, 2008 CUCM 6.1 - Phone Directory - how to add second number? - Oct 10, 2008 Invoking Service Programmatically - Oct 10, 2008 Unified CRM Connector - Oct 10, 2008
Voice & Video: General
Sip trunk between to CCM 4.13 server - Oct 14, 2008 Qos issue - Oct 13, 2008 EVM-HD-8FXS problem - Oct 13, 2008 Music-On-Hold - Oct 13, 2008 MPX 1.2 with upgraded CCM directory authentication problem - Oct 13, 2008

Related Information

- [Voice Technology Support](#)
- [Voice and Unified Communications Product Support](#)
- **Recommended Reading:** [Troubleshooting Cisco IP Telephony](#) 
- [Technical Support & Documentation - Cisco Systems](#)

Home	How to Buy	Login	Profile	Feedback	Site Map	Help
----------------------	----------------------------	-----------------------	-------------------------	--------------------------	--------------------------	----------------------

[Contacts & Feedback](#) | [Help](#) | [Site Map](#)

© 2007 - 2008 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)