



The bridge to possible

Cisco Community Meet The Authors

Meet the Authors Event- Leveraging SBCs to Empower a
Changing World of Collaboration

Steve Holl, Kaustubh Inamdar, Gonzalo Salgueiro, Arun Arunachalam & Kyzer Davis.
World-class Collaboration experts.

February 16th, 2021



Welcome to the new “Meet Authors event”

Learn from the IT expert that literally wrote the books & content
“Learn more about the latest trends in cybersecurity and the alternatives to enhance your security career”



Meet
Author



Learn the
Story behind



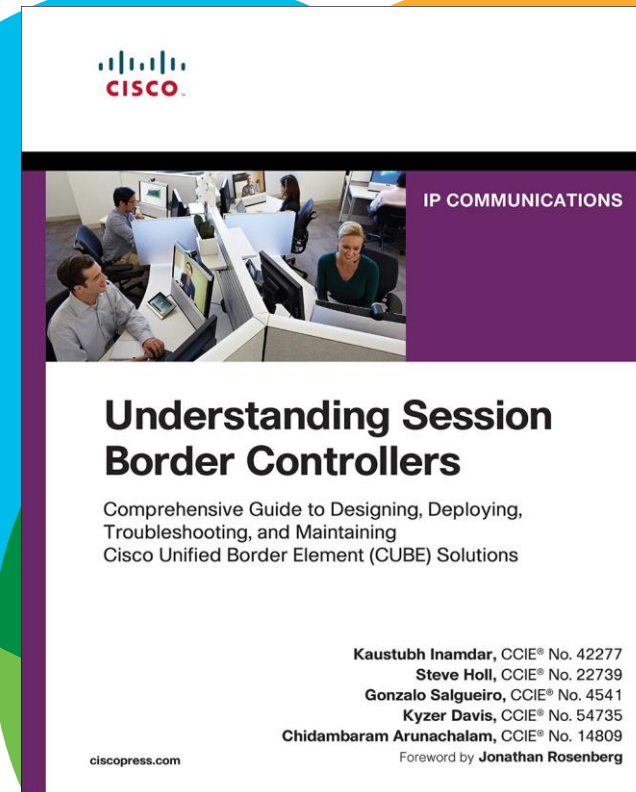
Trends &
Key Content



Clarify
Questions

Win a free signed copy!

Signed by the authors



2 free copies

Meet the Authors



Steve Holl

Collaboration Lead
CCIE #22739



Kyzer Davis

Technical Lead
CCIE #54735



Kaustubh Inamd

Technical Leader
CCIE #4227



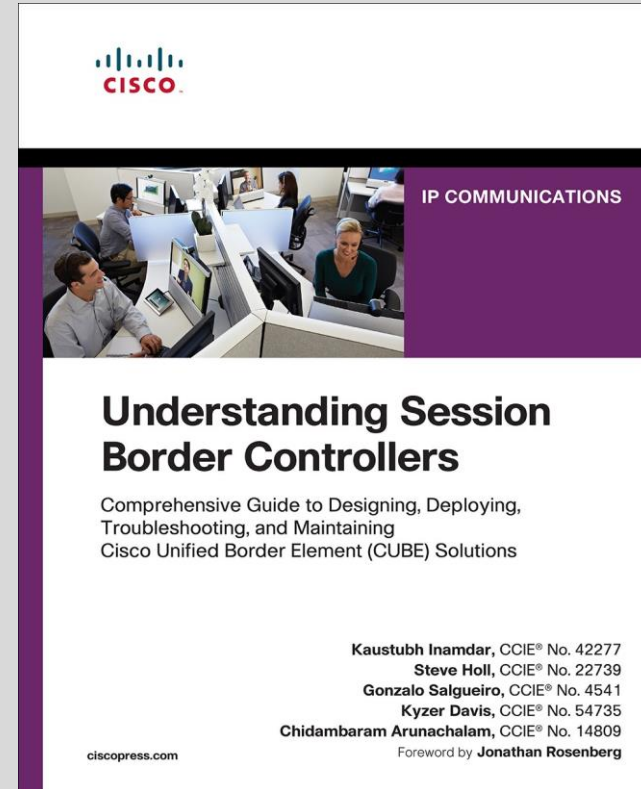
Gonzalo Salgueiro

Distinguished Engineer
CCIE #4541



Arun Arunachalam

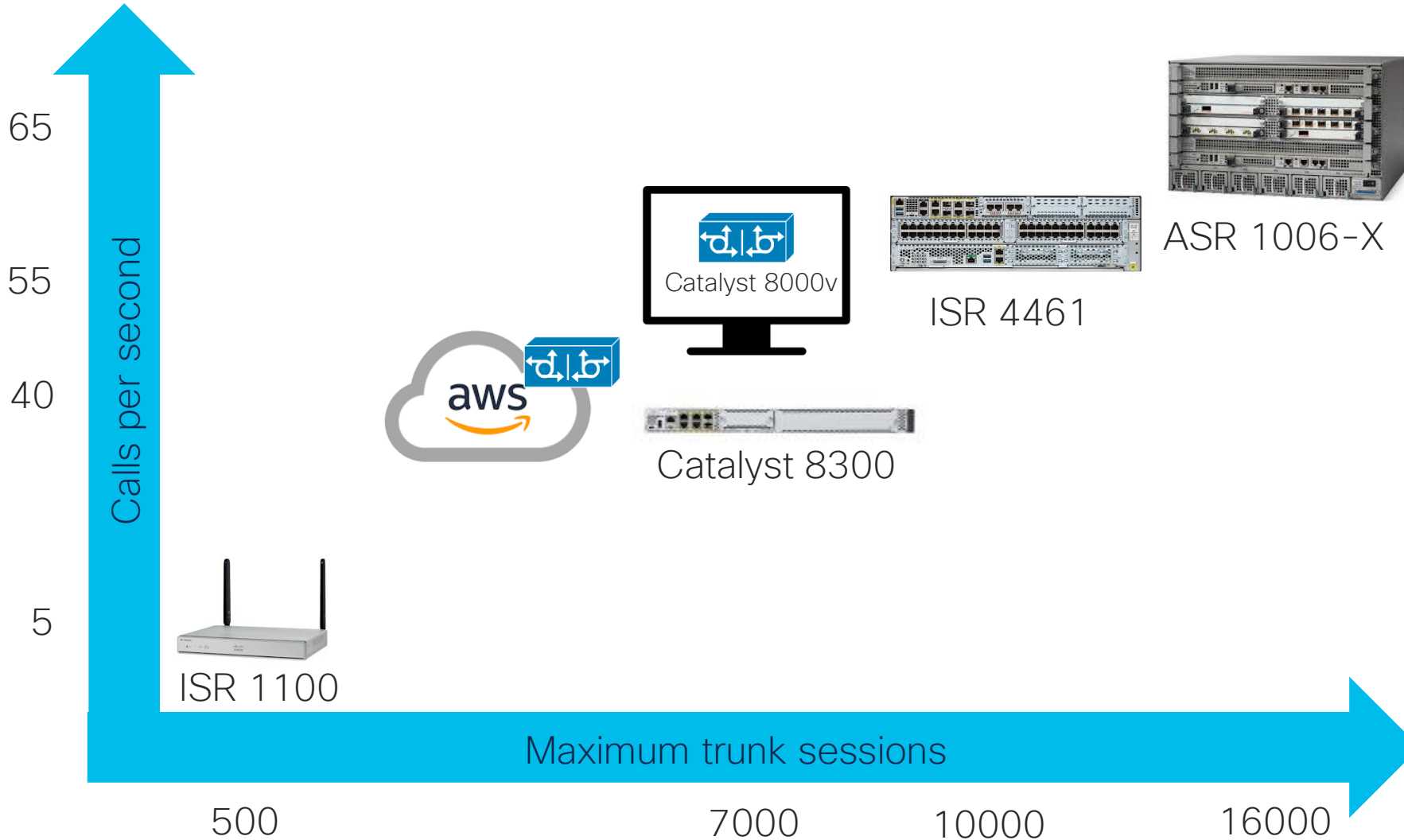
Principal engineer
CCIE #14809



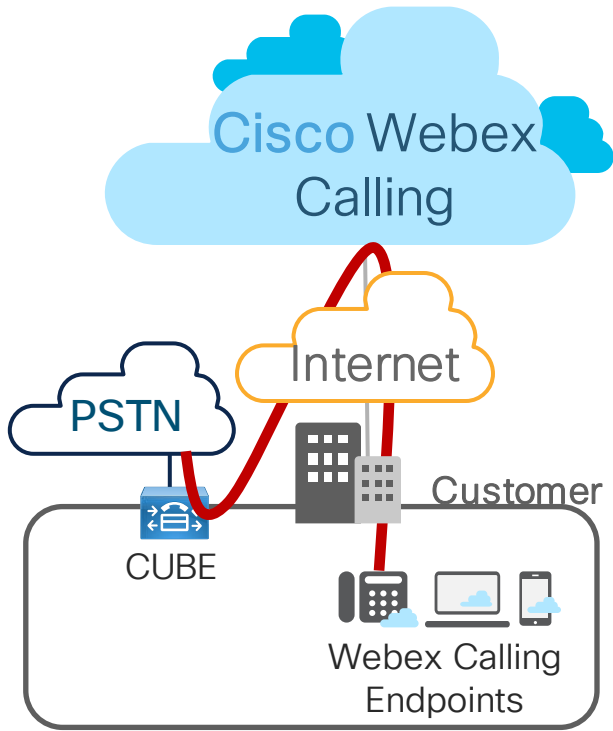
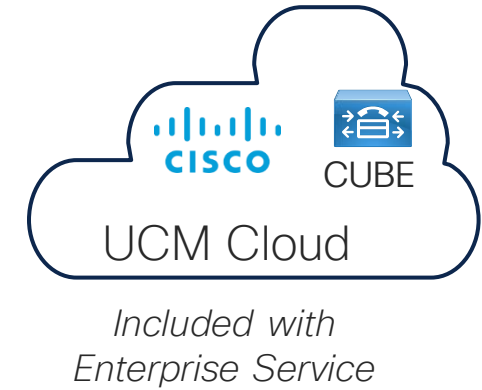
Experts who combined have over 70 years of collaboration experience

Platforms & Cloud Integrations

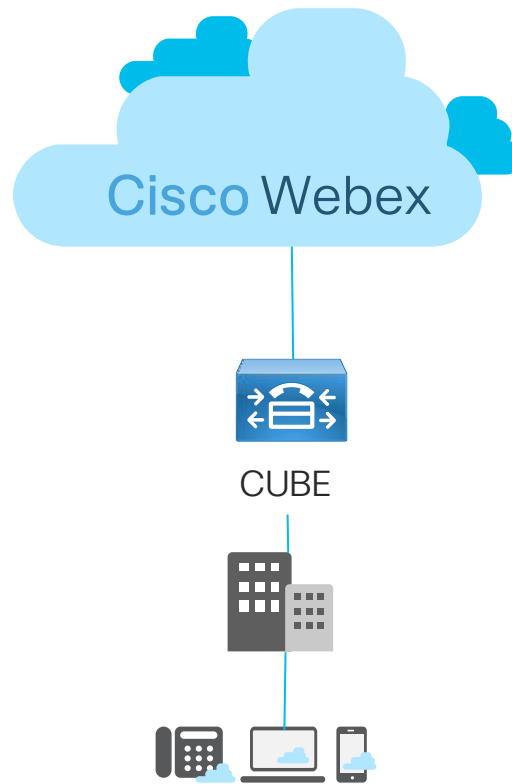
New CUBE Platforms



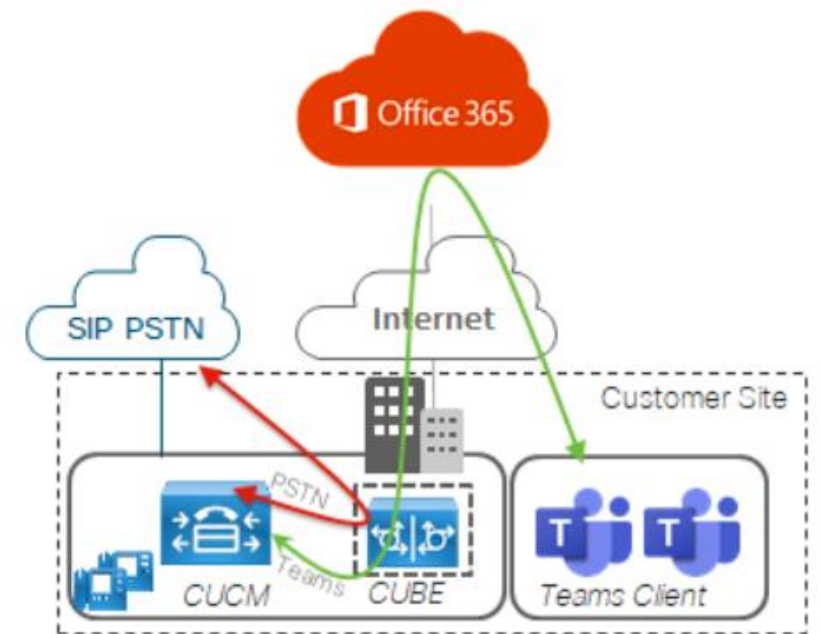
CUBE Cloud Calling Integrations



Webex Calling Local Gateway



Webex Edge Audio



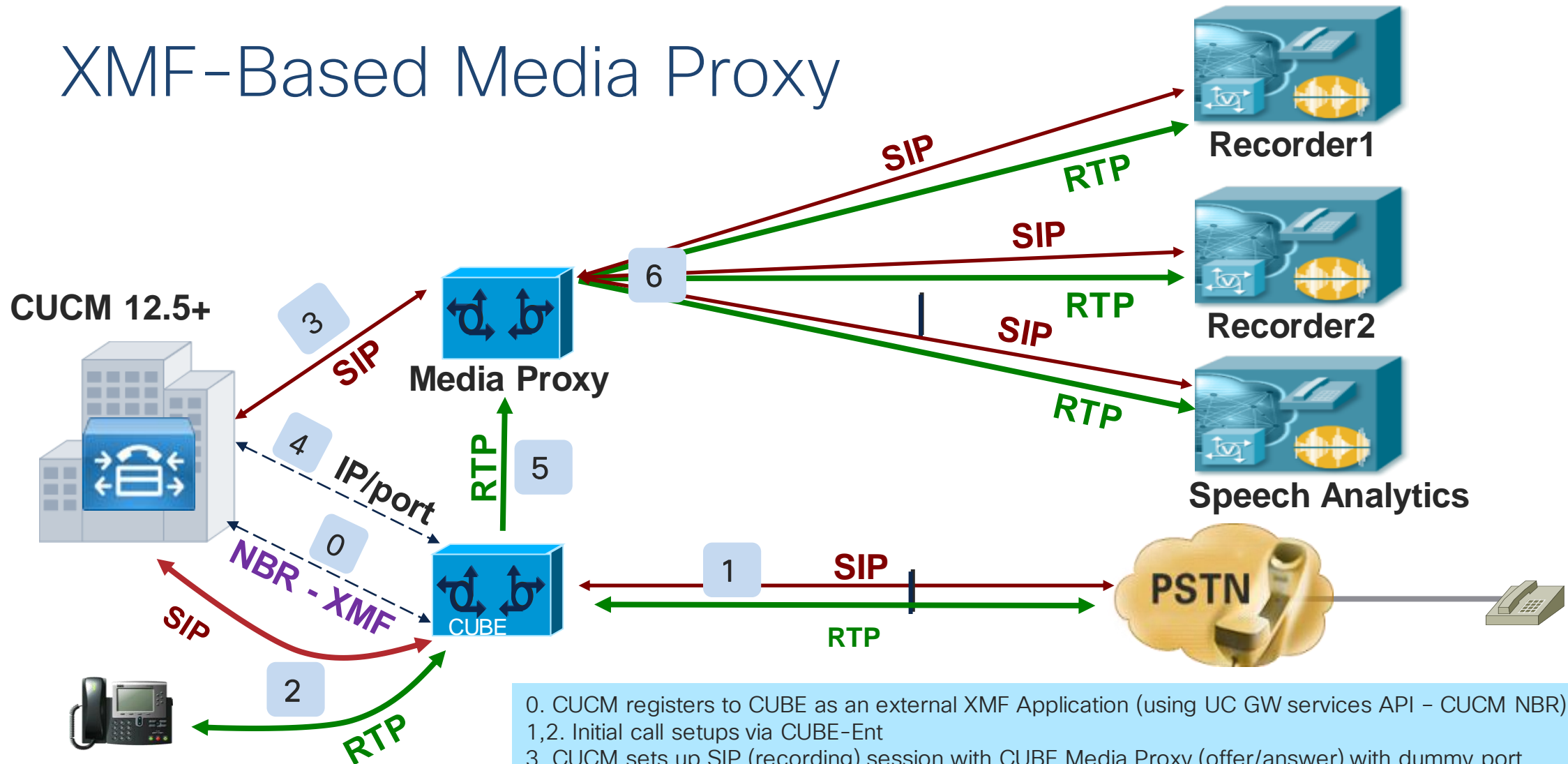
Microsoft Teams Direct Routing

What's New with CUBE?

Most Anticipated New Features

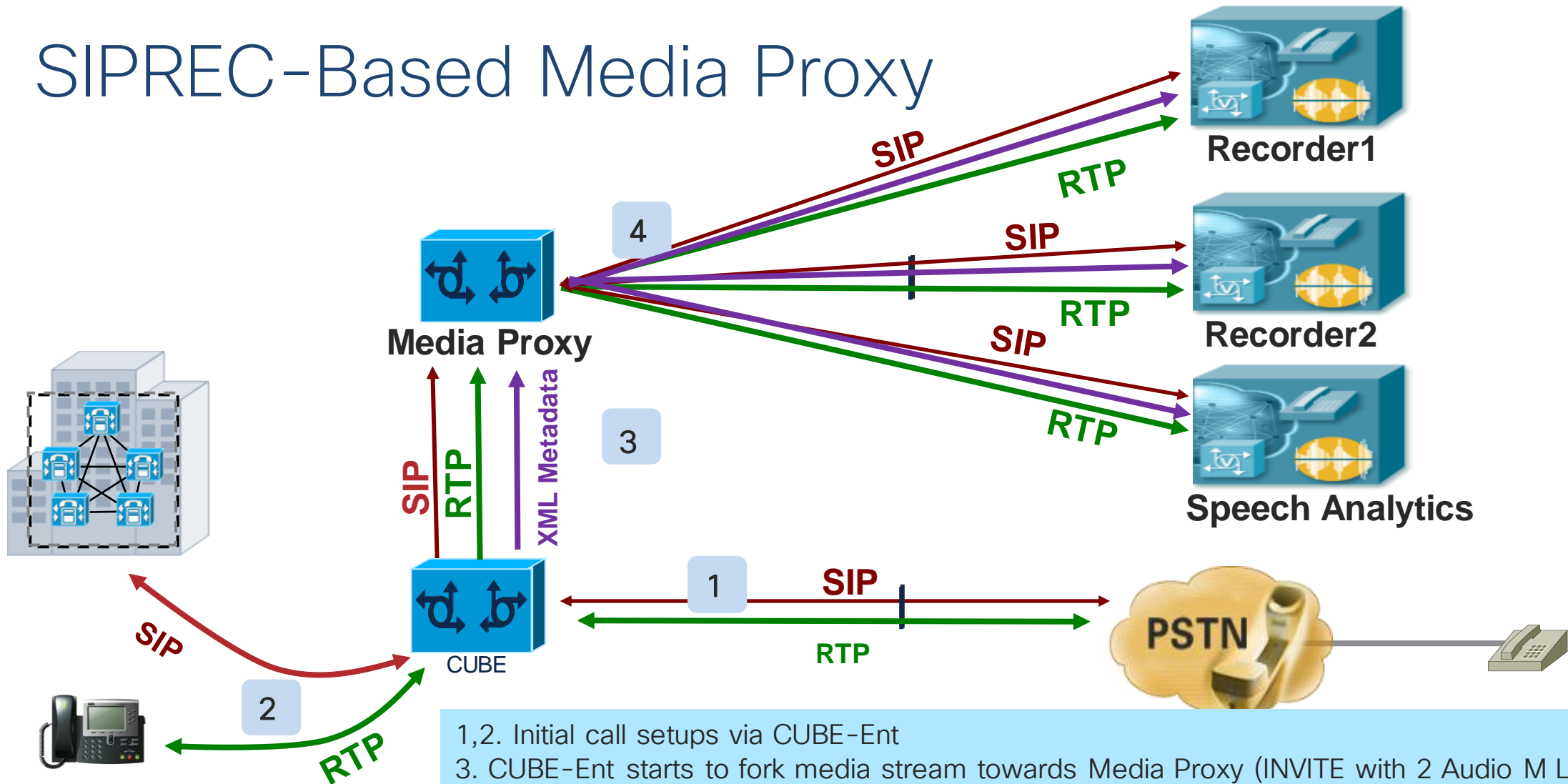
- OPUS Codec Support for IP-IP calls (17.3.1a)
- Dial-peer live binding with active calls (17.3.1a)
- Huntstop for Server Groups (17.4.1a)
- CUBE Fax detection (17.2.1r)
 - REFER or Re-INVITE
- CUBE Media Proxy for multiple recorders (16.10.1a)
 - XMF and SIPREC (17.3.1a)

XMF-Based Media Proxy



0. CUCM registers to CUBE as an external XMF Application (using UC GW services API - CUCM NBR)
 1,2. Initial call setups via CUBE-Ent
 3. CUCM sets up SIP (recording) session with CUBE Media Proxy (offer/answer) with dummy port
 4. MP destination IP/port obtained in Step-3 relayed by CUCM to CUBE via XMF API interface (HTTP)
 5. CUBE-Ent starts to fork media streams to the MP (target ip/port received in Step-4). MP accepts RTP because of Media latching in the inbound leg from CUCM
 6. MP sets up SIP recording sessions with the 3 Recorders for multi-fork. The ingress media stream from CUBE-Ent is then multi-forked by MP towards the 3 recorders simultaneously using the destination ip/ports as negotiated in the SIP offer/answer b/w MP and the Recorders.

SIPREC-Based Media Proxy



- 1,2. Initial call setups via CUBE-Ent
3. CUBE-Ent starts to fork media stream towards Media Proxy (INVITE with 2 Audio M Lines + XML Metadata)
4. Media Proxy accepts incoming SIPREC request from CUBE Ent and initiates an INVITE (2 Audio M Lines + XML Metadata) towards the Primary recorder - Recorder 1 above
Once a successful session with the Primary recorder has been established, MediaProxy sends an INVITE towards the rest of the recorders.

Serviceability Enhancements

- VoIP Trace (17.3.2, 17.4.1a)
- Clearing Hung RTP Sessions (17.4.1a)
- Collaboration Solution Analyzer Support

Call Tracing

VoIP Trace Enables Always-on Logging Functionality

Analyze the issue right away
without having to enable debug and wait for the issue to reoccur

Enabled by default!

```
voice service voip  
  trace  
  memory-limit <MB>
```

Commands to clear the VoIP Trace buffers

```
voice service voip  
  no trace  
  trace
```

Best Practice

Create a log with both configuration and trace outputs

Method 1:

```
term len 0  
show version  
show run  
show voip trace all
```

Method 2:

EEM Applet

EEM Applet to Generate VoIP Trace Log Bundle

Configuration

```
event manager applet VoIP-Trace
event none
action 1.0 info type routename
action 1.1 file open file1 voip-trace-bundle.log w
action 1.2 file puts file1 nonewline "$_info_routename#show version"
action 1.3 file close file1
action 1.4 cli command "enable"
action 1.5 cli command "show version | append voip-trace-bundle.log"
action 1.6 file open file1 voip-trace-bundle.log a
action 1.7 file puts file1 nonewline "$_info_routename#show run"
action 1.8 file close file1
action 1.9 cli command "enable"
action 2.0 cli command "show run | append voip-trace-bundle.log"
action 2.1 file open file1 voip-trace-bundle.log a
action 2.2 file puts file1 nonewline "$_info_routename#show voip trace all"
action 2.3 file close file1
action 2.4 cli command "enable"
action 2.5 cli command "show voip trace all | append voip-trace-bundle.log"
```

Execution

```
CUBE# event manager run VoIP-Trace
CUBE# copy voip-trace-bundle.log <target url>
```

VoIP Trace + Collaboration Solution Analyz 🧐 =

The screenshot shows the Cisco TAC Tool interface for log analysis. At the top left, there is a Cisco logo and the text "Tools Catalog / Cisco TAC Tool". At the top right, the user name "Arun Arunachalam" is displayed along with navigation icons. The main content area is titled "Log analysis" and features an "Upload log files" section. A file named "CSA-CUBE-VoIP-Trac..." (2.4 MB) is shown as selected. Below the file list is an "Upload files" button. To the right, there are two informational sections: "About the tool" and "When to use".

Tools Catalog / Cisco TAC Tool

Arun Arunachalam

Log analysis

Upload log files

CSA-CUBE-VoIP-Trac...
2.4 MB

1 Selected (Total: 2.4 MB)

Upload files

? About the tool

This tool analyzes the log files from multiple products in collaboration space and displays details about great amount of communications flows (calls, MRA logins, RTP/TCP/UDP streams, XMPP, STUN, etc.), configuration overview and diagnostic signatures highlighting known issues found and next action plan to resolve them. [More info](#)

? When to use


Use this tool when troubleshooting any issue on your collaboration servers or endpoints. Diagnostic signatures will suggest next action plan in case any known issues were found. Alternatively, use the tool output to visualize and better understand the communication flows and configuration to troubleshoot the issue further. [More info](#)

i Files are being uploaded to the same storage used and controlled by a service request and hence meets the same security requirements. For analysis the files are fetched in a sandbox unique and only accessible by the cco id and kept there for 8 hours after which they are automatically removed.

<https://cway.cisco.com/csa/>

CSA: Run Analysis

Available files ▼ More info

Select	Filename	Size	Product type	
<input checked="" type="checkbox"/>	CSA-CUBE-VoIP-Trace-17.3.2-Logs.txt	2.532 MB	CUBE	

Select all Run Analysis Delete all

System information ▼ More info

General information

HOST CONFIGURATION

Version	Cisco IOS XE Software, Version 17.03.02
Hostname	cube1
Hardware	ISR4451-X/K9

CSA: Select the Call of Interest

CUBE calls overview

Call overview was loaded based on TODO logs. Clicking on a call below will trigger the full analysis, which will require additional time.

Search:

From DN / URI	To DN / URI	Call-Id	SIP Call-Id	Peer Call-Id	guid	Call initiated (UTC)
4115	9195552015	63	7df09880-fb5114fe-2319b-c86e12ac@172.18.110.200	66	7DF098800002	2021-11-18 12:35:12
4115	2015	66	563FCC6F-28D111EB-8019F9BC-9CACEB3F@172.18.110.203	63	7DF098800002	2021-11-18 12:35:12
4116	9195552016	64	7df09880-fb5114fe-2319c-c86e12ac@172.18.110.200	67	7DF098800002	2021-11-18 12:35:12
4116	2016	67	564104BF-28D111EB-8023F9BC-9CACEB3F@172.18.110.203	64	7DF098800002	2021-11-18 12:35:12
4117	9195552017	65	7df09880-fb5114fe-2319d-c86e12ac@172.18.110.200	70	7DF098800002	2021-11-18 12:35:12
4117	2017	70	564327E3-28D111EB-8031F9BC-9CACEB3F@172.18.110.203	65	7DF098800002	2021-11-18 12:35:12
4118	9195552018	68	7df09880-fb5114fe-2319e-c86e12ac@172.18.110.200	76	7DF098800002	2021-11-18 12:35:12
4118	2018	76	5648A5E4-28D111EB-803BF9BC-9CACEB3F@172.18.110.203	68	7DF098800002	2021-11-18 12:35:12
4119	9195552019	69	7df09880-fb5114fe-2319f-c86e12ac@172.18.110.200	77	7DF098800002	2021-11-18 12:35:12
4119	2019	77	56491B4E-28D111EB-803DF9BC-9CACEB3F@172.18.110.203	69	7DF098800002	2021-11-18 12:35:12

Showing 1 to 10 of 160 entries

Previous **1** 2 3 4 5 6 7 8 9 10 ... 16 Next

CSA: Get Inbound and Outbound Call Leg Details

Call detail

From: 4118@172.18.110.200 To: 9195552018@172.18.110.203

Call leg info Signaling Ladder diagram [Download pcap](#)

SIP - incoming Use for signaling and ladder [Ladder tags](#)

General information

SIP call leg type	Call
From	4118@172.18.110.200
To	9195552018@172.18.110.203
Signaling source	172.18.110.200 : 5060
Signaling destination	172.18.110.203 : 5060
Call-ID	7df09880-fb5114fe-2319e-c86e12ac@172.18.110.200
Call leg connects	✓ 2021-11-18 12:35:13 UTC

No RTP streams linked for this call leg

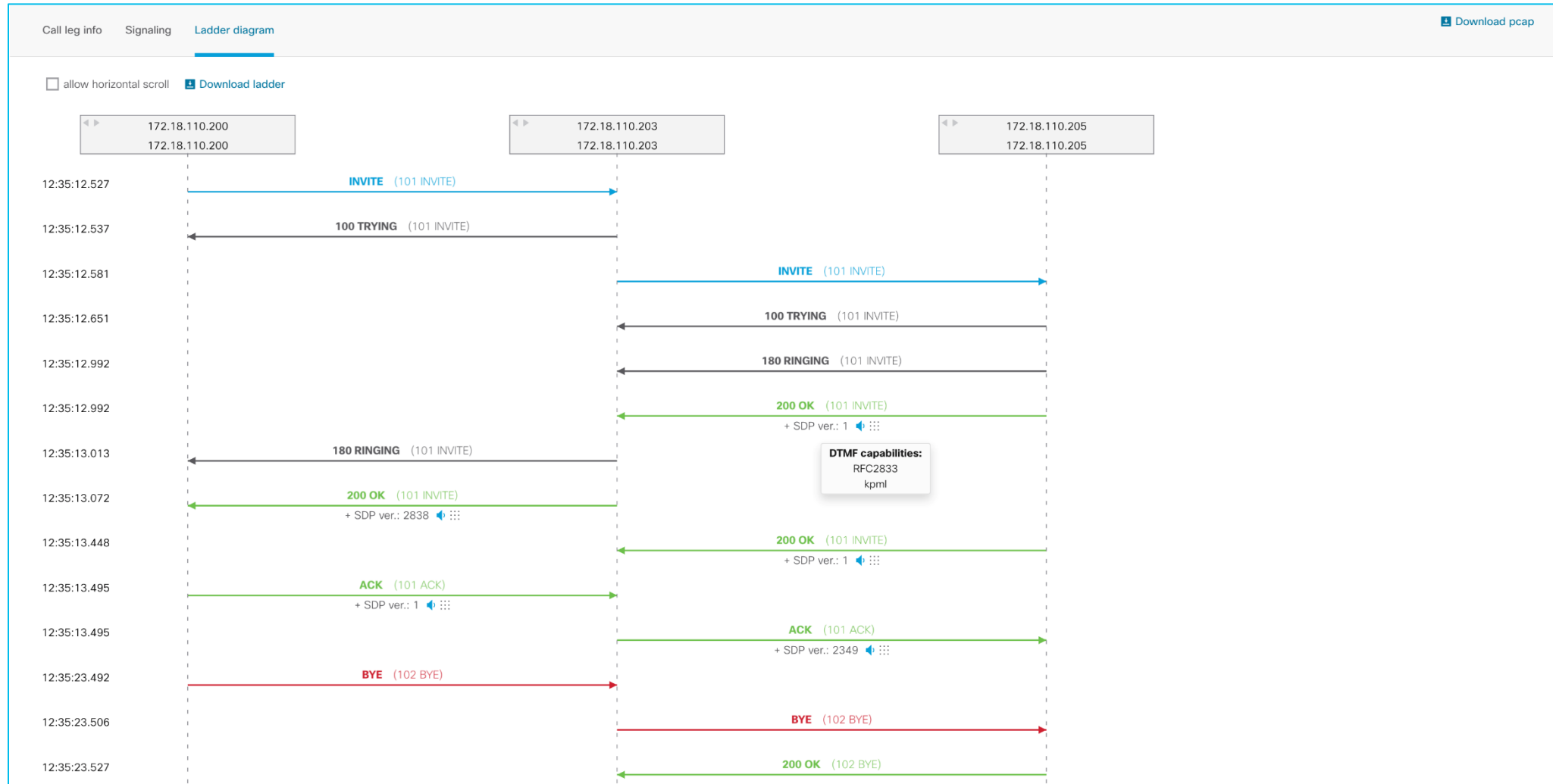
SIP - outgoing Use for signaling and ladder [Ladder tags](#)

General information

SIP call leg type	Call
From	4118@172.18.110.203
To	9195552018@172.18.110.203
Signaling source	172.18.110.203 : 5060
Signaling destination	172.18.110.205 : 5060
Call-ID	5648A5E4-28D111EB-803BF9BC-9CACEB3F@172.18.110.203
Call leg connects	✓ 2021-11-18 12:35:13 UTC

No RTP streams linked for this call leg

CSA: View the Signaling Ladder Diagram



CSA: View the SIP Messages of Interest

Call detail

From: 4118@172.18.110.200 To: 9195552018@172.18.110.203

Call leg info **Signaling** Ladder diagram [Download pcap](#)

Time (UTC)	Incoming legs	Outgoing legs	CSeq	Source	Destination	Message
12:35:12.527	→ Invite		101 INVITE	172.18.110.200 : 5060	172.18.110.203 : 5060	INVITE sip:9195552018@172.18.110.203:5060 SIP/2.0
12:35:12.537	← 100 Trying		101 INVITE	172.18.110.203 : 5060	172.18.110.200 : 5060	SIP/2.0 100 Trying
12:35:12.581		→ Invite	101 INVITE	172.18.110.203 : 5060	172.18.110.205 : 5060	INVITE sip:2018@172.18.110.205:5060 SIP/2.0
12:35:12.651		← 100 Trying	101 INVITE	172.18.110.205 : 5060	172.18.110.203 : 5060	SIP/2.0 100 Trying
12:35:12.992		← 180 Ringing	101 INVITE	172.18.110.205 : 5060	172.18.110.203 : 5060	SIP/2.0 180 Ringing
12:35:12.992		← 200 OK+ SDP ver.: 1	101 INVITE	172.18.110.205 : 5060	172.18.110.203 : 5060	SIP/2.0 200 OK
12:35:13.013	← 180 Ringing		101 INVITE	172.18.110.203 : 5060	172.18.110.200 : 5060	SIP/2.0 180 Ringing
12:35:13.072	← 200 OK+ SDP ver.: 2838		101 INVITE	172.18.110.203 : 5060	172.18.110.200 : 5060	SIP/2.0 200 OK
12:35:13.448		← 200 OK+ SDP ver.: 2849	101 INVITE	172.18.110.205 : 5060	172.18.110.203 : 5060	SIP/2.0 200 OK
12:35:13.495	→ Ack+ SDP ver.: 1		101 ACK	172.18.110.200 : 5060	172.18.110.203 : 5060	ACK sip:9195552018@172.18.110.203:5060;transport=tcp SIP/2.0
12:35:13.495		→ Ack+ SDP ver.: 2849	101 ACK	172.18.110.203 : 5060	172.18.110.205 : 5060	ACK sip:2018@172.18.110.205:5060 SIP/2.0
12:35:23.492	→ Bye		102 BYE	172.18.110.200 : 5060	172.18.110.203 : 5060	BYE sip:9195552018@172.18.110.203:5060;transport=tcp SIP/2.0
12:35:23.506		→ Bye	102 BYE	172.18.110.203 : 5060	172.18.110.205 : 5060	BYE sip:2018@172.18.110.205:5060 SIP/2.0
12:35:23.527		← 200 OK	102 BYE	172.18.110.205 : 5060	172.18.110.203 : 5060	SIP/2.0 200 OK

Audio capabilities:

- 0 - PCMU
- 8 - PCMA
- 15 - G728
- 18 - G729

101 - telephone-event

Listening IP and port:
172.18.110.112:41606

Mode:
sendrecv

Bandwidth:
TIAS:64000

CSA: View the SIP Message Details

Time (UTC)	Incoming legs	Outgoing legs	CSeq	Source	Destination	Message
12:35:12.527	→ Invite		101 INVITE	172.18.110.200 : 5060	172.18.110.203 : 5060	☞ INVITE sip:9195552018@172.18.110.203:5060 SIP/2.0
12:35:12.537	← 100 Trying		101 INVITE	172.18.110.203 : 5060	172.18.110.200 : 5060	☞ SIP/2.0 100 Trying
12:35:12.581		→ Invite	101 INVITE	172.18.110.203 : 5060	172.18.110.205 : 5060	☞ INVITE sip:2018@172.18.110.205:5060 SIP/2.0
12:35:12.651		← 100 Trying	101 INVITE	172.18.110.205 : 5060	172.18.110.203 : 5060	☞ SIP/2.0 100 Trying
12:35:12.992		← 180 Ringing	101 INVITE	172.18.110.205 : 5060	172.18.110.203 : 5060	☞ SIP/2.0 180 Ringing
						☞ SIP/2.0 200 OK Via: SIP/2.0/UDP 172.18.110.203:5060;branch=z9hG4bK5133C From: <sip:4118@172.18.110.203>;tag=6BBB4-726 To: <sip:2018@172.18.110.205>;tag=8284994-154a292d-1558-48b6-9155-e3e49acd9216-27524988 Date: Wed, 18 Nov 2020 12:35:12 GMT Call-ID: 5648A5E4-28D111EB-803BF9BC-9CACEB3F@172.18.110.203 CSeq: 101 INVITE Allow: INVITE, OPTIONS, INFO, BYE, CANCEL, ACK, PRACK, UPDATE, REFER, SUBSCRIBE, NOTIFY Allow-Events: presence, kpml Supported: replaces Server: Cisco-CUCM11.5 Call-Info: <urn:x-cisco-remotecc:callinfo>;x-cisco-video-traffic-class=DESKTOP Supported: X-cisco-srtp-fallback Supported: Geolocation Session-Expires: 1800;refresher=uas Require: timer Session-ID: 52cc629100405008000000a10000518;remote=66801f96403b6bb36394beb3e9b3eba0;logme P-Preferred-Identity: <sip:2018@172.18.110.205> Remote-Party-ID: <sip:2018@172.18.110.205>;party=called;screen=no;privacy=off Contact: <sip:2018@172.18.110.205:5060>;+sip.instance="<urn:uuid:00000000-0000-0000-0000-000A10000518>";+u.sip.devicename.ccm.cisco.com="SEP000A10000518";+u.sip.model.ccm.cisco.com="685" Content-Type: application/sdp Content-Length: 353
12:35:12.992		← 200 OK+ SDP ver.: 1	101 INVITE	172.18.110.205 : 5060	172.18.110.203 : 5060	v=0 o=CiscoSystemsCCM-SIP 8284994 1 IN IP4 172.18.110.205 s=SIP Call c=IN IP4 172.18.110.112 b=TIAS:64000 b=AS:64 t=0 0 m=audio 41606 RTP/AVP 0 8 15 18 101 b=TIAS:64000 a=label:55 a=rtpmap:0 PCMU/8000 a=rtpmap:8 PCMA/8000 a=rtpmap:15 G728/8000 a=rtpmap:18 G729/8000 a=fmtp:18 annex=no a=rtpmap:101 telephone-event/8000

Identify and Clear Hung RTP Connections

```
CUBE# show voip rtp stats
```

Media-Address Range	Min Port	Max Port	Ports Available	Ports Reserved	Ports In-use
Global Media Pool (ID :1)	8000	48198	19999	101	0


```
Port GCCB Status CallID Src Port Leak? No call
```

```
IP Address Based Media Pool (ID :2)
```

IP Address	Media Pool	Min Port	Max Port	Ports Available	Ports Reserved	Ports In-use
8.43.21.94	8.43.21.94	10000	40000	14900	101	3


```
Port GCCB Status CallID Src Port Leak? No call
```

```
10024 Null Y
```

```
10028 Null Y
```

```
10034 Null Y
```



```
Total=205, GCCB(Inserted=0, Deleted=0, Null=3, Possible Leaked=3, Blocked=202)
```

```
CUBE# clear voip rtp port 2 10024,10028,10034
```

```
Any port(s) associated with an active call will not be cleared.[confirm]
```

```
Cleared port 10024
```

```
Cleared port 10028
```

```
Cleared port 10034
```

```
CUBE#
```

Indicates that a RTP port is used but is not associated with an active call

Security Enhancements

Security Updates in CUBE

- Security Readiness Criteria (SRC) Closure for CUBE (16.11.1a)
 - Global Password Encryption with Type 6 AES encryption
 - SRTP keys removed from debugs
 - SAN and Common Name Validation
- TLS Server Name Indication (SNI) - RFC6066 (17.3.1a)

Password Encryption Changes - 16.11.1a+

- Dial-peer, SIP-UA, Tenants, and STUN authentication credentials/shared secrets will use the new Secure reversible encryption Type 6 AES format password

```
CUBE # conf t  
CUBE(config)# key config-key password-encrypt Password123  
CUBE(config)# password encryption aes
```

- If master key is not pre-configured, there will be an error shown when the password is configured like in the example below

```
CUBE(config-sip-ua)# authentication username cisco password 0 myPassword123  
  
Failed type 6 encryption on password
```

- If password type 0 is used, it will be stored as type 6 AES encrypted password in configuration.

```
CUBE #show run | include credentials  
credentials number 123456789 username cisco_LGU password 6 myPassword realm myRealm
```

SRTP Key Removal from debugs - 16.11.1a+

Before:

```
a=crypto:1 AEAD_AES_256_GCM inline:915/GeQpf+f0g3ESOWJZ0ws11zB13veD3yFeCqJdyM3cGrndDydQMvuk2kQ=  
a=crypto:2 AEAD_AES_128_GCM inline:FBYaCvjFU6DmAON96lJ3tsKjyXH0+kIhWP1Diw==  
a=crypto:3 AES_CM_128_HMAC_SHA1_80 inline:2v53snxkDbCFXgr9hu9Cu2Anvgj7EHPEu9amsJo+  
a=crypto:4 AES_CM_128_HMAC_SHA1_32 inline:i2WhE/LfNWj0ZeU+d+cdgwjZX7bHBP0cPQ6Fr2pE
```

Current:

```
a=crypto:1 AEAD_AES_256_GCM inline:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  
a=crypto:2 AEAD_AES_128_GCM inline:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  
a=crypto:3 AES_CM_128_HMAC_SHA1_80 inline:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  
a=crypto:4 AES_CM_128_HMAC_SHA1_32 inline:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

```
CUBE# show sip-ua calls  
<truncated>  
LocalCryptoSuite          : AES_CM_128_HMAC_SHA1_80  
Remote Crypto Suite       : AES_CM_128_HMAC_SHA1_80  
Local Crypto Key          : ASJRH+Y5HxfJUVUXD3JTjC111TSMQN2K5tLt740C  
Remote Crypto Key        : nHJO2SuM/R1/Hem34+IXgOyz/Y5HrGG34WlPt1rO
```

TLS SAN and CN Validation - 16.11.1a+

TLS connection is established only if the domain/IP configured as destination target under dial-peer/server-group/outbound proxy exactly matches with a Common Name (CN) or Subject Alternate Name (SAN) list provided by the server as part of the server certificate in the TLS handshake.

```
CUBE(config-sip-ua)# crypto signaling default trustpoint CUBETP ?  
  client-vtp          Set Client Verification Trustpoint  
  cn-san-validate   Enable CN/SAN validation for certificate  
  ecdsa-cipher        Use ECDSA Ciphers  
  strict-cipher       Use only ciphers mandated by SIP standards  
  
CUBE(config-sip-ua)# crypto signaling default trustpoint CUBETP cn-san-validate server
```

Note: If both CN and SAN is present in the server certificate, SAN will take precedence over CN

TLS Server Name Indication (SNI) - 17.3.1a+

New set of CLIs to enable outbound Server Name Indication (SNI) Extension in CUBE TLS Client Hello. RFC 6066
This value will be the server CUBE is trying to establish a TLS connection with; derived from the session-target or
outbound proxy. (IP Addresses not supported. Must be FQDN. SRV will send resolved DNS A Record)

```
CUBE(config)# voice class tls-profile 1
CUBE(config-class)# ?
VOICECLASS configuration commands:
  cipher          Configure a cipher-suite
  client-vtp      Assign a client verification trustpoint
  cn-san          Configure CN/SAN certificate options
  description     Description of the tls-profile group
  exit            Exit from voice class configuration mode
  help            Description of the interactive help system
  no              Negate a command or set its defaults
  sni            Enable TLS SNI (Server Name Indication) Extension
  trustpoint      Associate a trustpoint

CUBE(config-sip-ua)# crypto signaling default tls-profile 1
```

Sample Syntax for all commands:

```
crypto signaling default tls-profile 1
cipher {ecdsa-cipher | strict-cipher}
client-vtp trustpoint-name
cn-san validate server
trustpoint cube-trustpoint-name
sni send
```

Secure Telephony Identity Revisited (STIR)

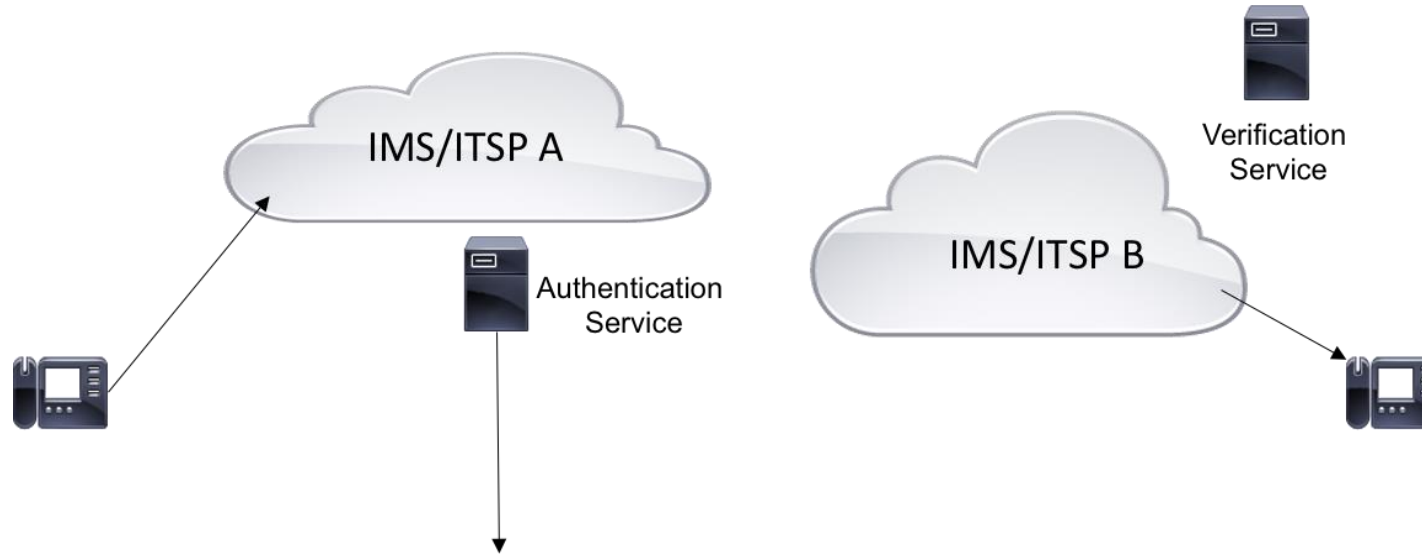
Secure Telephony Identity Revisited (STIR)

- STIR is an industry-wide effort to combat impersonation attacks and Robocalls.
- The Federal Communications Commission (FCC) requires service providers across the U.S. to implement & support STIR by June/July 2021.
- SIP-based framework to ensure ease of use across IP Multimedia Subsystem (IMS) & VoIP architectures.

Secure Telephony Identity Revisited (STIR)

- STIR used a new SIP header field called "Identity".
- Authenticated caller ID is inserted in the Identity header field in the form of a "PASSporT"
- A PASSporT is a JSON Web Token (JWT) that includes a header, payload & signature.
- The JWT payload contains "claims" - information such as the calling number, called number, time of the call, among others.

STIR Reference Architecture



Authentication Service creates and inserts a signed PASSporT in the SIP Identity header field – effectively providing cryptographic assurance of Caller ID.

STIR PASSporT Payload

```
{
  "attest": "A",
  "dest": {
    "tn": [
      "14045266060"
    ]
  },
  "iat": 1548859982,
  "orig": {
    "tn": "18001234567"
  },
  "origid": "3a47ca23-d7ab-446b-821d-33d5deedbed4"
}
```

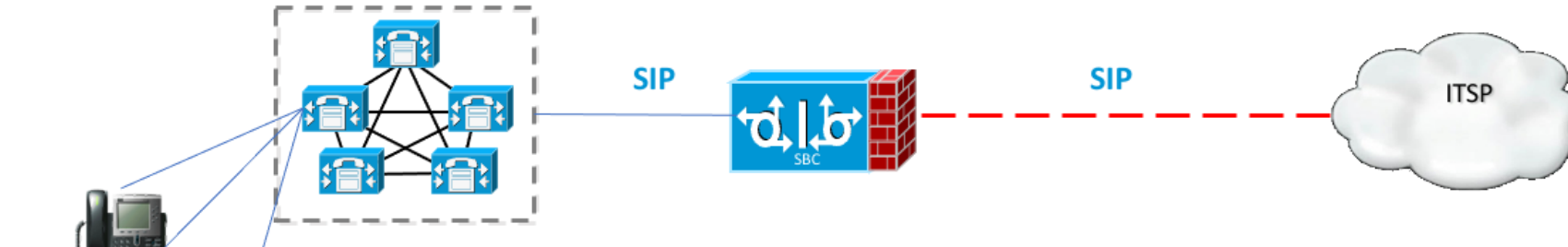
STIR & Rich Call Data

A PASSporT may be leveraged to transport additional information such as photos, logos, location information, etc. in a cryptographically secure way



Automatic SIP Trunking & Peering (ASAP)

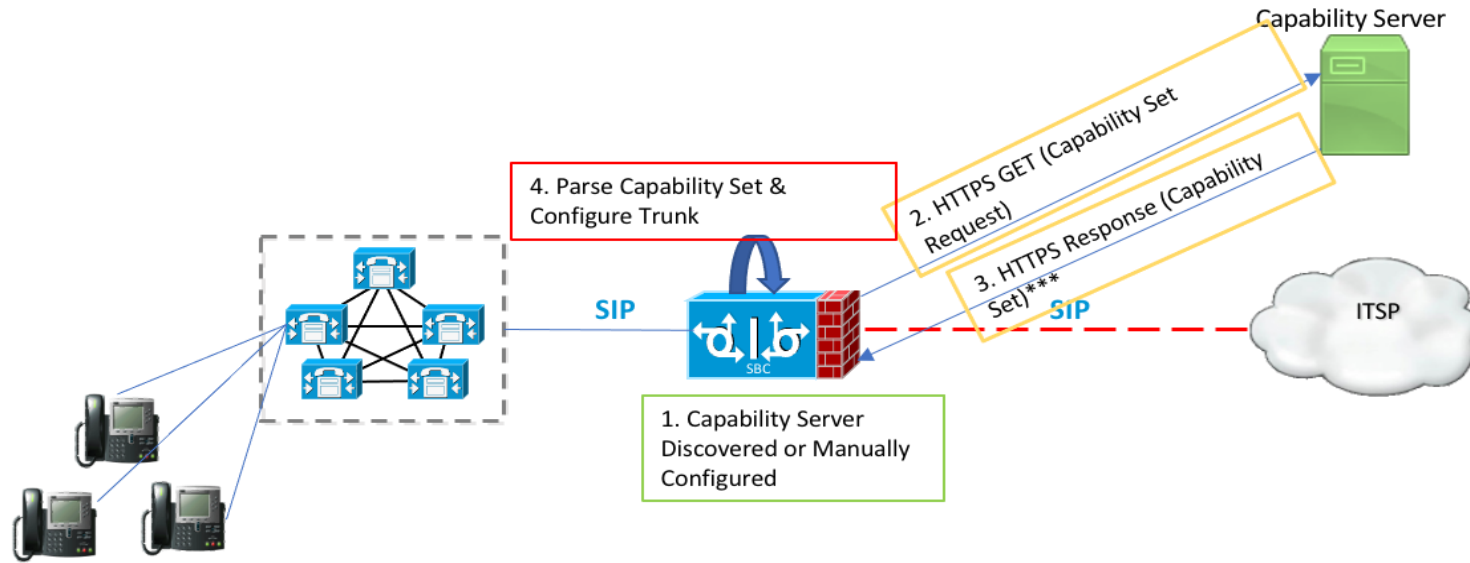
Automatic SIP Trunking & Peering (ASAP)



Bringing up service provider facing SIP trunks is a long process that could take several hours... sometimes days...

- Administrators usually uncover a significant number of problems when attempting to setup SIP trunking.
- Bringing up a SIP trunk often requires interaction with support teams.
- Deployment times increase significantly due to interoperability problems.

SIP Auto Peer



*** Body encoded in XML or JSON

Example Capability Set

```
<peering-info xmlns="urn:ietf:params:xml:ns:yang:ietf-peering"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:ietf:params:xml:ns:yang:ietf-peering ietf-peering.xsd">
  <variant>1.0</variant>
  <transport-info>
    <transport>TCP;TLS;UDP</transport>
    <registrar>registrar1.voip.example.com:5060</registrar>
    <registrar>registrar2.voip.example.com:5060</registrar>
    <registrarRealm>voip.example.com</registrarRealm>
    <callControl>callServer1.voip.example.com:5060</callControl>
    <callControl>192.168.12.25:5065</callControl>
    <dns>8.8.8.8</dns>
    <dns>208.67.222.222</dns>
    <outboundProxy>0.0.0.0</outboundProxy>
  </transport-info>
  <call-specs>
    <earlyMedia>true</earlyMedia>
    <signalingForking>false</signalingForking>
    <supportedMethods>INVITE;OPTIONS;BYE;CANCEL;ACK;PRACK;SUBSCRIBE;NOTIFY;REGISTER</supportedMethods>
  </call-specs>
  <media>
    <mediaTypeAudio>
      <mediaFormat>PCMU;rate=8000;ptime=20</mediaFormat>
      <mediaFormat> G729;rate=8000;annexb=yes</mediaFormat>
      <mediaFormat>G722;rate=8000;bitrate=56k,64k</mediaFormat>
    </mediaTypeAudio>
    <fax>
      <protocol>pass-through</protocol>
      <protocol>t38</protocol>
    </fax>
    <rtp>
      <RTPTrigger>true</RTPTrigger>
      <symmetricRTP>true</symmetricRTP>
    </rtp>
    <rtcp>
      <symmetricRTCP>true</symmetricRTCP>
      <RTCPFeedback>true</RTCPFeedback>
    </rtcp>
  </media>
  <dtmf>
    <payloadNumber>101</payloadNumber>
    <iteration>0</iteration>
  </dtmf>
  <security>
    <signaling>
      <type>TLS</type>
      <version>1.0;1.2</version>
    </signaling>
    <mediaSecurity>
      <keyManagement>SDES;DTLS-SRTP,version=1.2</keyManagement>
    </mediaSecurity>
  </security>
  <extensions>timer;rel100;gin;path</extensions>
</peering-response>
```

Submit Your
Questions Now!



Use the Q&A panel to submit your questions, our expert will respond.

Extra Resources and References

Cert Guide

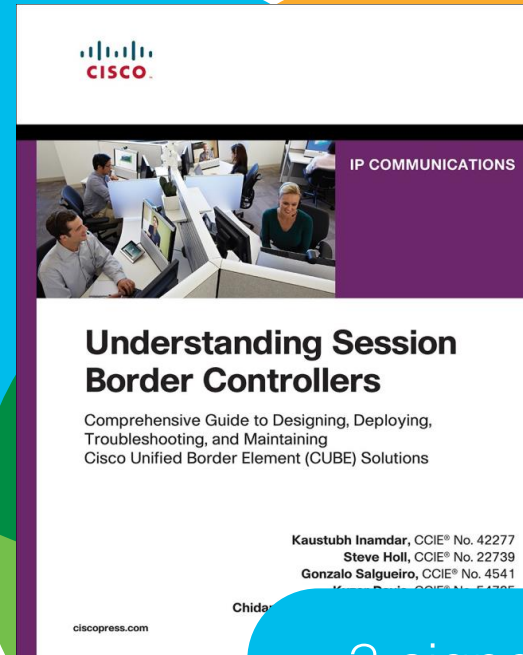
CCNP Collaboration Call Control and Mobility CLACCM 300-815 Official Cert Guide
[[Learn more](#)]

Other useful resources:

IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things
<https://www.ciscopress.com/store/iot-fundamentals-networking-technologies-protocols-9781587144561>

Fax, Modem, and Text for IP Telephony
<https://www.ciscopress.com/store/fax-modem-and-text-for-ip-telephony-9781587057588>

Congratulations winners!



2 signed books

We'll contact you via email

Thank you for Your Time!

Please help to complete the survey

Your opinion is important and help us to improve



Thanks For Joining today!



The bridge to possible