

Session Identifier

- Overview, on page 1
- Configure Support for Session Identifier, on page 3
- Tips to Troubleshoot, on page 3

Overview

Cisco Unified Border Element (CUBE) supports “Session Identifier” for end-to-end tracking of a SIP session in IP-based multimedia communication systems. Support for session identifier is in compliance with RFC 7206 and draft-ietf-insipid-session-id-15.

CUBE supports “Session Identifier” that overcomes the limitations with the existing call-identifiers and allows end-to-end tracking of a SIP session. To support session identifier, “Session-ID” header is added in the SIP request and response messages.

Note H.323 protocol is no longer supported from Cisco IOS XE Bengaluru 17.6.1a onwards. Consider using SIP for multimedia applications.

Note "Session Identifier" refers to the value of the identifier, whereas "Session-ID" refers to the header field used to convey the identifier.

The Session-ID comprises of Universally Unique Identifier (UUID) for each user agent participating in a call. Each call consists of two UUID known as local UUID and remote UUID. Local UUID is the UUID generated from the originating user agent and remote UUID is generated from the terminating user agent. The UUID values are presented as strings of lower-case hexadecimal characters, with the most significant octet of the UUID appearing first. Session Identifier comprises of 32 characters and remains same for the entire session. Refer to RFC 4122 for more information on UUID.

Example for Session ID header

```
Session-ID: ab30317f1a784dc48ff824d0d3715d86; remote=47755a9de7794ba387653f2099600ef2
```

In the above example:

Local UUID =

```
ab30317f1a784dc48ff824d0d3715d86
```

Remote UUID =

```
47755a9de7794ba387653f2099600ef2
```

Feature Behavior

- If all the user agents associated with CUBE support session-id, then CUBE allows pass-through of the Session ID header in all SIP request and response messages for the session.
- CUBE looks for the Session ID header present in the SIP messages and validates the SessionID header syntax as defined in draft-ietf-insipid-session-id-15. Session ID format earlier to draft-ietf-insipid-session-id-15 is considered as unsupported.
- If some of the user agents do not support session ID, CUBE generates local UUID on behalf of the user agent and sends the generated UUID in SIP request and response. CUBE generates UUID based on version 5 (SHA-1).
- If a Session ID is received in the format as defined in RFC 7329, CUBE considers it as unsupported. CUBE generates local UUID on behalf of the user agent and sends the generated UUID in SIP request and response.
- In a mid call scenario, where user a session is switched from supporting session identifier to non-supporting session identifier, CUBE saves the previous non-NULL session identifier and sends the saved non-NULL session identifier in re-invite messages as needed.
- For high availability, session ID is check pointed in active and re-created in standby.

Feature Information

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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1: Feature Information for Session Identifier Support

Feature Name	Releases	Feature Information
Support for Session Identifier	Baseline Functionality	A new keyword **session-id** is added to the following commands: • **show call active voice** • **show call active video** • **show call history voice** • **show call history video** • **show call active voice brief** • **show call active video brief**

Configure Support for Session Identifier

Session Identifier support is enabled on CUBE by default. No additional configuration required.

Tips to Troubleshoot

The following show commands helps you to troubleshoot any issues with session identifier.

- **show call active voice session-id** *WORD*
- **show call active voice brief session-id** *WORD*
- **show call active video session-id** *WORD*
- **show call active video brief session-id** *WORD*

WORD can be complete session identifier (local, remote, or both), or wildcard pattern of local or remote UUID. The valid wildcard patterns for search are *, 0-9, a-f, A-F.

The following session identifier is considered in the below examples:

```
SessionIDLocaluuid=db248b6cbdc547bbc6c6fdfb6916eeb
SessionIDRemoteuuid=4fd24d9121935531a7f8d750ad16e19
```

Valid Search Patterns

You can search for the session identifier using complete Session ID header as shown below:

```
Device# show call active voice session-id db248b6cbdc547bbc6c6fdfb6916eeb;
remote=4fd24d9121935531a7f8d750ad16e19
```

```
Telephony call-legs: 0
SIP call-legs: 1
H323 call-legs: 0
.
.
.
SessionIDLocaluuid=db248b6cbdc547bbc6c6fdfb6916eeb
SessionIDRemoteuuid=4fd24d9121935531a7f8d750ad16e19
.
.
SCCP call-legs: 0
Multicast call-legs: 0
Total call-legs: 1
```

You can search for the session identifier using complete local UUID as shown below:

```
Device#  show call active voice session-id db248b6cbdc547bbc6c6fdfb6916eeb
Telephony call-legs: 0
SIP call-legs: 1
H323 call-legs: 0
.
.
.
SessionIDLocaluuid=db248b6cbdc547bbc6c6fdfb6916eeb
SessionIDRemoteuuid=4fd24d9121935531a7f8d750ad16e19
.
.
SCCP call-legs: 0
Multicast call-legs: 0
Total call-legs: 1
```

You can search for the session identifier using complete remote UUID as shown below:

```
Device# show call active voice session-id 4fd24d9121935531a7f8d750ad16e19
Telephony call-legs: 0
SIP call-legs: 1
H323 call-legs: 0
.
.
.
SessionIDLocaluuid=db248b6cbdc547bbc6c6fdfb6916eeb
SessionIDRemoteuuid=4fd24d9121935531a7f8d750ad16e19
.
.
.
SCCP call-legs: 0
Multicast call-legs: 0
Total call-legs: 1
```

You can search for session id using wildcard pattern match as shown below:

```
Device# Device# show call active voice session-id 4fd2*
Telephony call-legs: 0
SIP call-legs: 2
H323 call-legs: 0
.
.
.
SessionIDLocaluuid=4fd24d9121935531a7f8d750ad16e19
SessionIDRemoteuuid=db248b6cbdc547bbc6c6fdfb6916eeb
SessionIDLocaluuid=db248b6cbdc547bbc6c6fdfb6916eeb
SessionIDRemoteuuid=4fd24d9121935531a7f8d750ad16e19
```

```
.
.
.
SCCP call-legs: 0
Multicast call-legs: 0
Total call-legs: 2


Device# show call active voice session-id *f*16e*
Telephony call-legs: 0
SIP call-legs: 2
H323 call-legs: 0
.
.
.
SessionIDLocaluuid=4fd24d9121935531a7f8d750ad16e19
SessionIDRemoteuuid=db248b6cbdc547bbc6c6fdfb6916eeb
SessionIDLocaluuid=db248b6cbdc547bbc6c6fdfb6916eeb
SessionIDRemoteuuid=4fd24d9121935531a7f8d750ad16e19
.
.
.
SCCP call-legs: 0
Multicast call-legs: 0
Total call-legs: 2


Device# show call active voice brief session-id *
Telephony call-legs: 0
SIP call-legs: 2
H323 call-legs: 0
.
.
.
SessionIDLocaluuid=4fd24d9121935531a7f8d750ad16e19
SessionIDRemoteuuid=db248b6cbdc547bbc6c6fdfb6916eeb
SessionIDLocaluuid=db248b6cbdc547bbc6c6fdfb6916eeb
SessionIDRemoteuuid=4fd24d9121935531a7f8d750ad16e19
.
.
.
SCCP call-legs: 0
Multicast call-legs: 0
Total call-legs: 2


Device# show call active voice session-id *; remote=*
Telephony call-legs: 0
SIP call-legs: 2
H323 call-legs: 0
.
.
.
SessionIDLocaluuid=4fd24d9121935531a7f8d750ad16e19
SessionIDRemoteuuid=db248b6cbdc547bbc6c6fdfb6916eeb
SessionIDLocaluuid=db248b6cbdc547bbc6c6fdfb6916eeb
SessionIDRemoteuuid=4fd24d9121935531a7f8d750ad16e19
.
.
.
SCCP call-legs: 0
Multicast call-legs: 0
Total call-legs: 2
```

```
Device# show call active voice session-id 4fd24d9*;remote=*16eeb
Telephony call-legs: 0
SIP call-legs: 2
H323 call-legs: 0
.
.
.
SessionIDLocaluuid=4fd24d9121935531a7f8d750ad16e19
SessionIDRemoteuuid=db248b6cbdc547bbc6c6fdfb6916eeb
.
.
.
SCCP call-legs: 0
Multicast call-legs: 0
Total call-legs: 2
```

Example: Invalid Search Patterns

The following wild card search patterns are invalid:

```
Device# show call active voice session-id ;remote=
Invalid Pattern. Pattern can have a string with ^[0-9a-fA-F*]+$ only  OR a string with
^[0-9a-fA-F*];remote=[0-9a-fA-F*]+$.


Device# show call active voice session-id *;remote=
Invalid Pattern. Pattern can have a string with ^[0-9a-fA-F*]+$ only  OR a string with
^[0-9a-fA-F*];remote=[0-9a-fA-F*]+$.


Device# show call active video session-id ;remote=*
Incorrect format for Session-ID Wildcard Pattern regular expression must be of the form
^[0-9A-Fa-f*]+$
Invalid Pattern. Pattern can have a string with ^[0-9a-fA-F*]+$ only  OR a string with
^[0-9a-fA-F*];remote=[0-9a-fA-F*]+$.


Device# show call active voice session-id 4fd24d9*remote=*16eeb
Incorrect format for Session-ID Wildcard Pattern regular expression must be of the form
^[0-9A-Fa-f*]+$
Invalid Pattern. Pattern can have a string with ^[0-9a-fA-F*]+$ only  OR a string with
^[0-9a-fA-F*];remote=[0-9a-fA-F*]+$.
```

Example: Search using Null session identifier

If one of the session identifier is null, you can search for the session identifiers using 0 as wildcard pattern. The following session identifier is considered in the below example:

```
SessionIDLocaluuid=00000000000000000000000000000000
SessionIDRemoteuuid=4fd24d9121935531a7f8d750ad16e19


Device# show call active voice session-id 0
Telephony call-legs: 0
SIP call-legs: 2
H323 call-legs: 0
.
.
.
SessionIDLocaluuid=00000000000000000000000000000000
SessionIDRemoteuuid=4fd24d9121935531a7f8d750ad16e19
.
.
```

```
.
SCCP call-legs: 0
Multicast call-legs: 0
Total call-legs: 2
```

Example: Correlation between Session Identifier and Call Identifier

The following session identifier is considered in the below examples:

```
SessionIDLocaluuid=db248b6cbdc547bbc6c6fdfb6916eeb
SessionIDRemoteuuid=4fd24d9121935531a7f8d750ad16e19
```

You can search for session identifier using the local UUID as shown below:

```
Device# show call active voice session-id d82c680a3eaecd5c29ac6ceeaa225061
Telephony call-legs: 0
SIP call-legs: 2
H323 call-legs: 0
Call agent controlled call-legs: 0
SCCP call-legs: 0
Multicast call-legs: 0
.
.
.
.
VOIP:
ConnectionId[0x8CDAC180 0x10000 0x1B7 0x5B56400A]
IncomingConnectionId[0x8CDAC180 0x10000 0x1B7 0x5B56400A]
CallID=1022
GlobalCallId=[0xC3DAB665 0x770C11E5 0x80318550 0x5A000ED7]
SessionIDLocaluuid=d82c680a3eaecd5c29ac6ceeaa225061
SessionIDRemoteuuid=6497636d0b747785241cfbf5aa225064
CallReferenceId=0
CallServiceType=Unknown
RTP Loopback Call=FALSE
RemoteIPAddress=10.64.86.91
RemoteUDPPort=16614
RemoteSignallingIPAddress=10.64.86.91
RemoteSignallingPort=5060
RemoteMediaIPAddress=10.127.17.142
RemoteMediaPort=16614
CoderTypeRate=g711ulaw
.
.
.
.
GlobalCallId=[0xC3DAB665 0x770C11E5 0x80318550 0x5A000ED7]
SessionIDLocaluuid=6497636d0b747785241cfbf5aa225064
SessionIDRemoteuuid=d82c680a3eaecd5c29ac6ceeaa225061
RemoteIPAddress=10.64.86.91
RemoteUDPPort=21978
RemoteSignallingIPAddress=10.64.86.91
RemoteSignallingPort=5060
RemoteMediaIPAddress=10.127.17.188
RemoteMediaPort=21978
```

From the above output, you get to know that 1022 (highlighted) is the call identifier associated with the local session identifier **d82c680a3eaecd5c29ac6ceeaa225061**. You can now use this call identifier to get further details and debugging of the desired call as shown below:

```
Device# show sip-ua calls callid 1022
```

```
SIP CALL INFO of CCAPI callid 1022
Call 1
SIP Call ID                : 8cdac180-627159d8-9cd-5b56400a@10.64.86.91
   State of the call       : STATE_ACTIVE (7)
   Substate of the call    : SUBSTATE_NONE (0)
   Calling Number          : 4443332212
   Called Number           : 4443332211
   Called URI              : sip:4443332211@10.64.86.132:5060
   Bit Flags               : 0xC0401C 0x10000100 0x80004
   CC Call ID              : 1022
   Source IP Address (Sig ): 10.64.86.132
   Destn SIP Req Addr:Port : [10.64.86.91]:5060
   Destn SIP Resp Addr:Port: [10.64.86.91]:5060
   Destination Name        : 10.64.86.91
   Number of Media Streams : 1
   Number of Active Streams: 1
   RTP Fork Object         : 0x0
   Media Mode              : flow-through
   Media Stream 1
     State of the stream      : STREAM_ACTIVE
     Stream Call ID           : 1022
     Stream Type              : voice-only (0)
     Stream Media Addr Type   : 1
     Negotiated Codec         : g711ulaw (160 bytes)
     Codec Payload Type       : 0
     Negotiated Dtmf-relay    : inband-voice
     Dtmf-relay Payload Type  : 0
     QoS ID                   : -1
     Local QoS Strength       : BestEffort
     Negotiated QoS Strength  : BestEffort
     Negotiated QoS Direction : None
     Local QoS Status         : None
     Media Source IP Addr:Port: [10.64.86.132]:16424
     Media Dest IP Addr:Port  : [10.127.17.142]:16614


Options-Ping    ENABLED:NO    ACTIVE:NO

SIP CALL INFO of peer leg CCAPI callid 1023
Call 2
SIP Call ID                : C3DEFC15-770C11E5-80348550-5A000ED7@10.64.86.132
   State of the call       : STATE_ACTIVE (7)
   Substate of the call    : SUBSTATE_NONE (0)
   Calling Number          : 4443332212
   Called Number           : 4443332211
   Called URI              : sip:4443332211@10.64.86.91:5060
   Bit Flags               : 0xC04018 0x90000100 0x80080
   CC Call ID              : 1023
   Source IP Address (Sig ): 10.64.86.132
   Destn SIP Req Addr:Port : [10.64.86.91]:5060
   Destn SIP Resp Addr:Port: [10.64.86.91]:5060
   Destination Name        : 10.64.86.91
   Number of Media Streams : 1
   Number of Active Streams: 1
   RTP Fork Object         : 0x0
   Media Mode              : flow-through
   Media Stream 1
     State of the stream      : STREAM_ACTIVE
     Stream Call ID           : 1023
     Stream Type              : voice-only (0)
     Stream Media Addr Type   : 1
     Negotiated Codec         : g711ulaw (160 bytes)
     Codec Payload Type       : 0
     Negotiated Dtmf-relay    : inband-voice
```

```
Dtmf-relay Payload Type  : 0
QoS ID                   : -1
Local QoS Strength       : BestEffort
Negotiated QoS Strength  : BestEffort
Negotiated QoS Direction : None
Local QoS Status         : None
Media Source IP Addr:Port: [10.64.86.132]:16426
Media Dest IP Addr:Port  : [10.127.17.188]:21978
```

Example for video Calls

The following session identifier is considered in the below example:

```
SessionIDLocaluuid=6f0a93a3a79451aebeb6d83f79a3359f
SessionIDRemoteuuid=a55b0f45861551b88f57d1fb5bb23f89
```

Note All the search patterns listed above for voice calls are also valid for video calls.

You can search for the session identifier using complete UUID (local, remote, or both) or use a wildcard pattern.

```
Device# show call active video session-id 6f*

Telephony call-legs: 0
SIP call-legs: 2
H323 call-legs: 0
Call agent controlled call-legs: 0
SCCP call-legs: 0
Multicast call-legs: 0
Total call-legs: 2

GENERIC:
SetupTime=56399650 ms (*16:58:12.964 IST Thu Aug 20 2015)
Index=1
PeerAddress=sipp
PeerSubAddress=
PeerId=1
PeerIfIndex=14
LogicalIfIndex=0
ConnectTime=56400660 ms (*16:58:13.974 IST Thu Aug 20 2015)
CallDuration=00:00:56 sec
CallState=4
CallOrigin=2
ChargedUnits=0
InfoType=video
TransmitPackets=0
TransmitBytes=0
ReceivePackets=0
ReceiveBytes=0
VOIP:
ConnectionId[0x6083CB92 0x466511E5 0xFFFFFFFF8018F617 0xFFFFFFFFA7C45A02]
IncomingConnectionId[0x6083CB92 0x466511E5 0xFFFFFFFF8018F617 0xFFFFFFFFA7C45A02]
CallID=11
GlobalCallId=[0x6083F24F 0x466511E5 0xFFFFFFFF801BF617 0xFFFFFFFFA7C45A02]
CallReferenceId=0
CallServiceType=Unknown
RTP Loopback Call=FALSE
SessionIDLocaluuid=6f0a93a3a79451aebeb6d83f79a3359f
SessionIDRemoteuuid=a55b0f45861551b88f57d1fb5bb23f89
RemoteIPAddress=10.64.86.70
```

```
RemoteSignallingIPAddress=10.64.86.70
RemoteSignallingPort=5061
RemoteMediaIPAddress=10.64.86.70
RemoteMediaPort=6003
RoundTripDelay=0 ms
tx_DtmfRelay=inband-voice
FastConnect=FALSE
```