

Understanding codec negotiation

Applies to VoipNow Professional, VoipNow 3, VoipNow 3.5 and newer!

Overview

A successful codec negotiation is an essential condition for a successful call between two parties. At the same time, an audio stream cannot be established between two parties unless a mutually supported codec is found and used. The supported codecs are published during the SIP setup of the call; they are mentioned in the SDP part of the INVITE/200 OK exchange.

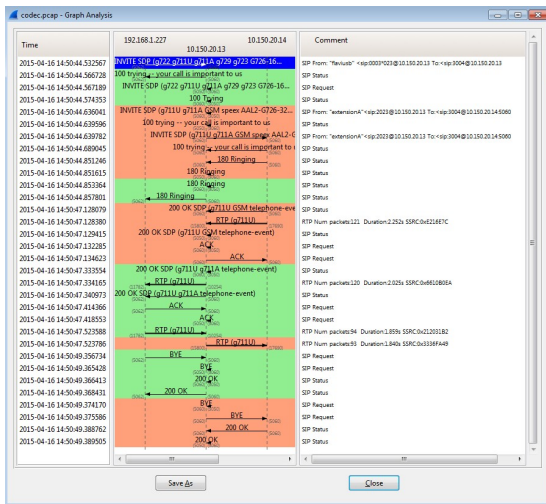
In VoipNow's case, the voice passes through Asterisk so an extra negotiation takes place.

Codec negotiation in VoipNow

As mentioned above, because the audio path includes Asterisk, an extra negotiation occurs. The SIP signalling also passes through Kamailio.

In the case of a call from the internal (private) network to the outside (public) network, the flow of the SIP signalling is as follows: **Internal caller >> Kamailio >> Asterisk >> Kamailio >> External callee.**

Here is a Wireshark capture of the flow (click to enlarge):



The following parties are involved:

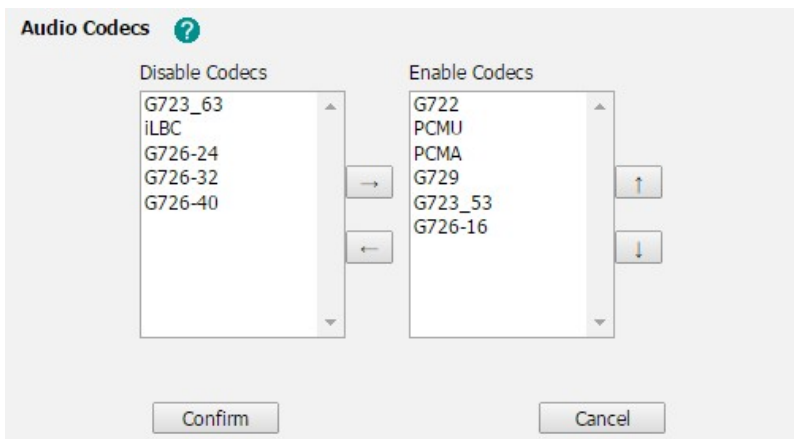
Caller: extensionA 0003*023 (public number 2023) with IP 192.168.1.227

VoipNow with IP 10.150.20.13 with Kamailio on port 5060 and asterisk on port 5050

Callee: phone number 3004 with IP 10.150.20.14

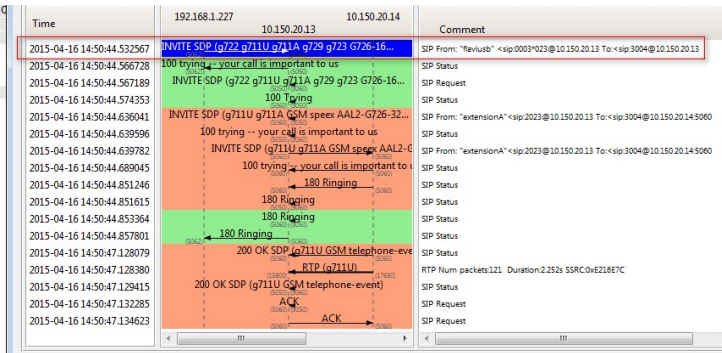
Let's look at some of the codec settings:

The caller is a Yealink device with the following codecs activated:



The device publishes these codecs in the SDP of the INVITE sent from 192.168.1.227:

```
Internet Protocol Version 4, Src: 192.168.1.227 (192.168.1.227), Dst: 10.150.20.13 (10.150.20.13)
User Datagram Protocol, Src Port: nala-lisise (5062), Dst Port: sip (5060)
Session Initiation Protocol (INVITE)
Request-Line: INVITE sip:3004@10.150.20.13:5060 SIP/2.0
Message Header
Message Body
Session Description Protocol
Session Description Protocol Version (v): 0
Owner/Creator, Session Id (o): - 20001 20001 IN IP4 192.168.1.227
Session Name (s): SDP data
Connection Information (c): IN IP4 192.168.1.227
Time Description, active time (t): 0 0
Media Description, name and address (m): audio 11782 RTP/AVP 9 0 8 18 4 112 101
Media Attribute (a): rtptime:9 G722/8000
Media Attribute (a): rtptime:8 PCMA/8000
Media Attribute (a): rtptime:18 G729/8000
Media Attribute (a): fmtp:18 annexb=no
Media Attribute (a): rtptime:4 G723/8000
Media Attribute (a): rtptime:112 G726-16/8000
Media Attribute (a): fmtp:101 0-15
Media Attribute (a): rtptime:101 telephone-event/8000
Media Attribute (a): pttime:20
Media Attribute (a): sendrecv
```



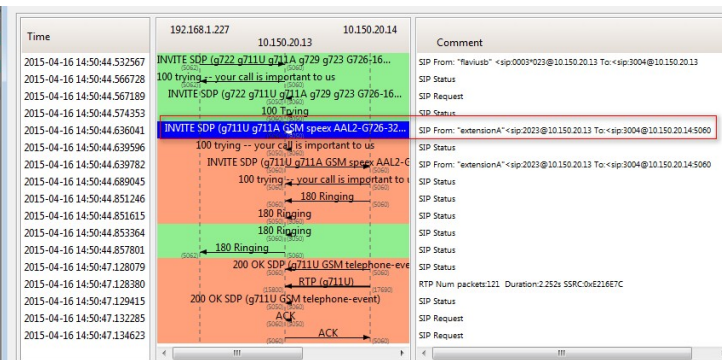
This INVITE reaches Kamailio and then is forwarded to Asterisk. The codecs remain unchanged and Asterisk recognizes the codecs supported by the caller.

Next, Asterisk initiates an INVITE towards the callee at address 10.150.20.14. It does so through a SIP channel using the following codecs:

Channel codecs *	<input checked="" type="checkbox"/> G.723	<input checked="" type="checkbox"/> G.711 u-Law	<input checked="" type="checkbox"/> G.711 A-Law	<input checked="" type="checkbox"/> GSM	<input type="checkbox"/> G.726 (RFC)
	<input type="checkbox"/> slin	<input type="checkbox"/> G.729a	<input checked="" type="checkbox"/> speex	<input type="checkbox"/> iLBC	<input type="checkbox"/> LPC-10
	<input type="checkbox"/> H.261	<input type="checkbox"/> H.263	<input type="checkbox"/> H.263 enhanced	<input checked="" type="checkbox"/> G.726	<input type="checkbox"/> H.264
	<input checked="" type="checkbox"/> G.722	<input type="checkbox"/> T.140 (redundant)	<input type="checkbox"/> T.140		

The INVITE containing all the codecs enabled on the channel looks like this:

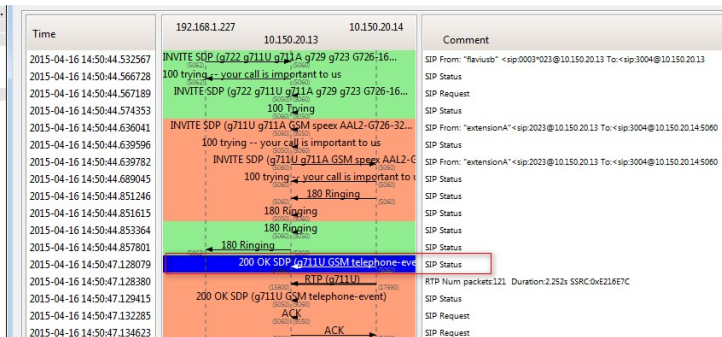
```
Internet Protocol Version 4, Src: 10.150.20.13 (10.150.20.13), Dst: 10.150.20.14 (10.150.20.14)
User Datagram Protocol, Src Port: mmcs (5050), Dst Port: sip (5060)
Session Initiation Protocol (INVITE)
Request-Line: INVITE sip:3004@10.150.20.14:5060 SIP/2.0
Message Header
Message Body
Session Description Protocol
Session Description Protocol Version (v): 0
Owner/Creator, Session Id (o): root 733891886 733891886 IN IP4 10.150.20.13
Session Name (s): VoIPNow
Connection Information (c): IN IP4 10.150.20.14
Time Description, active time (t): 0 0
Media Description, name and address (m): audio 15800 RTP/AVP 0 8 3 110 112 9 101
Media Attribute (a): rtptime:0 PCMU/8000
Media Attribute (a): rtptime:8 PCMA/8000
Media Attribute (a): rtptime:3 GSM/8000
Media Attribute (a): rtptime:110 speex/8000
Media Attribute (a): rtptime:112 AAL2-G726-32/8000
Media Attribute (a): rtptime:9 G722/8000
Media Attribute (a): rtptime:101 telephone-event/8000
Media Attribute (a): fmtp:101 0-16
Media Attribute (a): pttime:20
Media Attribute (a): sendrecv
```



The INVITE is sent to Kamailio which forwards it to the callee at 10.150.20.14 without changing the codecs. At this point, the callee is informed about the codecs supported by VoipNow.

The callee will now answer the call with an 200 OK message. This message will contain an SDP mentioning the codecs it supports:

```
Internet Protocol Version 4, Src: 10.150.20.14 (10.150.20.14), Dst: 10.150.20.13 (10.150.20.13)
User Datagram Protocol, Src Port: sip (5060), Dst Port: sip (5060)
Session Initiation Protocol (200)
Status-Line: SIP/2.0 200 OK
Message Header
Message Body
Session Description Protocol
Session Description Protocol Version (v): 0
Owner/Creator, Session Id (o): root 1429640280 1429640280 IN IP4 10.150.20.14
Session Name (s): VoIPNow
Connection Information (c): IN IP4 10.150.20.14
Time Description, active time (t): 0 0
Media Description, name and address (m): audio 17690 RTP/AVP 0 3 101
Media Attribute (a): rtptime:0 PCMU/8000
Media Attribute (a): rtptime:3 GSM/8000
Media Attribute (a): rtptime:101 telephone-event/8000
Media Attribute (a): fmtp:101 0-16
Media Attribute (a): pttime:20
Media Attribute (a): sendrecv
```



Now both VoipNow and the callee at 3004 know the codecs supported by the other parties. As you can see, the only common codec supported by both parties is G711 u-law (PCMU). This means G711 will be used for the voice stream between VoipNow and the callee:

Time	192.168.1.227	10.150.20.13	10.150.20.14	Comment
2015-04-16 14:50:44.639782		INVITE SDP (g711U g711A GSM speex AAL2-G		SIP From: "extensionA" <sip:2023@10.150.20.13> To: <sip:3004@10.150.20.14>
2015-04-16 14:50:44.689045		100 trying - your call is important to		SIP Status
2015-04-16 14:50:44.851246		180 Ringing		SIP Status
2015-04-16 14:50:44.851615		180 Ringing		SIP Status
2015-04-16 14:50:44.853364		180 Ringing		SIP Status
2015-04-16 14:50:44.857801		180 Ringing		SIP Status
2015-04-16 14:50:47.128079		200 OK SDP (g711U GSM telephone-eve		SIP Status
2015-04-16 14:50:47.128380		RTP (g711U)		RTP Num packets:121 Duration:2.252s SSRC:0xE216E7C
2015-04-16 14:50:47.129415		200 OK SDP (g711U GSM telephone-event)		SIP Status
2015-04-16 14:50:47.132285		ACK		SIP Request
2015-04-16 14:50:47.134623		ACK		SIP Request
2015-04-16 14:50:47.333554		200 OK SDP (g711U g711A telephone-event)		SIP Status
2015-04-16 14:50:47.334165		RTP (g711U)		RTP Num packets:120 Duration:2.025s SSRC:0x6610B0EA
2015-04-16 14:50:47.340973		200 OK SDP (g711U g711A telephone-event)		SIP Status
2015-04-16 14:50:47.414366		ACK		SIP Request
2015-04-16 14:50:47.418553		ACK		SIP Request
2015-04-16 14:50:47.523588		RTP (g711U)		RTP Num packets:94 Duration:1.859s SSRC:0x212031B2
2015-04-16 14:50:47.523786		RTP (g711U)		RTP Num packets:93 Duration:1.840s SSRC:0x3336FA49
2015-04-16 14:50:49.356734		BYE		SIP Request

Now it's time for Asterisk to send the 200 OK to the caller to let him know that the call was answered. This 200 OK message will also contain the codecs supported by VoipNow for the communication with the caller.

Note that these codecs do not have to be the same as for the other leg of the call.

These codecs are defined under the Provisioning and SIP menu of the 0003*023 extension:

Allowed codecs *

☒ G.711 u-Law
☒ G.711 A-Law
☐ GSM
☐ G.726 (RFC)
☐ G.729a

☐ iLBC

The resulting 200 OK SDP will look like this:

Time	192.168.1.227	10.150.20.13	10.150.20.14	Comment
2015-04-16 14:50:44.639782		INVITE SDP (g711U g711A GSM speex AAL2-G		SIP From: "extensionA" <sip:2023@10.150.20.13> To: <sip:3004@10.1
2015-04-16 14:50:44.689045		100 trying - your call is important to		SIP Status
2015-04-16 14:50:44.851246		180 Ringing		SIP Status
2015-04-16 14:50:44.851615		180 Ringing		SIP Status
2015-04-16 14:50:44.853364		180 Ringing		SIP Status
2015-04-16 14:50:44.857801		180 Ringing		SIP Status
2015-04-16 14:50:47.128079		200 OK SDP (g711U GSM telephone-eve		SIP Status
2015-04-16 14:50:47.128380		RTP (g711U)		RTP Num packets:121 Duration:2.252s SSRC:0xE216E7C
2015-04-16 14:50:47.129415		200 OK SDP (g711U GSM telephone-event)		SIP Status
2015-04-16 14:50:47.132285		ACK		SIP Request
2015-04-16 14:50:47.134623		ACK		SIP Request
2015-04-16 14:50:47.333554		200 OK SDP (g711U g711A telephone-event)		SIP Status
2015-04-16 14:50:47.334165		RTP (g711U)		RTP Num packets:120 Duration:2.025s SSRC:0x6610B0EA
2015-04-16 14:50:47.340973		200 OK SDP (g711U g711A telephone-event)		SIP Status
2015-04-16 14:50:47.414366		ACK		SIP Request
2015-04-16 14:50:47.418553		ACK		SIP Request
2015-04-16 14:50:47.523588		RTP (g711U)		RTP Num packets:94 Duration:1.859s SSRC:0x212031B2
2015-04-16 14:50:47.523786		RTP (g711U)		RTP Num packets:93 Duration:1.840s SSRC:0x3336FA49
2015-04-16 14:50:49.356734		BYE		SIP Request


Now both the caller (Yealink at 192.168.1.227) and VoipNow know the codecs supported by the other party. In this case, we have two common codecs, G711 u-law and G711 a-law (PCMU and PCMA). PCMU has a higher priority (it comes before PCMA in the SDP codec list), so it will be used for the voice stream between the caller and VoipNow:

Time	192.168.1.227	10.150.20.13	10.150.20.14	Comment
2015-04-16 14:50:47.128380		RTP (g711U)		RTP Num packets:121 Duration:2.252s SSRC:0xE216E7C
2015-04-16 14:50:47.129415	200 OK SDP (g711U GSM telephone-event)			SIP Status
2015-04-16 14:50:47.132285		ACK		SIP Request
2015-04-16 14:50:47.134623		ACK		SIP Request
2015-04-16 14:50:47.333554	200 OK SDP (g711U g711A telephone-event)			SIP Status
2015-04-16 14:50:47.334165		RTP (g711U)		RTP Num packets:120 Duration:2.025s SSRC:0x6610B0EA
2015-04-16 14:50:47.340973	200 OK SDP (g711U g711A telephone-event)			SIP Status
2015-04-16 14:50:47.414366		ACK		SIP Request
2015-04-16 14:50:47.418553		ACK		SIP Request
2015-04-16 14:50:47.523588		RTP (g711U)		RTP Num packets:94 Duration:1.859s SSRC:0x212031B2
2015-04-16 14:50:47.523786		RTP (g711U)		RTP Num packets:93 Duration:1.840s SSRC:0x3336FA49
2015-04-16 14:50:49.356734	BYE			SIP Request
2015-04-16 14:50:49.365428		BYE		SIP Request
2015-04-16 14:50:49.366413		200 OK		SIP Status
2015-04-16 14:50:49.368431	200 OK			SIP Status
2015-04-16 14:50:49.374170		BYE		SIP Request
2015-04-16 14:50:49.375586		BYE		SIP Request
2015-04-16 14:50:49.388762		200 OK		SIP Status
2015-04-16 14:50:49.389505		200 OK		SIP Status

The voice call is finally established with 4 streams, all on G711 u-law: Caller <<->> VoipNow and Voipnow <<->> Callee, both bidirectional.

The two legs of the call can use different streams. This means that Asterisk will perform transcoding (which also uses a bit more CPU)

Other considerations

 The codecs that you can check/uncheck in for the extensions can be selected from **Unified Communications >> Zero Priority >> SIP** area.

Related articles

- [Understanding codec negotiation](#)
- [How to monitor VoipNow with Homer](#)
- [Troubleshooting calls and debug steps](#)
- [How to check if an extension is registered](#)
- [How to debug Asterisk and Kamailio](#)