



Gonzalo Escarrá's Blog

Lync, networking, UC, VoIP, and some other hacks.

T.38 Fax over IP call on Wireshark

Posted on March 22, 2016 by Gonzalo Escarrá

Ever wondered what a proper T.38 Fax over IP (FoIP) transmission looks like running through Wireshark? Maybe you're troubleshooting a call flow, or never seen a T.38 capture. Below I'll try to explain the call flow and steps to look out for when troubleshooting T.38 calls. Here's an Outbound FAX call originating from a FXS port in a Cisco CUBE, and going towards Flowroute.

- Initial SIP INVITE and early media receipt (ringback). Note this is all RTP.

9	35.421673	50.198.1.182	216.115.69.144	SIP/SDP	26	1234 Request: INVITE sip:11111111111111111111111111111111@sip.flowroute.com:5060
10	35.497817	216.115.69.144	50.198.1.182	SIP	8	327 Status: 100 Trying
11	35.844063	199.199.10.15	50.198.1.182	RTP	8	214 PT=ITU-T G.711 PCMU, SSRC=0x5B34EE4D, Seq=24389, Time=2943698477
12	35.862483	199.199.10.15	50.198.1.182	RTP	8	214 PT=ITU-T G.711 PCMU, SSRC=0x5B34EE4D, Seq=24390, Time=2943698637
13	35.881938	199.199.10.15	50.198.1.182	RTP	8	214 PT=ITU-T G.711 PCMU, SSRC=0x5B34EE4D, Seq=24391, Time=2943698797
14	35.903469	199.199.10.15	50.198.1.182	RTP	8	214 PT=ITU-T G.711 PCMU, SSRC=0x5B34EE4D, Seq=24392, Time=2943698957
15	35.921874	199.199.10.15	50.198.1.182	RTP	8	214 PT=ITU-T G.711 PCMU, SSRC=0x5B34EE4D, Seq=24393, Time=2943699117
16	35.942393	199.199.10.15	50.198.1.182	RTP	8	214 PT=ITU-T G.711 PCMU, SSRC=0x5B34EE4D, Seq=24394, Time=2943699277
17	35.962292	199.199.10.15	50.198.1.182	RTP	8	214 PT=ITU-T G.711 PCMU, SSRC=0x5B34EE4D, Seq=24395, Time=2943699437

- SDP from the INVITE shows media offered is all voice (RTP)
 - Session Description Protocol
 - Session Description Protocol Version (v): 0
 - Owner/Creator, Session Id (o): CiscoSystemsSIP-GW-UserAgent 217 1006 IN IP4 172.20.4.10
 - Session Name (s): SIP Call
 - Connection Information (c): IN IP4 50.198.1.182
 - Time Description, active time (t): 0 0
 - Media Description, name and address (m): audio 18824 RTP/AVP 0 101 13
 - Connection Information (c): IN IP4 50.198.1.182
 - Media Attribute (a): rtpmap:0 PCMU/8000
 - Media Attribute (a): rtpmap:101 telephone-event/8000
 - Media Attribute (a): fmp:101 0-16
 - Media Attribute (a): rtpmap:13 CN/8000
 - Media Attribute (a): pt:time:20
- 183 Session in Progress, and we start sending media too (again, RTP). Later on comes the 200 OK, meaning the call was answered on the remote end.

31	36.230910	216.115.69.144	50.198.1.182	SIP/SDP	8	705 Status: 183 Session Progress
32	36.245187	199.199.10.15	50.198.1.182	RTP	8	214 PT=ITU-T G.711 PCMU, SSRC=0x5B34EE4D, Seq=24409, Time=2943693677
33	36.262049	199.199.10.15	50.198.1.182	RTP	8	214 PT=ITU-T G.711 PCMU, SSRC=0x5B34EE4D, Seq=24410, Time=2943693837
34	36.279528	50.198.1.182	199.199.10.15	RTP	46	214 PT=ITU-T G.711 PCMU, SSRC=0x0B2040A0, Seq=5932, Time=626046988
35	36.282447	199.199.10.15	50.198.1.182	RTP	8	214 PT=ITU-T G.711 PCMU, SSRC=0x5B34EE4D, Seq=24411, Time=2943693997
36	36.300663	50.198.1.182	199.199.10.15	RTP	46	214 PT=ITU-T G.711 PCMU, SSRC=0x0B2040A0, Seq=5933, Time=626041148
37	36.303827	199.199.10.15	50.198.1.182	RTP	8	214 PT=ITU-T G.711 PCMU, SSRC=0x5B34EE4D, Seq=24412, Time=2943694157
38	36.321653	50.198.1.182	199.199.10.15	RTP	46	214 PT=ITU-T G.711 PCMU, SSRC=0x0B2040A0, Seq=5934, Time=626041308
39	36.324336	199.199.10.15	50.198.1.182	RTP	8	214 PT=ITU-T G.711 PCMU, SSRC=0x5B34EE4D, Seq=24413, Time=2943694317
40	36.339285	50.198.1.182	199.199.10.15	RTP	46	214 PT=ITU-T G.711 PCMU, SSRC=0x0B2040A0, Seq=5935, Time=626041468

- Things changing now... in-dialog (RE)INVITE from Cisco CUBE to SIP trunk... RTP and T.38 packets mixed because

Recent Posts

- ["On a Call" light notification for the new COVID-19 world June 22, 2020](#)
- [Tesla, Smart Meters and Automation via APIs December 9, 2018](#)
- [Hooking up Twilio SIP to Skype for Business June 15, 2017](#)
- [LS Data MCU events 41025 and 41026 starting in May-June? June 13, 2017](#)
- [Skype Front End not starting on dual-homed VM August 22, 2016](#)

Calendar

January 2021

M	T	W	T	F	S	S
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

<< Jun

the remote end has not accepted our INVITE yet, but we start sending media either way.

Subscribe to Blog via Email

Enter your email address to subscribe to this blog and receive notifications of new posts by email.

Email Address

Subscribe

- And the SDP of the new INVITE now shows all T.38 media now.

```

Session Initiation Protocol (INVITE)
  > Request-Line: INVITE sip:8553922666@199.199.10.4:5060 SIP/2.0
  > Message Header
  > Message Body
    > Session Description Protocol
      > Session Description Protocol Version (v): 0
      > Owner/Creator, Session Id (o): CiscoSystemsSIP-GW-UserAgent 217 1007 IN IP4 172.20.4.10
      > Session Name (s): SIP Call
      > Connection Information (c): IN IP4 50.198.1.182
      > Time Description, active time (t): 0 0
      > Media Description, name and address (m): image 18824 udptl t38
      > Connection Information (c): IN IP4 50.198.1.182
      > Media Attribute (a): T38FaxVersion:3
      > Media Attribute (a): T38MaxBitRate:33600
      > Media Attribute (a): T38FaxFillBitRemoval:0
      > Media Attribute (a): T38FaxTranscodingMMR:0
      > Media Attribute (a): T38FaxTranscodingJBIG:0
      > Media Attribute (a): T38FaxRateManagement:transferredTCF
      > Media Attribute (a): T38FaxMaxBuffer:200
      > Media Attribute (a): T38FaxMaxDatagram:320
      > Media Attribute (a): T38FaxUdpEC:t38UDPRedundancy
  
```

- Once we get the 200 OK from Flowroute, it's all T.38 media both ways.

```

519 43.728484 216.115.69.144 50.198.1.182 SIP/SDP 8 692 Status: 200 OK |
520 43.731731 50.198.1.182 199.199.10.15 T.38 46 214 UDP: UDPTLPacket Seq=32768 t30ind: v17-12000-short-training hdlc-fcs-BAD-sig-end
521 43.735199 50.198.1.182 199.199.10.15 RTCP 26 110 Receiver Report Source description Goodbye
522 43.737412 50.198.1.182 216.115.69.144 SIP 26 532 Request: ACK sip:8553922666@199.199.10.4:5060 |
523 43.760049 50.198.1.182 199.199.10.15 T.38 46 48 UDP: UDPTLPacket Seq=00000 t30ind: no-signal
524 43.820190 50.198.1.182 199.199.10.15 T.38 46 48 UDP: UDPTLPacket Seq=00000 t30ind: no-signal
525 43.900336 50.198.1.182 199.199.10.15 T.38 46 48 UDP: UDPTLPacket Seq=00000 t30ind: no-signal
526 43.949507 50.198.1.182 199.199.10.15 T.38 46 50 UDP: UDPTLPacket Seq=00001 t30ind: v21-preamble
527 44.019622 50.198.1.182 199.199.10.15 T.38 46 48 UDP: UDPTLPacket Seq=00001 t30ind: v21-preamble
528 44.089665 50.198.1.182 199.199.10.15 T.38 46 48 UDP: UDPTLPacket Seq=00001 t30ind: v21-preamble
529 44.159750 50.198.1.182 199.199.10.15 T.38 46 48 UDP: UDPTLPacket Seq=00001 t30ind: v21-preamble
530 44.228027 50.198.1.182 199.199.10.15 T.38 46 48 UDP: UDPTLPacket Seq=00001 t30ind: v21-preamble
531 44.259948 50.198.1.182 199.199.10.15 T.38 46 48 UDP: UDPTLPacket Seq=00001 t30ind: v21-preamble
  
```

- Now the flow gets interesting, more Fax-ey. Wireshark will decode the HDLC data and show interesting bits here
 - TSI, is our Fax station number programmed in the machine.

```

61 UDP: UDPTLPacket Seq=00024 data:v21: hdlc-data[04] hdlc-fcs-OK (HDLC Reassembled: TSI - Transmitting Subscriber Identification - Number:1839615000)
  
```

- DCS, our Fax machine communicates the capabilities, and starts training.

```

61 UDP: UDPTLPacket Seq=00031 data:v21: hdlc-data[00] hdlc-fcs-OK (HDLC Reassembled: DCS - Digital Command Signal - DSR:14 400 bit/s, ITU-T V.17)
58 UDP: UDPTLPacket Seq=00032 data:v21: hdlc-sig-end
52 UDP: UDPTLPacket Seq=00033 t30ind: no-signal
48 UDP: UDPTLPacket Seq=00033 t30ind: no-signal
50 UDP: UDPTLPacket Seq=00034 t30ind: v17-14400-long-training
48 UDP: UDPTLPacket Seq=00034 t30ind: v17-14400-long-training
48 UDP: UDPTLPacket Seq=00034 t30ind: v17-14400-long-training
48 UDP: UDPTLPacket Seq=00034 t30ind: v17-14400-long-training
48 UDP: UDPTLPacket Seq=00034 t30ind: v17-14400-long-training
48 UDP: UDPTLPacket Seq=00034 t30ind: v17-14400-long-training
48 UDP: UDPTLPacket Seq=00034 t30ind: v17-14400-long-training
48 UDP: UDPTLPacket Seq=00034 t30ind: v17-14400-long-training
48 UDP: UDPTLPacket Seq=00034 t30ind: v17-14400-long-training
  
```

- If we look inside the packet's data, our DCS has a lot more information about our Fax machine's settings

and resolution

ITU-T Recommendation T.30

Address: 0xff

Control: final frames within the procedure (0xc8)

.100 0001 = Facsimile Control: Digital Command Signal (65)

0... = Store and forward Internet fax- Simple mode (ITU-T T.37): Not set
..0. = Real-time Internet fax (ITU T T.38): Not set
...0 = 3rd Generation Mobile Network: Not set
.1.. = Receiver fax operation: Set
..00 01.. = Data signalling rate: 14 400 bit/s, ITU-T V.17 (0x01)
.... ..1. = R8x7.7 lines/mm and/or 200x200 pels/25.4 mm: Set
.... ...1 = Two dimensional coding capability: Set
00.. = Recording width: Scan line length 215 mm +- 1% (0x00)
..01 = Recording length capability: Unlimited (0x01)
.... 100. = Minimum scan line time: 05 ms (0x04)
.... ...1 = Extension indicator: information continues through the next octet
.0.. = Compress/Uncompress mode: Compressed mode
..0. = Error correction mode: Not set
...0 = Frame size: 256 octets
.... ..0. = T.6 coding capability: Not set
.... ...0 = Extension indicator: last octet

- Then we get an FTT, means the remote end “Failed to Train” . Not usually a sign something is wrong, but more a capability mismatch. The remote fax may accept only lower baud rates, and will fail to train any higher. This is normal unless it’ s the only response we get back from the remote end.

```
60 UDP: UDPTLPacket Seq=00005 data:v21: hdlc-fcs-ok-sig-end (HDLC Reassembled: FTT - Failure To Train)
60 UDP: UDPTLPacket Seq=00006 t30ind: no-signal
60 UDP: UDPTLPacket Seq=00006 t30ind: no-signal
60 UDP: UDPTLPacket Seq=00006 t30ind: no-signal
48 UDP: UDPTLPacket Seq=00074 t30ind: no-signal
48 UDP: UDPTLPacket Seq=00074 t30ind: no-signal
48 UDP: UDPTLPacket Seq=00074 t30ind: no-signal
```

- We see the same process of TSI, DCS and FTT until we hit the right baud rate... in our case it’ s 9600... Once we get that, we receive a CFR.

```
60 UDP: UDPTLPacket Seq=00017 data:v21: hdlc-fcs-ok-sig-end (HDLC Reassembled: CFR - Confirmation To Receive)
60 UDP: UDPTLPacket Seq=00018 t30ind: no-signal
60 UDP: UDPTLPacket Seq=00018 t30ind: no-signal
60 UDP: UDPTLPacket Seq=00018 t30ind: no-signal
48 UDP: UDPTLPacket Seq=00222 t30ind: no-signal
```

- Followed by a short training to sync-up and data (because we did long training before the CFR)

```
50 UDP: UDPTLPacket Seq=00223 t30ind: v17-9600-short-training
48 UDP: UDPTLPacket Seq=00223 t30ind: v17-9600-short-training
48 UDP: UDPTLPacket Seq=00223 t30ind: v17-9600-short-training
48 UDP: UDPTLPacket Seq=00223 t30ind: v17-9600-short-training
48 UDP: UDPTLPacket Seq=00223 t30ind: v17-9600-short-training
96 UDP: UDPTLPacket Seq=00224 data:v17-9600: t4-non-ecm-data[ffffffffffffff...]
148 UDP: UDPTLPacket Seq=00225 data:v17-9600: t4-non-ecm-data[ffffffffffffff...]
154 UDP: UDPTLPacket Seq=00226 data:v17-9600: t4-non-ecm-data[ffffffffffffff...]
154 UDP: UDPTLPacket Seq=00227 data:v17-9600: t4-non-ecm-data[ffffffffffffff...]
154 UDP: UDPTLPacket Seq=00228 data:v17-9600: t4-non-ecm-data[ffffffffffffff...]
154 UDP: UDPTLPacket Seq=00229 data:v17-9600: t4-non-ecm-data[ffffffffffffff...]
154 UDP: UDPTLPacket Seq=00230 data:v17-9600: t4-non-ecm-data[ffffffffffffff...]
154 UDP: UDPTLPacket Seq=00231 data:v17-9600: t4-non-ecm-data[ffffffffffffff...]
154 UDP: UDPTLPacket Seq=00232 data:v17-9600: t4-non-ecm-data[ffffffffffffff...]
154 UDP: UDPTLPacket Seq=00233 data:v17-9600: t4-non-ecm-data[ffffffffffffff...]
```

- And our actual FAX data which will vary

```
154 UDP: UDPTLPacket Seq=00567 data:v17-9600: t4-non-ecm-data[61c00017231d09...]
154 UDP: UDPTLPacket Seq=00568 data:v17-9600: t4-non-ecm-data[f0d907905a8000...]
154 UDP: UDPTLPacket Seq=00569 data:v17-9600: t4-non-ecm-data[3c000000000000...]
154 UDP: UDPTLPacket Seq=00570 data:v17-9600: t4-non-ecm-data[7b970a522c0ad5...]
154 UDP: UDPTLPacket Seq=00571 data:v17-9600: t4-non-ecm-data[9ae5938820da8b...]
154 UDP: UDPTLPacket Seq=00572 data:v17-9600: t4-non-ecm-data[00000016f1741b...]
154 UDP: UDPTLPacket Seq=00573 data:v17-9600: t4-non-ecm-data[db7800162d8bff...]
154 UDP: UDPTLPacket Seq=00574 data:v17-9600: t4-non-ecm-data[0000001690870d...]
154 UDP: UDPTLPacket Seq=00575 data:v17-9600: t4-non-ecm-data[77a4062e5bc1e6...]
154 UDP: UDPTLPacket Seq=00576 data:v17-9600: t4-non-ecm-data[a226e6e0000000...]
154 UDP: UDPTLPacket Seq=00577 data:v17-9600: t4-non-ecm-data[0000001664288b...]
154 UDP: UDPTLPacket Seq=00578 data:v17-9600: t4-non-ecm-data[735a404e76f436...]
154 UDP: UDPTLPacket Seq=00579 data:v17-9600: t4-non-ecm-data[be8caeee6e9f6a...]
154 UDP: UDPTLPacket Seq=00580 data:v17-9600: t4-non-ecm-data[00000015eb9da7...]
154 UDP: UDPTLPacket Seq=00581 data:v17-9600: t4-non-ecm-data[00000015aa5af0...]
154 UDP: UDPTLPacket Seq=00582 data:v17-9600: t4-non-ecm-data[9a96f000000000...]
154 UDP: UDPTLPacket Seq=00583 data:v17-9600: t4-non-ecm-data[1075c000000000...]
```

- At the end of the data, Wireshark reassembles the packets and tells us whether there was a loss or not. In our case, we' re good!

```
118 UDP: UDPTLPacket Seq=00674 data:v17-9600: t4-non-ecm-sig-end (t4-data Reassembled: No packet lost)[46e5f6e000000000...]
66 UDP: UDPTLPacket Seq=00675 t30ind: no-signal
48 UDP: UDPTLPacket Seq=00675 t30ind: no-signal
48 UDP: UDPTLPacket Seq=00675 t30ind: no-signal
48 UDP: UDPTLPacket Seq=00675 t30ind: no-signal
50 UDP: UDPTLPacket Seq=00676 t30ind: v21-preamble
48 UDP: UDPTLPacket Seq=00676 t30ind: v21-preamble
```

- We send an EOP to signal the end of the transmission

```
61 UDP: UDPTLPacket Seq=00679 data:v21: hdlc-data[f4] hdlc-fcs-OK (HDLC Reassembled: EOP - End Of Procedure)
```

- The remote end does an MCF to acknowledge receipt (this is how your Fax machine knows the fax is "good" on the other end)

```
60 UDP: UDPTLPacket Seq=00023 data:v21: hdlc-fcs-OK-sig-end (HDLC Reassembled: MCF - Message Confirmation)
```

- And then we send a DCN to logically hang up the HDLC stream, but we wait for the remote end...

```
61 UDP: UDPTLPacket Seq=00685 data:v21: hdlc-data[df] hdlc-fcs-OK (HDLC Reassembled: DCN - Disconnect)
```

- Remote end hangs up the call... and we' re done...

```
1600 94.717649 216.115.69.144 50.198.1.182 SIP 8 741 Request: BYE sip:1:0000000000000000@50.198.1.182:5060 |
1601 95.124657 216.115.69.144 50.198.1.182 SIP 8 741 Request: BYE sip:1:0000000000000000@50.198.1.182:5060 |
1602 95.919071 216.115.69.144 50.198.1.182 SIP 8 741 Request: BYE sip:1:0000000000000000@50.198.1.182:5060 |
1603 97.516790 50.198.1.182 216.115.69.144 SIP 26 674 Status: 200 OK |
1604 97.548261 216.115.69.144 50.198.1.182 SIP 8 741 Request: BYE sip:1:0000000000000000@50.198.1.182:5060 |
1605 97.551775 50.198.1.182 216.115.69.144 SIP 0 512 Status: 200 OK |
```

And that was it. Many exchanges and training but in the end our page was sent over a SIP trunk, negotiating T.38, training with the remote fax machine at 9600 baud, and transmitting one page in about a minute.

Share:



Uncategorized

← Remote Wireshark capture for Sophos UTM over SSH Skype for Business Server June 2016 CU →

One thought on "T.38 Fax over IP call on Wireshark"



rinin says:

March 29, 2018 at 9:43 pm

Hi Gonzalo,
Nice work you have there. Could you share me the pcap file that was used in this article? my email is rinin_farina@yahoo.com. Thanks in advance!

Reply

Leave a Reply

Enter your comment here

Powered by GovPress, the WordPress theme for government.