SharkFest '19 Europe

Analysing VoIP Protocols

Discover Wireshark's numerous features to troubleshoot VoIP

Rolf Leutert

Leutert NetServices
Switzerland
www.netsniffing.ch

Rolf Leutert, El. Eng. HTL
Leutert NetServices
Zürich-Airport, Switzerland

- Network Analysis & Troubleshooting
- Protocol Trainings TCP/IP, WLAN, VoIP, IPv6
- Wireshark® Certified Network Analyst 2010
- Wireshark® Instructor since 2006
- Sniffer® certified Instructor since 1990

leutert@netsniffing.ch
www.netsniffing.ch

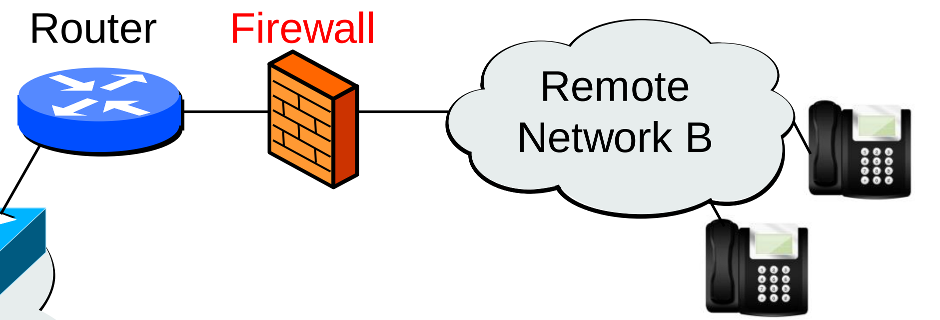
The original “Sniffer”

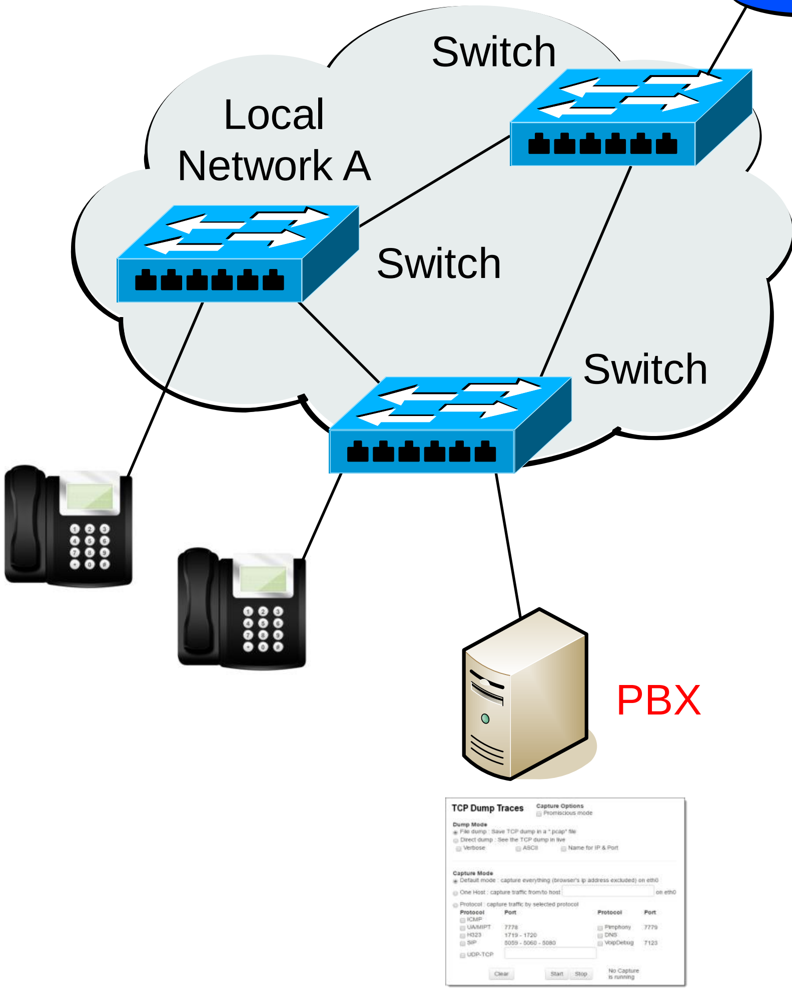
Sniffer® has been registered as trademark in 1989

- First **Network General Sniffer** in Switzerland
- Bought 1988 by Swissair airline to analyse **Token-Ring**
- Compaq Portable, DOS Version 1.30 / 256 KByte Capture Buffer
- Price US $ 30’000 (and more for each decoder)
- No trainings available (Sniffer University started in 1997)

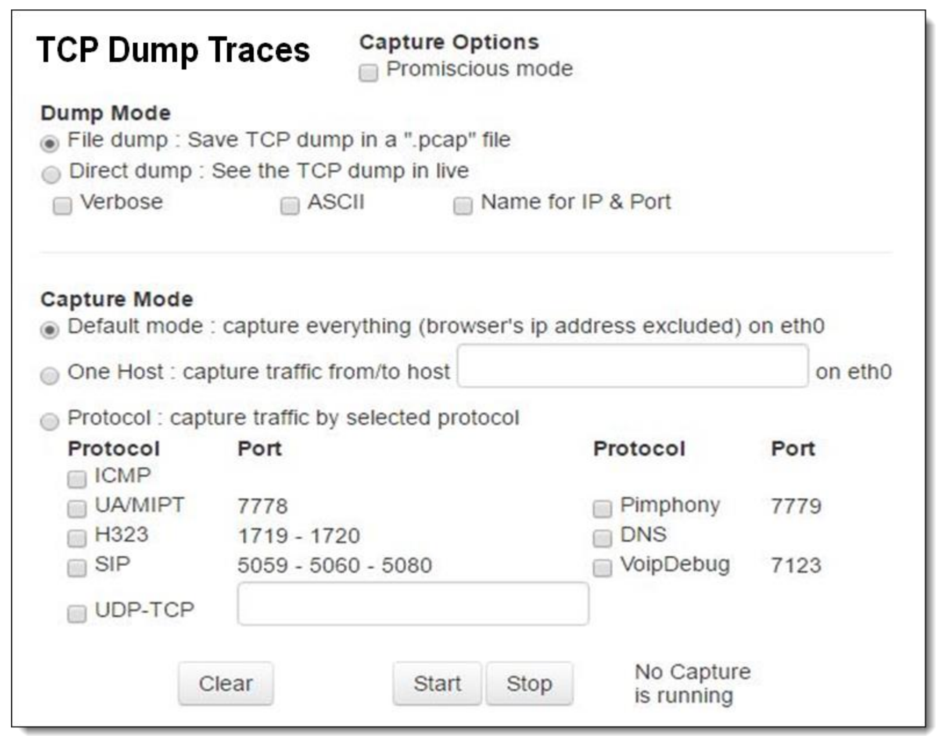
Session Content

- Where to capture VoIP traffic
- Analysing SIP signalling
- Use Case: Call interrupted after 32 seconds
- Analysing HFA an UA signalling
- Analysing SDP negotiation
- Use Case: Bad VoIP port negotiation
- Analysing RTP traffic
- RTP QOS, Delay, Jitter & Packet Loss
- RTP protocol forcing

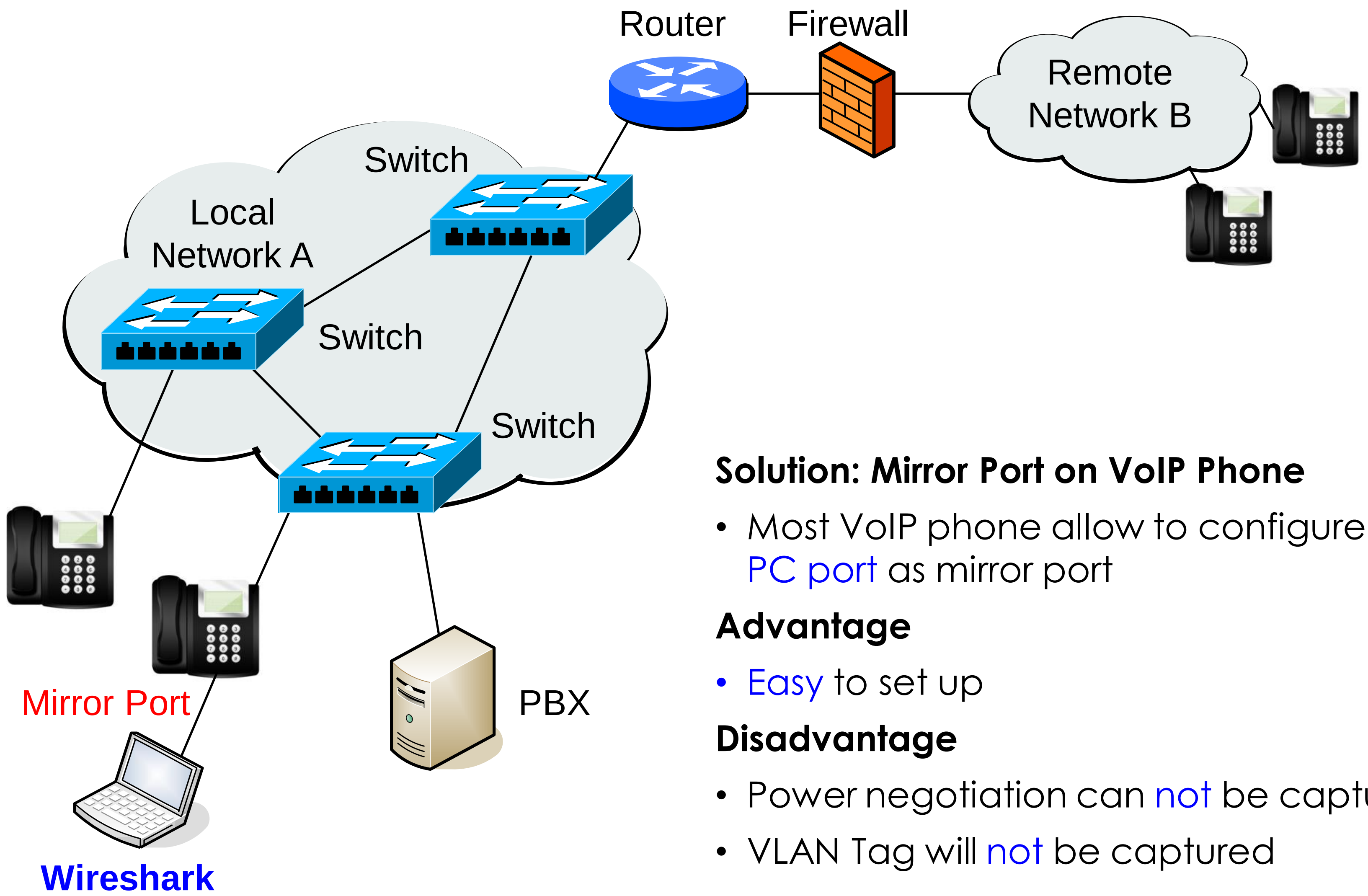




Solution: Capturing on PBX



Capturing in VoIP environment



Solution: Mirror Port on VoIP Phone

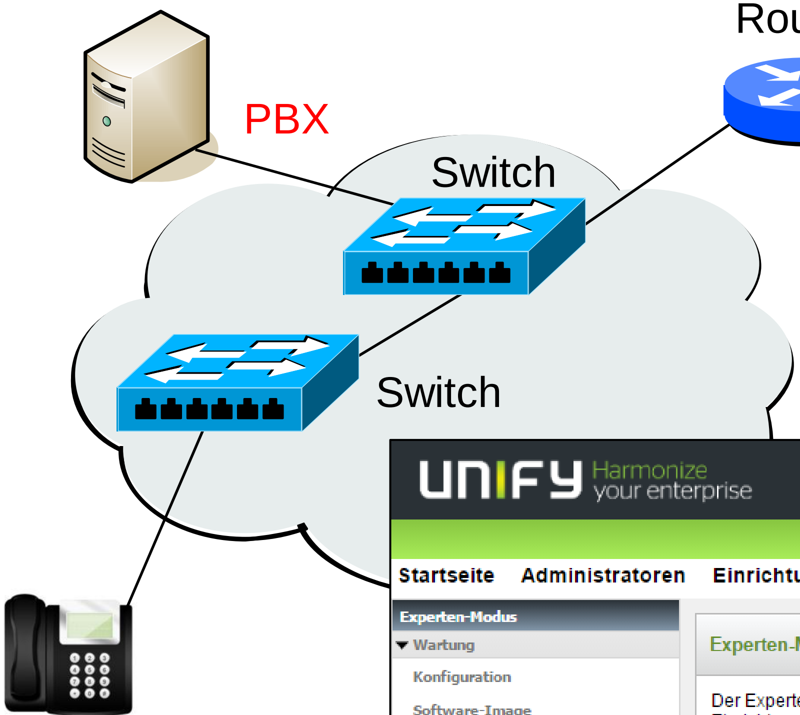
- Most VoIP phone allow to configure the PC port as mirror port

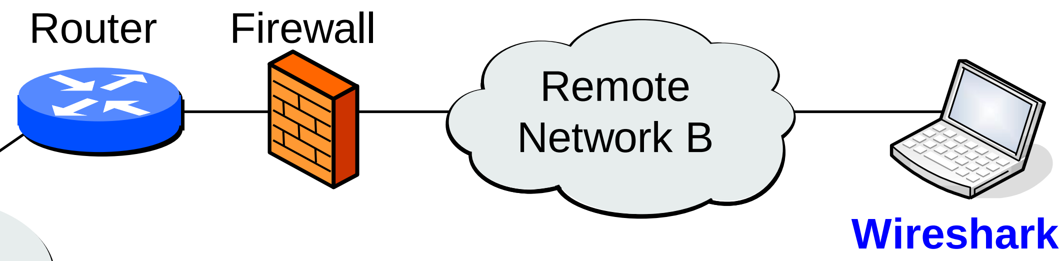
Advantage

- Easy to set up

Disadvantage

- Power negotiation can not be captured
- VLAN Tag will not be captured

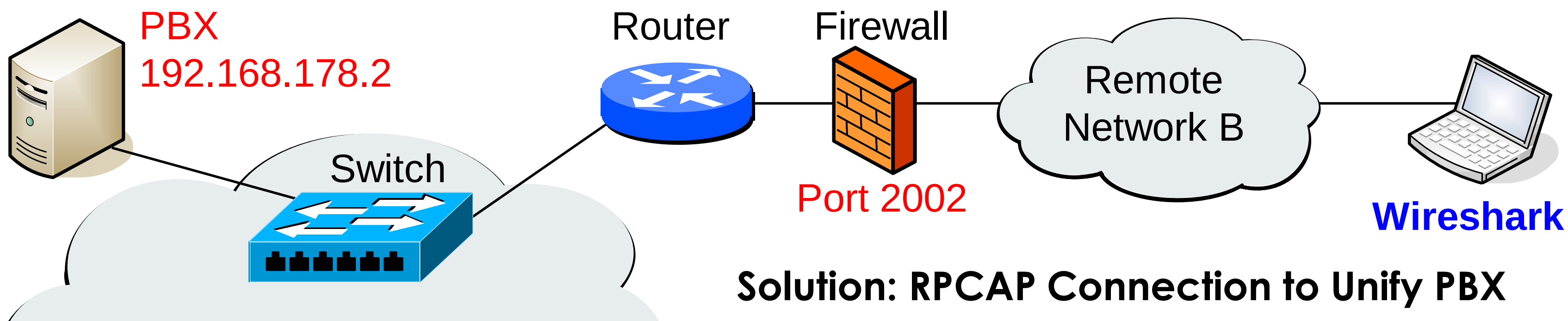




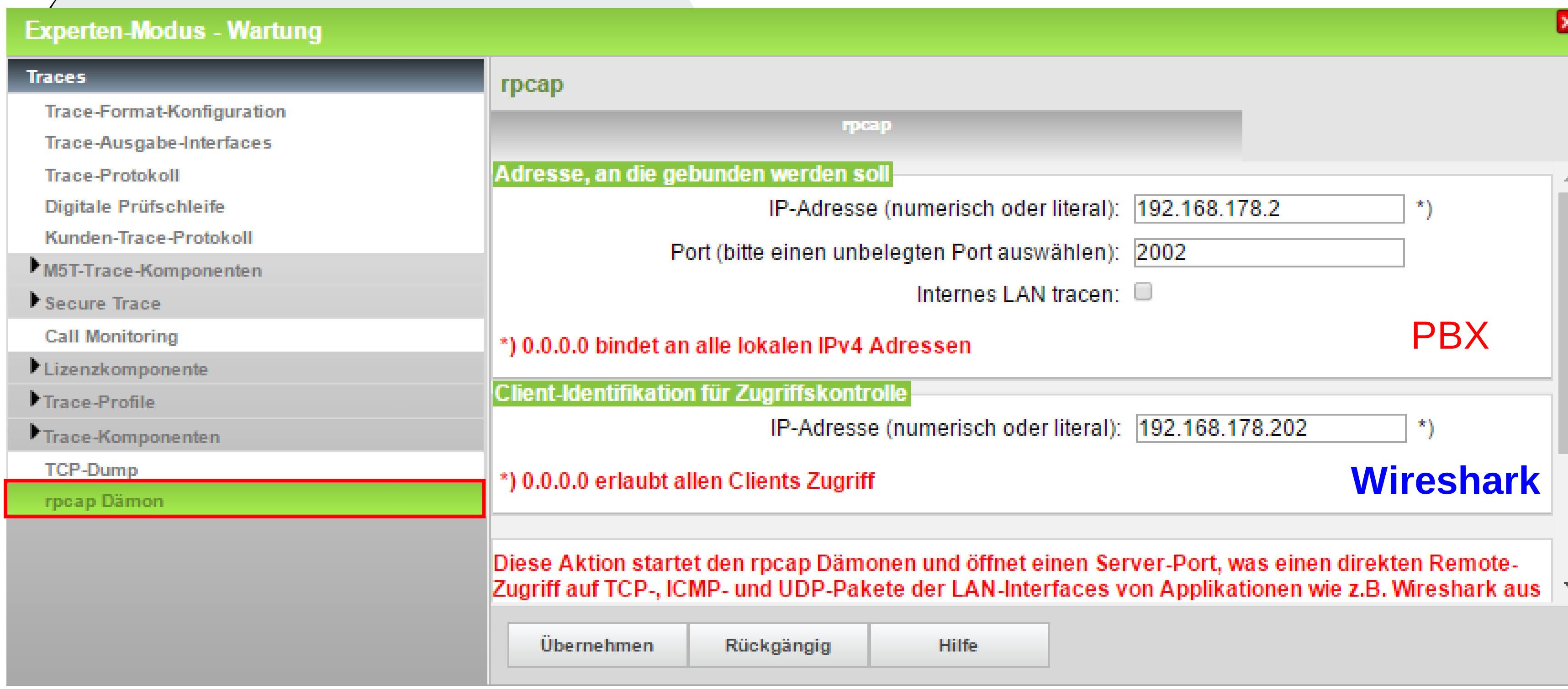
Solution: Remote Packet Capture (RPCAP)
Connection to Unify PBX

Screenshot: Unify PBX

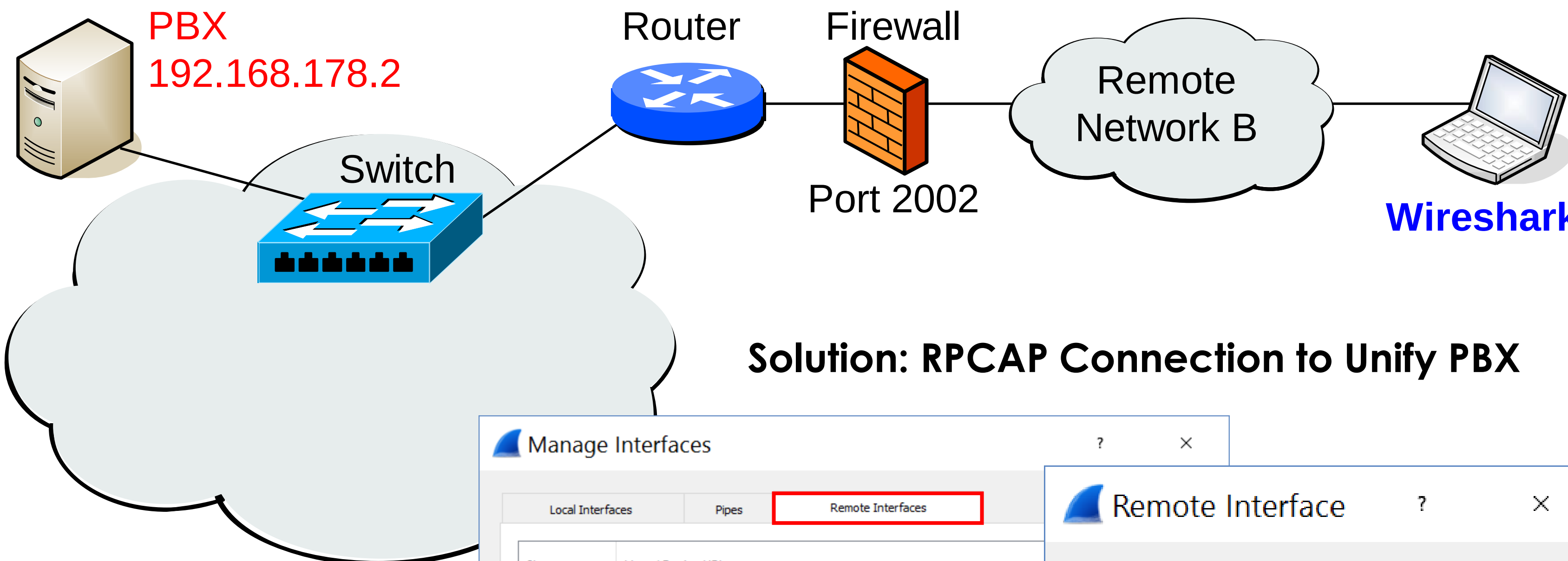
Capturing in VoIP environment



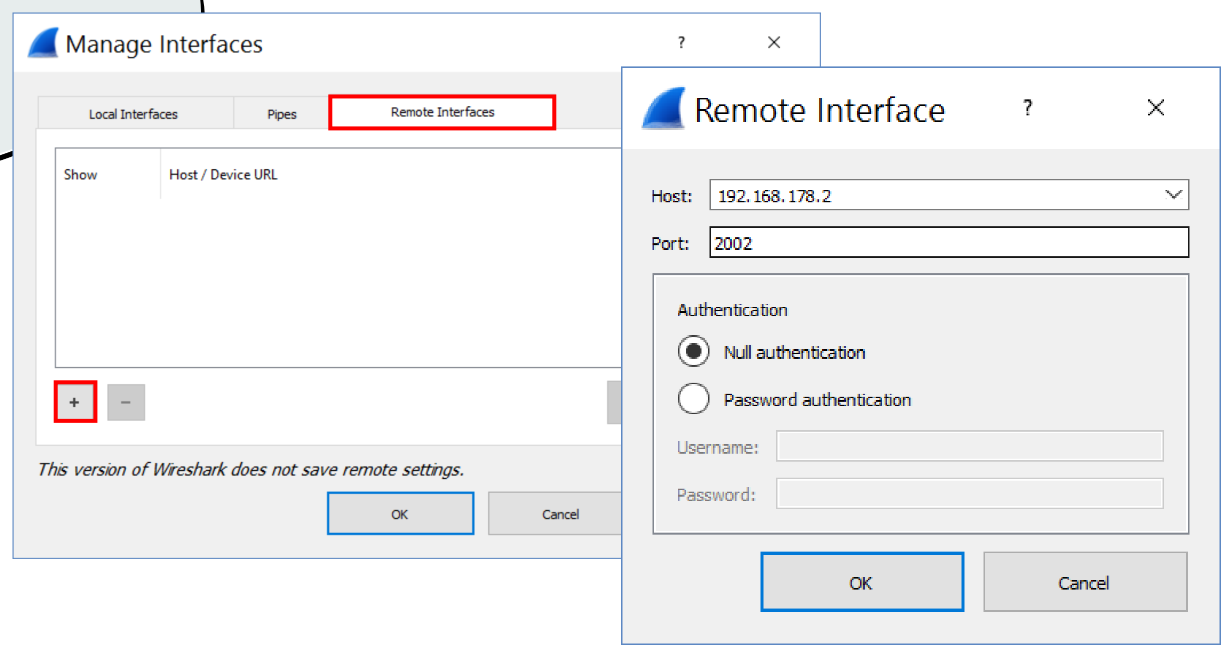
Solution: RPCAP Connection to Unify PBX



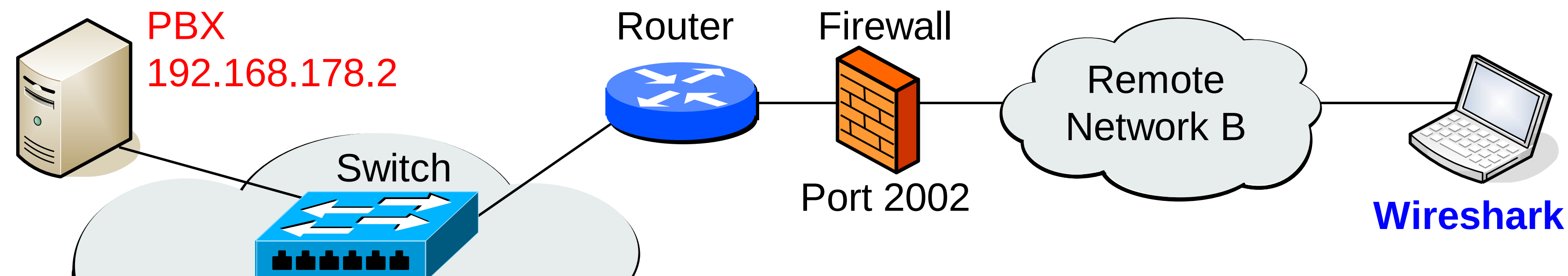
Capturing in VoIP environment



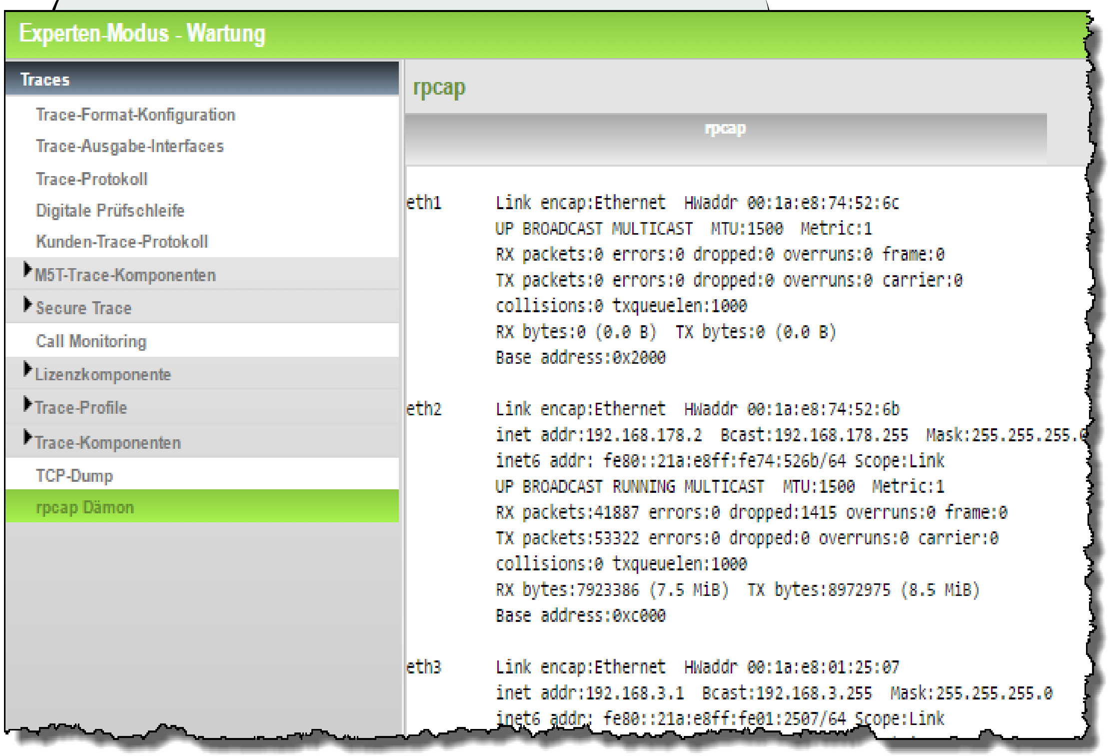
Solution: RPCAP Connection to Unify PBX

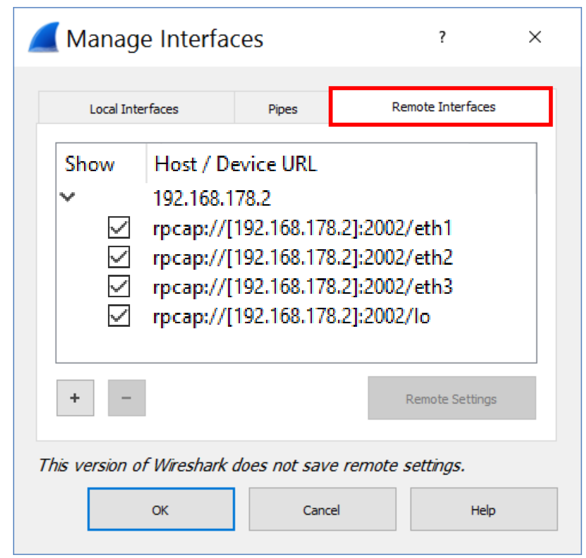


Wireshark:
→ Capture Options
→ Manage Interfaces
→ Remote Interfaces
→ Add Interface with +

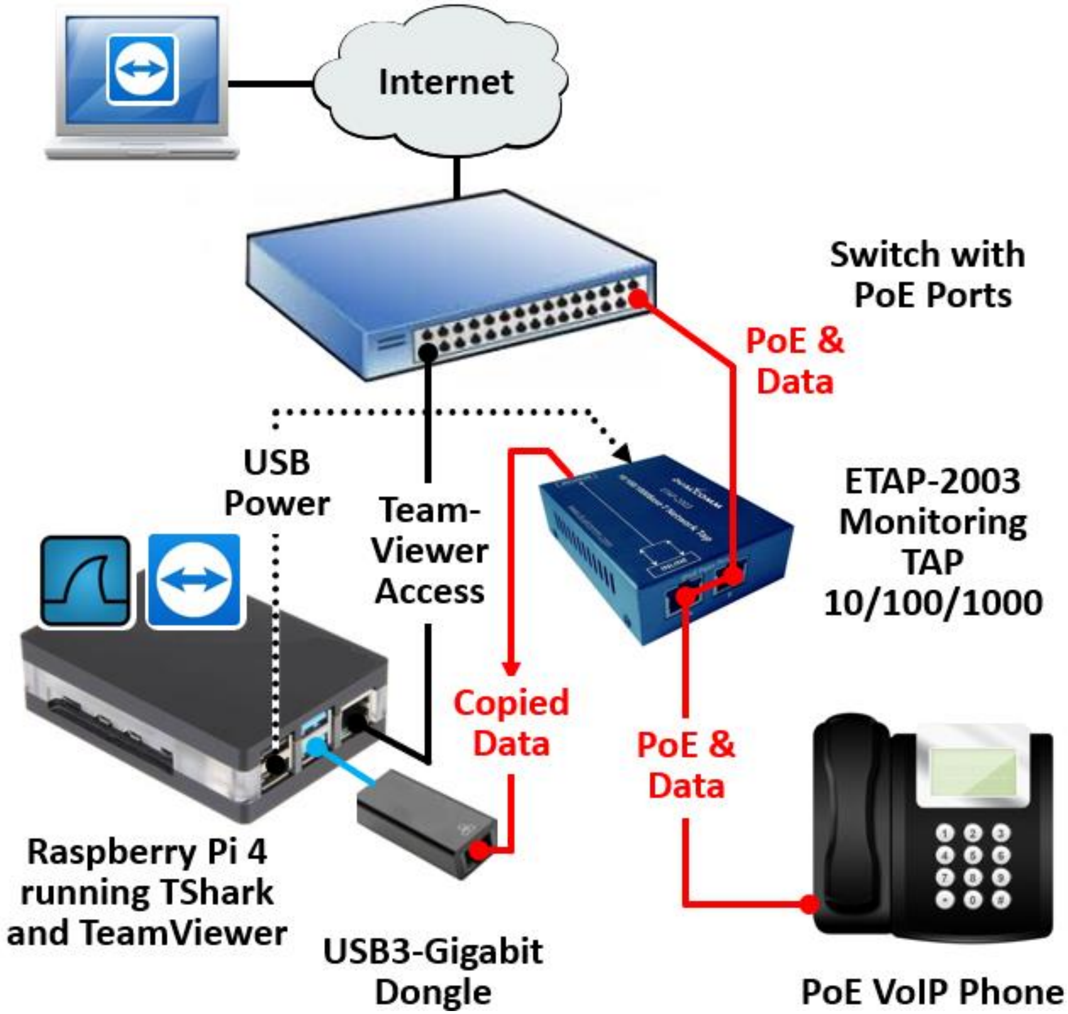


Solution: RPCAP Connection to Unify PBX





Capturing in VoIP environment



Inexpensive solution for remote capturing with Raspberry Pi 4

- Raspberry with two Ethernet interfaces: one for capturing, one for TeamViewer access

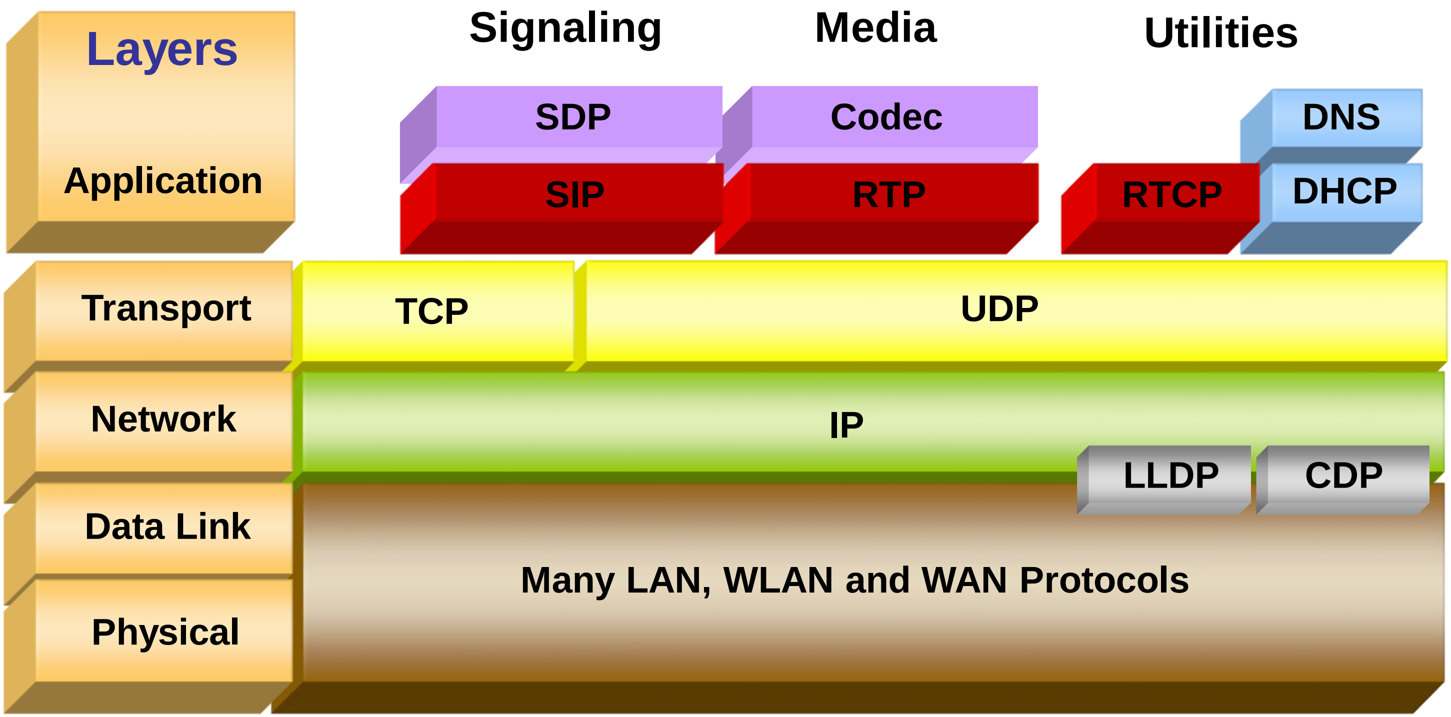
Advantage

- Allows remote configuration
- Allows long term capturing
- Can be installed at multiple customer sites
- Using TShark to preserve CPU

Disadvantage

- Raspberry performance is limited, solution is not suitable to capture server traffic
- Requires internet connection for TeamViewer access.

- **SIP** Session Initiation Protocol, create, modify, terminate sessions
- **SDP** Session Description Protocol, describing multimedia sessions
- **RTP** Real-Time Transport Protocol, audio and video packet format
- **RTCP** Real-Time Control Protocol, quality of reception data feedback
- **Codec** Analog/digital encodings; G.711, G.729, µ-Law, A-Law, AMR etc.



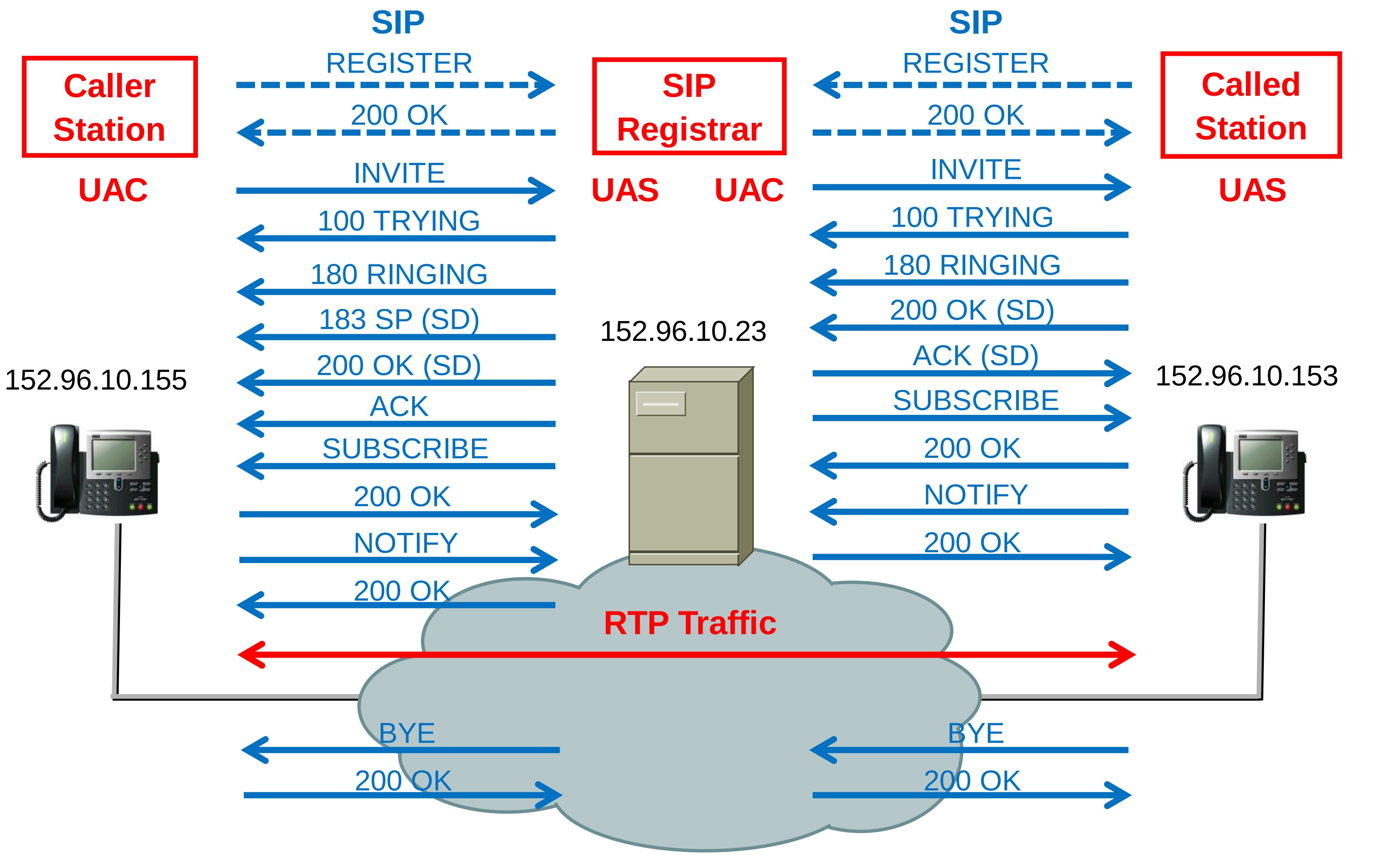
SIP basic Requests (called methods):

- **REGISTER** - Registers the address listed in the To header field with a SIP server
- **INVITE** - Indicates a client is being invited to participate in a call session
- **ACK** - Confirms that the client has received a final response to an INVITE request
- **BYE** - Terminates a call and can be sent by either the caller or the called
- **CANCEL** - Cancels any pending request but does not terminate a call
- **OPTIONS** - Queries the capabilities of servers
- **PRACK** - Provisional acknowledgement
- **SUBSCRIBE** - User wishes to receive information about the status of a service session
- **NOTIFY**- Status of the service session for which the Requestor has subscribed
- **PUBLISH** - Publishes an event to the Server
- **INFO** - Sends mid-session information that does not modify the session state
- **REFER** - Asks recipient to issue SIP request (call transfer)
- **MESSAGE** - Transports instant messages using SIP
- **UPDATE** - Modifies the state of a session without changing the state of the dialog

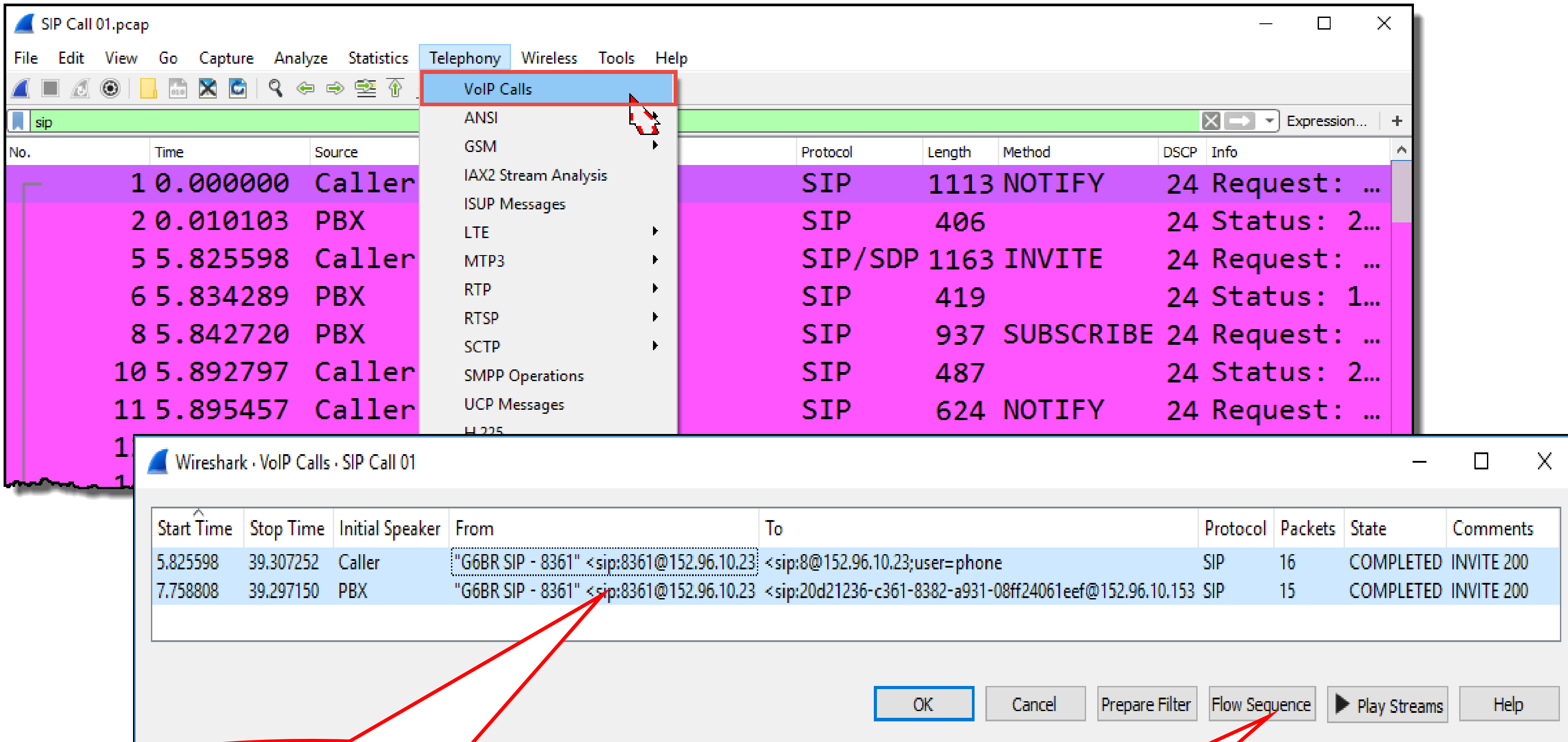
SIP basic Response codes:

	Description	Examples
1xx	Informational – Request received, continuing to process request.	100 Trying 180 Ringing 181 Call is Being Forwarded
2xx	Success – Action was successfully received, understood and accepted.	200 OK
3xx	Redirection – Further action needs to be taken in order to complete the request.	300 Multiple Choices 301 Moved Permanently 302 Moved Temporarily
4xx	Client Error – Request contains bad syntax or cannot be fulfilled at this server.	401 Unauthorized 406 Not Acceptable 407 Proxy Authentication Required 408 Request Timeout 415 Unsupported Media type
5xx	Server Error – Server failed to fulfill an apparently valid request.	502 Bad Gateway 503 Service Unavailable 505 Version Not Supported
6xx	Global Failure – Request is invalid at any server.	600 Busy Everywhere 603 Decline

Basic SIP Call Flows:



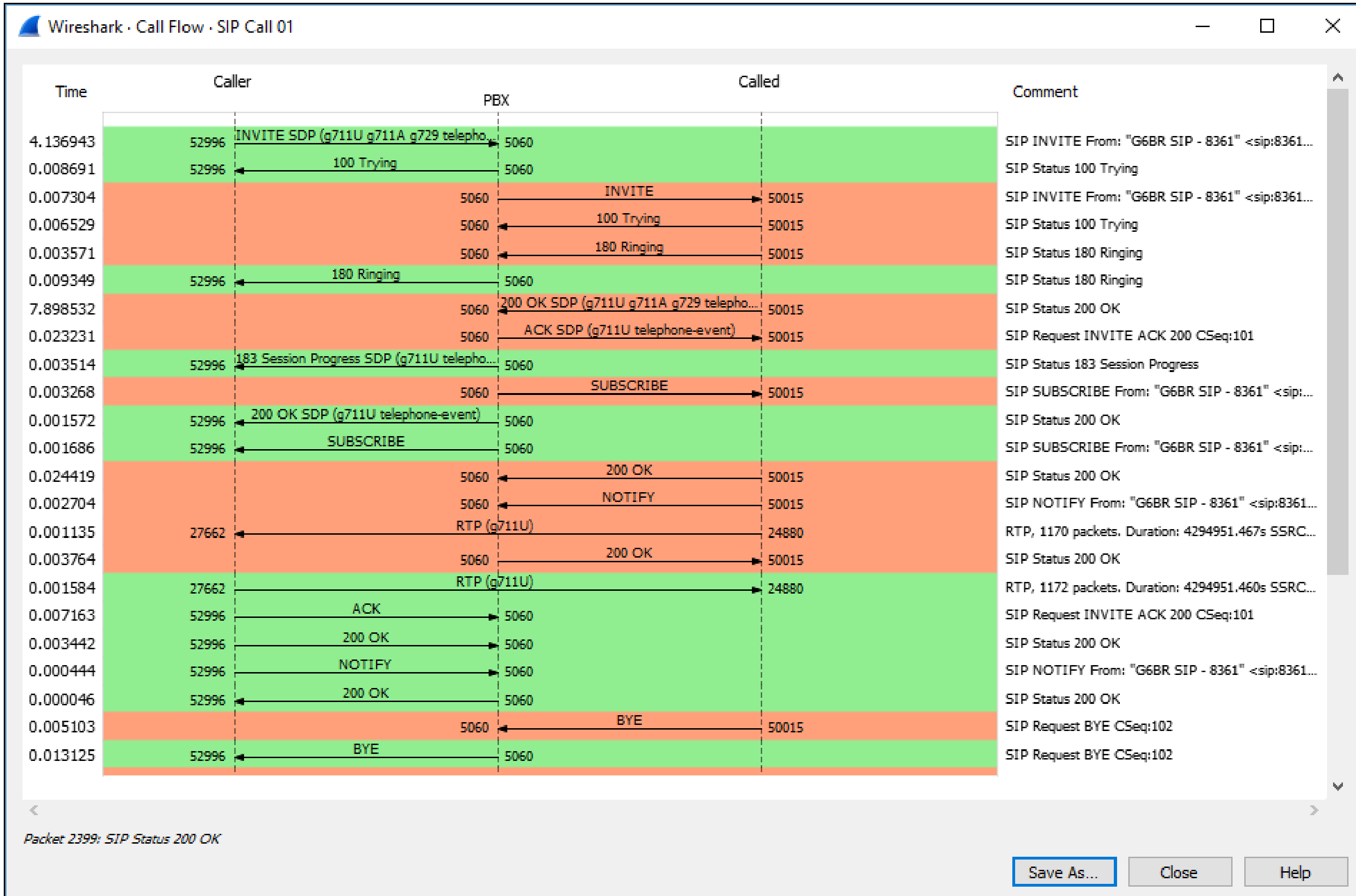
Graphical presentation of SIP calls:



1. Mark both Calls with Shift + Right Mouse

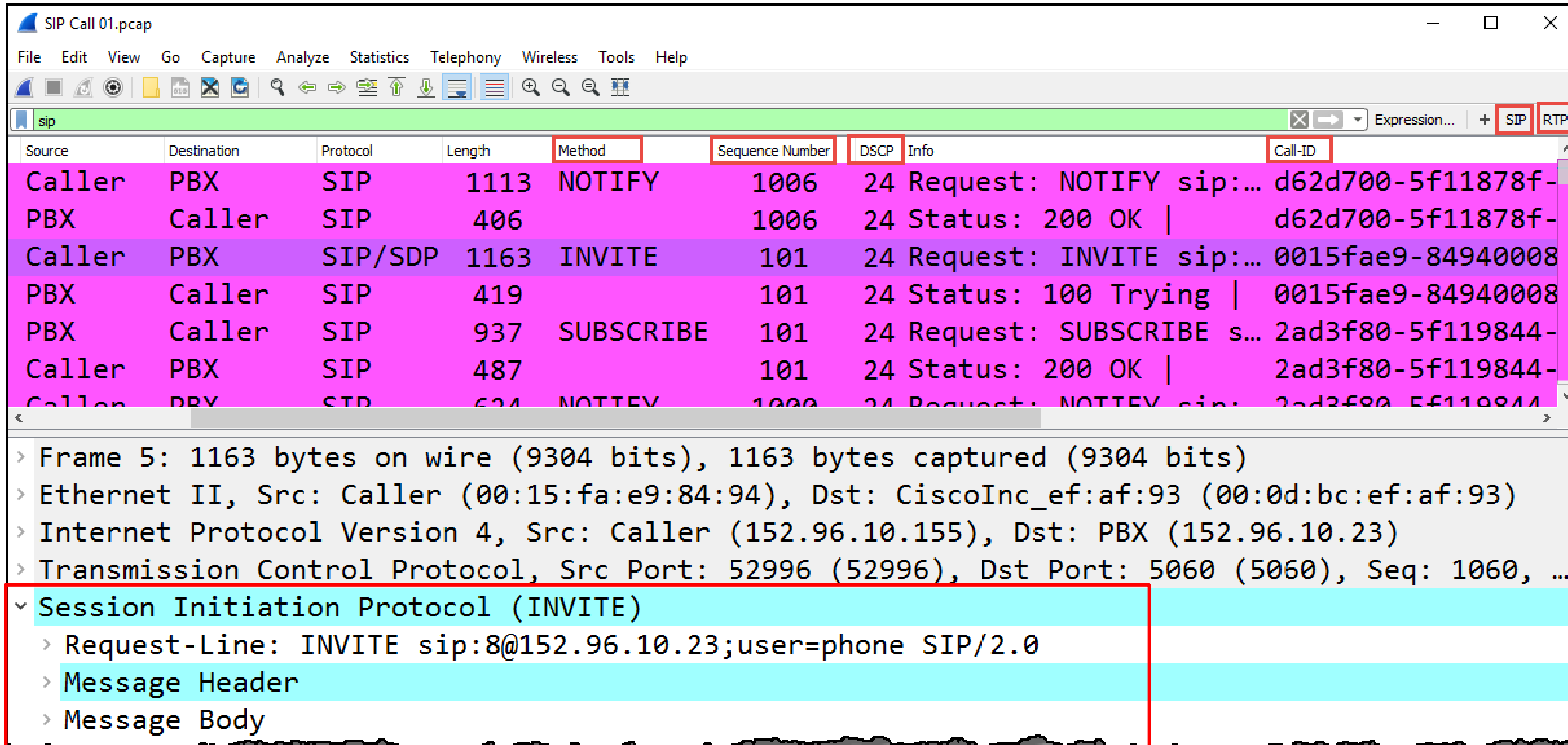
2. Select Flow Sequence

Graphical presentation of SIP calls:



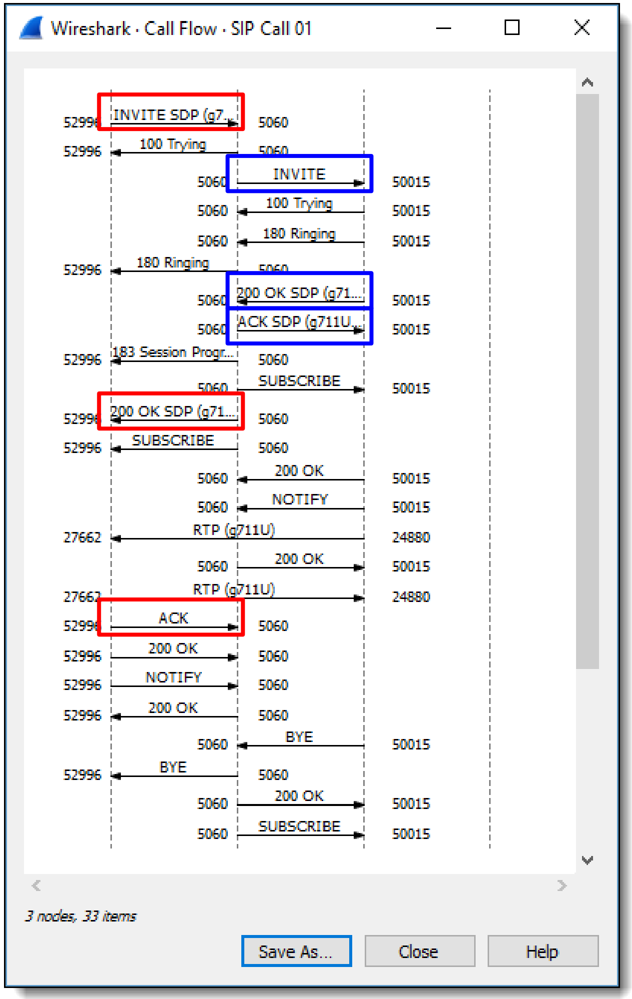
- Clicking on an arrow line jumps to the appropriate packet in the Packet List

Adding several columns for SIP analysis:



SIP special message ACK:

- **SIP ACK** (Acknowledge) is an exception to the SIP Request and Response rules
- **SIP ACK** is a **Request without a Response**
- **SIP ACK** is sent as a **Confirmation** to a successfully established **Invite** transaction.
- **SIP ACK** Message is carrying the same sequence number and call-id as the invite transaction.
- **SIP ACK** is not confirmed with a response



Use Case: Call interrupted after 32 seconds

Call_interupted_after_32sec.pcapng

Datei Bearbeiten Ansicht Navigation Aufzeichnen Analyse Statistiken Telephonie Wireless Tools Hilfe

Anzeigefilter anwenden ... <Ctrl-/> | Ausdruck... | + | SIP | RTP | SIP or RTP

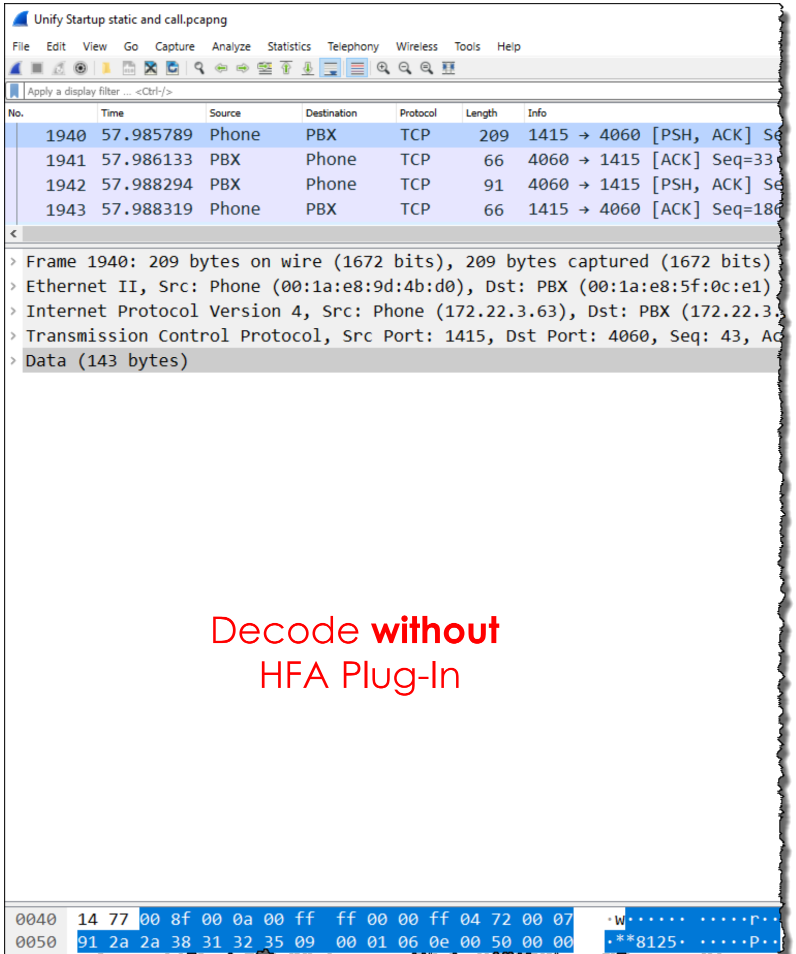
No.	Time	Delta Time	Source	Destination	Time to live	Protocol	Length	Sequence Number	Method	Info
1	0.000000	0.000000	Provider	PBX	55	SIP/SDP	980	1	INVITE	Request: INVITE sip:+4936824520@176.94.60.34:
2	0.000148	0.000148	PBX	Provider	64	TCP	66			5060 → 48380 [ACK] Seq=1 Ack=915 Win=1272 Len=
3	0.180673	0.180525	PBX	Provider	64	SIP	669	1		Status: 180 Ringing |
4	0.290555	0.109882	Provider	PBX	55	TCP	66			5060 → 33195 [ACK] Seq=1 Ack=604 Win=32768 Len
5	4.064327	3.773772	PBX	Provider	64	SIP/SDP	996	1		Status: 200 OK |
6	4.130327	0.066000	Provider	PBX	240	RTP	214			PT=ITU-T G.711 PCMA, SSRC=0x21C0148D, Seq=0,
7	4.133857	0.003530	PBX	Provider	64	RTP	214			PT=ITU-T G.711 PCMA, SSRC=0xBF576815, Seq=0,
8	4.150407	0.016550	Provider	PBX	240	RTP	214			PT=ITU-T G.711 PCMA, SSRC=0x21C0148D, Seq=1,
9	4.153869	0.003462	PBX	Provider	64	RTP	214			PT=ITU-T G.711 PCMA, SSRC=0xBF576815, Seq=1,
10	4.170393	0.016524	Provider	PBX	240	RTP	214			PT=ITU-T G.711 PCMA, SSRC=0x21C0148D, Seq=2,
11	4.173851	0.003458	PBX	Provider	64	RTP	214			PT=ITU-T G.711 PCMA, SSRC=0xBF576815, Seq=2,
12	4.174582	0.000731	Provider	PBX	55	TCP	66			5060 → 33195 [ACK] Seq=1 Ack=1534 Win=32768 Le
13	4.190680	0.016098	Provider	PBX	240	RTP	214			PT=ITU-T G.711 PCMA, SSRC=0x21C0148D, Seq=3,
14	4.193893	0.003213	PBX	Provider	64	RTP	214			PT=ITU-T G.711 PCMA, SSRC=0xBF576815, Seq=3,
15	4.210558	0.016665	Provider	PBX	240	RTP	214			PT=ITU-T G.711 PCMA, SSRC=0x21C0148D, Seq=4,
16	4.213882	0.003324	PBX	Provider	64	RTP	214			PT=ITU-T G.711 PCMA, SSRC=0xBF576815, Seq=4,
17	4.230293	0.016411	Provider	PBX	240	RTP	214			PT=ITU-T G.711 PCMA, SSRC=0x21C0148D, Seq=5,
18	4.233838	0.003545	PBX	Provider	64	RTP	214			PT=ITU-T G.711 PCMA, SSRC=0xBF576815, Seq=5,

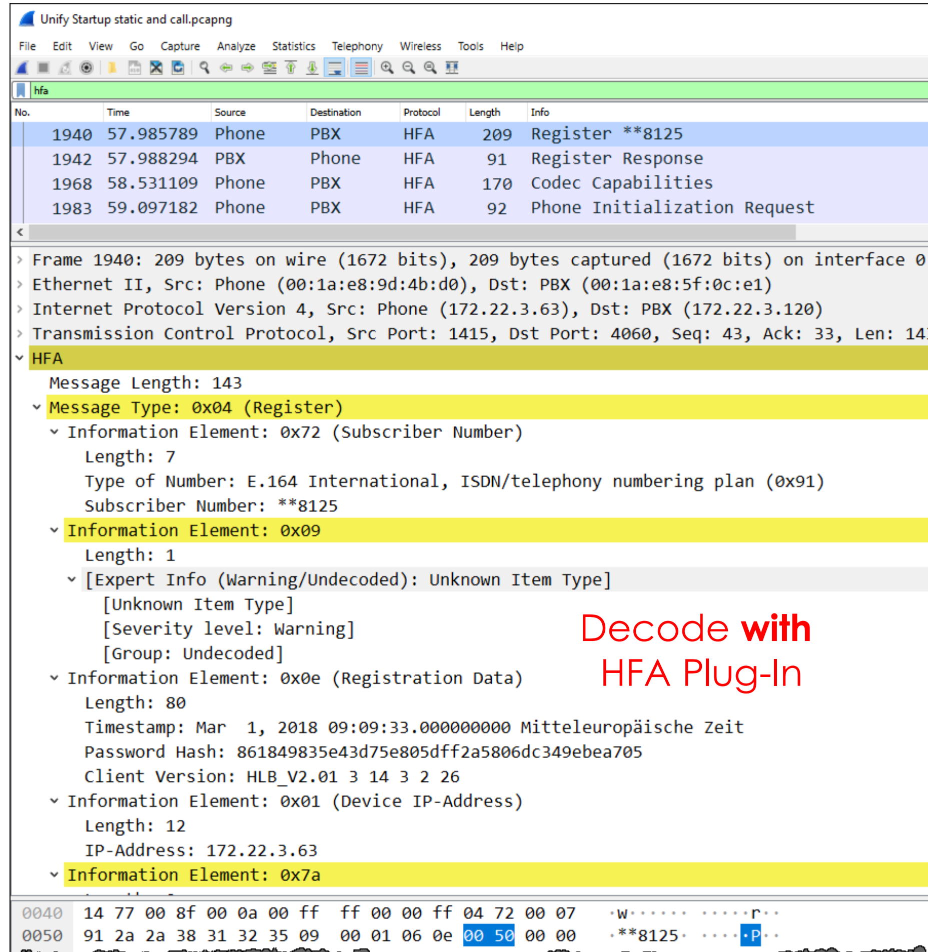
> Frame 1: 980 bytes on wire (7840 bits), 980 bytes captured (7840 bits) on interface 0
> Ethernet II, Src: Sophos_4c:50:f0 (7c:5a:1c:4c:50:f0), Dst: UnifySof_89:b8:0b (00:1a:e8:89:b8:0b)
> Internet Protocol Version 4, Src: Provider (176.95.49.1), Dst: PBX (192.168.70.100)
> Transmission Control Protocol, Src Port: 48380, Dst Port: 5060, Seq: 1, Ack: 1, Len: 914
> Session Initiation Protocol (INVITE)

Call_interupted_after_32sec.pcapng | Pakete: 3234 · Angezeigt: 3234 (100.0%) | Profil: LNS SIP

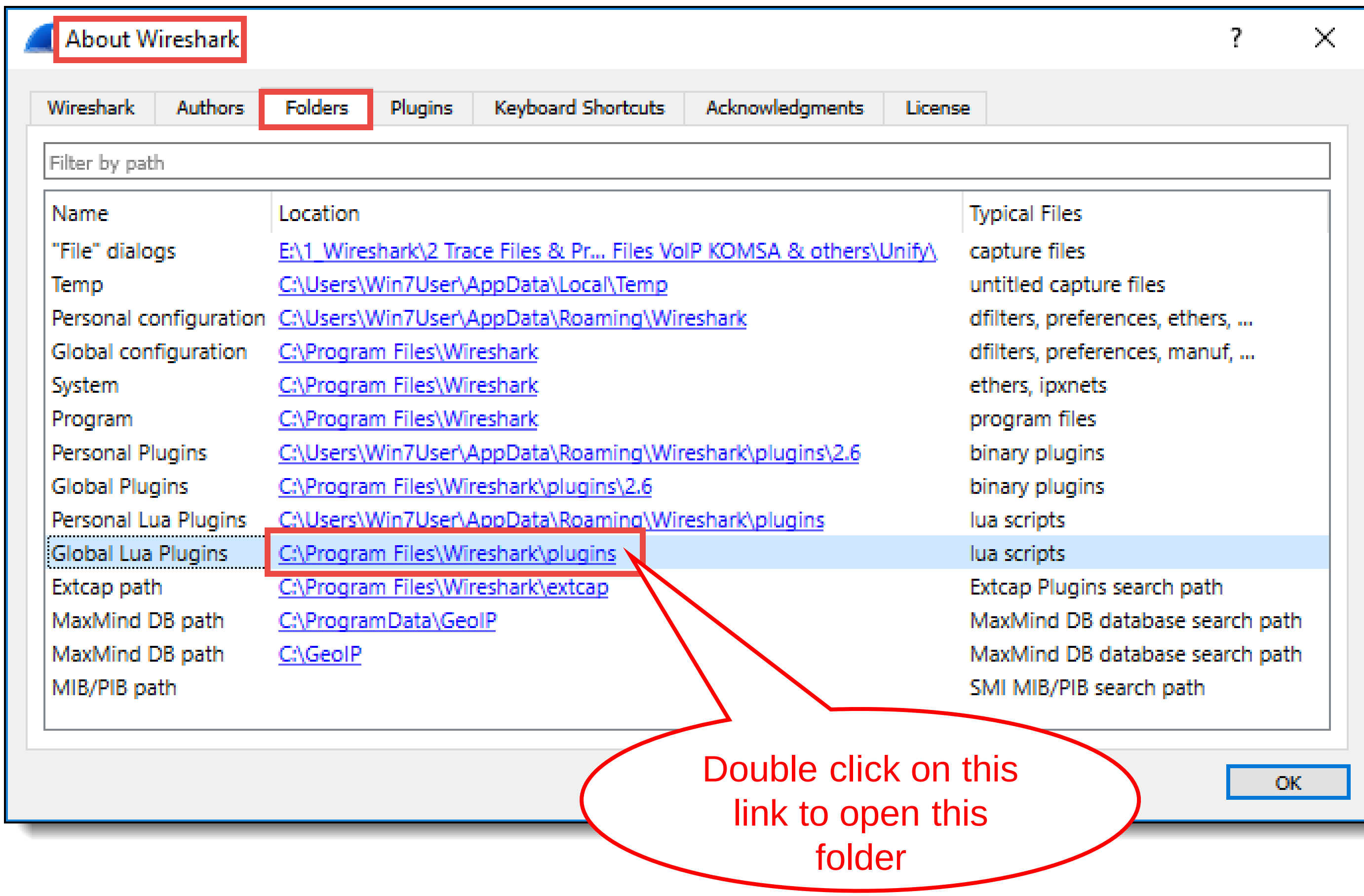
Analyzing Unify HFA protocol

- HFA (CorNet-IP) is TCP based and used for signaling in pure **HiPath** environment
- HFA can be decoded only by installing an additional **Plug-In** for Wireshark





- The **H**iPath **F**eature **A**ccess (HFA) is proprietary and not included in Wireshark
- Download **hfa.lua** Plug-In from https://github.com/jonas-koeritz/hfa-dissector
- Copy the **hfa.lua** to the Wireshark Plugins folder, close and restart Wireshark



Wireshark decodes the following UA protocols:

- **UAUDP** Universal Alcatel over UDP, **NOE** New Office Environment, **UA3G** and **UASIP**

UA Call432 & Incoming Call.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

uaudp

No.	Time	Source	Destination	Length	Protocol	Sequence No (sent)	Sequence No (expected)	Info
1	1509551125.523	Phone	PBX	56	UAUDP	121	303	Data - NOE Protocol (CS): Notify EVT_KEY_PRES
2	1509551125.523	PBX	Phone	60	UAUDP	303	122	Data ACK
3	1509551125.525	PBX	Phone	325	UAUDP	303	122	Data - UA3G Message: Main Voice Mode: Handsfr
5	1509551125.527	Phone	PBX	56	UAUDP	121	304	Data ACK
8	1509551125.824	Phone	PBX	56	UAUDP	122	304	Data - NOE Protocol (CS): Notify EVT_KEY_PRES
9	1509551125.824	PBX	Phone	60	UAUDP	304	123	Data ACK
10	1509551125.824	PBX	Phone	60	UAUDP	304	123	Data - NOE Protocol (CS): SetProperty TextBox
11	1509551125.825	Phone	PBX	56	UAUDP	122	305	Data ACK
13	1509551126.033	Phone	PBX	56	UAUDP	123	305	Data - NOE Protocol (CS): Notify EVT_KEY_PRES
14	1509551126.034	PBX	Phone	60	UAUDP	305	124	Data ACK
16	1509551126.043	PBX	Phone	60	UAUDP	305	124	Data - NOE Protocol (CS): SetProperty TextBox
17	1509551126.043	Phone	PBX	56	UAUDP	123	306	Data ACK

- Frame 1: 56 bytes on wire (448 bits), 56 bytes captured (448 bits) on interface 0
- Ethernet II, Src: Phone (00:80:9f:b1:44:63), Dst: AlcatelB_bf:ea:ae (00:80:9f:bf:ea:ae)
- Internet Protocol Version 4, Src: Phone (172.28.51.1), Dst: PBX (172.28.10.12)
- User Datagram Protocol, Src Port: 32512, Dst Port: 32640
- Universal Alcatel/UDP Encapsulation Protocol, Data
- Universal Alcatel Protocol, Terminal -> System

Session **D**escription **P**rotocol (SDP) features:

- **SDP** is intended for describing multimedia sessions for the purposes of session announcement, session invitation, and other forms of multimedia session initiation

- **SDP** is used by many protocols to describe media sessions such as SIP, MGCP, RTSP, BICC, H.248/MEGACO

- **SDP** does not deliver media itself but is used for negotiation between end points

- **SDP** contains three main sections, detailing the session, timing, and media descriptions.

- **SDP** is simple and flexible because it is Text-based

- **SDP** describes the session by a series of fields, one per line.

Introduction

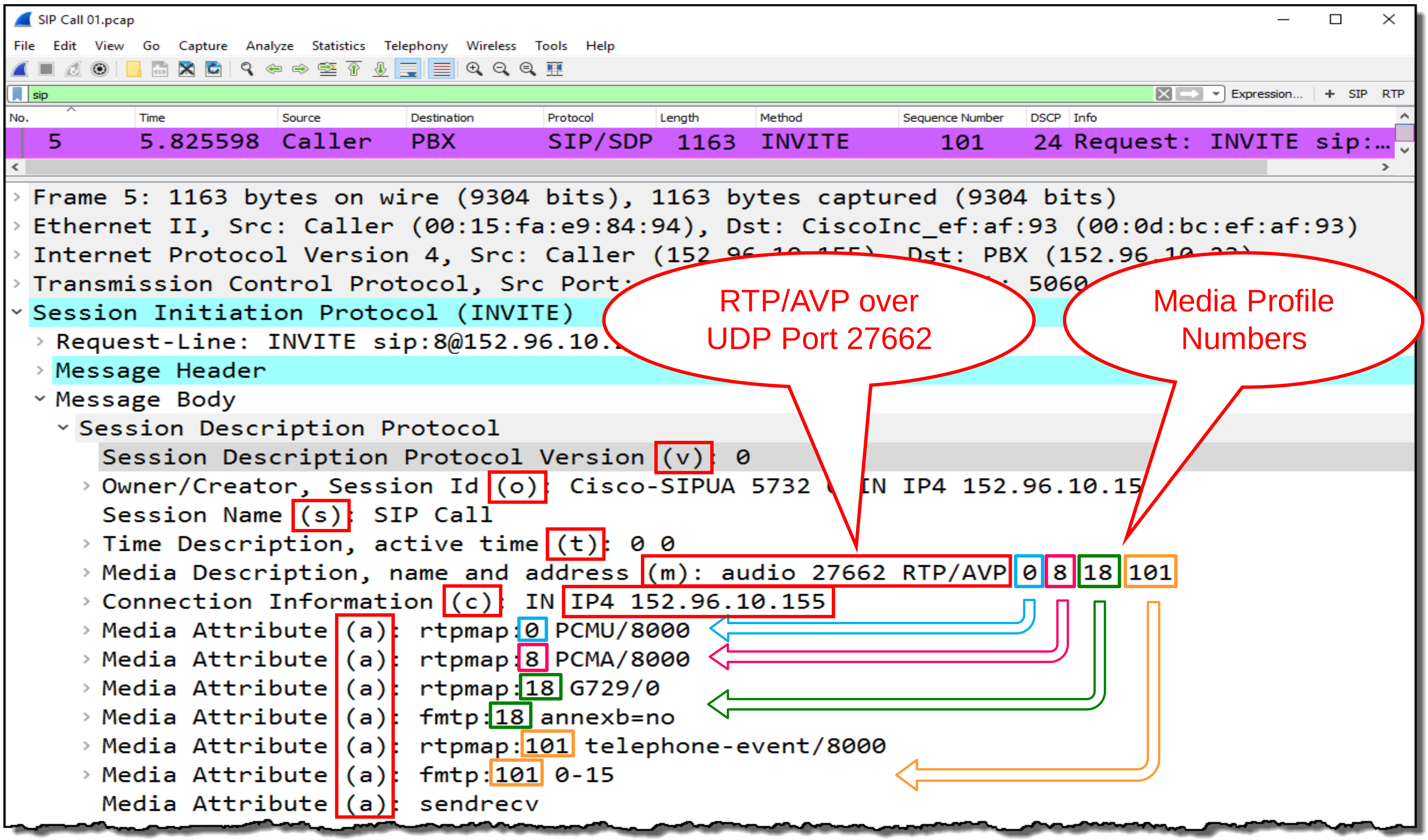
SIP Call 01.pcap

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

sip.Call-ID == "0015fae9-84940008-4ff1580c-85340a0d@152.96.10.155"

Source	Destination	Protocol	Length	Sequence Number	Info
Caller	PBX	SIP/SDP	1163	101	Request: INVITE sip:8@152.96.10.23;user=ph
PBX	Caller	SIP	419	101	Status: 100 Trying |
PBX	Caller	SIP	813	101	Status: 180 Ringing |
PBX	Caller	SIP/SDP	1068	101	Status: 183 Session Progress |
PBX	Caller	SIP/SDP	1053	101	Status: 200 OK |
PBX	Caller	SIP	934	101	Request: SUBSCRIBE sip:b56daa0c-e25d-7a68-
Caller	PBX	SIP	630	101	Request: ACK sip:8@152.96.10.23:5060;trans
Caller	PBX	SIP	558	101	Status: 200 OK |
Caller	PBX	SIP	692	102	Request: NOTIFY sip:152.96.10.23:5060 |
PBX	Caller	SIP	441	102	Status: 200 OK |
PBX	Caller	SIP	547	102	Request: BYE sip:b56daa0c-e25d-7a68-f5b6-2
Caller	PBX	SIP	450	102	Status: 200 OK |
PBX	Caller	SIP	632	103	Request: SUBSCRIBE sip:b56daa0c-e25d-7a68-
Caller	PBX	SIP	555	103	Status: 200 OK |
Caller	PBX	SIP	1003	103	Request: NOTIFY sip:152.96.10.23:5060 |
PBX	Caller	SIP	441	103	Status: 200 OK |

SDP Variables:



Use Case: Voice channel call be established

Codec Mismatch.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

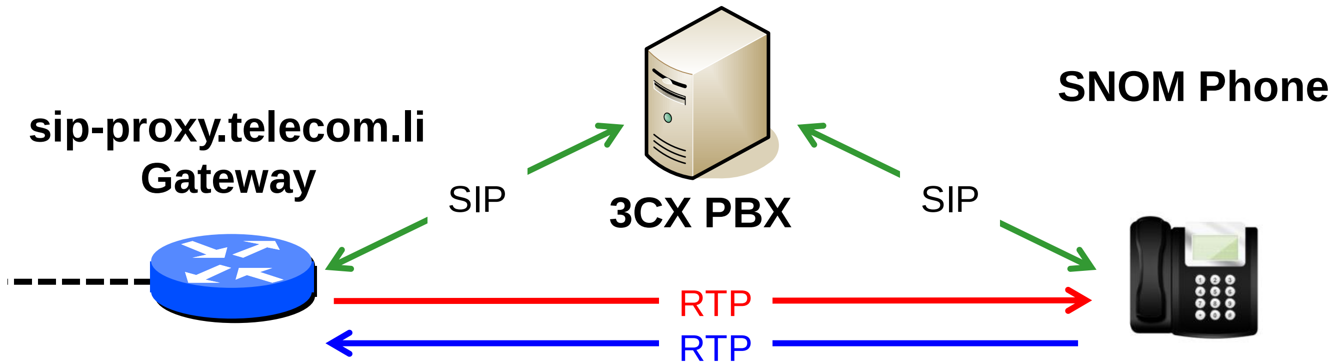
Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Sequence Number	Method	Info
1	0.000000	SNOM Phone	3CX PBX	SIP	99762	REGISTER	Request: REGISTER si
2	0.100748	3CX PBX	SNOM Phone	SIP	99762		Status: 407 Proxy Au
3	0.122080	SNOM Phone	3CX PBX	SIP	99763	REGISTER	Request: REGISTER si
4	0.222033	3CX PBX	SNOM Phone	SIP	99763		Status: 200 OK (2 b
5	5.688349	sip-proxy.telecom.li	3CX PBX	SIP/SDP	2	INVITE	Request: INVITE sip:
6	5.692957	3CX PBX	SNOM Phone	SIP/SDP	2	INVITE	Request: INVITE sip:
7	5.715344	SNOM Phone	3CX PBX	SIP	2		Status: 100 Trying |
8	5.746859	SNOM Phone	3CX PBX	SIP/SDP	2		Status: 200 OK |
9	5.767967	3CX PBX	sip-proxy.telecom.li	SIP	2		Status: 100 Trying |
10	5.767973	3CX PBX	sip-proxy.telecom.li	SIP/SDP	2		Status: 200 OK |
11	5.768013	3CX PBX	SNOM Phone	SIP	2	ACK	Request: ACK sip:523
12	5.790120	sip-proxy.telecom.li	3CX PBX	SIP	2	ACK	Request: ACK sip:004
13	5.802239	sip-proxy.telecom.li	SNOM Phone	RTP			PT=ITU-T G.711 PCMA,
14	5.804689	SNOM Phone	sip-proxy.telecom.li	RTP			PT=ITU-T G.711 PCMU,
15	5.821539	sip-proxy.telecom.li	SNOM Phone	RTP			PT=ITU-T G.711 PCMA,

> Frame 1: 609 bytes on wire (4872 bits), 609 bytes captured (4872 bits) on interface 0
> Ethernet II, Src: SnomTech_61:4b:6f (00:04:13:61:4b:6f), Dst: Elitegro_14:c5:89 (94:c6:91:14:c5:89)
> Internet Protocol Version 4, Src: SNOM Phone (192.168.1.58), Dst: 3CX PBX (192.168.1.110)
> User Datagram Protocol, Src Port: 5060, Dst Port: 5060
> Session Initiation Protocol (REGISTER)

Analysing SDP negotiation

Use Case: Voice channel call be established



Frame 5 — INVITE → SDP A-law, Port 24320

INVITE → Frame 6 — SDP A-law / µ-law, Port 24320

← TRYING — Frame 7

← 200 OK — Frame 8 — SDP µ-law / A-law, Port 50004

Frame 9 ← TRYING

Frame 10 ← 200 OK — SDP µ-law / A-law, Port 50004

ACK → Frame 11

Frame 12 — ACK →

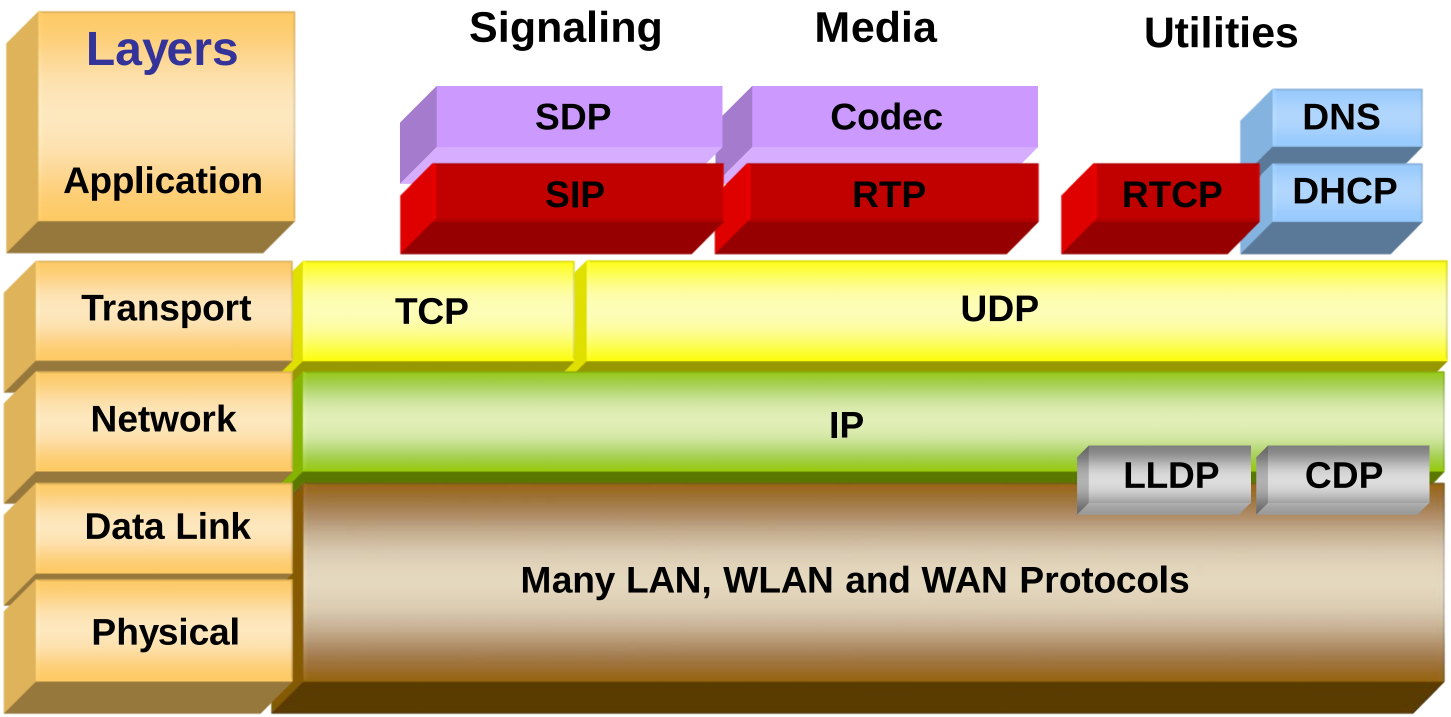
Port 24320 — RTP A-law → Port 50004

Port 24320 ← RTP µ-law — Port 50004

Fail reason:
Codec mismatch!

+

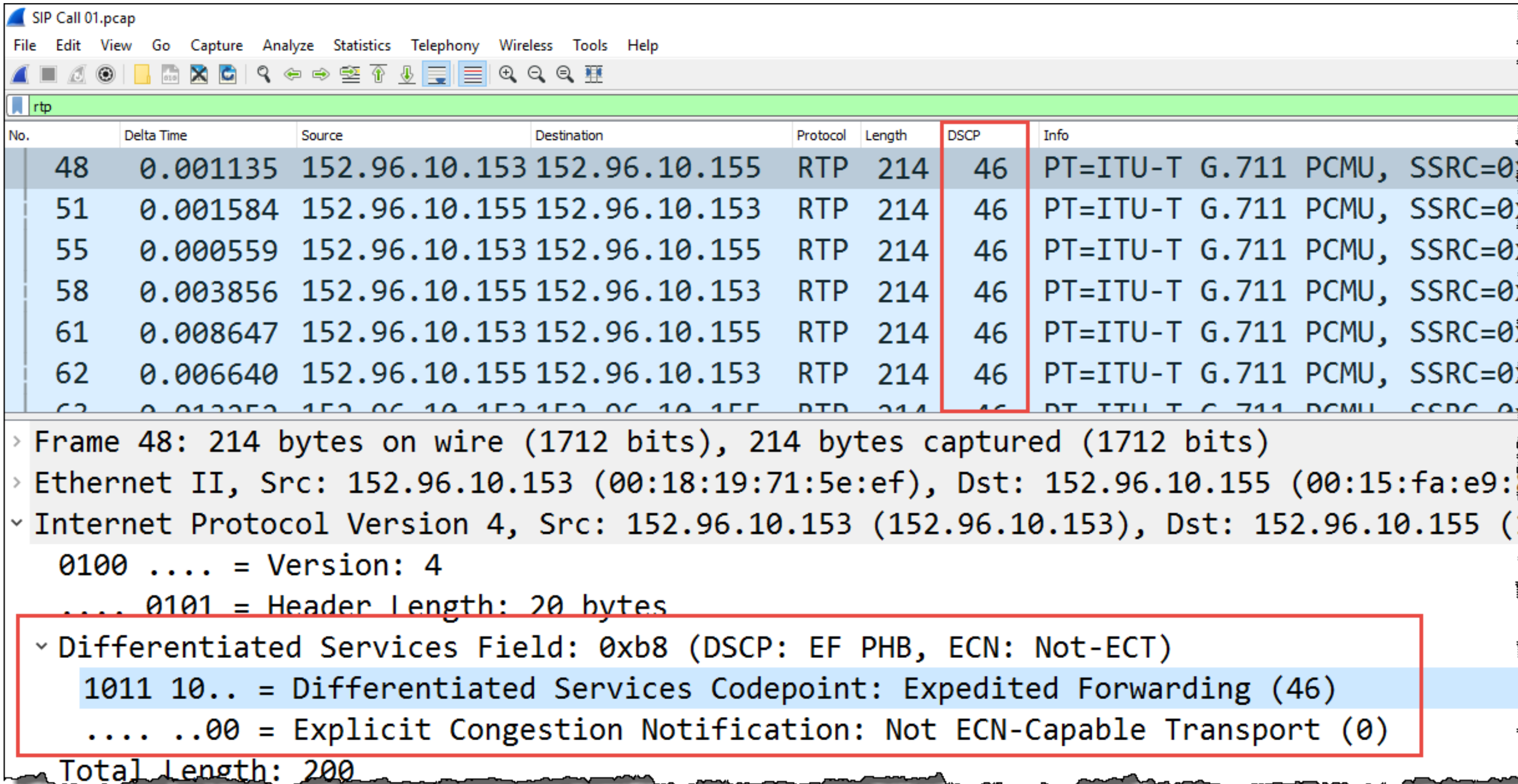
- **SIP** Session Initiation Protocol, create, modify, terminate sessions
- **SDP** Session Description Protocol, describing multimedia sessions
- **RTP** Real-Time Transport Protocol, audio and video packet format
- **RTCP** Real-Time Control Protocol, quality of reception data feedback
- **Codec** Analog/digital encodings; G.711, G.729, µ-Law, A-Law, AMR etc.



Real-Time **T**ransport **P**rotocol (RTP) features:

- **RTP** is the carrier protocol for real-time applications (Voice, Video etc.)
- **RTP** is using two streams, one in each direction and is based on UDP
- **RTP** does not address resource reservation and does not guarantee quality-of-service
- **RTP** does not provide any features to ensure timely delivery or mechanism to avoid jitter
- **RTP** contains the codec used (Payload Type), the sequence number (for sequential reassembly), the timestamp and the SSRC
- **RTP** carries the Synchronization source identifier SSRC which uniquely identifies the source of a stream.

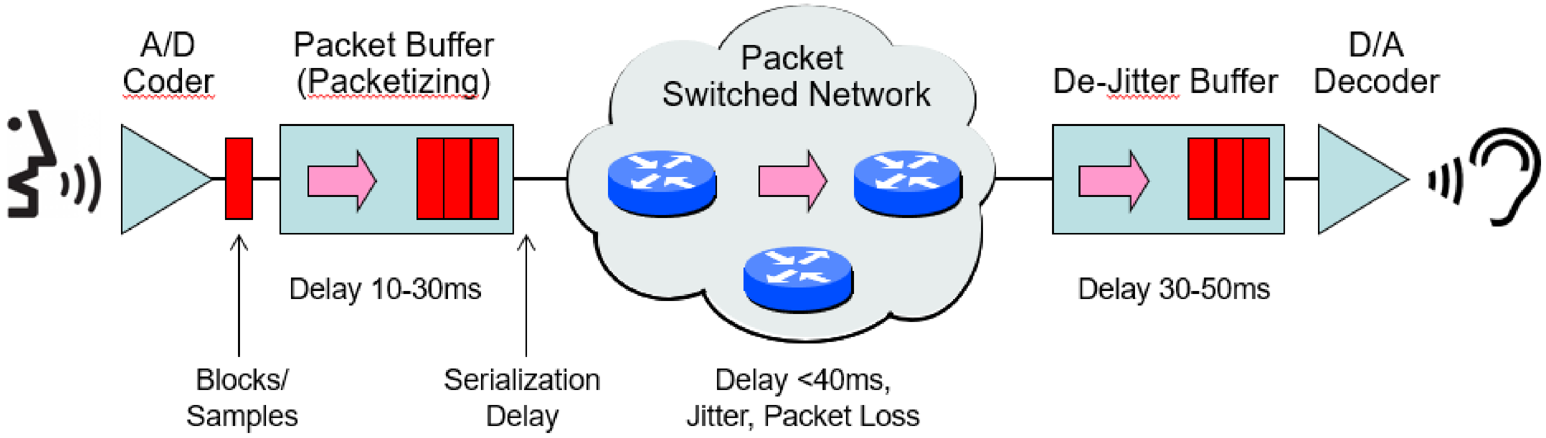
Adding QOS column



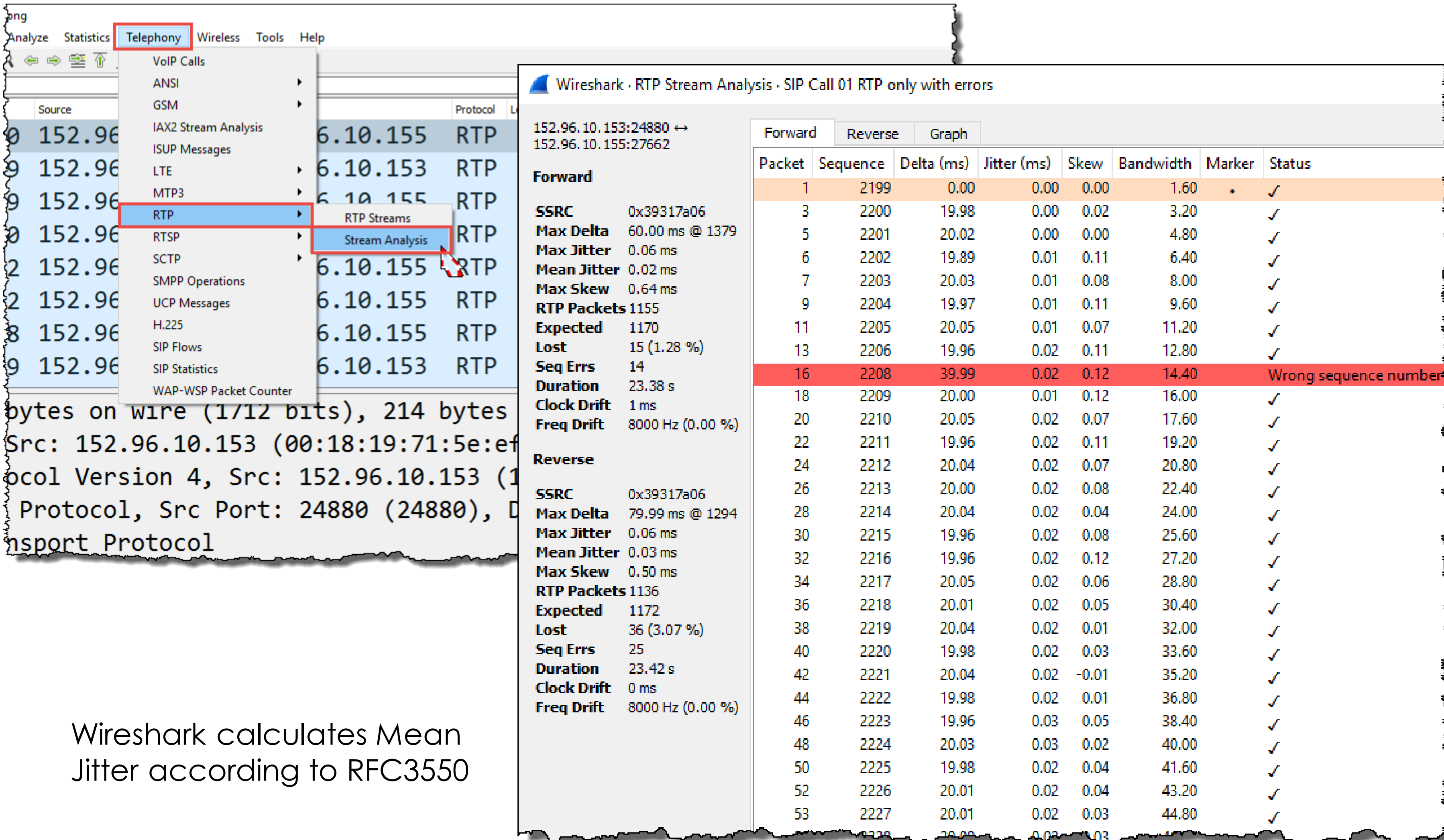
DiffServ (Differentiated Services Code Point) Header
Bits 0-5: DSCP (Differentiated Services Code Point)
Bits 6-7: ECN (Explicit Congestion Notification - IP Flow Control)

Delay, Jitter & Packet Loss

- Coder Delay Caused by the digital signal processor (DSP) to compress a block of digitized voice samples.
- Packetization Delay Time taken to fill a packet payload with speech blocks. Also called accumulation delay, dependant of number of blocks loaded in one packet.
- Serialization Delay Time required to clock a voice or data frame onto the network interface. It is directly related to the clock rate on the trunk (Bandwidth).
- Network Delay Caused by buffers of router and switches, ideally <40ms. QOS must be used to prioritize forwarding of voice packets.
- De-Jitter Delay Receive buffer to transform variable delay into a fixed delay.



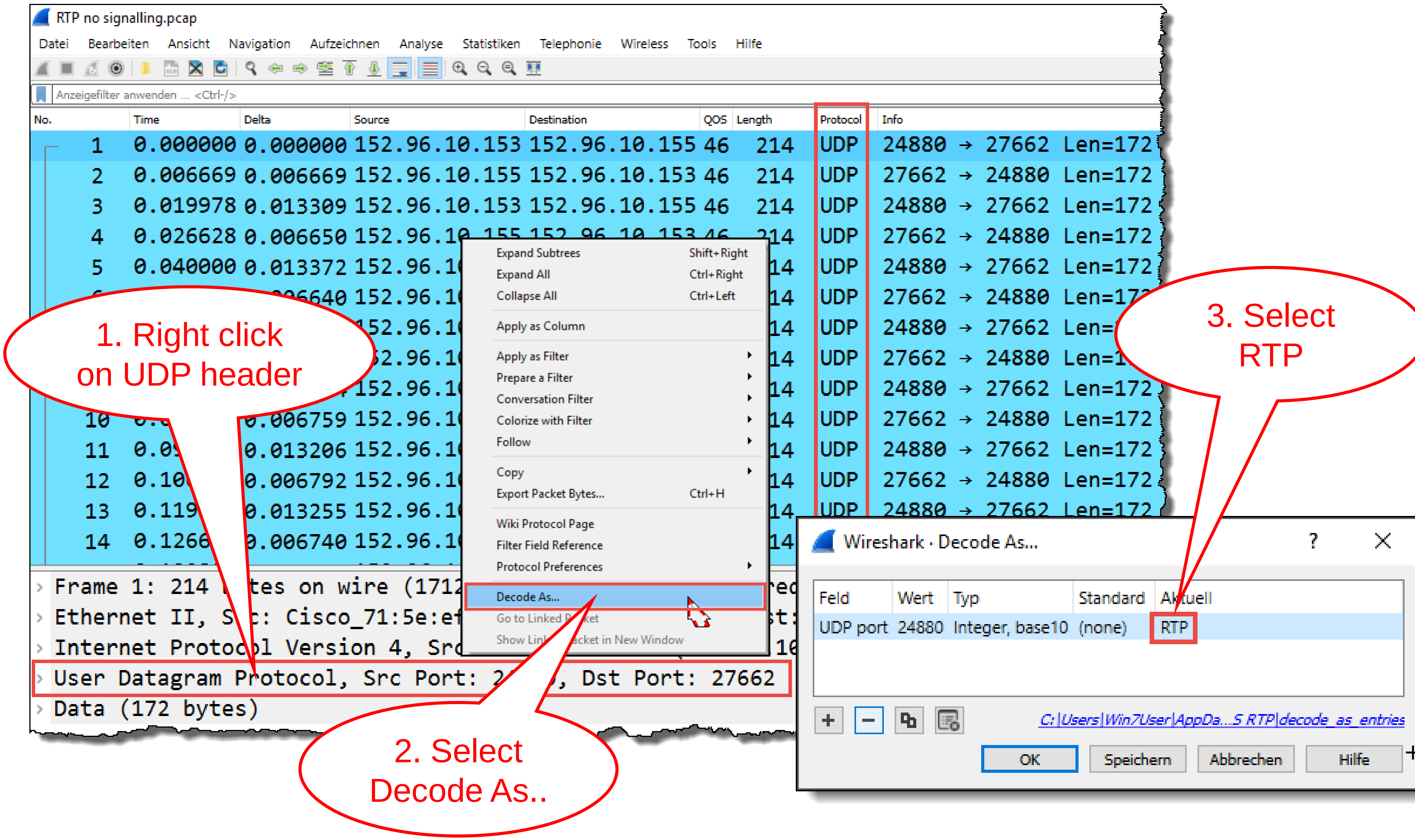
Wireshark can analyze Jitter, Packet Loss etc.



Packet	Sequence	Delta (ms)	Jitter (ms)	Skew	Bandwidth	Marker	Status
1	2199	0.00	0.00	0.00	1.60	•	✓
3	2200	19.98	0.00	0.02	3.20		✓
5	2201	20.02	0.00	0.00	4.80		✓
6	2202	19.89	0.01	0.11	6.40		✓
7	2203	20.03	0.01	0.08	8.00		✓
9	2204	19.97	0.01	0.11	9.60		✓
11	2205	20.05	0.01	0.07	11.20		✓
13	2206	19.96	0.02	0.11	12.80		✓
16	2208	39.99	0.02	0.12	14.40		Wrong sequence number
18	2209	20.00	0.01	0.12	16.00		✓
20	2210	20.05	0.02	0.07	17.60		✓
22	2211	19.96	0.02	0.11	19.20		✓
24	2212	20.04	0.02	0.07	20.80		✓
26	2213	20.00	0.02	0.08	22.40		✓
28	2214	20.04	0.02	0.04	24.00		✓
30	2215	19.96	0.02	0.08	25.60		✓
32	2216	19.96	0.02	0.12	27.20		✓
34	2217	20.05	0.02	0.06	28.80		✓
36	2218	20.01	0.02	0.05	30.40		✓
38	2219	20.04	0.02	0.01	32.00		✓
40	2220	19.98	0.02	0.03	33.60		✓
42	2221	20.04	0.02	-0.01	35.20		✓
44	2222	19.98	0.02	0.01	36.80		✓
46	2223	19.96	0.03	0.05	38.40		✓
48	2224	20.03	0.03	0.02	40.00		✓
50	2225	19.98	0.02	0.04	41.60		✓
52	2226	20.01	0.02	0.04	43.20		✓
53	2227	20.01	0.02	0.03	44.80		✓

Wireshark calculates Mean Jitter according to RFC3550

Forcing Wireshark to decode RTP



Hope you learned something useful!

VoIP Trainings with Wireshark available all over Europe