
Praktikum Rechnernetze

Protokoll zu Versuch 2 (Protokollanalyse mit Wireshark)
von Gruppe 1

Jakob Waibel, Daniel Hiller, Elia Wüstner, Felix Pojtinger

2021-10-19

Inhaltsverzeichnis

1	Einführung	3
1.1	Mitwirken	3
1.2	Lizenz	3
2	Wireshark	4
2.1	Einführung	4
2.2	Ping	6
2.3	DHCP	7
2.4	DNS	9
2.5	ARP	10
2.6	Layer-2-Protokolle	11
2.7	HTTP und TCP	13
2.8	MAC	13
2.9	STP	16
2.10	SNMP	16
2.11	Streaming and Downloads	16
2.12	Telnet und SSH	17
2.13	Wireshark-Filter	18

1 Einführung

1.1 Mitwirken

Diese Materialien basieren auf [Professor Kiefers “Praktikum Rechnernetze”-Vorlesung der HdM Stuttgart](#).

Sie haben einen Fehler gefunden oder haben einen Verbesserungsvorschlag? Bitte eröffnen Sie ein Issue auf GitHub (github.com/pojntfx/uni-netpractice-notes):



Abbildung 1: QR-Code zum Quelltext auf GitHub

Wenn Ihnen die Materialien gefallen, würden wir uns über einen GitHub-Stern sehr freuen.

1.2 Lizenz

Dieses Dokument und der enthaltene Quelltext ist freie Kultur bzw. freie Software.



Abbildung 2: Badge der AGPL-3.0-Lizenz

Uni Network Practice Notes (c) 2021 Jakob Waibel, Daniel Hiller, Elia Wüstner, Felix Pojtinger

SPDX-License-Identifier: AGPL-3.0

2 Wireshark

2.1 Einführung

An welchem Koppellement im Systemschrank sollte der Hardware-Analysator/Netzwerk-Sniffer sinnvollerweise angeschlossen werden und warum? Welche grundsätzlichen Möglichkeiten gibt es noch?

- Switch, damit Nachrichten auf Layer 2 auch abgefangen werden können
- Grundsätzlich könnte, vor allem auch in Heimnetzwerken, der Router hierzu verwendet werden, da hier oft Router und Switch zu einem Gerät kombiniert sind.

Starten Sie Wireshark und capturern Sie den aktuellen Traffic. Dokumentieren Sie zunächst, was alles auf Wireshark einprasselt.

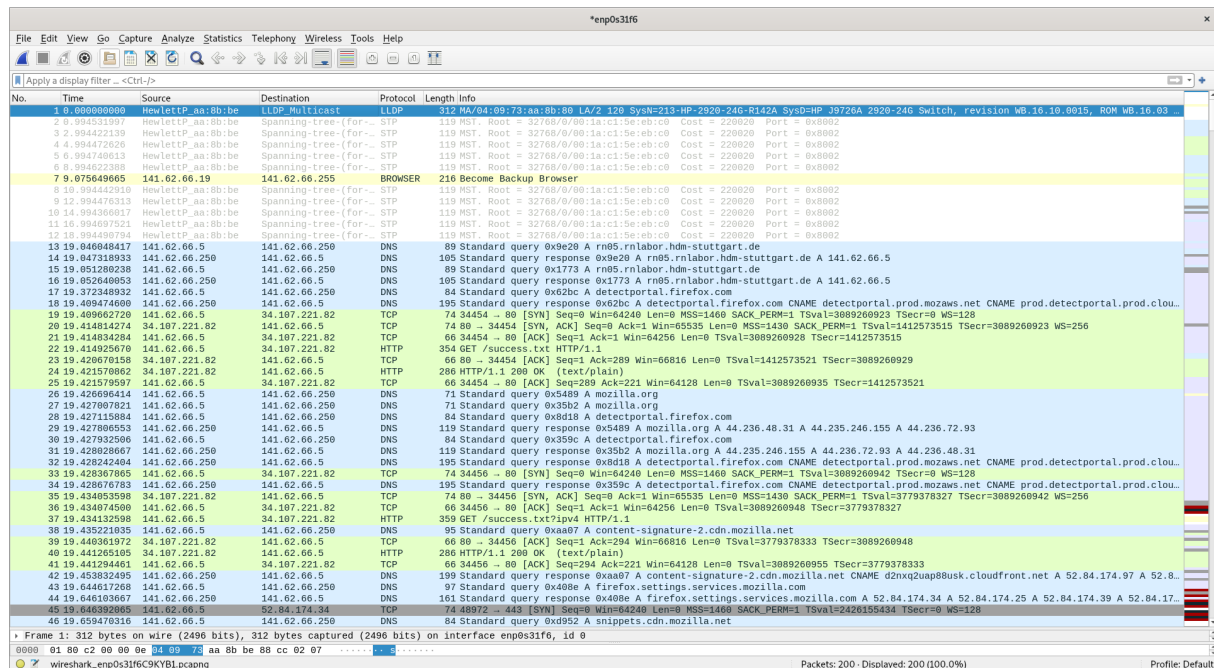


Abbildung 3: Screenshot von Wireshark

Zu erkennen sind Pakete von mehreren Protokollen:

- LLDP
- Spanning-Tree-Protokoll (STP)
- DNS
- TCP

- HTTP

Die letzten beiden Protokolle (TCP, HTTP) lassen sich durch das Öffnen des Browsers erklären.

Wie lautet der Filter, mit dem Sie ihre eigene Verbindung ins Labor ausklammern? Welche Möglichkeiten gibt es?

Hierzu gibt es mehrere Optionen:

```
1 !ip.addr == 141.62.66.5
2 not ip.addr == 141.62.66.5
3 !ip.addr eq 141.62.66.5
```

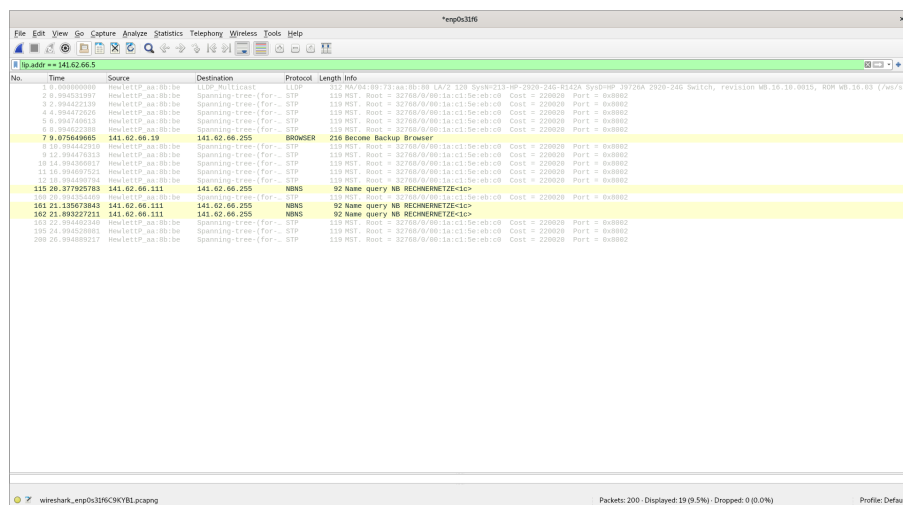


Abbildung 4: Ausklammern der eig. IP, Option 1

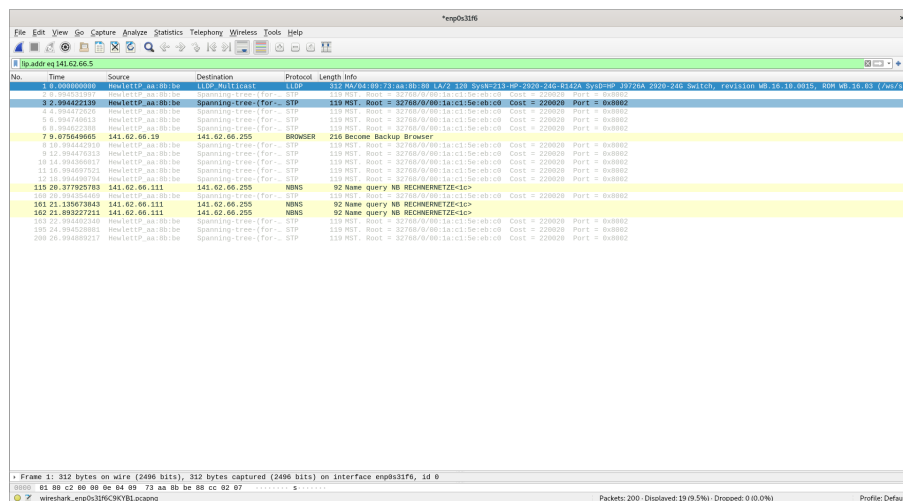
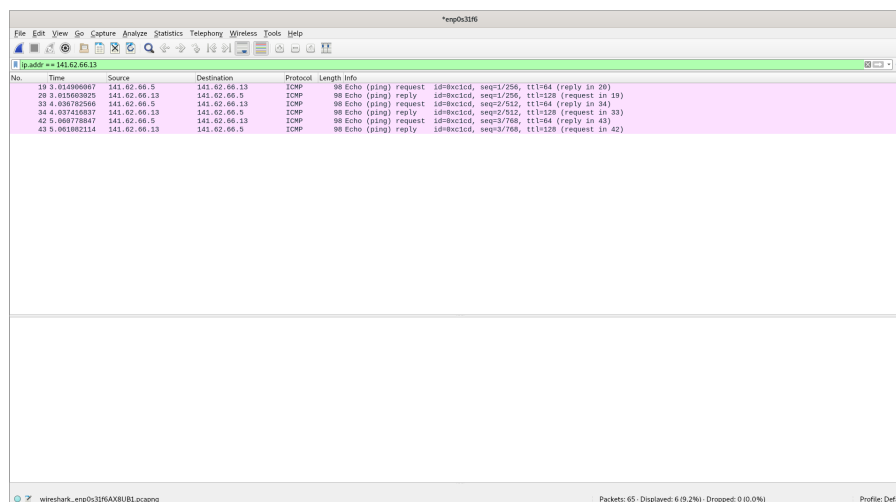


Abbildung 5: Ausklammern der eig. IP, Option 2

2.2 Ping

Senden Sie einen Ping zu nachfolgenden Empfängern und zeichnen Sie die entsprechenden Protokolle gezielt mit Wireshark auf. Vergleichen Sie die Protokollabläufe: wer sendet welches Protokoll warum an wen? Pingen Sie an

Einen Rechner Ihrer Wahl im Labornetz:



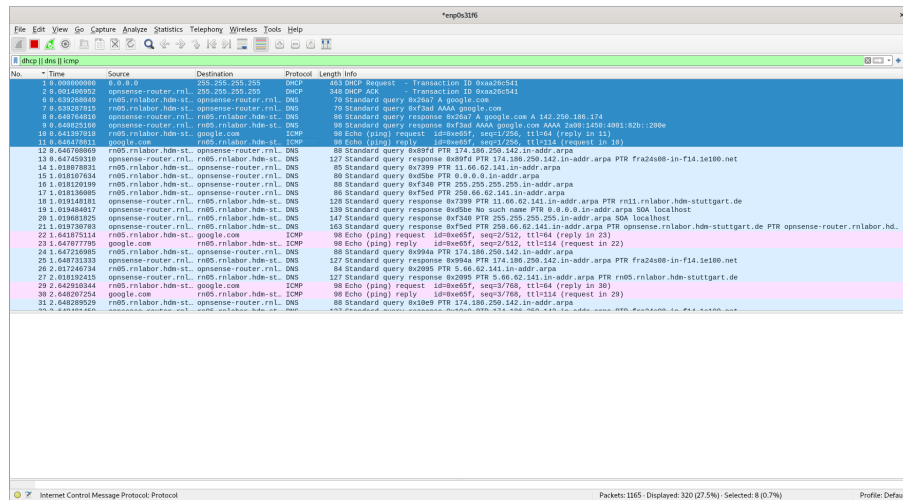
The image shows a Wireshark packet capture window titled '*tcp03116'. The filter bar at the top shows 'ip.addr == 141.62.66.13'. The packet list contains six entries, alternating between requests and replies. The packet details pane is empty. The status bar at the bottom indicates 'Packets: 65 - Displayed: 6 (9.2%) - Dropped: 0 (0.0%)' and 'Profile: Default'.

No.	Time	Source	Destination	Protocol	Length	Info
19	0.014066087	141.62.66.5	141.62.66.13	ICMP	98	Echo (ping) request id=8xc1cd, seq=1/256, ttl=64 (reply in 28)
28	0.015063920	141.62.66.13	141.62.66.5	ICMP	98	Echo (ping) reply id=8xc1cd, seq=1/256, ttl=128 (request in 19)
33	4.000782066	141.62.66.5	141.62.66.13	ICMP	98	Echo (ping) request id=8xc1cd, seq=2/512, ttl=64 (reply in 34)
34	4.007418937	141.62.66.13	141.62.66.5	ICMP	98	Echo (ping) reply id=8xc1cd, seq=2/512, ttl=128 (request in 33)
42	5.000778947	141.62.66.5	141.62.66.13	ICMP	98	Echo (ping) request id=8xc1cd, seq=3/768, ttl=64 (reply in 43)
43	5.001802114	141.62.66.13	141.62.66.5	ICMP	98	Echo (ping) reply id=8xc1cd, seq=3/768, ttl=128 (request in 42)

Abbildung 6: Wireshark-Output zu einem Rechner im Labornetz

Einen beliebigen Server im Internet (Google)

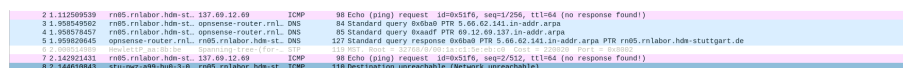
Wir haben hierzu die Name Resolution aktiviert, damit die IPs zur Domain [google.com](https://www.google.com) zugeordnet werden können.



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.1	255.255.255.255	DHCP	128	DHCP Request - Transaction ID 0xaa26c541
2	0.00140692	opnsense-router.rn1	255.255.255.255	DHCP	248	DHCP ACK - Transaction ID 0xaa26c541
8	0.00260484	rn05.rn1labor.hdm-st	opnsense-router.rn1	DNS	70	Standard query 8x5d5f A google.com
7	0.003297810	rn05.rn1labor.hdm-st	opnsense-router.rn1	DNS	70	Standard query 8x5fad AAA google.com
8	0.003704551	opnsense-router.rn1	rn05.rn1labor.hdm-st	DNS	85	Standard query response 8x5d5f A google.com A 142.250.135.174
9	0.048023160	opnsense-router.rn1	rn05.rn1labor.hdm-st	DNS	88	Standard query response 8x5fad AAA google.com AAA 2089:1458:4091:02b:120e
16	0.041307055	rn05.rn1labor.hdm-st	google.com	TCP	60	Echo (ping) request -> 8x5d5f, seq=1256, ttl=64 (reply in 14)
17	0.04137011	google.com	rn05.rn1labor.hdm-st	TCP	60	Echo (ping) reply -> 8x5d5f, seq=1256, ttl=112 (request in 16)
18	0.04709066	rn05.rn1labor.hdm-st	opnsense-router.rn1	DNS	88	Standard query 8x5d5f PTR 174.186.250.142 in-addr.arpa
19	0.047459310	opnsense-router.rn1	rn05.rn1labor.hdm-st	DNS	127	Standard query response 8x5d5f PTR 174.186.250.142 in-addr.arpa PTR fra24608-in-f14.1e100.net
14	0.00789024	rn05.rn1labor.hdm-st	opnsense-router.rn1	DNS	85	Standard query 8x7309 PTR 11.06.02.141 in-addr.arpa
15	0.00187634	rn05.rn1labor.hdm-st	opnsense-router.rn1	DNS	89	Standard query 8x5d5e PTR 0.0.0.0 in-addr.arpa
16	0.00120139	rn05.rn1labor.hdm-st	opnsense-router.rn1	DNS	88	Standard query 8x5fad PTR 250.255.255.255 in-addr.arpa
17	0.001316095	rn05.rn1labor.hdm-st	opnsense-router.rn1	DNS	86	Standard query 8x5fad PTR 250.66.62.141 in-addr.arpa
18	0.00194109	opnsense-router.rn1	rn05.rn1labor.hdm-st	DNS	120	Standard query response 8x7309 PTR 11.06.02.141 in-addr.arpa PTR rn11.rn1labor.hdm-stuttgart.de
19	0.00146481	opnsense-router.rn1	rn05.rn1labor.hdm-st	DNS	138	Standard query response 8x5d5e no such name PTR 0.0.0.0 in-addr.arpa 00A localhost
20	0.00180325	opnsense-router.rn1	rn05.rn1labor.hdm-st	DNS	147	Standard query response 8x5fad PTR 255.255.255.255 in-addr.arpa 00A localhost
21	0.00170781	opnsense-router.rn1	rn05.rn1labor.hdm-st	DNS	163	Standard query response 8x5fad PTR 250.66.62.141 in-addr.arpa PTR opnsense.rn1labor.hdm-stuttgart.de PTR opnsense-router.rn1labor.hdm-stuttgart.de
22	0.00187514	rn05.rn1labor.hdm-st	google.com	ICMP	60	Echo (ping) request -> 8x5d5f, seq=2512, ttl=64 (reply in 23)
23	0.00197770	google.com	rn05.rn1labor.hdm-st	ICMP	60	Echo (ping) reply -> 8x5d5f, seq=2512, ttl=114 (request in 22)
24	0.001721095	rn05.rn1labor.hdm-st	opnsense-router.rn1	DNS	88	Standard query 8x59aa PTR 174.186.250.142 in-addr.arpa
25	0.001871333	opnsense-router.rn1	rn05.rn1labor.hdm-st	DNS	127	Standard query response 8x59aa PTR 174.186.250.142 in-addr.arpa PTR fra24608-in-f14.1e100.net
26	0.001740734	rn05.rn1labor.hdm-st	opnsense-router.rn1	DNS	84	Standard query 8x2095 PTR 5.06.62.141 in-addr.arpa
27	0.001924215	opnsense-router.rn1	rn05.rn1labor.hdm-st	DNS	127	Standard query response 8x2095 PTR 5.06.62.141 in-addr.arpa PTR rn05.rn1labor.hdm-stuttgart.de
28	0.001910344	rn05.rn1labor.hdm-st	google.com	ICMP	60	Echo (ping) request -> 8x5d5f, seq=3768, ttl=64 (reply in 29)
29	0.001907294	google.com	rn05.rn1labor.hdm-st	ICMP	60	Echo (ping) reply -> 8x5d5f, seq=3768, ttl=114 (request in 28)
30	0.048209528	rn05.rn1labor.hdm-st	opnsense-router.rn1	DNS	88	Standard query 8x5d59 PTR 174.186.250.142 in-addr.arpa

Abbildung 7: Wireshark-Output zu einem Ping nach google.com

Eine beliebige nicht existierenden IP-Adresse



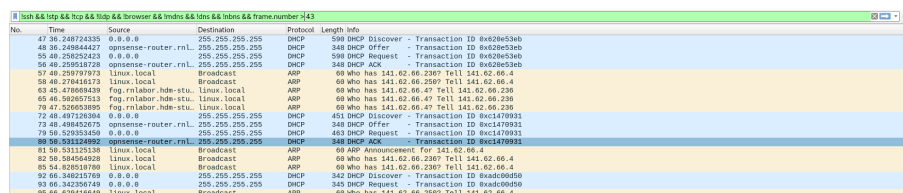
No.	Time	Source	Destination	Protocol	Length	Info
2	1.12505930	rn05.rn1labor.hdm-st	137.69.12.69	ICMP	98	Echo (ping) request -> 8x515f, seq=1256, ttl=64 (no response found)
3	1.905849502	rn05.rn1labor.hdm-st	opnsense-router.rn1	DNS	84	Standard query 8x5ba0 PTR 5.06.62.141 in-addr.arpa
4	1.905701603	rn05.rn1labor.hdm-st	opnsense-router.rn1	DNS	85	Standard query 8x5ba0 PTR 0.0.0.0 in-addr.arpa
5	1.905820845	opnsense-router.rn1	rn05.rn1labor.hdm-st	DNS	127	Standard query response 8x5ba0 PTR 5.06.62.141 in-addr.arpa PTR rn05.rn1labor.hdm-stuttgart.de
7	2.142921433	rn05.rn1labor.hdm-st	137.69.12.69	ICMP	98	Echo (ping) request -> 8x515f, seq=2512, ttl=64 (no response found)
8	2.148650845	stu-mz-499-hub-3-8	rn05.rn1labor.hdm-st	ICMP	118	Destination unreachable (Network unreachable)

Abbildung 8: Wireshark-Output zu einem Ping nach 137.69.12.69

2.3 DHCP

Analysieren Sie die Abläufe bei DHCP (im Labor installiert). Ihre Teilgruppe am Nachbartisch bootet den PC am Arbeitsplatz, protokollieren Sie die DHCP-Abläufe sowie sonstigen Netzwerkverkehr, den der PC bis zum Erhalt der IP-Adresse erzeugt.

TODO: Add descriptions



No.	Time	Source	Destination	Protocol	Length	Info
47	36.248712335	0.0.0.0	255.255.255.255	DHCP	240	DHCP Discover - Transaction ID 8x628e3eb
48	36.248844427	opnsense-router.rn1	255.255.255.255	DHCP	248	DHCP Offer - Transaction ID 8x628e3eb
55	48.209524223	0.0.0.0	255.255.255.255	DHCP	240	DHCP Request - Transaction ID 8x628e3eb
56	48.209518728	opnsense-router.rn1	255.255.255.255	DHCP	248	DHCP ACK - Transaction ID 8x628e3eb
57	48.259797979	Linux.local	Broadcast	ARP	60	Who has 141.62.66.230? Tell 141.62.66.4
58	48.278416173	Linux.local	Broadcast	ARP	60	Who has 141.62.66.230? Tell 141.62.66.4
63	45.478696439	rog.rn1labor.hdm-stu	Linux.local	ARP	60	Who has 141.62.66.4? Tell 141.62.66.230
65	46.569657553	rog.rn1labor.hdm-stu	Linux.local	ARP	60	Who has 141.62.66.4? Tell 141.62.66.230
79	47.526538895	rog.rn1labor.hdm-stu	Linux.local	ARP	60	Who has 141.62.66.4? Tell 141.62.66.230
72	48.197126384	0.0.0.0	255.255.255.255	DHCP	451	DHCP Discover - Transaction ID 8xc1470931
73	48.49842675	opnsense-router.rn1	255.255.255.255	DHCP	248	DHCP Offer - Transaction ID 8xc1470931
79	58.529534500	0.0.0.0	255.255.255.255	DHCP	463	DHCP Request - Transaction ID 8xc1470931
80	58.53124992	opnsense-router.rn1	255.255.255.255	DHCP	248	DHCP ACK - Transaction ID 8xc1470931
81	58.531215136	Linux.local	Broadcast	ARP	60	ARP Announcement For 141.62.66.4
82	59.584564928	Linux.local	Broadcast	ARP	60	Who has 141.62.66.230? Tell 141.62.66.4
85	54.820518700	Linux.local	Broadcast	ARP	60	Who has 141.62.66.230? Tell 141.62.66.4
92	66.348215709	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover - Transaction ID 8xad00450
93	66.342386749	0.0.0.0	255.255.255.255	DHCP	342	DHCP Request - Transaction ID 8xad00450
95	66.829416649	Linux.local	Broadcast	ARP	60	Who has 141.62.66.230? Tell 141.62.66.4

Abbildung 9: Gesamter Bootprozess

Abbildung 10: Bootprozess: DHCP-Requests des BIOS zum Netzwerkboot

Abbildung 11: Bootprozess: DHCP-Requests des Netzwerbootloaders iPXE

TODO: Add answer

Mittels der folgenden Commands wurde eine IP-Adresse freigegeben und eine neue angefordert.

- 8

No.	Time	Source	Destination	Protocol	Length	Info
19	15.392845861	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover - Transaction ID 0x79ef81d
20	15.393517126	0.0.0.0	255.255.255.255	DHCP	342	DHCP Request - Transaction ID 0x79ef81d
21	15.468881886	linux.local	Broadcast	ARP	68	Who has 141.62.66.250? Tell 141.62.66.4

TODO: Add description (no BIOS and iPXE DHCP requests)

2.4 DNS

Dokumentieren Sie den Ablauf bei einer DNS-Abfrage

Fall 1: DNS-Server 141.62.66.250:

Mittels folgendem Command wurde eine DNS-Abfrage gemacht:

```
1 $ dig @141.62.66.250 google.com
2 google.com.      163 IN  A      142.250.186.174
```

No.	Time	Source	Destination	Protocol	Length	Info
11	1.357358000	rn05.rnlabor.hdm-st.	opnsense-router.rnl.	DNS	93	Standard query 0xa276 A google.com OPT
12	1.371692978	opnsense-router.rnl.	rn05.rnlabor.hdm-st.	DNS	97	Standard query response 0xa276 A google.com A 142.250.186.174 OPT

Abbildung 12: Ablauf der Anfrage

TODO: Add interpretation

Fall 2: DNS-Server 1.1.1.1 (Cloudflare):

Mittels folgendem Command wurde eine DNS-Abfrage gemacht:

```
1 $ dig @1.1.1.1 +noall +answer google.com
2 google.com.      231 IN  A      142.250.185.110
```

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	rn05.rnlabor.hdm-st.	one.one.one.one	DNS	93	Standard query 0x6247 A google.com OPT
2	0.005982835	one.one.one.one	rn05.rnlabor.hdm-st.	DNS	97	Standard query response 0x6247 A google.com A 142.250.185.110 OPT
4	1.205820780	rn05.rnlabor.hdm-st.	opnsense-router.rnl.	DNS	84	Standard query 0xda2b PTR 5.66.62.141.in-addr.arpa
5	1.205848397	rn05.rnlabor.hdm-st.	opnsense-router.rnl.	DNS	88	Standard query 0x0083 PTR 1.1.1.1.in-addr.arpa
6	1.207179251	opnsense-router.rnl.	rn05.rnlabor.hdm-st.	DNS	127	Standard query response 0xda2b PTR 5.66.62.141.in-addr.arpa PTR rn05.rnlabor.hdm-stuttgart.de
7	1.207611330	opnsense-router.rnl.	rn05.rnlabor.hdm-st.	DNS	109	Standard query response 0x0083 PTR 1.1.1.1.in-addr.arpa PTR one.one.one.one

Abbildung 13: Ablauf der Anfrage

TODO: Add interpretation

Fall 3: DNS-Server 8.8.8.9 (DNS-Dienst ist dort nicht installiert):

Mittels folgendem Command wurde eine DNS-Abfrage gemacht:

```
1 $ dig @8.8.8.9 +noall +answer google.com
2 ;; connection timed out; no servers could be reached
```

No.	Time	Source	Destination	Protocol	Length	Info
3	0.572496372	rn05.rnlabor.hdm-st...	8.8.8.9	DNS	93	Standard query 0x73f9 A google.com OPT
5	1.088436116	rn05.rnlabor.hdm-st...	opnsense.rnlabor.hdm...	DNS	84	Standard query 0xc668 PTR 5.66.62.141.in-addr.arpa
6	1.088465401	rn05.rnlabor.hdm-st...	opnsense.rnlabor.hdm...	DNS	88	Standard query 0x74b6 PTR 9.8.8.8.in-addr.arpa
7	1.089961823	opnsense.rnlabor.hdm...	rn05.rnlabor.hdm-st...	DNS	127	Standard query response 0xc668 PTR 5.66.62.141.in-addr.arpa PTR rn05.rnlabor.hdm-stuttgart.de
8	1.089962625	opnsense.rnlabor.hdm...	rn05.rnlabor.hdm-st...	DNS	148	Standard query response 0x74b6 No such name PTR 9.8.8.8.in-addr.arpa SOA ns1.google.com
13	2.087996907	rn05.rnlabor.hdm-st...	opnsense.rnlabor.hdm...	DNS	86	Standard query 0x4f6b PTR 250.66.62.141.in-addr.arpa
17	2.089269013	opnsense.rnlabor.hdm...	rn05.rnlabor.hdm-st...	DNS	163	Standard query response 0x4f6b PTR 250.66.62.141.in-addr.arpa PTR opnsense-router.rnlabor.hdm-stuttgart.de PTR opnsense.rnlabor.hdm...
22	3.087916968	rn05.rnlabor.hdm-st...	opnsense.rnlabor.hdm...	DNS	86	Standard query 0x859b PTR 19.75.254.169.in-addr.arpa
23	3.087945863	rn05.rnlabor.hdm-st...	opnsense.rnlabor.hdm...	DNS	84	Standard query 0xfec6 PTR 251.0.0.224.in-addr.arpa
24	3.087950319	rn05.rnlabor.hdm-st...	opnsense.rnlabor.hdm...	DNS	88	Standard query 0x1f24 PTR 255.255.254.169.in-addr.arpa
25	3.088893145	opnsense.rnlabor.hdm...	rn05.rnlabor.hdm-st...	DNS	145	Standard query response 0x859b No such name PTR 19.75.254.169.in-addr.arpa SOA localhost
26	3.089011764	opnsense.rnlabor.hdm...	rn05.rnlabor.hdm-st...	DNS	141	Standard query response 0xfec6 No such name PTR 251.0.0.224.in-addr.arpa SOA sns.dns.icann.org
27	3.089125772	opnsense.rnlabor.hdm...	rn05.rnlabor.hdm-st...	DNS	147	Standard query response 0x1f24 No such name PTR 255.255.254.169.in-addr.arpa SOA localhost

Abbildung 14: Ablauf der Anfrage

TODO: Add interpretation

Wie erkennen Sie mit Wireshark, dass “versehentlich” ein falscher DNS-Server eingetragen wurde?

TODO: Add interpretation (based on case 3)

2.5 ARP

Lösen Sie eine ARP-Anfrage aus und protokollieren Sie die Datenpakete.

Hierzu wurde ein Rechner, welcher zuvor nicht im lokalen ARP-Cache war, neugestartet.

No.	Time	Source	Destination	Protocol	Length	Info
214	110.515578213	linux-2.local	Broadcast	ARP	42	who has 141.62.66.6? Tell 141.62.66.5
215	110.515667288	linux-3.local	linux-2.local	ARP	60	141.62.66.6 is at 4c:52:62:0e:54:2b
231	115.673164735	linux-3.local	linux-2.local	ARP	60	who has 141.62.66.5? Tell 141.62.66.6
232	115.673186783	linux-2.local	linux-3.local	ARP	42	141.62.66.5 is at 4c:52:62:0e:54:0b

Abbildung 15: Ablauf der Anfrage

Wann wird eine ARP-Anfrage gestartet?

TODO: Add interpretation

Welcher Rahmentyp wird für die Anfrage verwendet?

TODO: Add description (Ethernet II)

No.	Time	Source	Destination	Protocol	Length	Info
214	119.515575213	linux-2.local	Broadcast	ARP	42	Who has 141.62.66.6? Tell 141.62.66.6
215	119.515967298	linux-3.local	linux-2.local	ARP	60	141.62.66.6 is at 4c:52:62:0e:54:2b
231	115.673164735	linux-3.local	linux-2.local	ARP	60	Who has 141.62.66.5? Tell 141.62.66.6
232	115.673186793	linux-2.local	linux-3.local	ARP	42	141.62.66.5 is at 4c:52:62:0e:54:8b

Frame 214: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface enp0s31f6, id 0

Ethernet II, Src: linux-2.local (4c:52:62:0e:54:8b), Dst: Broadcast (ff:ff:ff:ff:ff:ff)

- Destination: Broadcast (ff:ff:ff:ff:ff:ff)
- Source: linux-2.local (4c:52:62:0e:54:8b)
- Type: ARP (0x0806)

Address Resolution Protocol (request)

Abbildung 16: Verwendetes Ethernet-Frame

Beobachten Sie die Veränderung in der ARP-Tabelle Ihres Rechners

Zuvor:

```
1 $ ip neigh show
2 141.62.66.6 dev enp0s31f6 lladdr 4c:52:62:0e:54:2b STALE
3 141.62.66.250 dev enp0s31f6 lladdr 00:0d:b9:4f:b8:14 STALE
4 141.62.66.13 dev enp0s31f6 lladdr 4c:52:62:0e:54:5d STALE
5 141.62.66.236 dev enp0s31f6 lladdr 26:c5:04:8a:fa:eb STALE
```

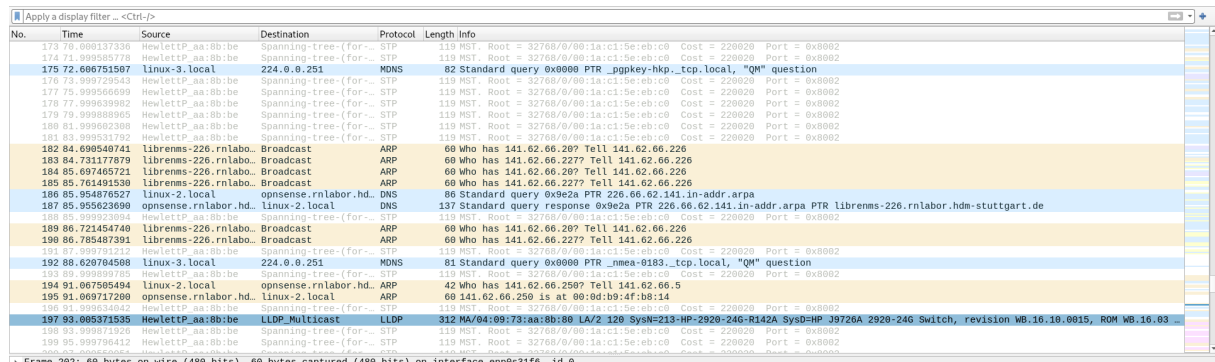
Danach:

```
1 $ ip neigh show
2 141.62.66.6 dev enp0s31f6 lladdr 4c:52:62:0e:54:2b STALE
3 141.62.66.250 dev enp0s31f6 lladdr 00:0d:b9:4f:b8:14 STALE
4 141.62.66.4 dev enp0s31f6 lladdr 4c:52:62:0e:53:eb STALE
5 141.62.66.13 dev enp0s31f6 lladdr 4c:52:62:0e:54:5d STALE
6 141.62.66.236 dev enp0s31f6 lladdr 26:c5:04:8a:fa:eb STALE
```

2.6 Layer-2-Protokolle

Gelegentlich werden vom Analyzer Broadcasts erkannt. Wer sendet sie, warum und in welchen zeitlichen Abständen?

Die Broadcasts sind ARP-Requests.



The image shows a Wireshark packet capture window with a display filter set to 'eth 0'. The packet list shows various network protocols including STP, MDNS, ARP, and DNS. The packet details pane shows the selected packet (No. 175) as a 'Standard query 0x0000 PTR _pgpkey-hkp._tcp.local, "QM" question'. The packet bytes pane shows the raw data of the selected packet.

No.	Time	Source	Destination	Protocol	Length	Info
173	79.009137336	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
174	71.009086379	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
175	72.606751507	Linux-3.local	224.0.0.251	MDNS	82	Standard query 0x0000 PTR _pgpkey-hkp._tcp.local, "QM" question
176	73.999729543	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
177	75.999566699	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
178	77.999639992	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
179	79.999889965	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
180	81.999602388	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
181	83.999631792	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
182	84.699549741	Librems-226.rnlabo	Broadcast	ARP	60	Who has 141.62.66.227? Tell 141.62.66.226
183	84.731177879	Librems-226.rnlabo	Broadcast	ARP	60	Who has 141.62.66.227? Tell 141.62.66.226
184	85.697465721	Librems-226.rnlabo	Broadcast	ARP	60	Who has 141.62.66.287? Tell 141.62.66.226
185	85.761491530	Librems-226.rnlabo	Broadcast	ARP	60	Who has 141.62.66.227? Tell 141.62.66.226
186	85.954876527	Linux-2.local	opnsense.rnlabor.hd	DNS	86	Standard query 0x9e2a PTR 226.66.62.141.in-addr.arpa
187	85.955623699	opnsense.rnlabor.hd	Linux-2.local	DNS	137	Standard query response 0x9e2a PTR 226.66.62.141.in-addr.arpa PTR Librems-226.rnlabor.hdm-stuttgart.de
188	86.629784598	Linux-3.local	224.0.0.251	MDNS	81	Standard query 0x0000 PTR _nmea-0183._tcp.local, "QM" question
189	86.721454740	Librems-226.rnlabo	Broadcast	ARP	60	Who has 141.62.66.287? Tell 141.62.66.226
190	86.785487391	Librems-226.rnlabo	Broadcast	ARP	60	Who has 141.62.66.227? Tell 141.62.66.226
191	87.999791212	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
192	88.629784598	Linux-3.local	224.0.0.251	MDNS	81	Standard query 0x0000 PTR _nmea-0183._tcp.local, "QM" question
193	89.999899785	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
194	91.067595494	Linux-2.local	opnsense.rnlabor.hd	ARP	42	Who has 141.62.66.250? Tell 141.62.66.5
195	91.069717290	opnsense.rnlabor.hd	Linux-2.local	ARP	60	141.62.66.250 is at 00:0d:b9:4f:b8:14
196	91.999631792	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
197	93.005371535	HewlettP_aa:8b:be	LLDP_Multicast	LLDP	312	MA/04:09:73:aa:8b:80 LA/2 120 SysN=213-HP-2920-24G-R142A SysD=HP J9726A 2920-24G Switch, revision WB.16.10.0015, ROM WB.16.03
198	93.999611926	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
199	95.999796412	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002

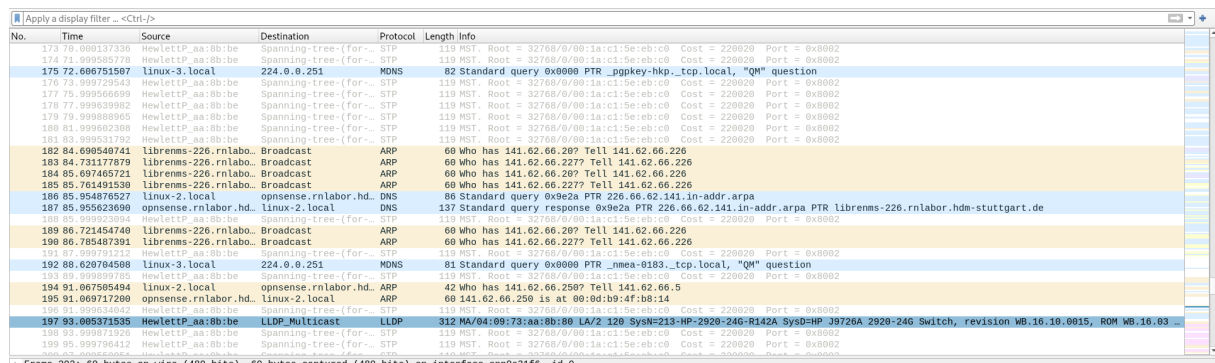
Abbildung 17: Aufzeichnung der ARP-Requests

TODO: Add interpretation

Haben Sie noch weitere Protokolle “eingefangen”, die offensichtlich im Labor Rechnernetze keinen Sinn machen?

NMEA 0183.

TODO: Add interpretation



The image shows a Wireshark packet capture window with a display filter set to 'eth 0'. The packet list shows various network protocols including STP, MDNS, ARP, and DNS. The packet details pane shows the selected packet (No. 175) as a 'Standard query 0x0000 PTR _pgpkey-hkp._tcp.local, "QM" question'. The packet bytes pane shows the raw data of the selected packet.

No.	Time	Source	Destination	Protocol	Length	Info
173	79.009137336	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
174	71.009086379	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
175	72.606751507	Linux-3.local	224.0.0.251	MDNS	82	Standard query 0x0000 PTR _pgpkey-hkp._tcp.local, "QM" question
176	73.999729543	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
177	75.999566699	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
178	77.999639992	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
179	79.999889965	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
180	81.999602388	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
181	83.999631792	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
182	84.699549741	Librems-226.rnlabo	Broadcast	ARP	60	Who has 141.62.66.287? Tell 141.62.66.226
183	84.731177879	Librems-226.rnlabo	Broadcast	ARP	60	Who has 141.62.66.227? Tell 141.62.66.226
184	85.697465721	Librems-226.rnlabo	Broadcast	ARP	60	Who has 141.62.66.287? Tell 141.62.66.226
185	85.761491530	Librems-226.rnlabo	Broadcast	ARP	60	Who has 141.62.66.227? Tell 141.62.66.226
186	85.954876527	Linux-2.local	opnsense.rnlabor.hd	DNS	86	Standard query 0x9e2a PTR 226.66.62.141.in-addr.arpa
187	85.955623699	opnsense.rnlabor.hd	Linux-2.local	DNS	137	Standard query response 0x9e2a PTR 226.66.62.141.in-addr.arpa PTR Librems-226.rnlabor.hdm-stuttgart.de
188	86.629784598	Linux-3.local	224.0.0.251	MDNS	81	Standard query 0x0000 PTR _nmea-0183._tcp.local, "QM" question
189	86.721454740	Librems-226.rnlabo	Broadcast	ARP	60	Who has 141.62.66.287? Tell 141.62.66.226
190	86.785487391	Librems-226.rnlabo	Broadcast	ARP	60	Who has 141.62.66.227? Tell 141.62.66.226
191	87.999791212	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
192	88.629784598	Linux-3.local	224.0.0.251	MDNS	81	Standard query 0x0000 PTR _nmea-0183._tcp.local, "QM" question
193	89.999899785	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
194	91.067595494	Linux-2.local	opnsense.rnlabor.hd	ARP	42	Who has 141.62.66.250? Tell 141.62.66.5
195	91.069717290	opnsense.rnlabor.hd	Linux-2.local	ARP	60	141.62.66.250 is at 00:0d:b9:4f:b8:14
196	91.999631792	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
197	93.005371535	HewlettP_aa:8b:be	LLDP_Multicast	LLDP	312	MA/04:09:73:aa:8b:80 LA/2 120 SysN=213-HP-2920-24G-R142A SysD=HP J9726A 2920-24G Switch, revision WB.16.10.0015, ROM WB.16.03
198	93.999611926	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002
199	95.999796412	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220928 Port = 0x8002

Abbildung 18: Aufzeichnung der ARP-Requests; hier ist das Protokoll zu sehen

Wie sieht es mit UPnP im Labor aus? Auf welchen Maschinen von welchem Hersteller läuft der Dienst? Mit welchem Wireshark-Filter „fischen“ Sie den Traffic heraus?

TODO: Re-start this experiment once the network is back up

No.	Time	Source	Destination	Protocol	Length	Info
826	235.113864599	fe80::5e49:79ff:fe6...	ff02::c	SSDP	365	NOTIFY * HTTP/1.1
827	235.115078419	fe80::5e49:79ff:fe6...	ff02::c	SSDP	375	NOTIFY * HTTP/1.1
828	235.115520626	fe80::5e49:79ff:fe6...	ff02::c	SSDP	411	NOTIFY * HTTP/1.1
829	235.117651813	fe80::5e49:79ff:fe6...	ff02::c	SSDP	411	NOTIFY * HTTP/1.1
839	240.109859521	fe80::5e49:79ff:fe6...	ff02::c	SSDP	363	NOTIFY * HTTP/1.1
840	240.110184287	fe80::5e49:79ff:fe6...	ff02::c	SSDP	372	NOTIFY * HTTP/1.1
841	240.110442125	fe80::5e49:79ff:fe6...	ff02::c	SSDP	435	NOTIFY * HTTP/1.1
842	240.113785421	fe80::5e49:79ff:fe6...	ff02::c	SSDP	372	NOTIFY * HTTP/1.1
843	240.114125389	fe80::5e49:79ff:fe6...	ff02::c	SSDP	411	NOTIFY * HTTP/1.1
844	240.117673873	fe80::5e49:79ff:fe6...	ff02::c	SSDP	372	NOTIFY * HTTP/1.1
845	240.118024377	fe80::5e49:79ff:fe6...	ff02::c	SSDP	431	NOTIFY * HTTP/1.1
846	240.120316833	fe80::5e49:79ff:fe6...	ff02::c	SSDP	399	NOTIFY * HTTP/1.1
847	240.122478594	fe80::5e49:79ff:fe6...	ff02::c	SSDP	443	NOTIFY * HTTP/1.1
848	240.124712871	fe80::5e49:79ff:fe6...	ff02::c	SSDP	427	NOTIFY * HTTP/1.1
849	240.126997425	fe80::5e49:79ff:fe6...	ff02::c	SSDP	425	NOTIFY * HTTP/1.1
850	240.129151475	fe80::5e49:79ff:fe6...	ff02::c	SSDP	439	NOTIFY * HTTP/1.1
851	241.110212315	fe80::5e49:79ff:fe6...	ff02::c	SSDP	364	NOTIFY * HTTP/1.1
852	241.110541617	fe80::5e49:79ff:fe6...	ff02::c	SSDP	373	NOTIFY * HTTP/1.1
853	241.110892288	fe80::5e49:79ff:fe6...	ff02::c	SSDP	436	NOTIFY * HTTP/1.1
854	241.114209272	fe80::5e49:79ff:fe6...	ff02::c	SSDP	373	NOTIFY * HTTP/1.1
855	241.114551951	fe80::5e49:79ff:fe6...	ff02::c	SSDP	412	NOTIFY * HTTP/1.1

Frame 826: 365 bytes on wire (2920 bits), 365 bytes captured (2920 bits) on interface enp8s31f0, id 0
 Ethernet II, Src: AVMAudio_6a:a9:78 (5c:a9:79:6a:a9:78), Dst: IPv6cast_0c (33:33:00:00:00:0c)
 Internet Protocol Version 6, Src: fe80::5e49:79ff:fe6a:a978, Dst: ff02::c
 User Datagram Protocol, Src Port: 1900, Dst Port: 1900
 Simple Service Discovery Protocol

Abbildung 19: Aufzeichnung des SSDP-Protokolls

2.7 HTTP und TCP

Initiieren Sie eine HTTP-TCP-Sitzung (beliebige Website) und zeichnen Sie die Protokollabläufe auf

TODO: Add description

Können Sie den 3-Way-Handshake erkennen? Markieren Sie ihn in der Dokumentation. Welche TCP-Optionen sind beim Handshake aktiviert und welche Bedeutung haben sie?

TODO: Add description

Dokumentieren und erläutern Sie die Verwendung der Portnummern bei der Dienstanfrage und der Beantwortung des Dienstes durch den Server.

TODO: Add description

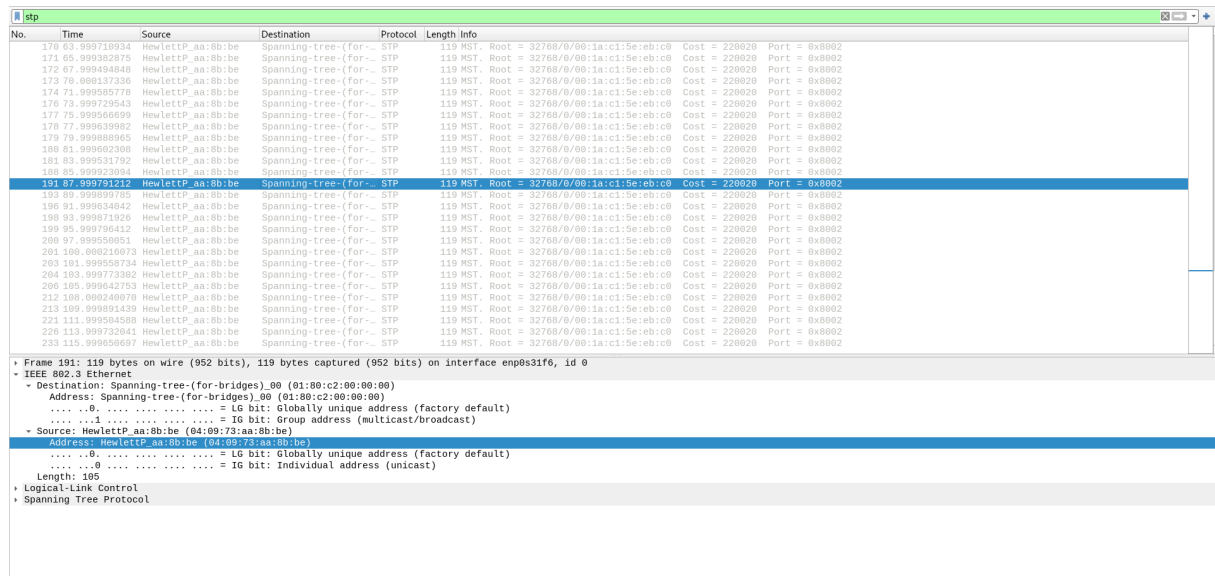
Klicken Sie auf der Website ein anderes Bild / Link an. Beobachten und dokumentieren Sie: wie verändert sich der TCP-Ablauf?

TODO: Add description

2.8 MAC

Wie lauten die MAC-Adressen der im Labor befindlichen Ethernet-Switches? Wie haben Sie die Switches identifizieren können. Welche Möglichkeiten der Identifizierung gibt es?

TODO: Add interpretation



No.	Time	Source	Destination	Protocol	Length	Info
170	63.999718934	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
171	65.999382875	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
172	67.999404846	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
173	70.009137336	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
174	71.999595778	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
176	73.999729543	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
177	75.999566699	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
178	77.999639982	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
179	79.998889965	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
180	81.999602306	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
181	83.999531792	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
186	85.999923894	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
191	67.999731217	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
193	88.999697105	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
196	91.999634042	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
198	93.999871926	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
199	95.999766412	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
200	97.999559051	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
201	100.000216073	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
203	101.999508734	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
204	103.999773302	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
206	105.999642753	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
212	108.000240078	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
213	109.999891439	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
221	111.999504586	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
226	113.999732041	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002
233	115.999508597	HewlettP_aa:8b:be	Spanning-tree-(for-...	STP	119	MST. Root = 32768/0/00:1a:c1:5e:eb:c0 Cost = 220020 Port = 0x8002

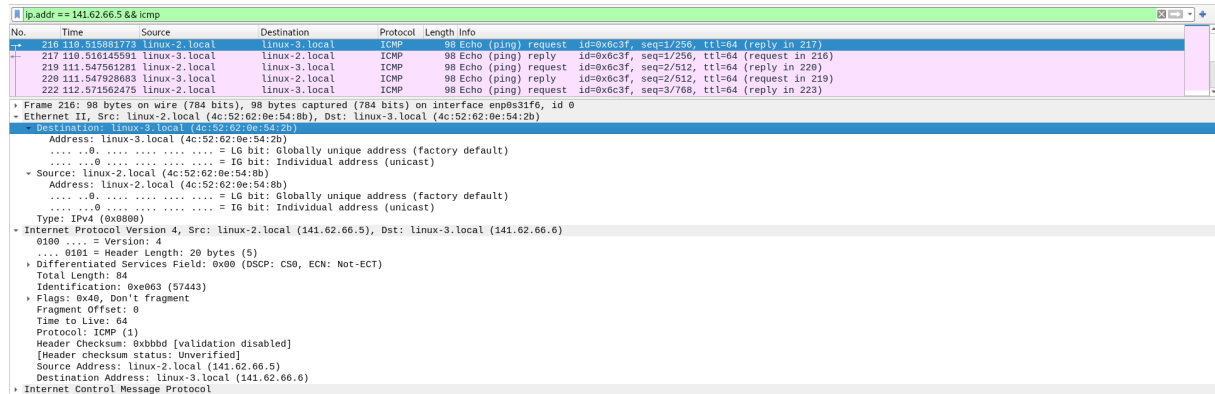
Frame 191: 119 bytes on wire (952 bits), 119 bytes captured (952 bits) on interface enp0s31f6, id 0

- IEEE 802.3 Ethernet
 - Destination: Spanning-tree-(for-bridges)_00 (01:00:c2:00:00:00)
 - Address: Spanning-tree-(for-bridges)_00 (01:00:c2:00:00:00)
 -0 = LG bit: Globally unique address (factory default)
 -1 = IG bit: group address (multicast/broadcast)
 - Source: HewlettP_aa:8b:be (04:00:73:aa:8b:be)
 - Address: HewlettP_aa:8b:be (04:00:73:aa:8b:be)
 -0 = LG bit: Globally unique address (factory default)
 -0 = IG bit: Individual address (unicast)
 - Length: 105
 - Logical-Link Control
 - Spanning Tree Protocol

Abbildung 20: Aufzeichnung des STP-Protokolls

Welche MAC-Adresse hat ihr Nachbarrechner?

TODO: Add interpretation



No.	Time	Source	Destination	Protocol	Length	Info
216	110.515881773	linux-2.local	linux-3.local	ICMP	98	Echo (ping) request id=0xc63f, seq=1/256, ttl=64 (reply in 217)
217	110.516145591	linux-3.local	linux-2.local	ICMP	98	Echo (ping) reply id=0xc63f, seq=1/256, ttl=64 (request in 216)
219	111.547561281	linux-2.local	linux-3.local	ICMP	98	Echo (ping) request id=0xc63f, seq=2/512, ttl=64 (reply in 220)
220	111.547826089	linux-3.local	linux-2.local	ICMP	98	Echo (ping) reply id=0xc63f, seq=2/512, ttl=64 (request in 219)
222	112.571562475	linux-2.local	linux-3.local	ICMP	98	Echo (ping) request id=0xc63f, seq=3/768, ttl=64 (reply in 223)

Frame 216: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface enp0s31f6, id 0

- Ethernet II, Src: linux-2.local (4c:52:62:0e:54:2b), Dst: linux-3.local (4c:52:62:0e:54:2b)
 - Destination: linux-3.local (4c:52:62:0e:54:2b)
 -0 = LG bit: Globally unique address (factory default)
 -0 = IG bit: Individual address (unicast)
 - Source: linux-2.local (4c:52:62:0e:54:2b)
 - Address: linux-2.local (4c:52:62:0e:54:2b)
 -0 = LG bit: Globally unique address (factory default)
 -0 = IG bit: Individual address (unicast)
 - Type: IPv4 (0x0800)
- Internet Protocol Version 4, Src: linux-2.local (141.62.66.5), Dst: linux-3.local (141.62.66.6)
 - 0000 = Version: 4
 - 0101 = Header Length: 20 bytes (5)
 - Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
 - Total Length: 84
 - Identification: 0xe063 (57443)
 - Flags: 0x40, Don't fragment
 - Fragment Offset: 0
 - Time to Live: 64
 - Protocol: ICMP (1)
 - Header Checksum: 0xbdbd [validation disabled]
[Header checksum status: Unverified]
 - Source Address: linux-2.local (141.62.66.5)
 - Destination Address: linux-3.local (141.62.66.6)
- Internet Control Message Protocol

Abbildung 21: MAC-Adresse des Nachbarrechners

Welche MAC-Adresse hat der Labor-Router?

TODO: Add interpretation

Abbildung 22: MAC-Adresse des Labor-Routers

TODO: Add interpretation

Da der Rechner außerhalb des Labor-Netzes ist, kann dessen Mac nicht bestimmt werden.

Abbildung 23: MAC-Adresse des externen Rechners

2.9 STP

Filtern Sie auf das Protokoll BPDU/STP. Wer sendet es und welchen Sinn hat dieses Protokoll?

TODO: Add interpretation

No.	Time	Source	Destination	Protocol	Length	Info
393	182.00015690	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
394	184.00105920	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
395	186.00005607	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
397	188.00020236	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
398	190.000136348	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
406	192.000508647	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
407	194.000071189	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
408	196.000399863	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
411	198.000053659	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
412	200.000207349	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
413	202.000107163	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
417	204.000254351	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
418	206.000015952	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
423	208.000037935	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
424	210.000205071	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
425	212.000277733	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
426	214.001000472	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
427	216.000076791	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
429	218.000208922	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
430	220.000146954	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
433	222.000177244	HewlettP_aa:8b:be	Spanning-tree-(for-... STP	119 MST	Root = 32768/0/00:1a:c1:5e:eb:c0	Cost = 220020 Port = 0x0002
* Frame 426: 119 bytes on wire (952 bits), 119 bytes captured (952 bits) on interface enp8s31f6, id 0 * IEEE 802.3 Ethernet * Destination: Spanning-tree-(for-bridges).00 (01:80:c2:00:00:00) Address: Spanning-tree-(for-bridges).00 (01:80:c2:00:00:00)0. = LG bit: Globally unique address (factory default)1. = IG bit: Group address (multicast/broadcast) * Source: HewlettP_aa:8b:be (04:09:73:aa:8b:be) Address: HewlettP_aa:8b:be (04:09:73:aa:8b:be)0. = LG bit: Globally unique address (factory default)0. = IG bit: Individual address (unicast) Length: 105 * Logical-Link Control * Spanning Tree Protocol Protocol Identifier: Spanning Tree Protocol (0x0000) Protocol Version Identifier: Multiple Spanning Tree (3) BPDU Type: Rapid/Multiple Spanning Tree (0x02) * BPDU flags: 0x3e, Forwarding, Learning, Port Role: Designated, Proposal * Root Identifier: 32768 / 0 / 00:1a:c1:5e:eb:c0 Root Path Cost: 220020 * Bridge Identifier: 32768 / 0 / 04:09:73:aa:8b:80 Port Identifier: 0x0002 Message Age: 3 Max Age: 20 Hello Time: 2 Forward Delay: 15 Version 1 Length: 0						

Abbildung 24: Capture mit Filter für STP

2.10 SNMP

Auf welchen Komponenten im Netzwerk wird das Protokoll SNMP ausgeführt?

TODO: Add interpretation (there were no packets to be found at the time of the experiment)

2.11 Streaming and Downloads

Starten Sie einen Download einer größeren Datei aus dem Internet und stoppen Sie ihn während der Übertragung. Dokumentieren Sie, wie der Stop-Befehl innerhalb der Protokolle umgesetzt wird

TODO: Add description

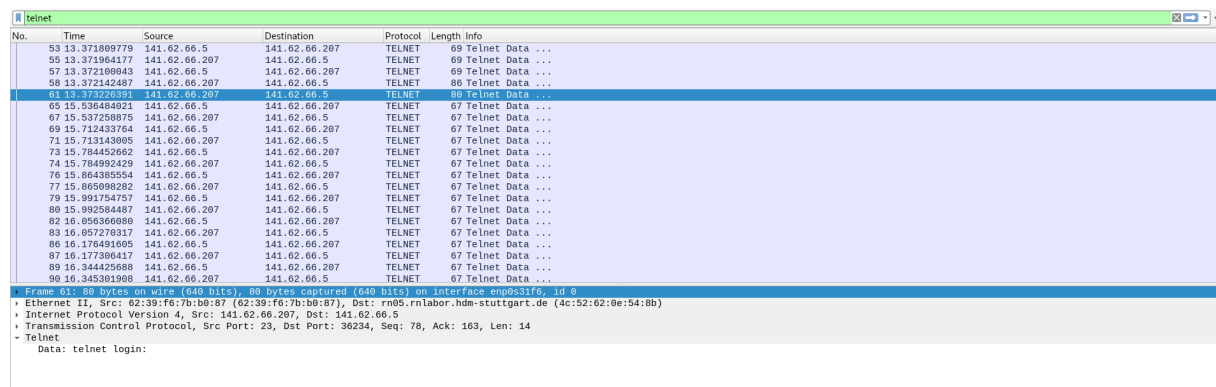
Protokollieren sie ein Video-Streaming Ihrer Wahl. Welche TCP-Ports werden wozu benutzt? Filtern Sie alle Rahmen, in denen sich das TCP-Window geändert hat

TODO: Add description

2.12 Telnet und SSH

Protokollieren Sie den Ablauf einer TELNET-Verbindung zur IP-Adresse 141.62.66.207 (login: praktikum; passwd: versuch). Können Sie Passwörter im Wireshark-Trace identifizieren? Wie verhält sich im Vergleich dazu eine SSH-Verbindung zum gleichen Server?

TODO: Add interpretation



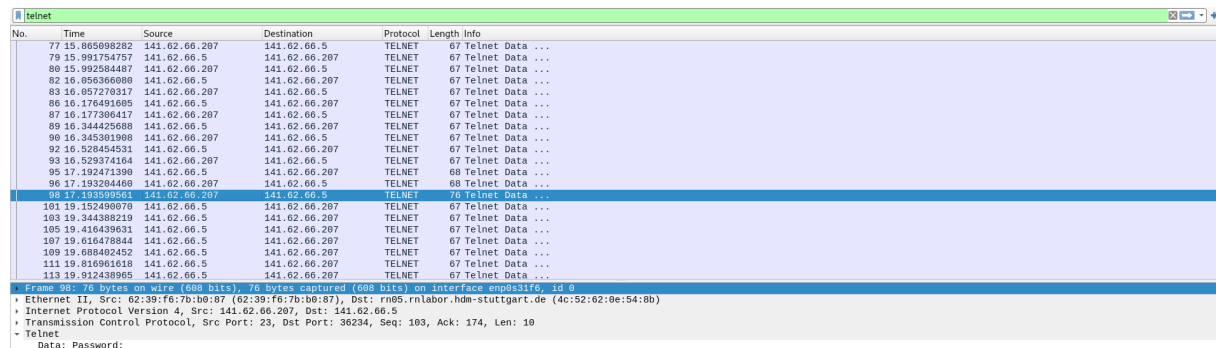
No.	Time	Source	Destination	Protocol	Length	Info
53	13.371899779	141.62.66.5	141.62.66.207	TELNET	69	Telnet Data ...
55	13.371964177	141.62.66.207	141.62.66.5	TELNET	69	Telnet Data ...
57	13.372109843	141.62.66.5	141.62.66.207	TELNET	69	Telnet Data ...
59	13.372142487	141.62.66.207	141.62.66.5	TELNET	69	Telnet Data ...
61	13.37220331	141.62.66.207	141.62.66.5	TELNET	69	Telnet Data ...
65	15.536484021	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
67	15.537258875	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
69	15.712433754	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
71	15.713143895	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
73	15.784452662	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
74	15.784992429	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
76	15.864385554	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
77	15.865998282	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
79	15.991754757	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
80	15.992504487	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
82	16.056366080	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
83	16.057278317	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
86	16.176491695	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
87	16.177396417	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
89	16.344425688	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
90	16.345391988	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...

Frame 90: 70 bytes on wire (560 bits), 69 bytes captured (552 bits) on interface enp0s31f0, id 0

- Ethernet II, Src: 62:39:f6:7b:b8:87 (62:39:f6:7b:b8:87), Dst: rnm95.rnlabor.hdm-stuttgart.de (4c:52:62:0e:54:8b)
- Internet Protocol Version 4, Src: 141.62.66.207, Dst: 141.62.66.5
- Transmission Control Protocol, Src Port: 23, Dst Port: 36234, Seq: 78, Ack: 163, Len: 14
- Telnet
 - Data: telnet login:

Abbildung 25: Capture des Telnet-Logins

Können Sie Passwörter im Wireshark-Trace identifizieren?

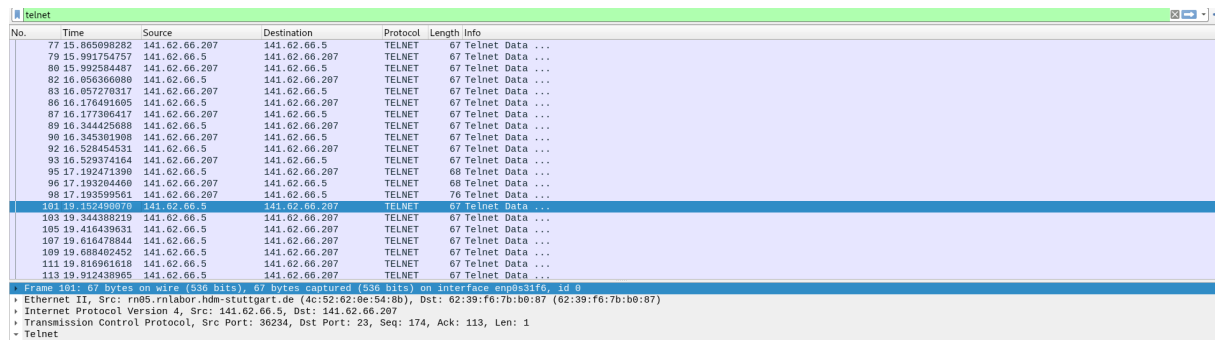


No.	Time	Source	Destination	Protocol	Length	Info
77	15.865998282	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
79	15.991754757	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
80	15.992504487	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
82	16.056366080	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
83	16.057278317	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
86	16.176491695	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
87	16.177396417	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
89	16.344425688	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
90	16.345391988	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
92	16.529454531	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
93	16.529374164	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
95	17.192471390	141.62.66.5	141.62.66.207	TELNET	68	Telnet Data ...
96	17.193204460	141.62.66.207	141.62.66.5	TELNET	68	Telnet Data ...
98	17.193599561	141.62.66.207	141.62.66.5	TELNET	70	Telnet Data ...
101	19.152490070	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
103	19.344389219	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
105	19.416439631	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
107	19.616478844	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
109	19.688402452	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
111	19.816961618	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
113	19.912438965	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...

Frame 98: 70 bytes on wire (560 bits), 70 bytes captured (560 bits) on interface enp0s31f0, id 0

- Ethernet II, Src: 62:39:f6:7b:b8:87 (62:39:f6:7b:b8:87), Dst: rnm95.rnlabor.hdm-stuttgart.de (4c:52:62:0e:54:8b)
- Internet Protocol Version 4, Src: 141.62.66.207, Dst: 141.62.66.5
- Transmission Control Protocol, Src Port: 23, Dst Port: 36234, Seq: 103, Ack: 174, Len: 10
- Telnet
 - Data: Password:

Abbildung 26: Capture des Telnet-Passworts



The image shows a Wireshark packet capture of Telnet traffic. The packet list on the left shows several Telnet packets. Packet 101 is selected, showing details of an Ethernet II frame, an Internet Protocol Version 4 packet, and a Transmission Control Protocol packet. The packet data pane at the bottom shows the raw data of the Telnet packet, which is a single character 'v'.

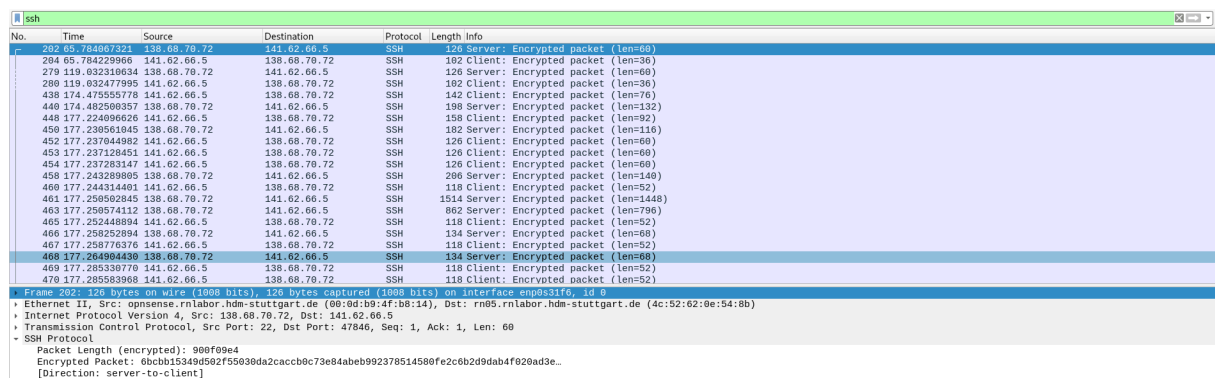
No.	Time	Source	Destination	Protocol	Length	Info
77	15.865098282	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
79	15.991754757	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
80	15.992584487	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
82	16.856360808	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
83	16.057270317	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
86	16.170481695	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
87	16.177386417	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
89	16.344425688	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
90	16.345381986	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
92	16.528454531	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
93	16.529374164	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
95	17.192471390	141.62.66.5	141.62.66.207	TELNET	68	Telnet Data ...
96	17.193284469	141.62.66.207	141.62.66.5	TELNET	68	Telnet Data ...
98	17.193595561	141.62.66.207	141.62.66.5	TELNET	76	Telnet Data ...
101	19.102488970	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
103	19.344388219	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
105	19.416439631	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
107	19.616478844	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
109	19.688482452	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
111	19.816961618	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
113	19.912438965	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...

Frame 101: 67 bytes on wire (536 bits), 67 bytes captured (536 bits) on interface enp0s31f6, id 0
Ethernet II, Src: r05.rnlabor.hdm-stuttgart.de (4c:52:62:0e:54:b6), Dst: 62:39:f6:7b:b0:87 (62:39:f6:7b:b0:87)
Internet Protocol Version 4, Src: 141.62.66.5, Dst: 141.62.66.207
Transmission Control Protocol, Src Port: 36234, Dst Port: 23, Seq: 174, Ack: 113, Len: 1
Telnet
Data: v

Abbildung 27: Capture eines Chars des Telnet-Passworts

Wie verhält sich im Vergleich dazu eine SSH-Verbindung zum gleichen Server?

TODO: Add interpretation



The image shows a Wireshark packet capture of SSH traffic. The packet list on the left shows several SSH packets. Packet 202 is selected, showing details of an Ethernet II frame, an Internet Protocol Version 4 packet, and an SSH packet. The packet data pane at the bottom shows the raw data of the SSH packet, which is an encrypted packet.

No.	Time	Source	Destination	Protocol	Length	Info
202	05.784807321	138.68.70.72	141.62.66.5	SSH	120	Server: Encrypted packet (len=60)
204	05.784229966	141.62.66.5	138.68.70.72	SSH	162	Client: Encrypted packet (len=36)
278	119.632319634	138.68.70.72	141.62.66.5	SSH	120	Server: Encrypted packet (len=60)
280	119.632477995	141.62.66.5	138.68.70.72	SSH	162	Client: Encrypted packet (len=36)
438	174.475555778	141.62.66.5	138.68.70.72	SSH	142	Client: Encrypted packet (len=70)
440	174.492580657	138.68.70.72	141.62.66.5	SSH	198	Server: Encrypted packet (len=132)
448	177.224966626	141.62.66.5	138.68.70.72	SSH	158	Client: Encrypted packet (len=92)
450	177.238561045	138.68.70.72	141.62.66.5	SSH	182	Server: Encrypted packet (len=116)
452	177.237644982	141.62.66.5	138.68.70.72	SSH	126	Client: Encrypted packet (len=60)
453	177.237128451	141.62.66.5	138.68.70.72	SSH	126	Client: Encrypted packet (len=60)
454	177.237283147	141.62.66.5	138.68.70.72	SSH	126	Client: Encrypted packet (len=60)
458	177.243289885	138.68.70.72	141.62.66.5	SSH	206	Server: Encrypted packet (len=140)
460	177.244314491	141.62.66.5	138.68.70.72	SSH	118	Client: Encrypted packet (len=52)
461	177.250582948	138.68.70.72	141.62.66.5	SSH	154	Server: Encrypted packet (len=148)
463	177.250574112	138.68.70.72	141.62.66.5	SSH	862	Server: Encrypted packet (len=796)
465	177.252448894	141.62.66.5	138.68.70.72	SSH	118	Client: Encrypted packet (len=52)
466	177.258252894	138.68.70.72	141.62.66.5	SSH	134	Server: Encrypted packet (len=68)
467	177.258776376	141.62.66.5	138.68.70.72	SSH	118	Client: Encrypted packet (len=52)
468	177.264984438	138.68.70.72	141.62.66.5	SSH	134	Server: Encrypted packet (len=68)
469	177.265338770	141.62.66.5	138.68.70.72	SSH	118	Client: Encrypted packet (len=52)
470	177.265583868	141.62.66.5	138.68.70.72	SSH	118	Client: Encrypted packet (len=52)

Frame 202: 126 bytes on wire (1008 bits), 126 bytes captured (1008 bits) on interface enp0s31f6, id 0
Ethernet II, Src: opsense.rnlabor.hdm-stuttgart.de (00:0d:b9:4f:b8:14), Dst: r05.rnlabor.hdm-stuttgart.de (4c:52:62:0e:54:b6)
Internet Protocol Version 4, Src: 138.68.70.72, Dst: 141.62.66.5
Transmission Control Protocol, Src Port: 22, Dst Port: 47846, Seq: 1, Ack: 1, Len: 60
SSH Protocol
Packet Length (encrypted): 900f09e4
Encrypted Packet: c3-0bb52f5930da2caccb0c73e84abeb992378514500fe2c6b2d9dab4f020ad3e...
[Direction: server-to-client]

Abbildung 28: Capture eines verschlüsselten SSH-Pakets

2.13 Wireshark-Filter

Entwickeln, testen und dokumentieren Sie Wireshark-Filter zur Lösung folgender Aufgaben:

Nur IP-Pakete, deren TTL größer ist als ein von Ihnen sinnvoll gewählter Referenzwert

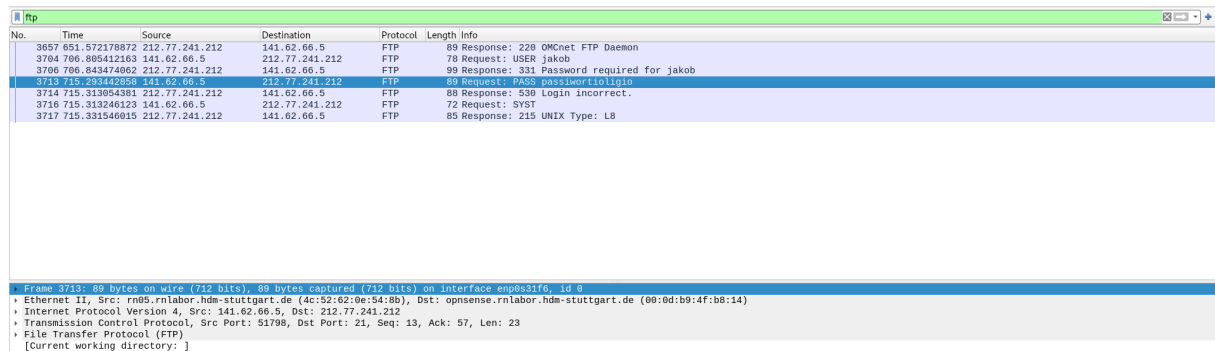
TODO: Add description

Nur IP-Pakete, die fragmentiert sind

TODO: Add description

Beim Login-Versuch auf ftp.bellevue.de mit von Ihnen wählbaren Account-Daten nur Rahmen herausfiltern, die das gewählte Passwort im Ethernet-Datenfeld enthalten

TODO: Add interpretation



No.	Time	Source	Destination	Protocol	Length	Info
3657	651.572178872	212.77.241.212	141.62.66.5	FTP	89	Response: 220 OMNet FTP Daemon
3704	766.895412163	141.62.66.5	212.77.241.212	FTP	78	Request: USER jakob
3706	766.843474962	212.77.241.212	141.62.66.5	FTP	99	Response: 331 Password required for jakob
3713	715.293442958	141.62.66.5	212.77.241.212	FTP	89	Request: PASS passwort101010
3714	715.313954381	212.77.241.212	141.62.66.5	FTP	88	Response: 530 Login incorrect.
3716	715.313246123	141.62.66.5	212.77.241.212	FTP	72	Request: SYST
3717	715.331546015	212.77.241.212	141.62.66.5	FTP	88	Response: 215 UNIX Type: L8

Frame 3713: 89 bytes on wire (712 bits), 89 bytes captured (712 bits) on interface enp0s31f6, id 0

- Ethernet II, Src: rn05.rnlabor.hdm-stuttgart.de (4c:52:62:0e:54:8b), Dst: opnsense.rnlabor.hdm-stuttgart.de (08:0d:b9:4f:b8:14)
- Internet Protocol Version 4, Src: 141.62.66.5, Dst: 212.77.241.212
- Transmission Control Protocol, Src Port: 51798, Dst Port: 21, Seq: 13, Ack: 57, Len: 23
- File Transfer Protocol (FTP)
- [Current working directory:]

Abbildung 29: Capture eines FTP-Pakets, welches ein Passwort enthält**Nur den Port 80-Verkehr zu Ihrer IP-Adresse (ankommend und abgehend)**

TODO: Add description

Nur Pakete mit einer IP-Multicast-Adresse

TODO: Add description