

Praktikum Rechnernetze

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

Jakob Waibel Daniel Hiller Elia Wüstner Felix Pojtinger

2021-10-26

Einführung

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):



Figure 1: QR-Code zum Quelltext auf GitHub

Lizenz

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



Figure 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

Wireshark

Einführung

An welchem Koppelement 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.



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:



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 Netzverkehr, den der PC bis zum Erhalt der IP-Adresse erzeugt.

Während des Startens werden drei DHCP-Requests für verschiedene Komponenten abgehandelt.

No.	Time	Source	Destination	Protocol	Length	Info
47	36.2408724335	0.0.0.0	255.255.255.255	DHCP	590	DHCP Discover - Transaction ID 0x620e53eb
48	36.2408844427	ognsense-router.rml...	255.255.255.255	DHCP	348	DHCP Offer - Transaction ID 0x620e53eb
55	40.250252423	0.0.0.0	255.255.255.255	DHCP	590	DHCP Request - Transaction ID 0x620e53eb
56	40.250518728	ognsense-router.rml...	255.255.255.255	DHCP	348	DHCP ACK - Transaction ID 0x620e53eb
57	40.2509797973	linux.local	Broadcast	ARP	60	Who has 141.62.66.236? Tell 141.62.66.4
58	40.278416173	linux.local	Broadcast	ARP	60	Who has 141.62.66.250? Tell 141.62.66.4
63	45.478669439	fog.rnrlabor.hdm-stu...	linux.local	ARP	60	Who has 141.62.66.47 Tell 141.62.66.236
65	46.478669439	fog.rnrlabor.hdm-stu...	1.1.1.1	ARP	60	Who has 141.62.66.47 Tell 1.1.1.1
70	47.526653889	fog.rnrlabor.hdm-stu...	linux.local	ARP	60	Who has 141.62.66.47 Tell 141.62.66.236
72	49.407126304	0.0.0.9	255.255.255.255	DHCP	451	DHCP Discover - Transaction ID 0xc1478931
73	49.498452675	ognsense-router.rml...	255.255.255.255	DHCP	348	DHCP Offer - Transaction ID 0xc1478931
79	50.529353450	0.0.0.0	255.255.255.255	DHCP	463	DHCP Request - Transaction ID 0xc1478931
80	50.531124992	ognsense-router.rml...	255.255.255.255	DHCP	348	DHCP ACK - Transaction ID 0xc1478931
81	50.531125138	linux.local	Broadcast	ARP	60	ARP Announcement for 141.62.66.4
82	50.5845464928	0.0.0.0	Broadcast	ARP	60	Who has 141.62.66.236? Tell 141.62.66.4
85	54.820515489	linux.local	Broadcast	ARP	60	Who has 141.62.66.236? Tell 141.62.66.4
92	56.342356749	0.0.0.0	255.255.255.255	DHCP	348	DHCP Discover - Transaction ID 0xadic98d59
93	66.342356749	0.0.0.0	255.255.255.255	DHCP	345	DHCP Offer - Transaction ID 0xadic98d59
95	66.6292416640	linux.local	Broadcast	ARP	60	Who has 141.62.66.250? Tell 141.62.66.4

Figure 9: Gesamter Bootprozess

Dokumentieren Sie den Ablauf bei einer DNS-Abfrage

Fall 1: DNS-Server 141.62.66.250:

Mittels folgendem Command wurde eine DNS-Abfrage gemacht:

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

dns && frame.number < 20						
No.	Time	Source	Destination	Protocol	Length	Info
11	1.357358668	rn05.rnlab0.hdm-st	opnsense-router.rn1	DNS	93	Standard query 0xa276 A google.com OPT
12	1.371692878	opnsense-router.rn1	rn05.rnlab0.hdm-st	DNS	97	Standard query response 0xa276 A google.com A 142.250.186.174 OPT

Figure 12: Ablauf der Anfrage

Hier nutzten wir den internen DNS Server und machen eine Anfrage auf google.com.

Fall 2: DNS-Server 1.1.1.1 (Cloudflare):

Mit folgendem Command kann die DNS-Aufgabe gelöst werden:

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

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

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	110.515867298	Linux-3.local	Linux-2.local	ARP	60	141.62.66.6 is at 4c:52:62:0e:54:2b
231	115.073154795	Linux-3.local	Linux-2.local	ARP	60	who has 141.62.66.5? Tell 141.62.66.6
232	115.073186793	Linux-2.local	Linux-3.local	ARP	42	141.62.66.6 is at 4c:52:62:0e:54:2b

Figure 15: Ablauf der Anfrage

Wann wird eine ARP-Anfrage gestartet?

Sobald ein Paket an die Zieladresse (in unserem Fall 141.62.66.6) gesendet werden soll, wird eine ARP-Anfrage in Form eines Broadcasts gestartet, um das Zielgerät im Netzwerk zu ermitteln, sofern sich diese nicht bereits im ARP-Cache befindet. Dieser kann mit ip neigh show ausgelesen werden. Mit ip neigh flush all

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

Das STP-Protokoll ist das Spanning Tree Protocol. Das STP-Protokoll verhindert Schleifenbildung; dies ist besonders dann von Nutzen, wenn Redundanzen vorhanden sind. Beim STP-Protokoll werden durch alle am Netz beteiligten Switches eine "Root Bridge" gewählt und redundante Links werden deaktiviert. Wie anhand der OUI der MAC-Adresse erkannt werden kann wird dieses hier von einem HP-Switch verwendet.

No.	Time	Source	Destination	Protocol	Length Info
393	182.000315869	HuaweiTP_aa:00:be	Spanning-tree-(For-	STP	119 MST Root = 32768/0/0/0:1a:c1:5e:eb:09 Cost = 220629 Port = 0x8002
394	184.001050892	HuaweiTP_aa:00:be	Spanning-tree-(For-	STP	119 MST Root = 32768/0/0/0:1a:c1:5e:eb:09 Cost = 220629 Port = 0x8002
395	186.000280922	HuaweiTP_aa:00:be	Spanning-tree-(For-	STP	119 MST Root = 32768/0/0/0:1a:c1:5e:eb:09 Cost = 220629 Port = 0x8002
397	188.000262080	HuaweiTP_aa:00:be	Spanning-tree-(For-	STP	119 MST Root = 32768/0/0/0:1a:c1:5e:eb:09 Cost = 220629 Port = 0x8002
398	190.000313040	HuaweiTP_aa:00:be	Spanning-tree-(For-	STP	119 MST Root = 32768/0/0/0:1a:c1:5e:eb:09 Cost = 220629 Port = 0x8002
400	192.000560847	HuaweiTP_aa:00:be	Spanning-tree-(For-	STP	119 MST Root = 32768/0/0/0:1a:c1:5e:eb:09 Cost = 220629 Port = 0x8002
407	194.000077119	HuaweiTP_aa:00:be	Spanning-tree-(For-	STP	119 MST Root = 32768/0/0/0:1a:c1:5e:eb:09 Cost = 220629 Port = 0x8002
408	196.000077120	HuaweiTP_aa:00:be	Spanning-tree-(For-	STP	119 MST Root = 32768/0/0/0:1a:c1:5e:eb:09 Cost = 220629 Port = 0x8002
411	198.000053059	HuaweiTP_aa:00:be	Spanning-tree-(For-	STP	119 MST Root = 32768/0/0/0:1a:c1:5e:eb:09 Cost = 220629 Port = 0x8002
412	200.000287784	HuaweiTP_aa:00:be	Spanning-tree-(For-	STP	119 MST Root = 32768/0/0/0:1a:c1:5e:eb:09 Cost = 220629 Port = 0x8002
413	202.000287163	HuaweiTP_aa:00:be	Spanning-tree-(For-	STP	119 MST Root = 32768/0/0/0:1a:c1:5e:eb:09 Cost = 220629 Port = 0x8002
417	204.000287164	HuaweiTP_aa:00:be	Spanning-tree-(For-	STP	119 MST Root = 32768/0/0/0:1a:c1:5e:eb:09 Cost = 220629 Port = 0x8002
418	206.000285992	HuaweiTP_aa:00:be	Spanning-tree-(For-	STP	119 MST Root = 32768/0/0/0:1a:c1:5e:eb:09 Cost = 220629 Port = 0x8002
423	208.000053793	HuaweiTP_aa:00:be	Spanning-tree-(For-	STP	119 MST Root = 32768/0/0/0:1a:c1:5e:eb:09 Cost = 220629 Port = 0x8002
424	210.000285871	HuaweiTP_aa:00:be	Spanning-tree-(For-	STP	119 MST Root = 32768/0/0/0:1a:c1:5e:eb:09 Cost = 220629 Port = 0x8002
425	212.000287232	HuaweiTP_aa:00:be	Spanning-tree-(For-	STP	119 MST Root = 32768/0/0/0:1a:c1:5e:eb:09 Cost = 220629 Port = 0x8002
426	214.001040847	HuaweiTP_aa:00:be	Spanning-tree-(For-	STP	119 MST Root = 32768/0/0/0:1a:c1:5e:eb:09 Cost = 220629 Port = 0x8002
427	216.000285871	HuaweiTP_aa:00:be	Spanning-tree-(For-	STP	119 MST Root = 32768/0/0/0:1a:c1:5e:eb:09 Cost = 220629 Port = 0x8002
429	218.000280922	HuaweiTP_aa:00:be	Spanning-tree-(For-	STP	119 MST Root = 32768/0/0/0:1a:c1:5e:eb:09 Cost = 220629 Port = 0x8002
430	220.000546853	HuaweiTP_aa:00:be	Spanning-tree-(For-	STP	119 MST Root = 32768/0/0/0:1a:c1:5e:eb:09 Cost = 220629 Port = 0x8002

SNMP

Auf welchen Komponenten im Netzwerk wird das Protokoll SNMP ausgeführt?

Es konnte kein SNMP-Traffic im Netzwerk gefunden werden. SNMP, das Simple Network Management Protocol, wird jedoch meist zur Wartung von verbundenen Geräte im Network verwendet, woraus sich schließen lässt, dass es auf Komponenten wie Switches, Routern oder Servern zum Einsatz kommen würde.

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



Figure 28: Capture beim Canceln des eines Downloads über HTTPS

Da der Download hier via HTTPS durchgeführt wurde, kann erkannt werden, dass die darunterliegende TCP-Verbindung unterbrochen wurde, indem die RST-Flag gesetzt wurde. Auch ein

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?

Wie zu erkennen ist, wird für eine Telnet-Verbindung eine TCP-Verbindung aufgebaut. Die Passwörter sind zu erkennen.

No.	Time	Source	Destination	Protocol	Length	Info
53	13:37:38.897798	141.62.66.5	141.62.66.207	TELNET	69	Telnet Data ...
55	13:37:38.917798	141.62.66.5	141.62.66.207	TELNET	69	Telnet Data ...
57	13:37:38.908443	141.62.66.5	141.62.66.207	TELNET	69	Telnet Data ...
58	13:37:21.424887	141.62.66.207	141.62.66.5	TELNET	86	Telnet Data ...
61	13:37:21.639111	141.62.66.207	141.62.66.5	TELNET	69	Telnet Data ...
65	13:59:44.021221	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
67	13:59:44.058098	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
69	15:7124333764	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
71	15:7131438985	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
73	15:7844526862	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
74	15:7844526863	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
76	15:8643885545	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
77	15:865980282	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
79	15:9917547577	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
80	15:9925844847	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
82	16:05:27.790171	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
83	16:05:27.790171	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
86	16:1764916985	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
87	16:1773866117	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...
89	16:3444256886	141.62.66.5	141.62.66.207	TELNET	67	Telnet Data ...
90	16:3459398886	141.62.66.207	141.62.66.5	TELNET	67	Telnet Data ...

Frame 1: 141.62.66.5 -> 141.62.66.207 [Tlnt] Telnet connection (Data) (69 bytes on wire (556 bits), 69 bytes captured (556 bits), 100% loss.)

Ethernet II, Src: rnlabor (08:39:f0:7b:0b:87) (ether), Dst: rnlab0.rnlabor.hdb-stuttgart.de (4c:52:62:0e:54:b8)

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:

