

Praktikum Rechnernetze

Protokoll zu Versuch 4 (IPv6) von Gruppe 1

Jakob Waibel Daniel Hiller Elia Wüstner Felicitas Pojtinger

2021-11-09

Einführung

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

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, Felicitas Pojtinger

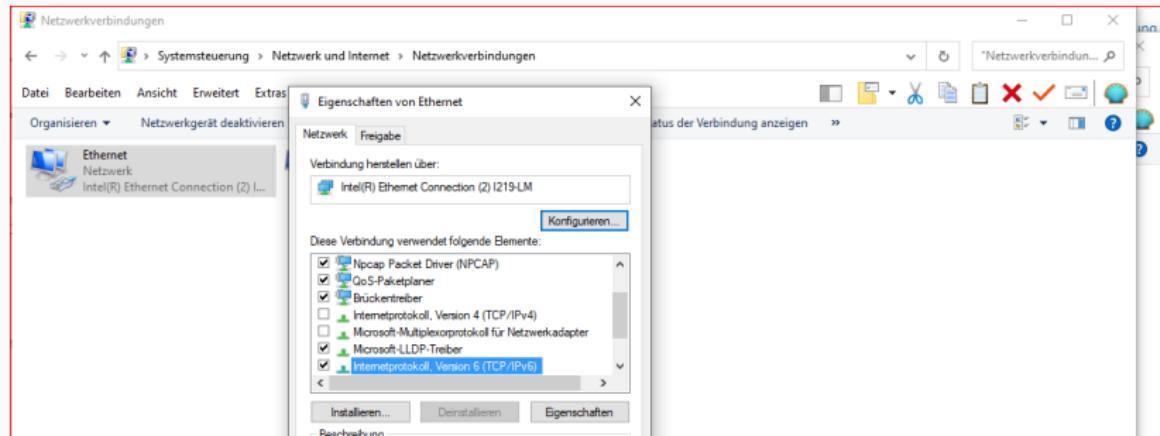
SPDX-License-Identifier: AGPL-3.0

IPv6-Addressen

IPv6-Addressen

Voreinstellung für die Aufgaben - deaktivieren von IPv4 und aktivieren von IPv6 unter Windows.

Um IPv4 zu deaktivieren und IPv6 zu aktivieren, muss man in den Netzwerkeinstellungen zum jeweiligen Adapter über den Pfad Systemsteuerung > Netzwerk und Internet > Netzwerkverbindungen > Adaptoreinstellungen navigieren. Hier wurde der Haken bei IPv6 (Internetprotokoll, Version6) gesetzt und bei IPv4 (Internetprotokoll, Version4) entfernt.



IPv6 und DNS

IPv6 und DNS

Identifizieren Sie mit Wireshark die Pakete mit denen der Router im Netz das Prefix mitteilt. Welches Protokoll wird dafür benutzt und um welchen Type handelt es sich und wie lautet die Zieladresse des Pakets?

Das verwendete Protokoll ist wie auch in den unten stehenden Screenshots zu sehen ICMPv6. Die Types sind Router Solicitation und Router Advertisement. Die Zieladresse des Pakets ist die Multicast-Adresse ff02 ::1 .

Router Solicitation:

No.	Time	Source	Destination	Protocol	Length	Info
14	19.3699860966	fe80::1:4e52:6fff:fed0_	ff02::1:ffff:fed0:6612	ICMPv6	88	Neighbor Solicitation for fe80::1:4e52:6fff:fed0:6612 from 4c:52:63:0e:54:2c
27	29.5099454466	::	ff02::1:ffff:fed0:6612	ICMPv6	90	Multicast Listener Report Message v2
30	38.3421085140	::	ff02::1:ffff:fed0:6612	ICMPv6	90	Multicast Listener Report Message v2
31	39.4388482361	::	ff02::1:ffff:0e:54ab	ICMPv6	88	Neighbor Solicitation for fe80::1:4e52:62ff:fed0:54ab
34	41.4622355057	fe80::1:4e52:62ff:fed0_	ff02::1:ffff:0e:54ab	ICMPv6	90	Multicast Listener Report Message v2
35	41.4622355057	fe80::1:4e52:62ff:fed0_	ff02::2	ICMPv6	78	Router Solicitation from 4c:52:02:0e:54:b8
36	31.474113948	fe80::1:4e52:62ff:fed0_	ff02::16	ICMPv6	90	Multicast Listener Report Message v2
39	31.762184739	fe80::1:4e52:62ff:fed0_	ff02::16	ICMPv6	118	Multicast Listener Report Message v2
42	32.2941560000	fe80::1:4e52:62ff:fed0_	ff02::16	ICMPv6	90	Multicast Listener Report Message v2
43	32.2941560000	fe80::1:4e52:62ff:fed0_	ff02::16	ICMPv6	118	Multicast Listener Report Message v2
44	32.2941560000	fe80::1:4e52:62ff:fed0_	ff02::16	ICMPv6	118	Multicast Listener Report Message v2
50	3995252689	fe80::1:4e52:62ff:fed0_	ff02::1:ffff:0e:54ab	ICMPv6	110	Router Advertisement from f8:d1:11:bd:66:12
51	35.4223123568	::	ff02::1:ffff:54ab	ICMPv6	88	Neighbor Solicitation for 2001:470:6d:4d0:4e52:62ff:fe0e:54ab
54	36.466118769	fe80::1:4e52:62ff:fed0_	ff02::16	ICMPv6	90	Multicast Listener Report Message v2
56	37.466118769	fe80::1:4e52:62ff:fed0_	ff02::16	ICMPv6	90	Multicast Listener Report Message v2
115	45.352524849	fe80::1:4e52:62ff:fed0_	ff02::1:ffff:0e:54ab	ICMPv6	88	Neighbor Solicitation for 2001:470:6d:4d0:4e52:62ff:fe0e:541b from f8:d1:11:bd:66:12
122	45.2773928238	fe80::1:4e52:62ff:fed0_	ff02::1:ffff:0e:54ab	ICMPv6	88	Neighbor Solicitation for 2001:470:6d:4d0:4e52:62ff:fe0e:54ab from f8:d1:11:bd:66:12
123	45.2773928238	fe80::1:4e52:62ff:fed0_	ff02::1:ffff:0e:54ab	ICMPv6	78	Neighbor Advertisement 2001:470:6d:4d0:4e52:62ff:fe0e:54ab (sol)

Neighbor Solicitation

Neighbor Solicitation

Starten Sie den „Kabelhai“ und pingen Sie ihren Nachbarrechner. Welches Protokoll/Type wird anstatt ARP zur Ermittlung der MAC-Adressen verwendet?

Windows

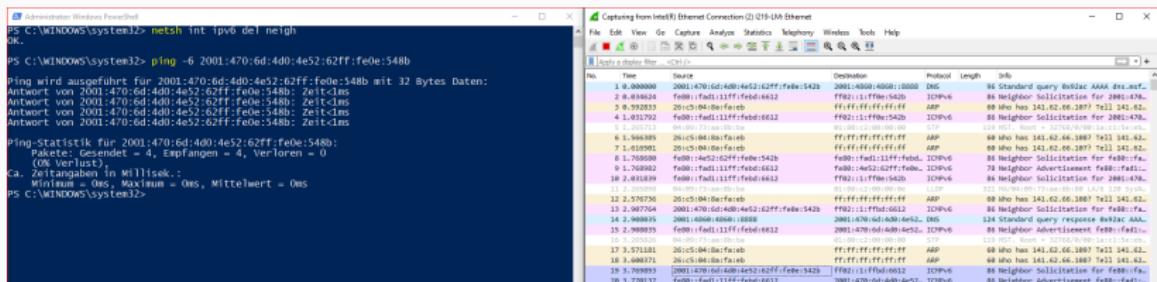


Abbildung 16: Solicitation und Advertisement-Pakete in Wireshark - Windows

Linux

```
$ sudo ip neigh flush dev enp0s31f6
$ ping6 fe80::fad1:11ff:febд:6612
PING fe80::fad1:11ff:febд:6612(fe80::fad1:11ff:febд:6612) 56
```

IPv6-Header

IPv6-Header

Starten Sie Wireshark und senden sie ein ping an einen IPv6-fähigen Webserver (www.ix.de, <http://www.heise.de>, <http://www.kame.net>), stoppen Sie Wireshark und schauen sich den Trace an.

Windows

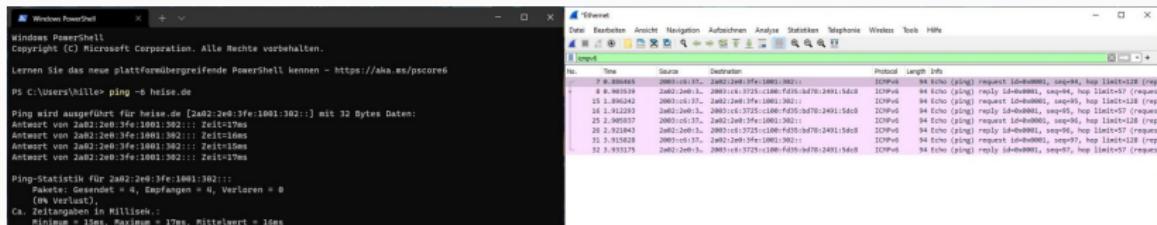


Abbildung 18: Ping Heise

Linux

```
$ ping www.kame.net
```

```
PING www.kame.net(2001:2f0:0:8800:226:2dff:fe0b:4311) (2001:2f0:0:8800:226:2dff:fe0b:4311)
64 bytes from 2001:2f0:0:8800:226:2dff:fe0b:4311 (2001:2f0:0:8800:226:2dff:fe0b:4311): icmp_seq=1 ttl=64 time=0.000 ms
64 bytes from 2001:2f0:0:8800:226:2dff:fe0b:4311 (2001:2f0:0:8800:226:2dff:fe0b:4311): icmp_seq=2 ttl=64 time=0.000 ms
```

Privacy Extension

Privacy Extension

Tragen Sie weitere Informationen zur „Privacy Extension“ (vor allem auch zur Konfiguration unter Windows und Ubuntu) zusammen und versuchen hier im Versuch die Einstellungen für die „Privacy Extension“ auf beiden Rechnern (Windows und Ubuntu) zu realisieren.

Privacy Extensions sind dafür da, Rückchluss auf Nutzer:innen schwerer zu machen, indem der Hostanteil der IPv6-Adressen anonymisiert wird.

Privacy Extensions entkoppeln Interface Identifier und MAC-Adresse und erzeugen diese nahezu zufällig. Mit diesen periodisch wechselnden Adressen werden dann ausgehende Verbindungen hergestellt, was den Rückschluss auf *einzelne* Nutzer:innen erschwert. Mit Hilfe der Privacy Extensions kann man also nicht mehr einzelne Nutzer:innen identifizieren. Was allerdings trotzdem möglich ist, ist das Identifizieren über den Präfix, welcher allerdings nur Informationen zum Netzwerk bereitstellt. Wenn der Provider den Präfix regelmäßig wechselt, dann kann auch die Identifikation über diesen erschwert werden.

Feste IPv6-Addressen

Feste IPv6-Addressen

Weisen Sie in dieser Aufgabe Ihrem Netzwerkinterface eine feste sinnvolle (heißt: Der Prefix ist weiterhin gültig) IPv6-Adresse zu.

Windows



Lease-Zeiten

Lease-Zeiten

Die Werte für “Maximale bevorzugte Gültigkeitsdauer” und “Maximale Gültigkeitsdauer” setzt man in Windows über die Schlüssel maxpreferredlifetime und maxvalidlifetime, die Zeitangaben in Tagen (d), Stunden (h), Minuten (m) und Sekunden (s) entgegennehmen. Wie sind diese Parameter bei Ihnen gesetzt?

Windows

```
netsh interface ipv6 show privacy
```

Parameter für temporäre Adressen

```
Temporäre Adresse verwenden      : enabled
Versuch, doppelte Adr. zu entdecken : 3
Maximale Gültigkeitsdauer       : 7d
Maximale bevorzugte Gültigkeitsdauer: 7d
Regenerationszeit                : 5s
Maximale Verzögerungszeit        : 10m
Verzögerungszeit                 : 6m23s
```

OS-Updates

OS-Updates

Lässt sich eigentlich Windows über IPv6 updaten? Was sagt Wireshark dazu?

Windows

Unter Windows wurde das Update ohne Probleme installiert. Windows Update verfügt über vollen IPv6-Support.

(<https://serverfault.com/questions/844107/windows-server-update-on-ipv6-only-network>). Dies konnte auch mittels Wireshark validiert werden:

The screenshot shows two windows: Wireshark on the left and the Windows Update interface on the right.

Wireshark (Left): A packet capture window showing several TCP segments from an IPv6 source (2001:db8::1) to an IPv6 destination (2001:db8::1). The ports are 8080 and 443. The segments contain ACK and SEQ values, indicating a download process. One segment includes the text "TCP segment of a retransmission".

Windows Update (Right): The Windows Update settings window. It shows an "Updates available" section with a blue circular arrow icon. Below it, a progress bar indicates "Downloading - 4%" for the "2021-11 Cumulative Update for Windows 11 for x64-based Systems (KB5007215)". Another update, "2021-11 Cumulative Update for .NET Framework 3.5 and 4.8 for Windows 11 for x64 (KB5006363)", is shown as "Installing - 0%". At the bottom, an update for "2021-11 Update for Windows 11 for x64-based Systems (KB5006295)" is listed as "Pending restart".

Network Traffic Summary:

- Frame 46494: 1554 bytes on wire (12112 bits), 1554 bytes captured (12112 bits) on interface \Device\NPF_{...}, id 0
Ethernet II, Src: Microsoft Edge [00:0c:29:00:00:50], Dst: Fellowes-Online-Local [52:54:00:12:3e:50]
Infrared Physical, Version: 6, Src: c-0001:c-edge.net [00:0c:29:00:00:50], Dst: Feodit:[6035:7516:c9ff] (Feodit:[6035:7516:c9ff])
- > 0000 0000 0000 0000 0000 0000 => Traffic Class: 0x00 (ISCP: CSB, EOB; Net-ECT)
- 0000 0000 0000 0000 0000 0000 => Flow Label: 0x0000000000000000
- 0000 0000 0000 0000 0000 0000 => Hop Limit: 255
- 0000 0000 0000 0000 0000 0000 => Destination Address: Feodit:[6035:7516:c9ff] (Feodit:[6035:7516:c9ff])
- > Transmission Control Protocol, Src Port: 80, Dst Port: 55540, Seq: 282243, Ack: 436, Len: 1440

Hex dump of the captured traffic shows standard TCP header fields like Source and Destination Port, Sequence and Acknowledgment numbers, and flags (SYN, ACK, FIN, PSH).