

# Praktikum Rechnernetze

Protokoll zu Versuch 4 (IPv6) von Gruppe 1

---

Jakob Waibel   Daniel Hiller   Elia Wüstner   Felix Pojtinger

2021-11-09

# 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](https://github.com/pojntfx/uni-netpractice-notes)):



**Abbildung 1:** QR-Code zum Quelltext auf GitHub

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

# IPv6-Adressen

---

# IPv6-Adressen

**Erkunden sie unter Windows und Ubuntu, wie viele IP-Adressen dem physikalischen Interface zugeordnet sind.**

```
$ ip addr
```

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noque
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
2: enp0s31f6: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu
    link/ether 4c:52:62:0e:54:8b brd ff:ff:ff:ff:ff
    inet 141.62.66.5/24 brd 141.62.66.255 scope glo
        valid_lft 13993sec preferred_lft 13993sec
```

```
# /etc/sysctl.conf
```

```
net.ipv6.conf.all.disable_ipv6 = 0
```

```
net.ipv6.conf.default.disable_ipv6 = 0
```

# 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?

Protokoll: ICMPv6 Type: Router Solicitation bzw. Router Advertisement

The image shows a Wireshark packet capture window titled "enp0s3196". The packet list on the left shows several ICMPv6 packets. The selected packet is packet 42, which is an ICMPv6 Router Solicitation from fe80::4e52:62ff:fe0b::2 to ff02::16. The packet details pane on the right shows the following information:

- Frame 42: 78 bytes on wire (608 bits), 78 bytes captured (608 bits) on interface enp0s3196, id 0
- Ethernet II, Src: FujitsuT\_0e:54:0b (4c:52:62:0e:54:0b), Dst: IPv6cast\_B2 (33:33:00:00:00:02)
- Internet Protocol Version 6, Src: fe80::4e52:62ff:fe0b:540b, Dst: ff02::12
- Internet Control Message Protocol v6
  - Type: Router Solicitation (133)
    - Code: 0
    - Checksum: 8x7956 [correct]
    - [Checksum Status: Good]
    - Reserved: 80000000
  - ICMPv6 Option (Source Link-layer address : 4c:52:62:0e:54:0b)
    - Type: Source Link-layer address (3)
    - Length: 1 (8 bytes)
    - Link-layer address: FujitsuT\_0e:54:0b (4c:52:62:0e:54:0b)



# Neighbor Solicitation

---

# Neighbor Solicitation

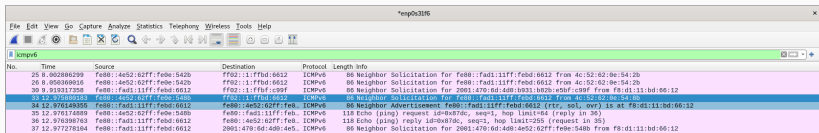
Starten Sie den „Kabelhai“ und pingen Sie ihren Nachbarrechner. Welches Protokoll/Type wird anstatt ARP zur Ermittlung der MAC-Adressen verwendet?

```
$ sudo ip neigh flush dev enp0s31f6
```

```
$ ping6 fe80::fad1:11ff:febd:6612
```

```
PING fe80::fad1:11ff:febd:6612 (fe80::fad1:11ff:febd:6612)
64 bytes from fe80::fad1:11ff:febd:6612: enp0s31f6:
^C
```

```
— fe80::fad1:11ff:febd:6612 ping statistics —
1 packets transmitted, 1 received, 0% packet loss,
rtt min/avg/max/mdev = 0.568/0.568/0.568/0.000 ms
```



The screenshot shows a Wireshark packet capture on the interface 'enp0s31f6'. The filter is set to 'icmpv6'. The packet list shows several ICMPv6 packets. The first three are Neighbor Solicitations from fe80::fad1:11ff:febd:6612 to fe80::fad1:11ff:febd:6612. The fourth is a Neighbor Advertisement from fe80::fad1:11ff:febd:6612 to fe80::fad1:11ff:febd:6612. The fifth is an Echo (ping) request from fe80::fad1:11ff:febd:6612 to fe80::fad1:11ff:febd:6612. The sixth is an Echo (ping) reply from fe80::fad1:11ff:febd:6612 to fe80::fad1:11ff:febd:6612.

No.	Time	Source	Destination	Protocol	Length	Info
25	8.862886299	fe80::4e52:62ff:febe:542b	ff02::1:ffbd:6612	ICMPv6	86	Neighbor Solicitation for fe80::fad1:11ff:febd:6612 from 4c:52:62:0e:54:2b
26	8.856368616	fe80::4e52:62ff:febe:542b	ff02::1:ffbd:6612	ICMPv6	86	Neighbor Solicitation for fe80::fad1:11ff:febd:6612 from 4c:52:62:0e:54:2b
30	9.91317258	fe80::fad1:11ff:febd:6612	ff02::1:ffbd:c99f	ICMPv6	86	Neighbor Solicitation for 2001:478:6d:408:b031:b02b:e5bf:c99f from f8:d1:11:bd:66:12
34	12.976148355	fe80::fad1:11ff:febd:6612	fe80::4e52:62ff:febe:542b	ICMPv6	86	Neighbor Advertisement fe80::fad1:11ff:febd:6612 (rtt, sol, ov) is at f8:d1:11:bd:66:12
35	12.976174888	fe80::4e52:62ff:febe:542b	fe80::fad1:11ff:febd:6612	ICMPv6	118	Echo (ping) request id=8d7dc, seq=1, hop limit=64 (reply in 3s)
36	12.976398763	fe80::fad1:11ff:febd:6612	fe80::4e52:62ff:febe:542b	ICMPv6	118	Echo (ping) reply id=8d7dc, seq=1, hop limit=255 (request in 3s)
37	12.977278164	fe80::fad1:11ff:febd:6612	2001:478:6d:408:b031:b02b:e5bf:c99f	ICMPv6	86	Neighbor Solicitation for 2001:478:6d:408:b031:b02b:e5bf:c99f from f8:d1:11:bd:66:12

# IPv6-Header

---

## IPv6-Header

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

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

```
PING www.kame.net (2001:2f0:0:8800:226:2dff:fe0b:4311): 64 bytes from 2001:2f0:0:8800:226:2dff:fe0b:4311: 64 bytes from 2001:2f0:0:8800:226:2dff:fe0b:4311: 64 bytes from 2001:2f0:0:8800:226:2dff:fe0b:4311: 64 bytes from 2001:2f0:0:8800:226:2dff:fe0b:4311: 64 bytes from 2001:2f0:0:8800:226:2dff:fe0b:4311: 64 bytes from 2001:2f0:0:8800:226:2dff:fe0b:4311: 64 bytes from 2001:2f0:0:8800:226:2dff:fe0b:4311: 64 bytes from 2001:2f0:0:8800:226:2dff:fe0b:4311: (2
```

```
^C
```

# 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.**

TODO: Add research results

**Mit welchen IPv6-Adressen sind sie nach dem Aktivieren der Privacy Extension im Internet unterwegs?**

TODO: Add research results

# Feste IPv6-Adressen

---

## Feste IPv6-Adressen

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

```
$ ip a
```

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noque
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s31f6: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu
    link/ether 4c:52:62:0e:54:8b brd ff:ff:ff:ff:ff
    inet6 2001:470:6d:4d0:4e52:62ff:fe0e:548b/64 sc
        valid_lft 86255sec preferred_lft 14255sec
    inet6 fe80::4e52:62ff:fe0e:548b/64 scope link
```



# 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?**

TODO: Add interpretation

**Halbieren Sie die “Maximale bevorzugte Gültigkeitsdauer” auf den Rechnern.**

TODO: Add interpretation

**Verringern Sie ebenso die Zeitspanne, in der Windows über eine temporäre IPv6-Adresse eingehende Pakete empfängt.**

TODO: Add interpretation

# OS-Updates

---

## OS-Updates

```
$ sudo ip addr del 141.62.66.5/24 dev enp0s31f6
$ sudo apt update
Hit:1 http://deb.debian.org/debian bullseye InRelease
Hit:2 http://security.debian.org bullseye-security
Get:3 http://deb.debian.org/debian bullseye-updates
Hit:4 http://ppa.launchpad.net/ansible/ansible/ubuntu
Fetched 39.4 kB in 5s (7,169 B/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
1 package can be upgraded. Run 'apt list --upgradable' to see details.
$ sudo apt upgrade -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```