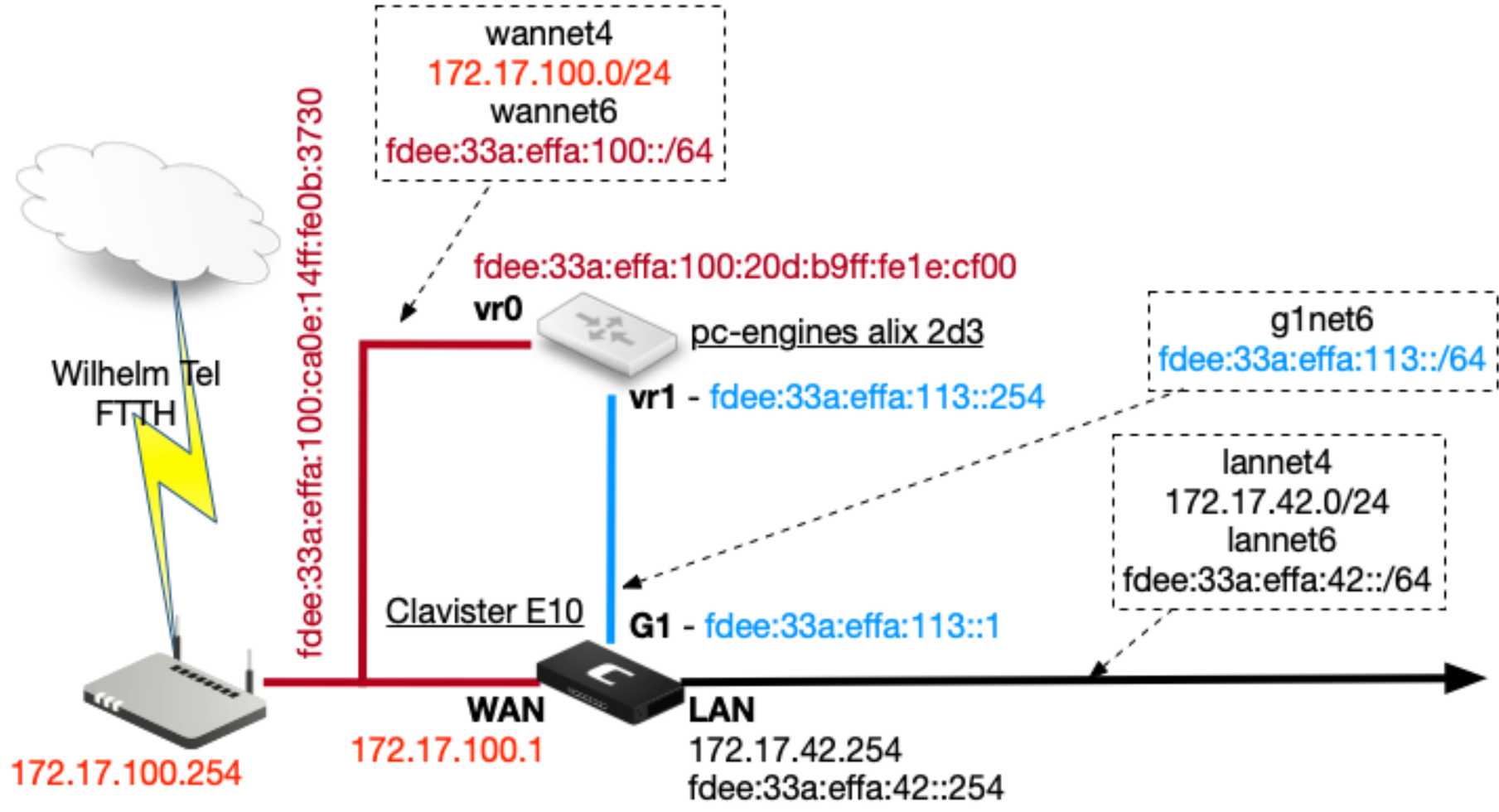
Von hinten durch die Brust ins Auge

IPv6 NAT gegen jegliche Vernunft

SAGE Hamburg - Frühjahr 2023

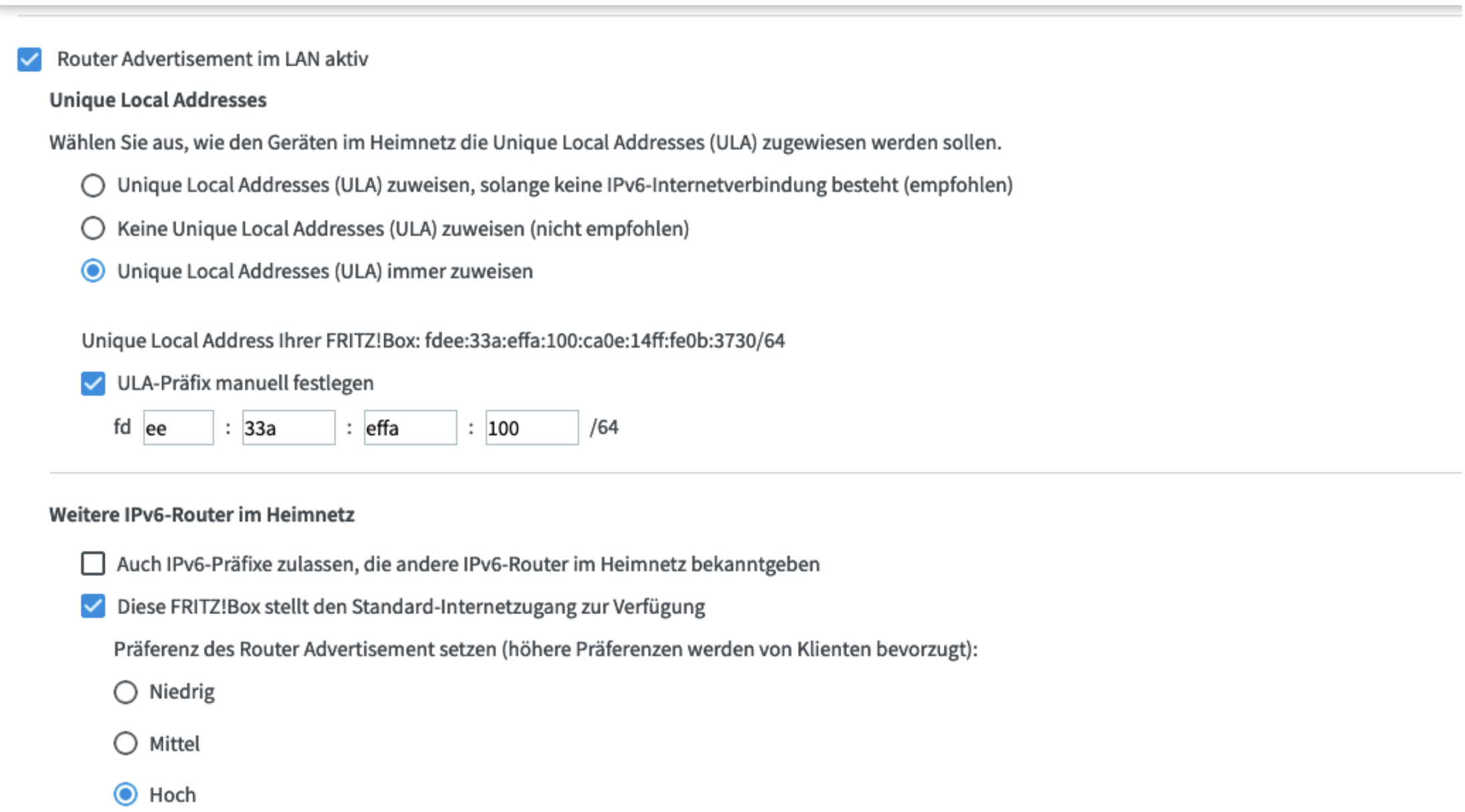
Netzplan Übersicht



		H. Michaelis	
Michaelis Anbindung privat Wilhelm Tel 250/50 FTTH		Netzplan	Entwurf
			Blatt 1/1
			18.02.2023

Fritz!Box als WT CPE

Fritz!Box Konfiguration #1



Fritz!Box Konfiguration #1

DHCPv6-Server im Heimnetz

◉ **DHCPv6-Server in der FRITZ!Box für das Heimnetz aktivieren:**

Wählen Sie aus, welche Informationen der DHCPv6-Server im Heimnetz bereitstellen soll.

○ **Nur DNS-Server zuweisen**

FRITZ!Box wird als DNS-Server via DHCPv6 bekannt gegeben.

○ **DNS-Server und IPv6-Präfix (IA_PD) zuweisen**

FRITZ!Box wird als DNS-Server via DHCPv6 bekannt gegeben. Teile des vom Internetanbieter zugewiesenen IPv6-Netzes werden an nachgelagerte Router we

◉ **DNS-Server, Präfix (IA_PD) und IPv6-Adresse (IA_NA) zuweisen**

FRITZ!Box wird als DNS-Server via DHCPv6 bekannt gegeben. Teile des vom Internetanbieter zugewiesenen IPv6-Netzes werden an nachgelagerte Router we

Falls mehrere DHCPv6-Server im Heimnetz aktiv sind, wird der DHCPv6-Server mit dem höheren Präferenzwert von den Heimnetzgeräten priorisiert.

Präferenz des FRITZ!Box DHCPv6-Servers: 200 (Wertebereich 0..255)

○ **DHCPv6-Server in der FRITZ!Box deaktivieren:**

Fritz!Box Konfiguration #3

Exposed Hosts:

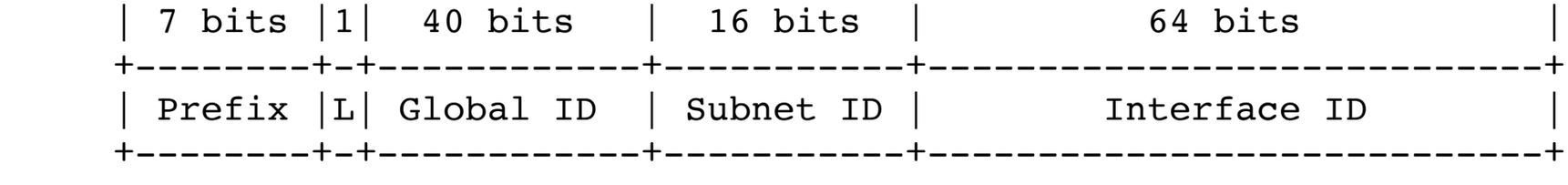
- Exposed Host IPv6: fdee:33a:effa:100:20d:b9ff:fe1e:cf00
  (pc-engines vr0 Port)
- Exposed Host IPv4: 172.17.100.1
  (Clavister E10 WAN Port)

Routen:

- Route IPv4: 172.17.0.0/16 -> 172.17.100.1
- Route IPv6: fdee:33a:effa::/48 -> fdee:33a:effa:100:20d:b9ff:fe1e:cf00

RFC 4193: Unique Local IPv6 Unicast Addresses (ULA)

The Local IPv6 addresses are created using a pseudo-randomly
allocated global ID. They have the following format:



Where:

Prefix	FC00::/7 prefix to identify Local IPv6 unicast addresses.
L	Set to 1 if the prefix is locally assigned. Set to 0 may be defined in the future. See Section 3.2 for additional information.
Global ID	40-bit global identifier used to create a globally unique prefix. See Section 3.2 for additional information.
Subnet ID	16-bit Subnet ID is an identifier of a subnet within the site.
Interface ID	64-bit Interface ID as defined in [ADDARCH].

ULA Registry

- https://www.sixxs.net/tools/grh/ula/list/ (read only)
- https://ula.ungleich.ch/

ULA considered harmful

- **Warning**: Pregnant women, the elderly, and children under 10 should avoid prolonged exposure to ULA. (https://www.modem.show/post/s02e03/)

- When not in use, ULA should be returned to its special container and kept under refrigeration. The ingredients of ULA include an unknown glowing green substance, which fell to Earth, presumably from outer space.

- Unintended Operational Issues With ULA (IETF draft-buraglio-v6ops-ula-05)

NAT Varianten

- NAT
- NAT66 - stateful - IETF draft-mrw-nat66-00.txt
- NPTv6 - stateless - RFC 6296
- NAT64 - IPv6 Addresse nach IPv4 Adresse
- NAT46 - IPv4 Adresse nach IPv6 Adresse

IPv6 NAT = Network Address Translation

- funktioniert wie gehabt bei IPv4: eine externe routbare Adresse wird nach draussen benutzt, innere Adressen werden per portmapping darauf gemultiplext.

- nach innen static portmapping: bekannte externe Ports (i.e. 22, ssh) werden auf interne addr:port gemappt

- die üblichen probleme, leidlich bekannt

IPv6 NPT = Network Prefix Translation

- es passiert das naheliegende: Ein IPv6 Prefix wird - mehr oder weniger bidirektional - nach Regeln oder nicht - auf ein anderes IPv6 Prefix gemappt.

Test mit OPNsense

- Showstopper: strongswan, Linux Software auf FreeBSD, Mist! Die beste Kombination wäre m.M. mit racoon / ipsec-tools gegeben.
- Sehr schön: GUI Konfiguration, Paketverwaltung, sehr vielseitig
- Lernkurve, pf und oder FreeBSD Verständnis hilfreich

PC-Engines: alix 2d13

- CPU: 500 MHz AMD Geode LX800
- DRAM: 256 MB DDR DRAM
- Storage: CompactFlash socket, 44 pin IDE header
- Power: DC jack or passive POE, min. 7V to max. 20V
- Three front panel LEDs, pushbutton
- Expansion: 1 miniPCI slot, LPC bus
- Connectivity: 3 Ethernet channels (Via VT6105M 10/100)
- I/O: DB9 serial port, dual USB port
- Board size: 6 x 6" (152.4 x 152.4 mm) - same as WRAP.1E
- Firmware: tinyBIOS

FreeBSD

- Gottes eigenes Betriebssystem!
- FreeBSD 11.0 stable (crochet)
- 8G Compact Flash
- serielle Konsole
- IPv6 only

/etc/rc.conf #1

```
# vr0 = EXTERN - zur fritz!box, mainboard
#-------------------------------------------------------------------
--

ifconfig_vr0_ipv6="inet6 accept_rtadv"

ipv6_gateway_enable="yes"
ipv6_cpe_wanif="vr0"

rtsold_enable="YES"
rtsold_flags="-d -R /usr/local/etc/rtsold_script.sh vr0"

# vr1 = INTERN - zur clavister
#———————————————————————————

ifconfig_vr1_ipv6="inet6 fdee:33a:effa:113::254/64"

ipv6_static_routes="int cust"
ipv6_route_int="fdee:33a:effa:42::/64 fdee:33a:effa:113::1"
ipv6_route_cust="xxxx:yyyy:zzzz:aaaa::/64 fdee:33a:effa:113::1"
```

/etc/rc.conf.local

```
# ============================================================================

# ----------------------------------------------------------------------------
# MAIL
# ----------------------------------------------------------------------------

postfix_enable="YES"

# ----------------------------------------------------------------------------
# dhcp6c (KAME)
# ----------------------------------------------------------------------------

dhcp6c_enable="YES"
dhcp6c_interfaces="vr0"
dhcp6c_flags="-D"

# ============================================================================
```

ps -ax

```
/usr/sbin/unbound -c /var/unbound/unbound.conf

/usr/sbin/rtsold -d -R /usr/local/etc/rtsold_script.sh vr0

/usr/local/sbin/dhcp6c -D -c /usr/local/etc/dhcp6c.conf -p /var/run/dhcp6c.pid -D vr0

/usr/sbin/syslogd -s -n -cc -C

/usr/sbin/ntpd -g -c /etc/ntp.conf -p /var/run/ntpd.pid -f /var/db/ntpd.drift

/usr/local/libexec/postfix/master -w

qmgr -l -t unix -u

/usr/sbin/sshd

/usr/sbin/cron -s
```

dhcp6c.conf

```
interface vr0 {
        send ia-na 0;
        send ia-pd 0;
        send domain-name-servers;
        script "/usr/local/etc/dhcp6c_script.sh";
};

id-assoc na 0 {
};

id-assoc pd 0 {
};
```

pf.conf.nat-template #1

```
#-----------------------------------------------------------------------
# vr1 = intern
# vr0 = extern
# icmpv6 type, see icmp6(4)
icmp6_pass = "{unreach toobig timex paramprob echoreq echorep routersol
               routeradv neighbrsol neighbradv}"

set loginterface vr1
set loginterface vr0

scrub on vr1 all
scrub on vr0 all

# no filtering on loopback
set skip on lo0

# outbound nat
#-----------------------------------------------------------------------
nat on vr0 inet6 from fdee:33a:effa::/48 to any -> IPV6ADDRESSPLACEHOLDER/128
```

pf.conf.nat-template #2

```
# inbound nat redir
# ------------------------------------------------------------------
# SSH
rdr on vr0 inet6 proto tcp from any to IPV6ADDRESSPLACEHOLDER port 22  ->
        fdee:33a:effa:42::1
# SMTP
rdr on vr0 inet6 proto tcp from any to IPV6ADDRESSPLACEHOLDER port 25  ->
        fdee:33a:effa:42::1
# HTTPS
rdr on vr0 inet6 proto tcp from any to IPV6ADDRESSPLACEHOLDER port 443 ->
        fdee:33a:effa:42::1
# SMTPS
rdr on vr0 inet6 proto tcp from any to IPV6ADDRESSPLACEHOLDER port 465 ->
        fdee:33a:effa:42::1
rdr on vr0 inet6 proto tcp from any to IPV6ADDRESSPLACEHOLDER port 587 ->
        fdee:33a:effa:42::1
# IMAPS
rdr on vr0 inet6 proto tcp from any to IPV6ADDRESSPLACEHOLDER port 993  ->
        fdee:33a:effa:42::1
```

pf.conf.nat-template #3

```
# alle erlaubten icmpv6'e
#-------------------------------------------------------------------
pass quick inet6 proto ipv6-icmp from any to any icmp6-type $icmp6_pass keep
state

# ipv6 tcp/udp erlauben
#-------------------------------------------------------------------
pass quick inet6 proto {tcp udp} from any to any keep state

# everthing else: log & block
#-------------------------------------------------------------------
block log inet  from any to any
block log inet6 from any to any
```

dhcp6c_script.sh

```
#!/bin/sh

/usr/bin/logger -i -t dhcp6c_script.sh runs update_pf.sh, $*

/usr/local/etc/update_pf.sh -n
```

rtsold_script.sh

```
#!/bin/sh

/usr/bin/logger -i -t rtsold_script.sh runs /sbin/resolvconf and
                  update_pf.sh, $*

/sbin/resolvconf $*

/usr/local/etc/update_pf.sh -n
```

update_pf.sh #1

```
# hole ifconfig-output, wir wollen
#       - nur ipv6 adressen
#       - keine fe80 oder fdee adressen
#       - keine deprecated adressen
#       - keine preferred lifetime = 0 adressen
ADDRLINE=`ifconfig vr0 | grep inet6 | grep -v -e "inet6 f" -e "deprecated" -e "pltime 0"`

# diese Zeile, zweiter token ist es !
THIS_ADDRV6=`echo ${ADDRLINE} | awk '{print $2}'`

if [ ! "${THIS_ADDRV6}" -o -z "${THIS_ADDRV6}" ]
then
        logger -i -t $p: ipv6 address is empty !
        exit 1
fi

N=`echo "$ADDRLINE" | wc -l`
if [ $N -gt 1 ]
then
        logger -i -t $p: more than one ipv6 address: $ADDRLINE

#       exit 1

fi
```

update_pf.sh #2

```
# vorherige gemerkte adresse holen
if [ -f ${PREVIP_FILE} ]
then
        PREV_ADDRV6=`cat ${PREVIP_FILE}`
        logger -i -t $p: prev ${PREV_ADDRV6}, this ${THIS_ADDRV6}
else
        PREV_ADDRV6=""
        logger -i -t $p: prev addr empty, new addr ${THIS_ADDRV6}
fi
```

```
if [ "${PREV_ADDRV6}" = "${THIS_ADDRV6}" ]
then
        if [ $force_flag -eq 0 ]
        then
                logger -i -t $p: ${THIS_ADDRV6} equals previous address, exit script
                exit 0
        else
                logger -i -t $p: force_flag active, force update address ${THIS_ADDRV6}
        fi
fi

logger -i -t $p: update dyndns address on hh01
```

update_pf.sh #3

```
# warten, bis neue Adresse = DNS Adresse
while true
do
        # update dyndns entry on kokolores
        fetch -6 -q -o /dev/null "https://xyz:qwertz@dyndns.kokolores.gov/cgi-bin/
             updatenamesrv?zuhauseprivat&$address6={THIS_ADDRV6}"
        DNS_ADDRV6=`/usr/bin/host -t aaaa host.extern.zuhauseprivat.org |
                    /usr/bin/awk '{print $5}'`
        logger -i -t $p: check DNS, new addr ${THIS_ADDRV6}, DNS addr ${DNS_ADDRV6}
        if [ "${THIS_ADDRV6}" = "${DNS_ADDRV6}" ]
        then
                break
        fi
        sleep 15
done

# die neue adresse fuers naechstemal sichern
echo "${THIS_ADDRV6}" > ${PREVIP_FILE}
```

update_pf.sh #4

```
# die adresse ins template reinfiedeln
cat /usr/local/etc/pf.conf.nat-template | \
    sed "s/IPV6ADDRESSPLACEHOLDER/${THIS_ADDRV6}/g" > /usr/local/etc/pf.conf.new

# altes file sichern
mv /etc/pf.conf /etc/pf.conf.old

# und neues ueberschreiben
mv /usr/local/etc/pf.conf.new /etc/pf.conf

# im pf aktivieren
pfctl -f /etc/pf.conf
pfctl -e

logger -i -t $p: NAT-mode, new addr ${THIS_ADDRV6} ready

exit 0
```

neighbour discovery

RFC 4861

- Rechner senden Router Solicitation Messages um Router Advertisements zu triggern
- Router senden periodisch Router Advertisements oder als Antwort auf Router Solicitation

RFC 5175

- Router Advertisement Flag Option

```
0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
|M|O|H|Prf|P|R|R|
+-+-+-+-+-+-+-+-+
```

Figure 1: Router Advertisement Flags

- o M – Managed Address Configuration Flag [RFC4861] (Adressen per DHCP verfügbar)
- o O – Other Configuration Flag [RFC4861] (andere Informationen per DHCP verfügbar)
- o H – Mobile IPv6 Home Agent Flag [RFC3775]
- o Prf – Router Selection Preferences [RFC4191] (High - Medium - Low)
- o P – Neighbor Discovery Proxy Flag [RFC4389]
- o R – Reserved

router discovery

```
17:18:33.361357 IP6 (hlim 255, next-header ICMPv6 (58) payload length: 16)
  fe80::20d:b9ff:fe1e:cf00 > ff02::2: [icmp6 sum ok] ICMP6, router solicitation,
  length 16 source link-address option (1), length 8 (1): 00:0d:b9:1e:cf:00

17:18:33.364043 IP6 (hlim 255, next-header ICMPv6 (58) payload length: 160)
 fe80::ca0e:14ff:fe0b:3730 > ff02::1: [icmp6 sum ok] ICMP6, router advertisement,
 length 160 hop limit 255, Flags [managed, other stateful], pref high,
 router lifetime 1800s, reachable time 0ms, retrans timer 0ms
prefix info option (3), length 32 (4): 2a04:4540:8c00:4800::/64,
  Flags [onlink, auto], valid time 7200s, pref. time 3600s
prefix info option (3), length 32 (4): fdee:33a:effa:100::/64,
 Flags [onlink, auto], valid time 7200s, pref. time 3600s
rdnss option (25), length 24 (3):  lifetime 1200s,
 addr: fdee:33a:effa:100:ca0e:14ff:fe0b:3730
mtu option (5), length 8 (1):  1492
route info option (24), length 8 (1):  ::/0, pref=high, lifetime=1800s
route info option (24), length 16 (2):  2a04:4540:8c00:4800::/56, pref=high,
 lifetime=1800s
route info option (24), length 16 (2):  fdee:33a:effa:100::/64, pref=high,
 lifetime=1800s
source link-address option (1), length 8 (1): c8:0e:14:0b:37:30
```

dhcp #1

```
17:18:36.022367 IP6 (hlim 1, next-header UDP (17) payload length: 68)
 fe80::20d:b9ff:fe1e:cf00.dhcpv6-client > ff02::1:2.dhcpv6-server:
 [udp sum ok] dhcp6 solicit
 (xid=ae475d
  (client-ID hwaddr/time type 1 time 539088934 000db91ecf00)
  (IA_NA IAID:0 T1:0 T2:0)
  (elapsed-time 0) (IA_PD IAID:0 T1:0 T2:0))

17:18:36.027606 IP6 (hlim 64, next-header UDP (17) payload length:178)
fe80::ca0e:14ff:fe0b:3730.dhcpv6-server >
fe80::20d:b9ff:fe1e:cf00.dhcpv6-client:
[udp sum ok] dhcp6 advertise
(xid=ae475d
 (client-ID hwaddr/time type 1 time 539088934 000db91ecf00)
 (server-ID hwaddr type 1 c80e140b3730)
 (preference 200) (DNS-server fdee:33a:effa:100:ca0e:14ff:fe0b:3730)
 (opt_86)
 (IA_NA IAID:0 T1:1800 T2:2880
  (IA_ADDR 2a04:4540:8c00:4800:20d:b9ff:fe1e:cf00 pltime:3600 vltime: 7200))
 (IA_PD IAID:0 T1:1800 T2:2880
  (IA_PD-prefix 2a04:4540:8c00:48fc::/62 pltime:3600 vltime:7200)))
```

dhcp #2

```
17:18:37.032403 IP6 (hlim 1, next-header UDP (17) payload length: 139)
 fe80::20d:b9ff:fe1e:cf00.dhcpv6-client > ff02::1:2.dhcpv6-server: [udp sum ok]
 dhcp6 request
 (xid=df715c
  (client-ID hwaddr/time type 1 time 539088934 000db91ecf00)
  (server-ID hwaddr type 1 c80e140b3730)
  (IA_NA IAID:0 T1:0 T2:0
   (IA_ADDR 2a04:4540:8c00:4800:20d:b9ff:fe1e:cf00 pltime:3600 vltime:7200))
  (elapsed-time 0)
  (IA_PD IAID:0 T1:0 T2:0
   (IA_PD-prefix 2a04:4540:8c00:48fc::/62 pltime:3600 vltime:7200)))

17:18:37.038850 IP6 (hlim 64, next-header UDP (17) payload length: 178)
 fe80::ca0e:14ff:fe0b:3730.dhcpv6-server > fe80::20d:b9ff:fe1e:cf00.dhcpv6-client:
 [udp sum ok] dhcp6 reply
  (xid=df715c
   (client-ID hwaddr/time type 1 time 539088934 000db91ecf00)
   (server-ID hwaddr type 1 c80e140b3730) (preference 200)
   (DNS-server fdee:33a:effa:100:ca0e:14ff:fe0b:3730)
   (opt_86)
   (IA_NA IAID:0 T1:1800 T2:2880
    (IA_ADDR 2a04:4540:8c00:4800:20d:b9ff:fe1e:cf00 pltime:3600 vltime:7200))
   (IA_PD IAID:0 T1:1800 T2:2880
    (IA_PD-prefix 2a04:4540:8c00:48fc::/62 pltime:3600 vltime:7200)))
```

neue IPv6 Adresse

```
Mar  7 02:25:44 zapp.wanext.hallstr.de rtsold_script.sh[11652]: runs
                /sbin/resolvconf and update_pf.sh, -a vr0:slaac
Mar  7 02:25:44 zapp.wanext.hallstr.de update_pf.sh:[11673]: prev
                2a04:4540:8c03:f500:20d:b9ff:fe1e:cf00, this
                2a04:4540:8c00:db00:20d:b9ff:fe1e:cf00
Mar  7 02:25:44 zapp.wanext.hallstr.de update_pf.sh:[11674]: update dyndns address
Mar  7 02:25:47 zapp.wanext.hallstr.de update_pf.sh:[11679]: check DNS,
                new addr 2a04:4540:8c00:db00:20d:b9ff:fe1e:cf00,
                DNS addr 2a04:4540:8c00:db00:20d:b9ff:fe1e:cf00
Mar  7 02:25:47 zapp.wanext.hallstr.de update_pf.sh:[11686]: NAT-mode, new addr
                2a04:4540:8c00:db00:20d:b9ff:fe1e:cf00 ready



Mar  7 04:18:40 zapp.wanext.hallstr.de dhcp6c[553]: prefix timeout for
                2a04:4540:8c03:f5fc::/62
Mar  7 04:18:40 zapp.wanext.hallstr.de dhcp6c[553]: remove a site prefix
                2a04:4540:8c03:f5fc::/62
Mar  7 04:18:40 zapp.wanext.hallstr.de dhcp6c[553]: IA PD-0 is invalidated
```