RUHR-UNIVERSITÄT BOCHUM

FIRST EXPERIENCES WITH DEPLOYING IPV6-MOSTLY

From the perspective of a University Campus

Introduction

- Robin Därmann
- Technically responsible for the data network of the Ruhr-University Bochum
- NOC Team with seven members
- Campus network with ~3.000 devices, ~140.000 Ethernet access ports, ~2.600 wireless APs (>1.000 more to come)
- A lot of self-made automation (perl mostly)

Motivation

- Trying to push IPv6 wherever possible since 2011
- Traditionally, the Dual-Stack approach
- Since 2022 we do some NAT44 because the lack of official IPv4 addresses
- NAT44 with IPv6 in parallel works, but this can be done „better“. Because...

NOT EVEN A SINGLE IPv4 ADDRESS HAS BEEN SAVED UNTIL NOW!

The long way...

- We can‘t do a hard cut (almost nobody can), so we have to migrate somehow
- „Better“ approach than simply going Dual-Stack would be to force nodes to do IPv6 whenever possible and to avoid NAT44

- NAT64 with DNS64 came to our mind, but it has caveats:
  - Like with NAT44, some protocols need special handling (i.e. IPsec)
  - Communication with IPv4 literals (i.e. **`ping 203.0.113.1`**) does not work
  - Legacy software wants to use IPv4-only sockets

...to IPv6-mostly

- Better approach: 464XLAT (RFC 6877)
  - Uses NAT64 on the provider side (PLAT) and a SIIT translator on the node (CLAT)
  - IPv4 translation to IPv6 directly on the node and back to IPv4 at the provider
  - No manipulated DNS like with DNS64 → no DNSSEC problems

- It also has caveats:
  - Originally designed for Mobile ISPs operating large IPv6-only networks
  - Therefore, some popular operating systems don‘t support it on WiFi or LAN

About IPv6-mostly

- Combination of a DHCPv4 Option, an IPv6 RA Option and NAT64 / 464XLAT
  - RFC 8925: IPv6-Only Preferred Option for DHCPv4
  - RFC 8781: Discovering PREF64 in Router Advertisements
- Saves IPv4 addresses because endpoints can (and will) decide to use only IPv6
- Nodes without support will simply run dual-stack, but with DNS64 most traffic will be IPv6

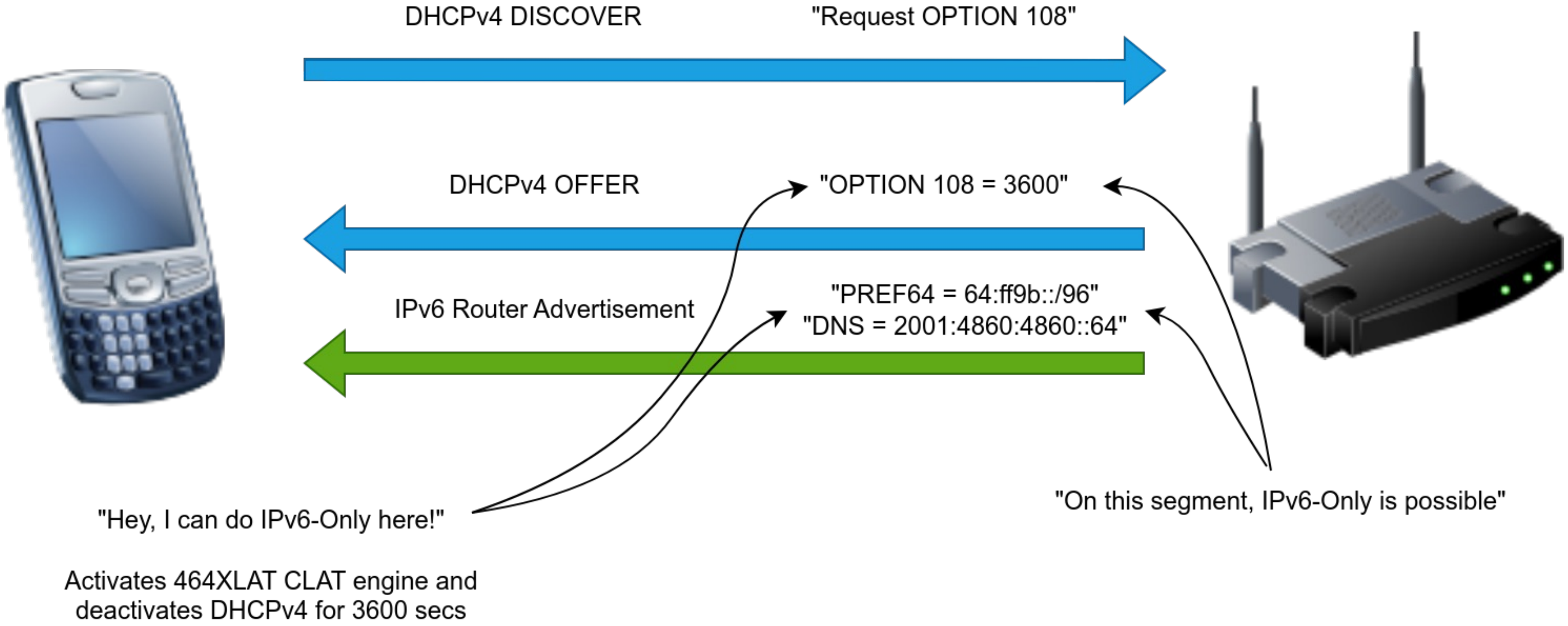
How does it work?

- DHCPv4 option 108 („IPv6-Only Preferred“) is introduced
  - requested in DHCPDISCOVER and DHCPREQUEST by willing DHCPv4 clients
  - sent back in DHCPOFFER by the DHCPv4 server, with a positive integer value
  - → client may disable IPv4 stack for the time specified in the option value (in seconds)

- Router Advertisement Option Type 38 („PREF64“) is introduced
  - sent in Router Advertisements, contains a usable NAT64 prefix
  - Used for local DNS64 and CLAT configuration
  - shares fate with other configuration parameters (i.e. DNS servers)

- Receivement of both option 108 and PREF64 is necessary for IPv6-mostly to work

Onboarding with support for IPv6-mostly



DHCPv4 DISCOVER

"Request OPTION 108"

DHCPv4 OFFER

"OPTION 108 = 3600"

IPv6 Router Advertisement

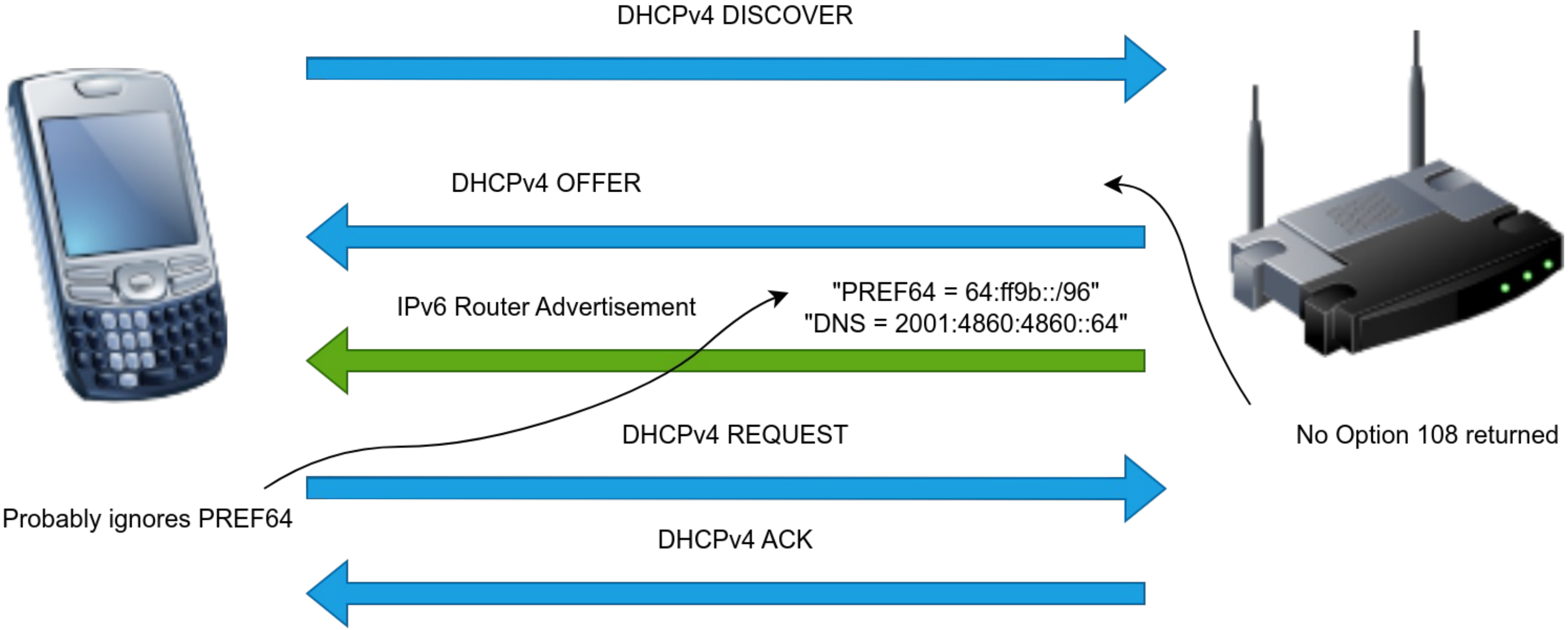
"PREF64 = 64:ff9b::/96"
"DNS = 2001:4860:4860::64"

"Hey, I can do IPv6-Only here!"

Activates 464XLAT CLAT engine and
deactivates DHCPv4 for 3600 secs

"On this segment, IPv6-Only is possible"

Onboarding without support for IPv6-mostly



Uses IPv4 and IPv6 with DNS64 capable nameservers

Statistics – who requests option 108?

- Campus-wide wireless segment with authentication via Captive Portal
- DHCPDISCOVER captured between and 2023-08-31 and 2023-11-01 (9 weeks)
- 10,794 different MACs / devices in total
  - 7,772 devices requested option 108 in DHCPDISCOVER (72%)
  - 3,022 devices did not request option 108 (28%)

Statistics – who doesn‘t want to play along?

- 3,022 devices did not request option 108 (28%)
- Quick check with Fingerbank API showed:
  - 1,932: „Operating System/Windows OS/Microsoft Windows Kernel 10.0“
  - 243: „Operating System/Apple OS“
  - 191: „Operating System/Google OS/Android OS“
  - 128: „Phone, Tablet or Wearable/Generic Android/Samsung Android“
  - 117: „Phone, Tablet or Wearable/Generic Android/Huawei Android“
  - 102: „Operating System/Linux OS“
  - ...
  - 4: „Internet of Things (IoT)/Appliance/iRobot/iRobot Roomba“

DHCPv4 Server Configuration

- ISC Kea:

```
"subnet4": { "option-data": [ "name": "v6-only-preferred", "data": "3600" ] }
```

- ISC DHCP Server (the older one):

```
option option-108 "3600";
```

- Any other:
  - You can send custom option with code 108 and unsigned integer value

Router Configuration

- Cisco IOS-XE >= 17.11.1:

```
interface Vlan888
 ip address 10.10.10.1 255.255.255.0
 ip helper-address 10.1.1.1
 ipv6 address 2001:db8:1::1/64
 ipv6 nd ra nat64-prefix 64:FF9B::/96
 ipv6 nd ra dns-search-list domain your-domain.tld
 ipv6 nd ra dns server 2001:4860:4860::64
 ipv6 nd ra dns server 2001:4860:4860::6464
```

- Arista EOS >= 4.31.0F:

```
 ipv6 nd ra pref64 64:ff9b::/96
```

Client Configuration / Issues

- Don‘t set up custom DNS servers!
- Don‘t disable IPv6 entirely on the client!
- Don‘t use multiple IPv6 prefixes on the link!
  - macOS will pick a random one for CLAT, without considering any deprecated or ULA prefix on the interface
- Don‘t disable SLAAC (i.e. `ipv6 nd prefix default no-autoconfig`)
- Have free addresses in your DHCPv4 pool!
  - DHCPv4 servers don‘t send DHCPOFFER if there are no free leases
- Cisco routers create an ARP entry if they relay a DHCPOFFER (at least seen on IOS-XE 17.12.1)
  - DHCPOFFER always includes a valid IPv4 address from the pool
  - ARP table gets full of unused entries that time out after a while

Conclusion

- Next step will be to roll out some wired segments with IPv6-mostly configuration and gather statistics
- Some (all, to be honest) OS vendors should implement the necessary components (464XLAT, CLAT, option 108)
- The numbers are getting better (more than 70% of devices request option 108 already)
- IPv6-mostly is the way to go

That‘s it

- Questions?

- Contact: robin.daermann@rub.de