RIPE NCC
RIPE NETWORK COORDINATION CENTRE

Improving IPv6-only experience on Linux

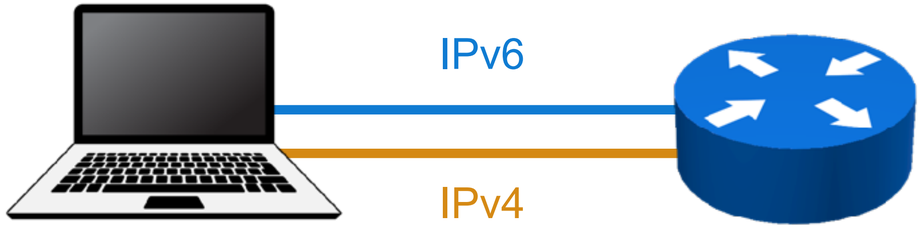
How to get rid of the last dependencies on IPv4

Ondřej Caletka | 3 February 2024 | FOSDEM 2024

IPv6? You mean Dual Stack!

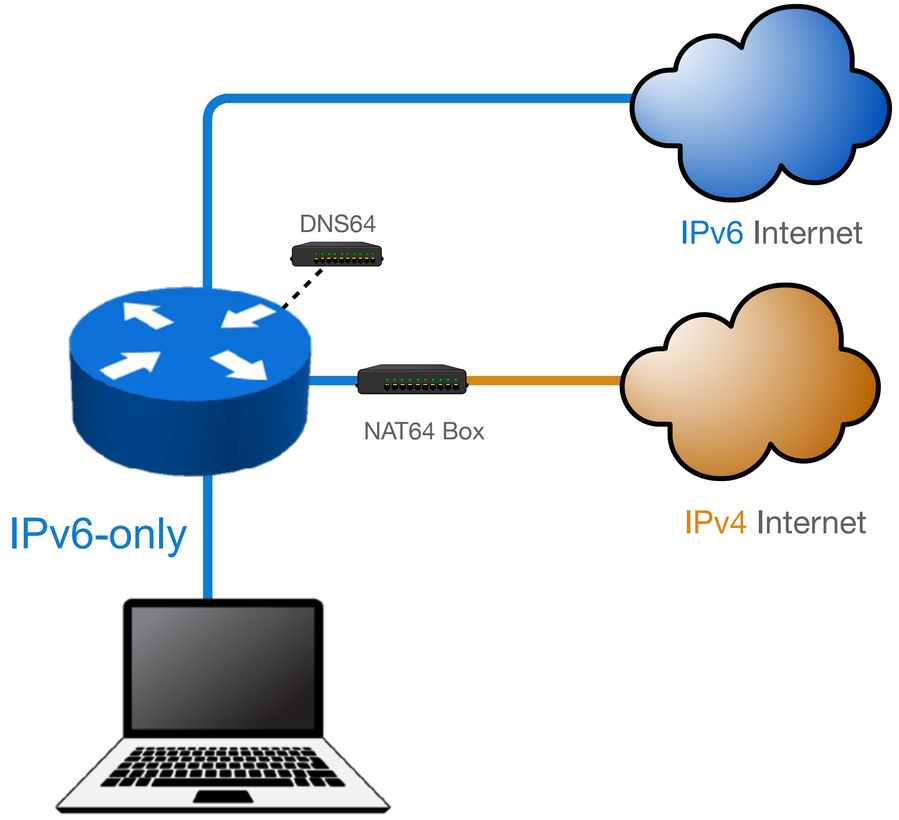
- IPv4-only and IPv6-only resources **directly accessible**
- IPv6 preferred for dual-stack resources
- Problems with IPv6 **masked** by Happy Eyeballs algorithm
- But it **does not address IPv4 scarcity**

Dual Stack



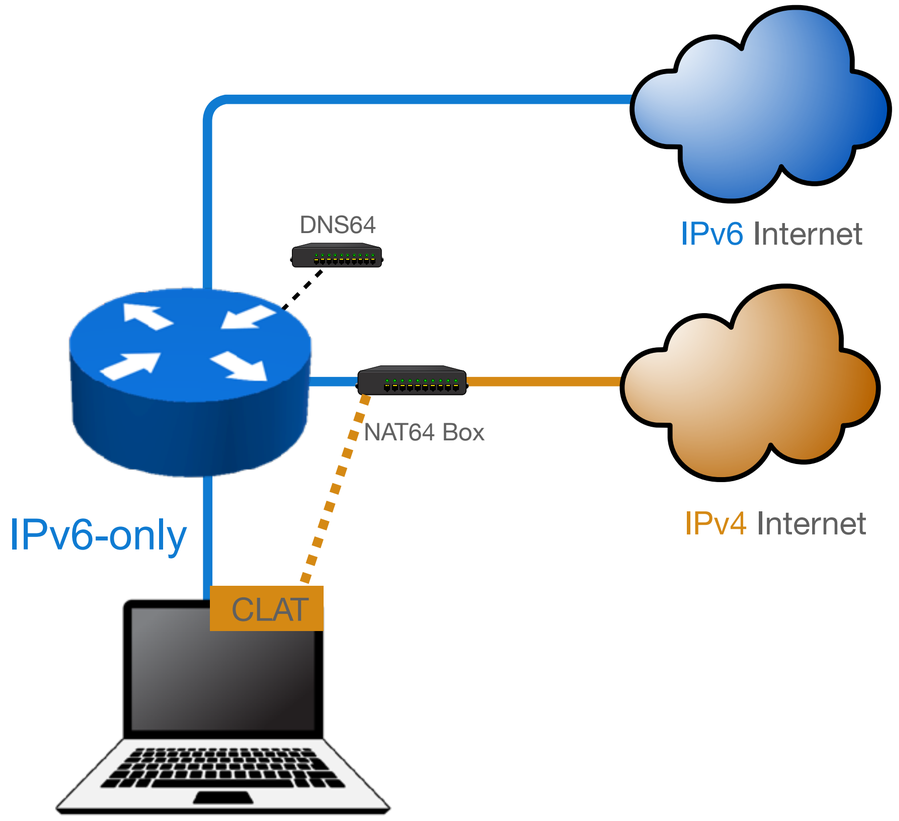
NAT64 allows IPv6-only networks

- IPv6 accessible **natively**
- IPv4 is **translated** into part of IPv6 address space
- Together with **DNS64**, everything seems to be **accessible over IPv6**
- **But sometimes you run into…**
  - IPv4 literals
  - Legacy software opening IPv4-only sockets
  - Dual-stack servers with broken IPv6



464XLAT closes the gap

- **CLAT** translator inside the host
- Translates residual IPv4 traffic to IPv6
- Translated IPv6 traffic get translated to IPv4 by NAT64 = **PLAT**
- Applications see *good old* dual-stack

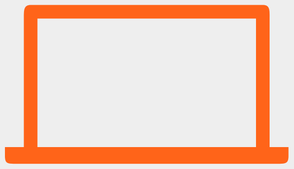


Can my device work on IPv6-only?

Fully

Android

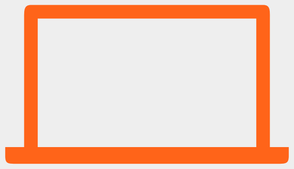
iOS



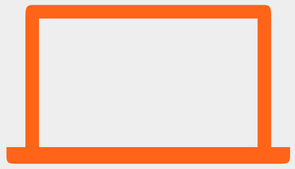
macOS

- CLAT is present*
- Some mobile networks run billions of IPv6-only phones for years already

Mostly



Windows



Linux

- No CLAT*
- Applications relying on IPv4 are **broken**

No way!

IoT

Smart home

- No IPv6 support*
- Native IPv4 **required**

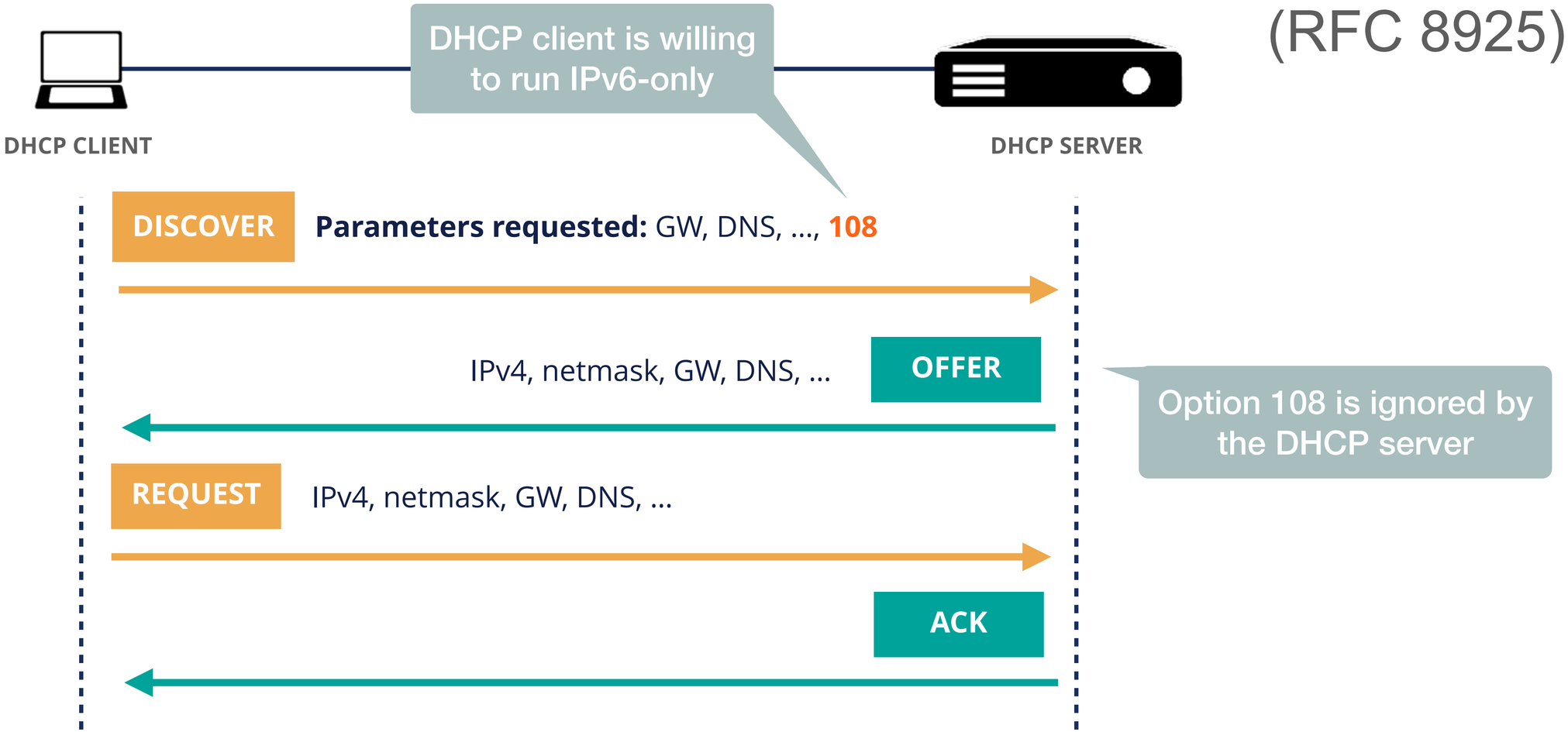
***) some statements are simplified**

Can we do IPv6-only?

At least for those devices that support it?

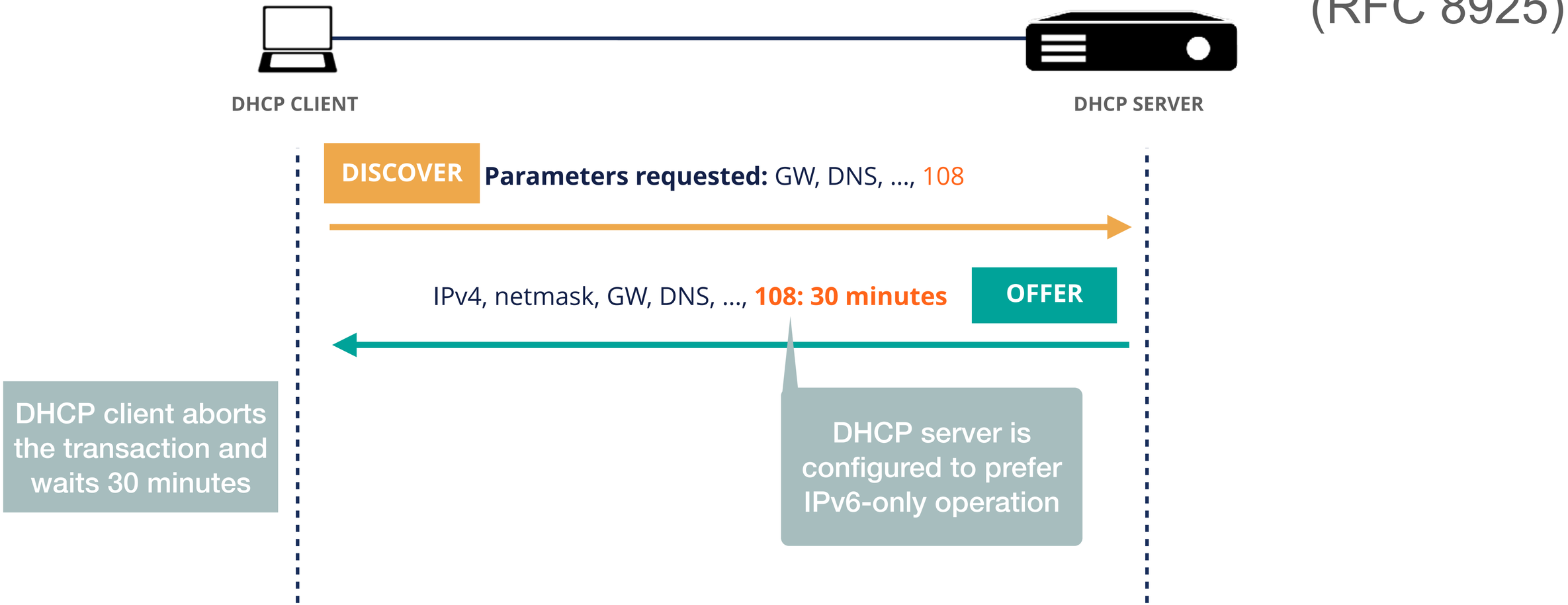
IPv6-only Preferred option of DHCP

(RFC 8925)



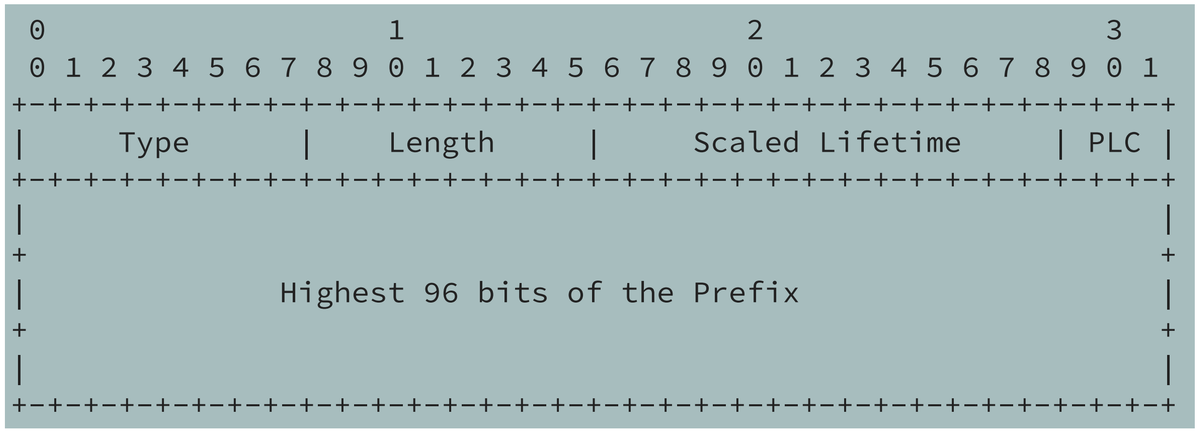
Using DHCP to turn IPv4 off

(RFC 8925)



What is an IPv6-mostly network?

- A network designed to **run IPv6-only**, but still providing **some IPv4** for legacy devices
- Must provide *perfectly* working IPv6 **with NAT64 support**
  - NAT64 prefix should be signalled using **PREF64** option of RAs
- Must provide **native IPv4**
  - DHCP server must offer option 108



PREF64 option of Router Advertisement

Linux on IPv6-only

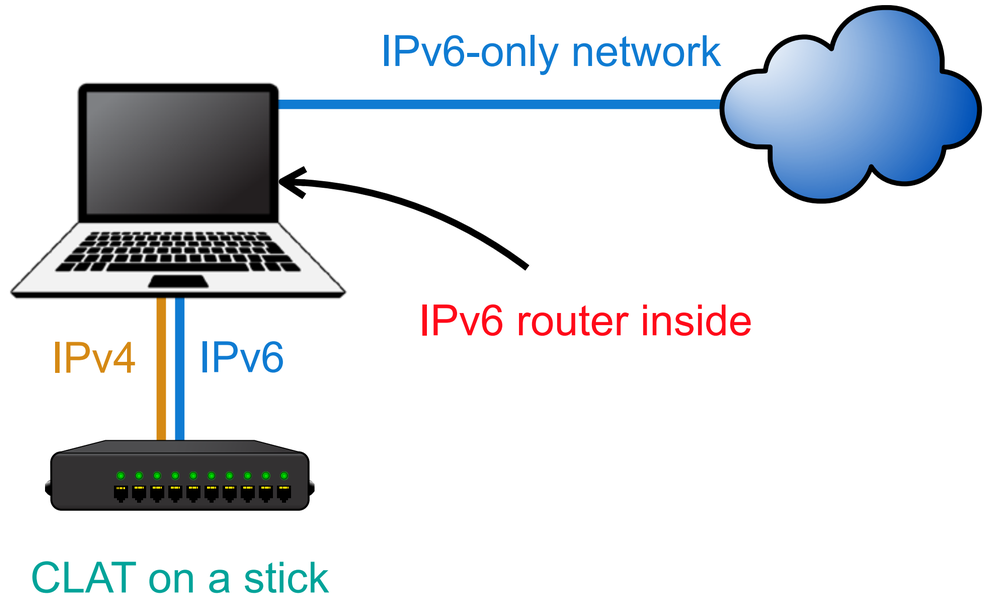
How to avoid the need to have
native IPv4 on Linux

What needs to be done

- **Option 1: fix everything** that has a hard dependency on IPv4
  - Mostly done, reaching 100% is **virtually impossible**
- **Option 2:** make sure **CLAT is there** for residual issues
  - The **most complex problem**
  - Requires **third-party software** and a **proper orchestration**
- **Then finally:** implement handling of DHCP option 108
  - So even IPv6-mostly network works as IPv6-only for Linux
  - This should not be enabled **before implementing Option 2**
  - We **already have this implemented** in `dhcpcd` and `systemd-networkd`

Running CLAT on Linux

- No **native kernel support** for address family translation
- Third party software like TAYGA, tundra-nat64, nat46, Jool
- Perl script `clatd`
  - detects if CLAT is needed
  - uses TAYGA or nat46 for actual translation
  - sets up addressing, forwarding, firewall rules
  - does not react to **renumbering**
  - does not support **multiple instances**

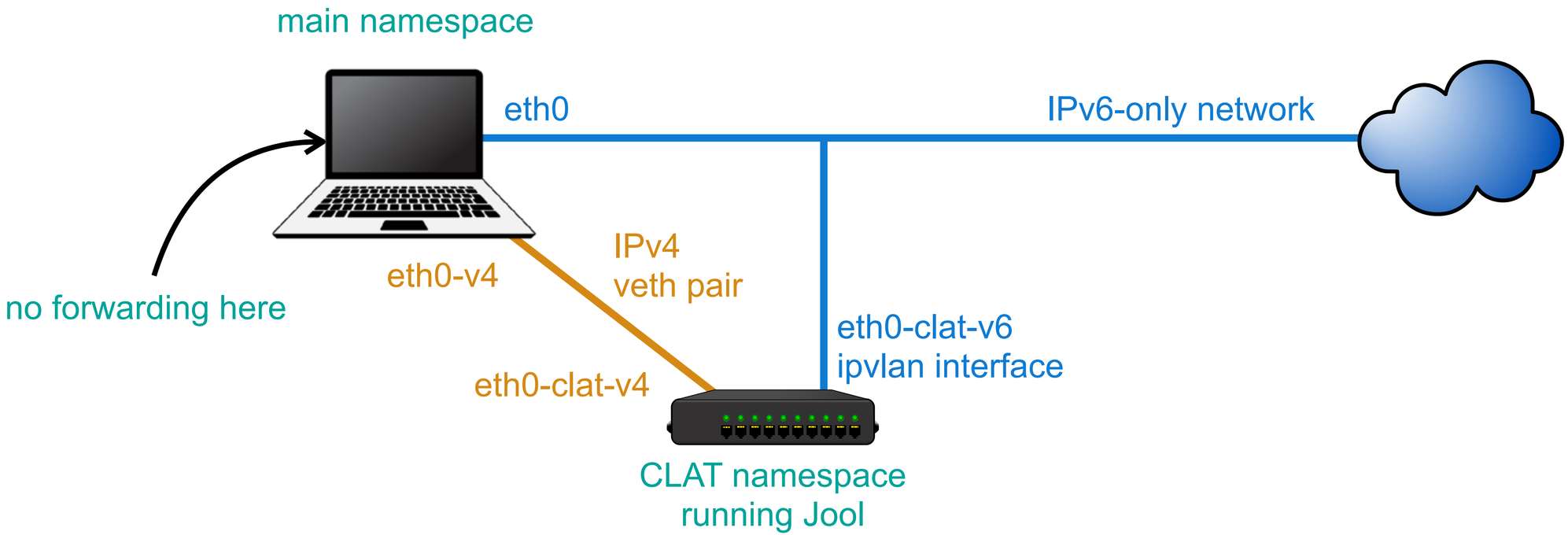


Ideal CLAT

- Supports **multiple instances**
  - One per interface
  - Should deal with **conflicting NAT64 prefixes**
- Sets itself up as soon as **NAT64 is detected**
  - Either using **PREF64** RA option or doing `ipv4only.arpa` DNS64 detection
  - Installs IPv4 default route with a **higher metric than potential native IPv4 route**
- **Reacts dynamically** to changing conditions
- **Does not touch** firewall of forwarding

Individual draft in IETF: CLAT Node Recommendations

Using ipvlan and namespaces



Using ipvlan and namespaces

- **No change to routing or firewall** of the main namespace
  - ipvlan will **branch a single IPv6 address** to the CLAT namespace
  - IPv4 is provided via a new network interface (`veth` link to the CLAT namespace)
- Supports **multiple instances**, even with conflicting prefixes
  - the only issue is to assign a **unique IPv4 address** from 192.0.0.0/29
- **Any translator** can run in the CLAT namespace
  - for instance: kernel-space **Jool** if available, userspace **tundra-nat64** as a fallback
- Simple teardown without any side effects
  - just **delete the network namespace**

What is missing

- A software responsible for **setting up**, (re-)**configuring** and **tearing down** the CLAT
  - detect NAT64 presence
  - set up a *(checksum-neutral)* IPv6 address for the CLAT
  - assigns a **free IPv4 address** from 192.0.0.0/29
  - react to subsequent RAs and adjust **configuration on the fly**

- Ideally integrated in common Linux distributions

Questions

Ondrej@Caletka.cz
https://Ondřej.Caletka.nl
@Oskar456@mastodon.social