



**RIPE NCC**  
RIPE NETWORK COORDINATION CENTRE

# Přístupové sítě typu IPv6-mostly

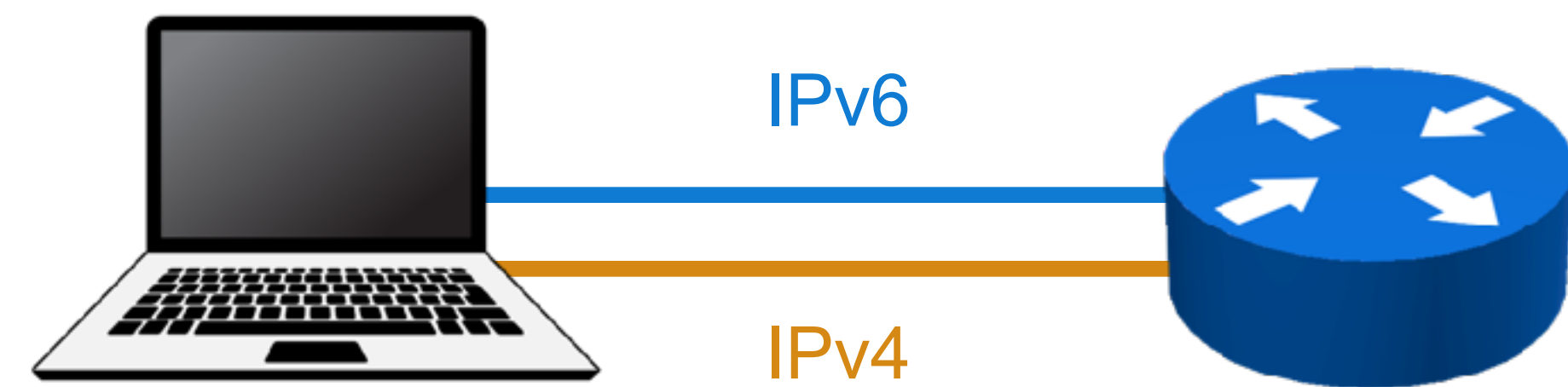
IPv6-only a dual stack v jedné síti

# Nejlepší přechodový mechanismus



- IPv4-only i IPv6-only zdroje **přímo dostupné**
- IPv6 preferováno pro zdroje s dual-stackem
- Problémy s IPv6 **skryty** algoritmem Happy Eyeballs
- Bohužel **neřeší nedostatek IPv4 adres**

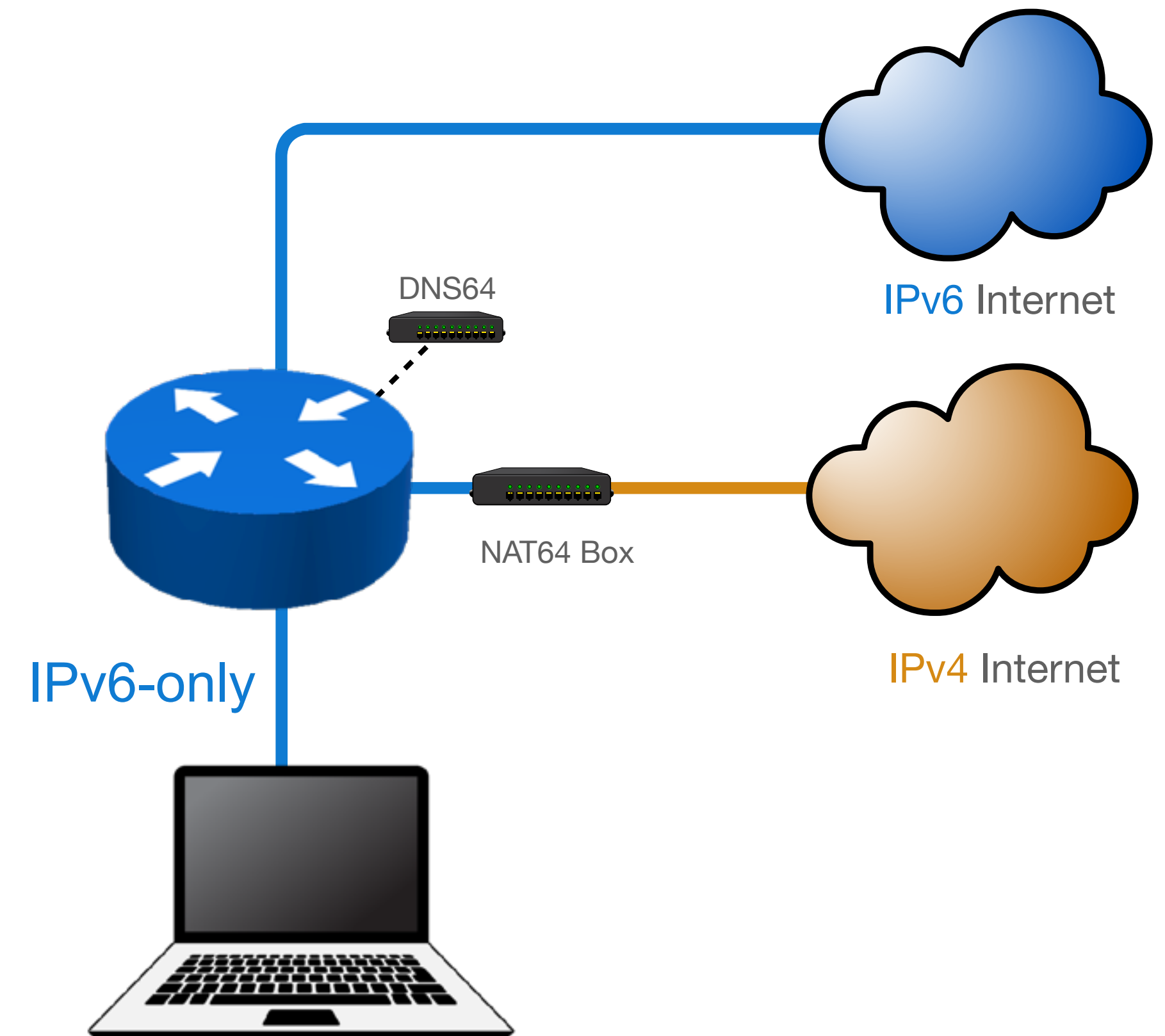
## Dual Stack



# NAT64 umožňuje sítě IPv6-only



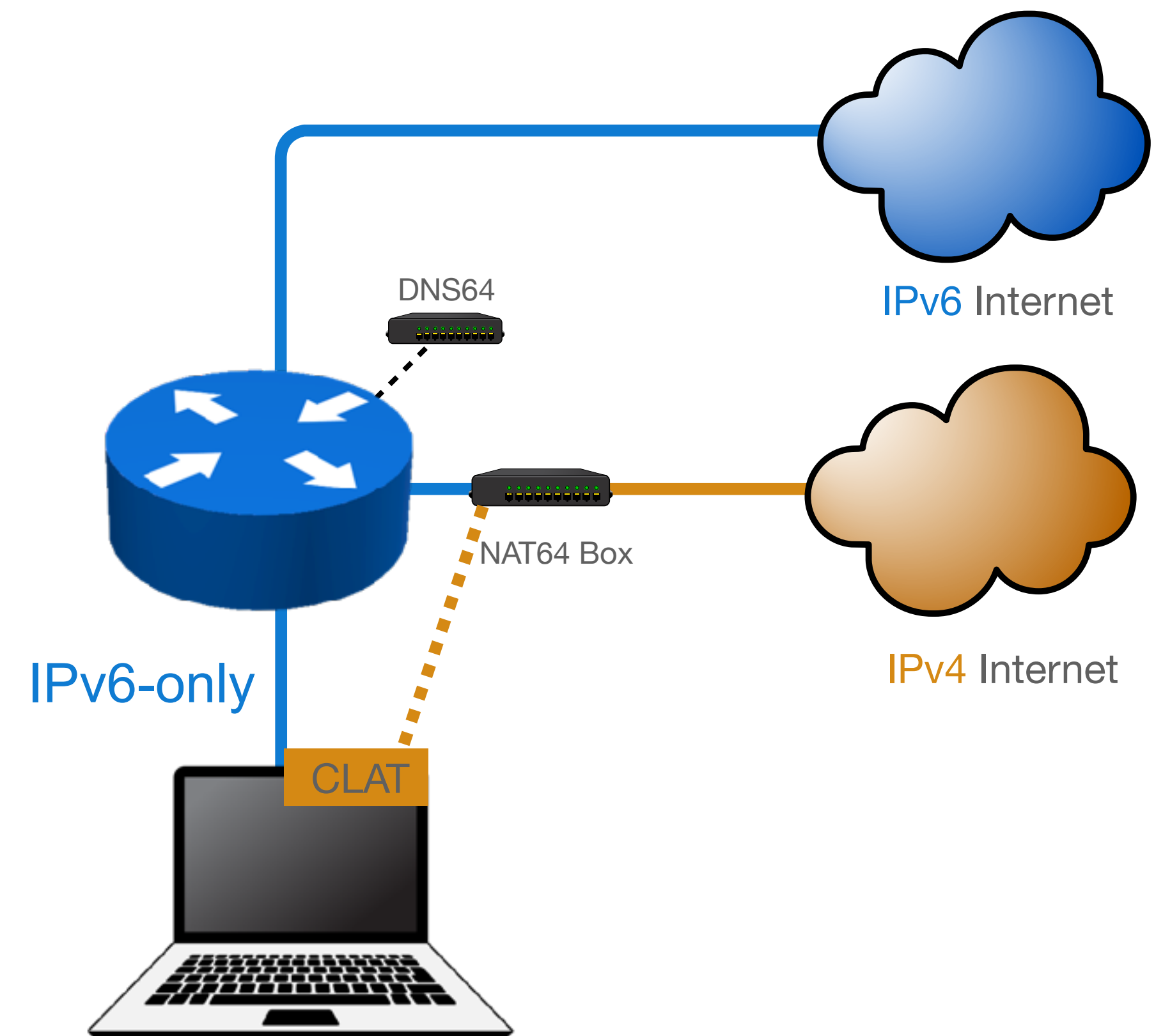
- IPv6 je dostupné nativně
- IPv4 je přeloženo do části adresního prostoru IPv6
- Díky **DNS64** se vše zdá být dostupné pouze pomocí IPv6
- **Funguje skvěle až na...**
  - IPv4 literály
  - Zastaralý software používající IPv4-only sokety
  - Dual-stackové servery s rozbitým IPv6



# Mobilní platformy jsou připravené



- Apple vynucuje podporu sítí s NAT64 pro všechny aplikace pro iOS
- Algoritmus **Happy Eyeballs 2.0** řeší IPv4 literály a rozbité IPv6 na dual-stackových serverech
- Pro tethering se používá **CLAT**
- Android používá pouze CLAT (464XLAT)
  - IPv4 je aplikacím dostupné dvěma překlady



# Desktopy mají na IPv6-only problémy



- Happy Eyeballs 2.0 neexistuje mimo Apple
  - I na Apple je dostupné pouze u vysokoúrovňových API (*např. Safari, ale ne Firefox*)
  - Chrome nedávno získalo funkci „Use NAT64 translation for IPv4 literals“
- **CLAT** není k dispozici na Windows, Linuxu, nebo ChromeOS
- Dobře známé **drobné problémy**:
  - Zastaralé aplikace s IPv4-only sokety
  - Nefunkční IPv4 literály (*s výjimkou Chrome*)
  - Problémy s dual-stack servery, kde IPv6 nefunguje
  - Tradiční Happy Eyeballs **nepomáhá** protože není k dispozici IPv4 pro fallback
  - Korporátní VPN často nefungují (nejčastěji *jen* kvůli konfiguraci)



# Lze IPv6-only provozovat?

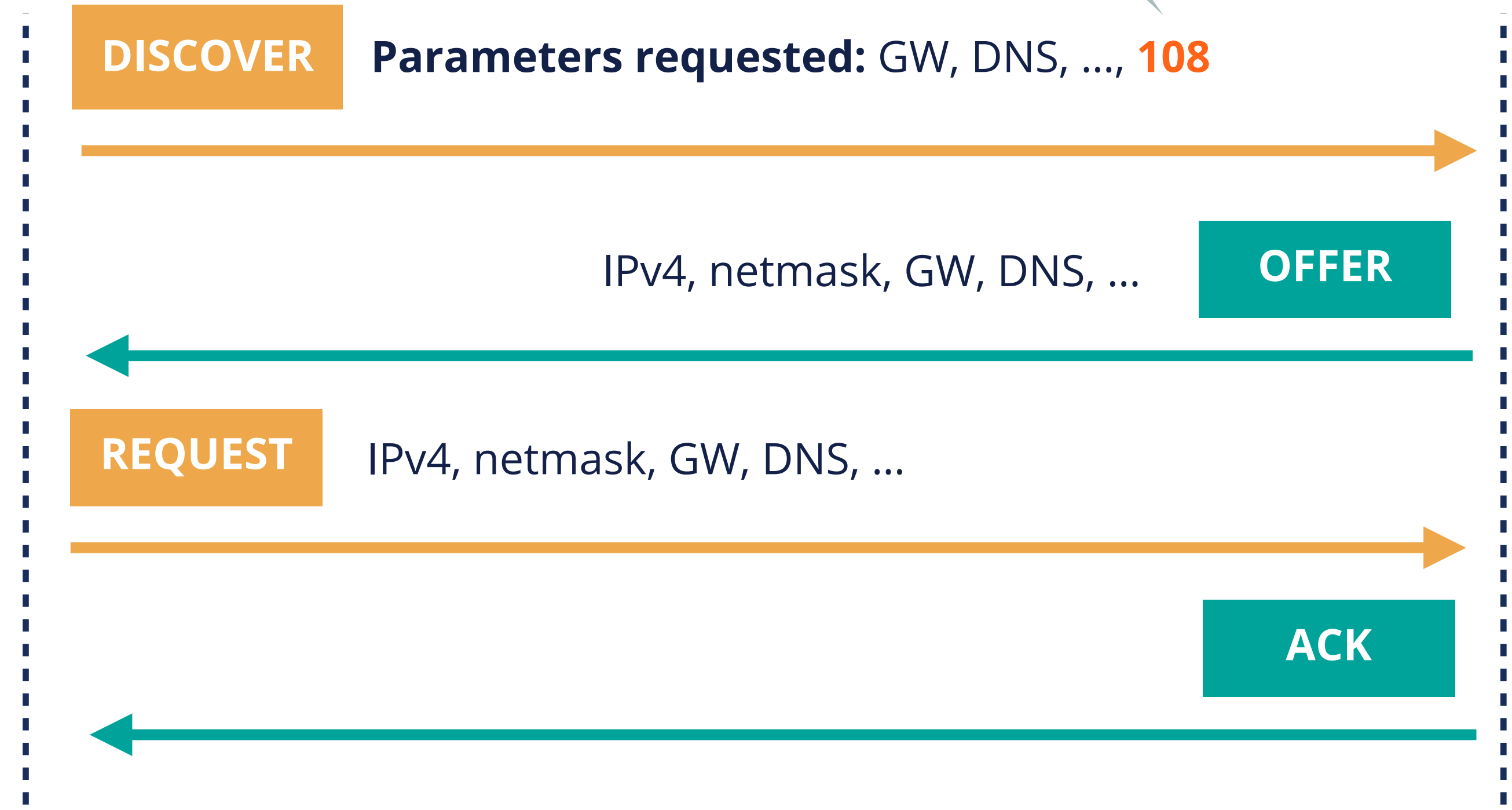
...aspoň pro moderní zařízení?



# Volba DHCP: IPv6-only Preferred



(RFC 8925)

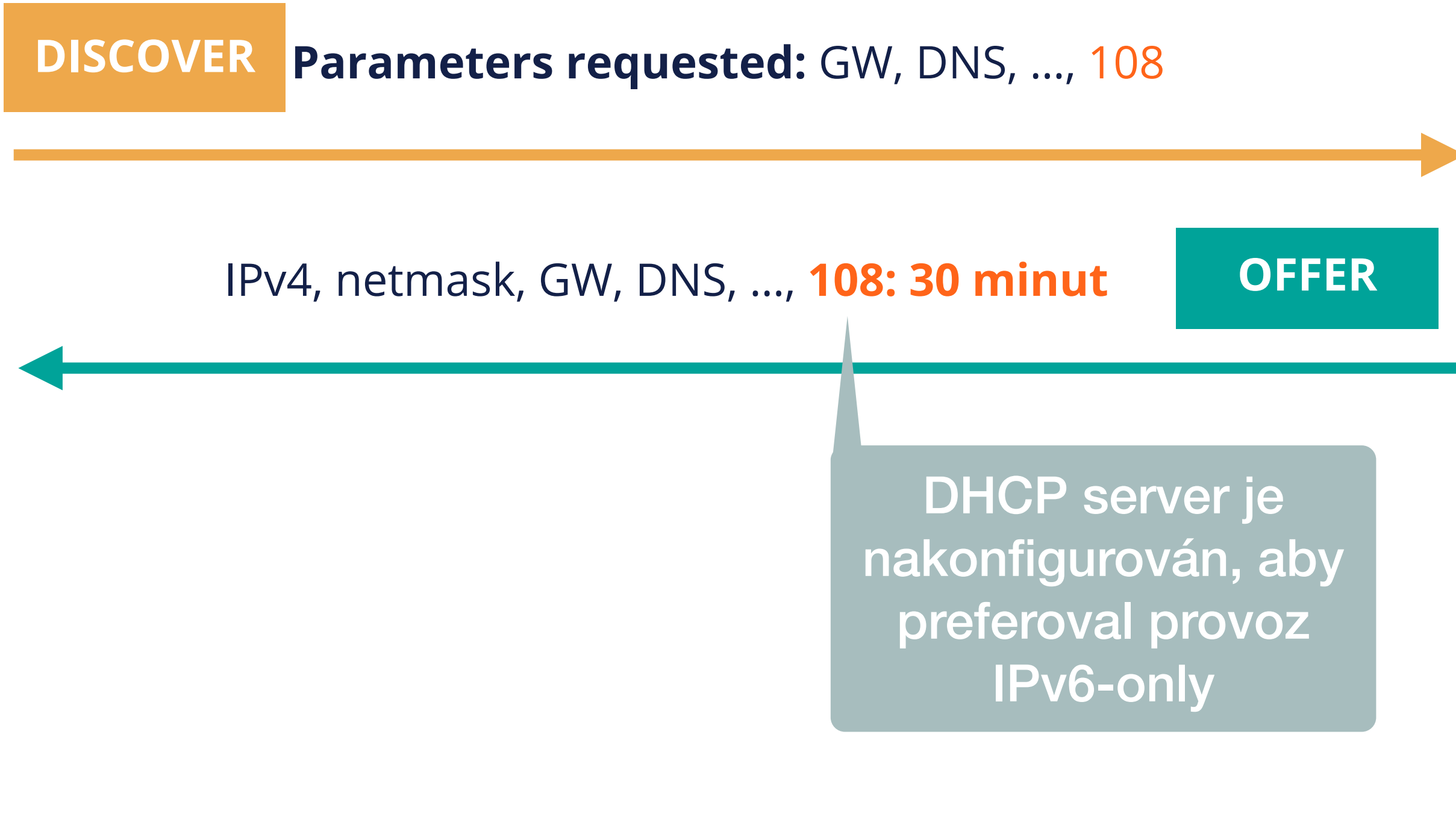


Volbu 108 DHCP server ignoruje

# Použití DHCP k vypnutí IPv4



(RFC 8925)



Klient DHCP ukončí předčasně transakci a čeká 30 minut

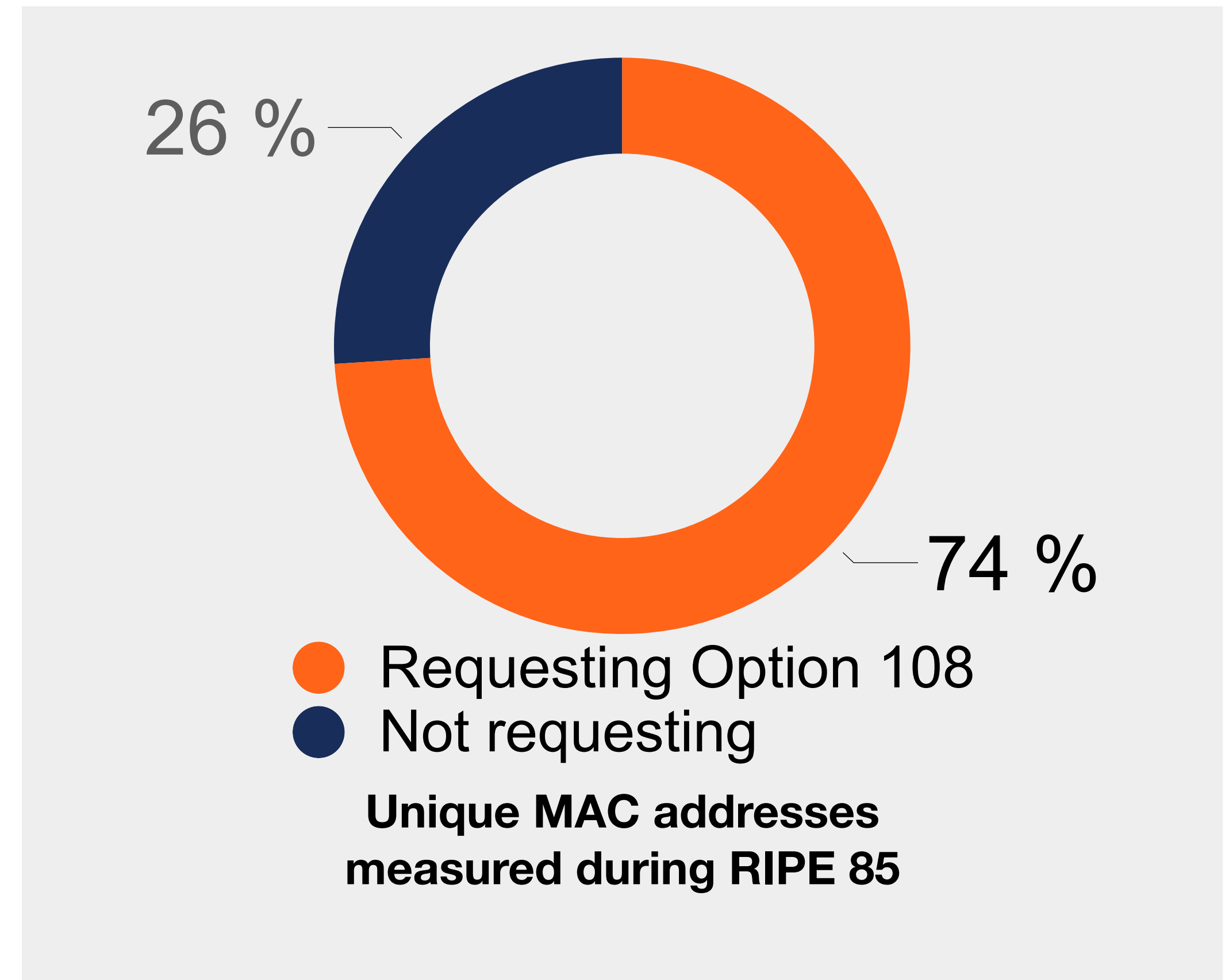
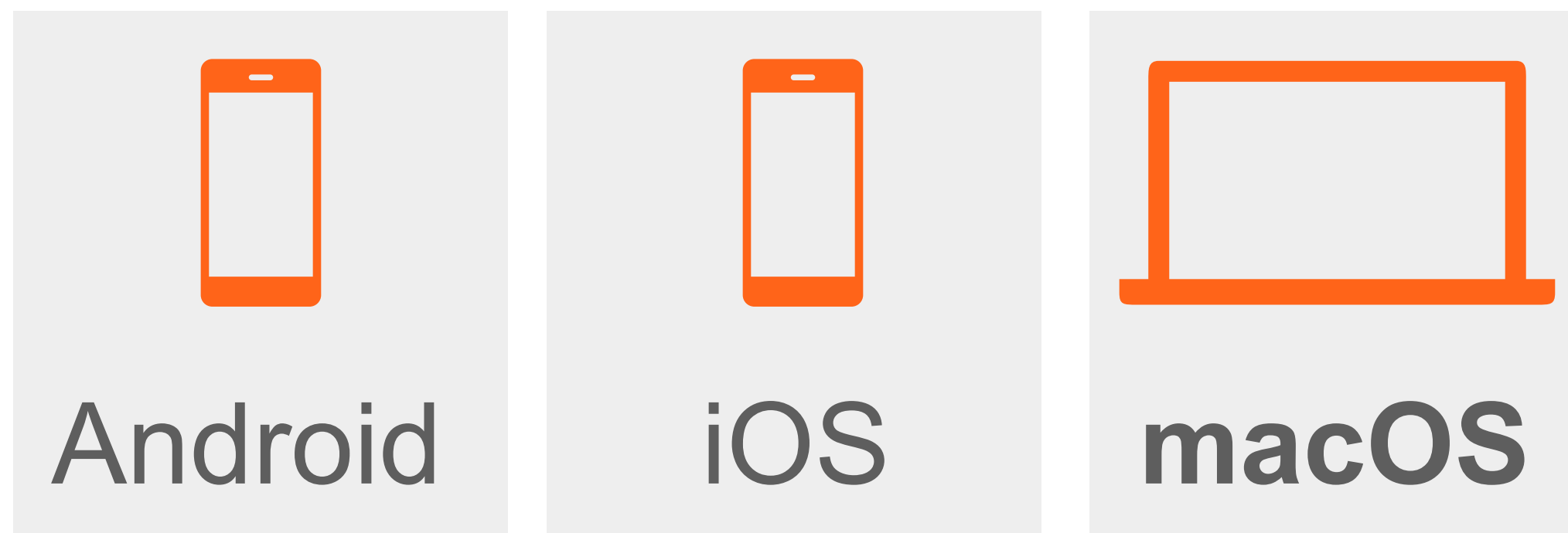
DHCP server je nakonfigurován, aby preferoval provoz IPv6-only



# Je volba DHCP 108 podporována?



Ano! Volbu 108 podporují současné verze:



Zařízení jsou **připravena**, sítě zatím ne.

# Zvláštní případ macOS?



- Umožňuje běh *libovolných* aplikací, včetně těch používajících zastaralá IPv4-only API
- Ukázalo se, že macOS má také **CLAT!**
  - Na macOS 12 je aktivován volbou 108 v DHCP zároveň s volbou RA PREF64
  - Od macOS 13, je aktivován bez zvláštních požadavků
  - Zároveň byla odstraněna podpora sítí čistě IPv6-only (bez NAT64)

```
~ ifconfig en0
en0: flags=8963<UP,BROADCAST,SMART,RUNNING,PROMISC,SIMPLEX,MULTICAST> mtu 1500
options=6463<RXCSUM, TXCSUM,TSO4,TSO6,CHANNEL_IO,PARTIAL_CSUM,ZEROINVERT_CSUM>
ether f0:18:98:31:36:c6
inet6 fe80::1477:9fe8:a21d:56a6%en0 prefixlen 64 secured scopeid 0x6
inet6 2a02:::80:c48:6e99:5e6c:e453 prefixlen 64 autoconf secured
inet6 2a02:::80:392d:6ea9:e5fd:ddd1 prefixlen 64 autoconf temporary
inet6 fdba:91fa:4142:80:813:d49b:cca9:9b87 prefixlen 64 autoconf secured
inet 192.0.0.1 netmask 0xffffffff broadcast 192.0.0.1
inet6 fdba:91fa:4142:80:fa:bf88:9a02:cbb1 prefixlen 64 clat46
nat64 prefix 64:ff9b:: prefixlen 96
nd6 options=201<PERFORMNUD,DAD>
media: autoselect
status: active

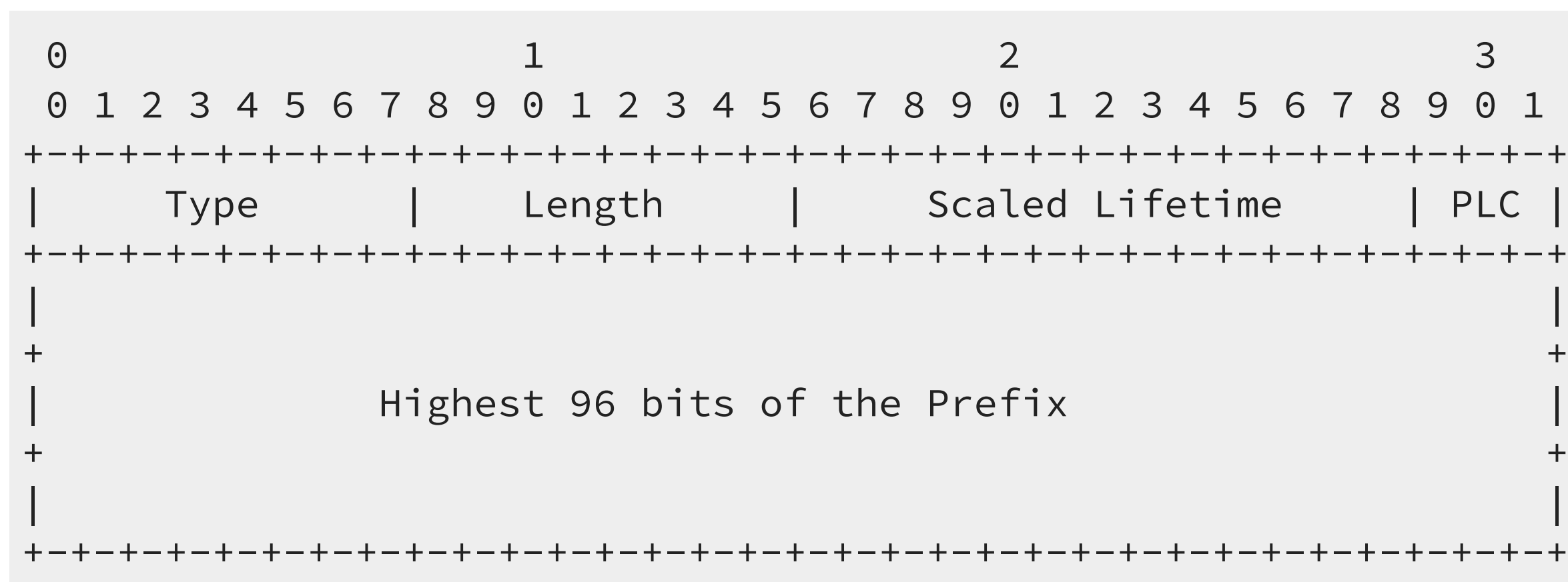
~ ping -c 5 1.1.1.1
PING 1.1.1.1 (1.1.1.1): 56 data bytes
64 bytes from 1.1.1.1: icmp_seq=0 ttl=56 time=5.045 ms
64 bytes from 1.1.1.1: icmp_seq=1 ttl=56 time=10.375 ms
64 bytes from 1.1.1.1: icmp_seq=2 ttl=56 time=11.156 ms
64 bytes from 1.1.1.1: icmp_seq=3 ttl=56 time=10.977 ms
64 bytes from 1.1.1.1: icmp_seq=4 ttl=56 time=10.280 ms

--- 1.1.1.1 ping statistics ---
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 5.045/9.567/11.156/2.286 ms
```

# Volba RA PREF64



- Volba ohlášení směrovače **obsahující prefix NAT64**
- Potřebná pro **konfiguraci CLATu**, lokální DNS64 nebo Happy Eyeballs 2.0 (*IPv4 literály*)
- **Sdílí osud** s dalšími konfiguračními volbami
  - je o něco důvěryhodnější než DNS64
- Podporováno aktuálními verzemi Android, iOS a macOS







# Provozování sítí IPv6-mostly

# Volba DHCP číslo 108 je snadná



- **Nativní podpora** v DHCP serveru Kea
- Většina DHCP serverů podporuje **uživatelské volby**
  - například: `dnsmasq -0 108,0:0:1:2c`
  - hodnota volby představuje čas, jak dlouho má být IPv4 deaktivováno
- **Speciální zpracování** volby 108 na straně serveru není nutné
- Server ale **musí mít volné adresy** v poolu
  - Jinak na dotaz klienta neodpoví

# Volba RA PREF64 je těžší



- Směrovače **nepodporují vlastní volby RA**
  - Stejný problém jsme už měli s adresami DNS serverů, teď je tu znovu
- Nějaká podpora se pomalu objevuje:
  - radvd (v nevydané verzi)
  - FRR (čekající pull request)
  - odhcpd (čekající pull request)
  - rad (součást OpenBSD)
  - MikroTik RouterOS v7.8 beta2



# Překvapení na macOS



Pokud je v síti víc prefixů, CLAT si náhodně jeden vybere, nerespektuje přitom omezený dosah (ULA) ani preferenci adres.

```
→ ~ ifconfig en0
en0: flags=8963<UP,BROADCAST,SMART,RUNNING,PROMISC,SIMPLEX,MULTICAST> mtu 1500
    options=6463<RXCSUM,TXCSUM,TS04,TS06,CHANNEL_IO,PARTIAL_CSUM,ZEROINVERT_CSUM>
    ether f0:18:98:31:36:c6
    inet6 fe80::1477:9fe8:a21d:56a6%en0 prefixlen 64 secured scopeid 0x6
    inet6 2a02:::80:c48:6e99:5e6c:e453 prefixlen 64 autoconf secured
    inet6 2a02:::80:392d:6ea9:e5fd:ddd1 prefixlen 64 autoconf temporary
    inet6 fd8a:91fa:4142:80:813:d49b:cca9:9b87 prefixlen 64 autoconf secured
    inet 192.0.0.1 netmask 0xffffffff broadcast 192.0.0.1
    inet6 fd8a:91fa:4142:80:fa:bf88:9a02:cbb1 prefixlen 64 clat46
    nat64 prefix 64:ff9b:: prefixlen 96
    nd6 options=201<PERFORMNUD,DAD>
    media: autoselect
    status: active
```



# Překvapení na macOS



Pokud si uživatel nastaví **vlastní IPv4 adresy DNS resolverů**, DNS nefunguje, i když utility jako `host` fungují normálně.

```
→ ~ scutil --dns | head
DNS configuration

resolver #1
  search domain[0] : mtg.ripe.net
  nameserver[0]   : 1.1.1.1
  flags           : Request A records, Request AAAA records
  reach           : 0x00000002 (Reachable)

resolver #2
  domain          : local
→ ~ host google.com
google.com has address 172.217.168.238
google.com has IPv6 address 2a00:1450:400e:811::200e
google.com mail is handled by 10 smtp.google.com.
→ ~ ping google.com
ping: cannot resolve google.com: Unknown host
→ ~
```



**Shrnutí**



# Pro

- **Pouze jedna síť**
- **IPv4 adresy nejsou plýtvány na nová zařízení**
  - Hodí se, když nepoužíváte NAT
- **Použití IPv4 bude minimální i pro stará zařízení**
  - DNS64 způsobí, že většina provozu půjde přes NAT64 namísto nativního IPv4

# Proti

- **Nejkomplikovanější síť**
- **IPv4 je stále potřeba**
- **Je potřeba NAT64**
- **Problematická interoperabilita mezi dual-stack a IPv6-only zařízeními**
  - *Nelze nastavit Chromecast z telefonu s Androidem*

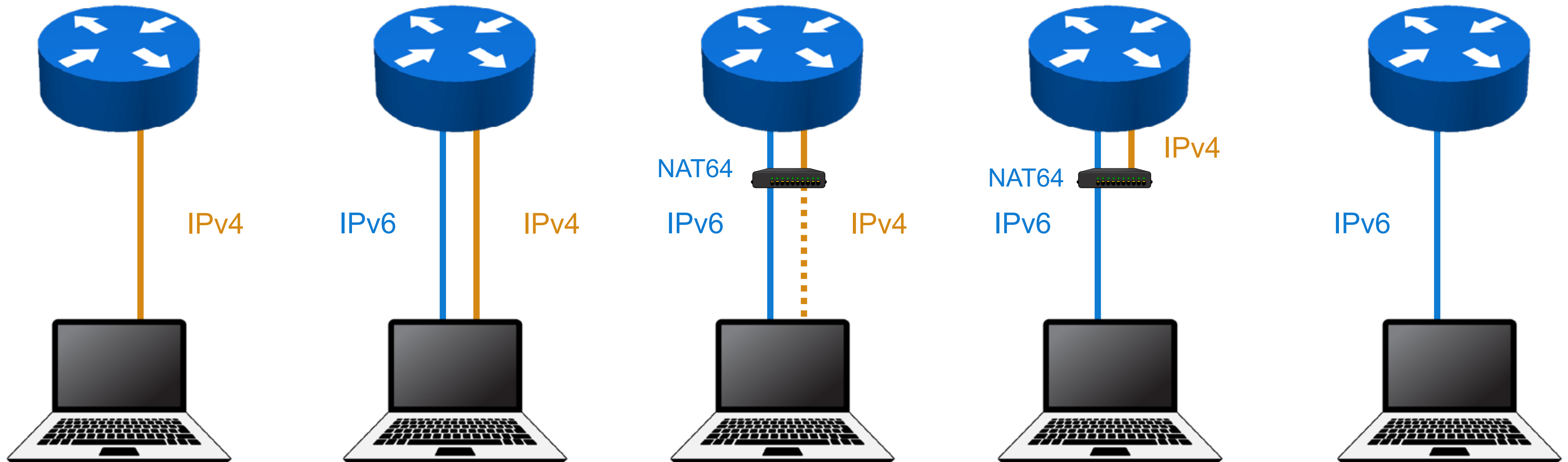




# Kdy zvážit IPv6-mostly

- **Nepoužíváte NAT** a pool DHCP se plní
- Používáte NAT ale **dochází privátní IPv4 adresy**
- V síti jsou převážně **mobilní** nebo **zařízení od Apple**
- Už máte **NAT64** a radi byste začali s **vypínáním IPv4**

# Fáze přechodu na IPv6







# Otázky

Ondrej.Caletka@ripe.net  
@ripencc

**IPv6-mostly v akci:**

RIPE 86, 22.-26. 5. Rotterdam

IPv6 Day, 6. 6. Praha