RIPE NCC
RIPE NETWORK COORDINATION CENTRE

RIPE Meeting Network behind the scenes

How we run a conference network for a networking conference

Ondřej Caletka | 2 November 2024 | OpenAlt 2024

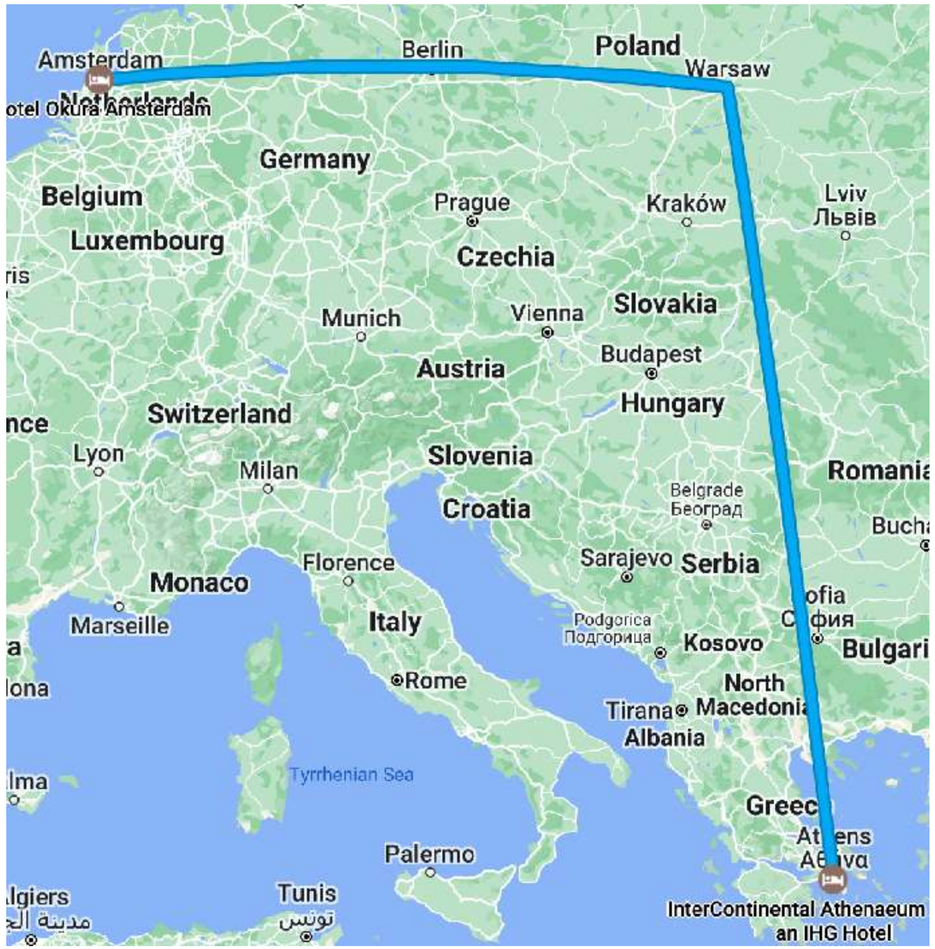
What is a RIPE meeting

- A week long event twice a year along RIPE NCC service region
- 600+ attendees from all over the world
- **Previously:** RIPE 89 Prague, 28 October - 1 November 2024
- **Next up:** RIPE 90 Lisbon, 12 - 16 May 2025

- **Custom temporary Wi-Fi network during the event**
  - **AS2121**
  - **193.0.24.0/21**
  - **2001:67c:64::/48**

Issues with Geolocation

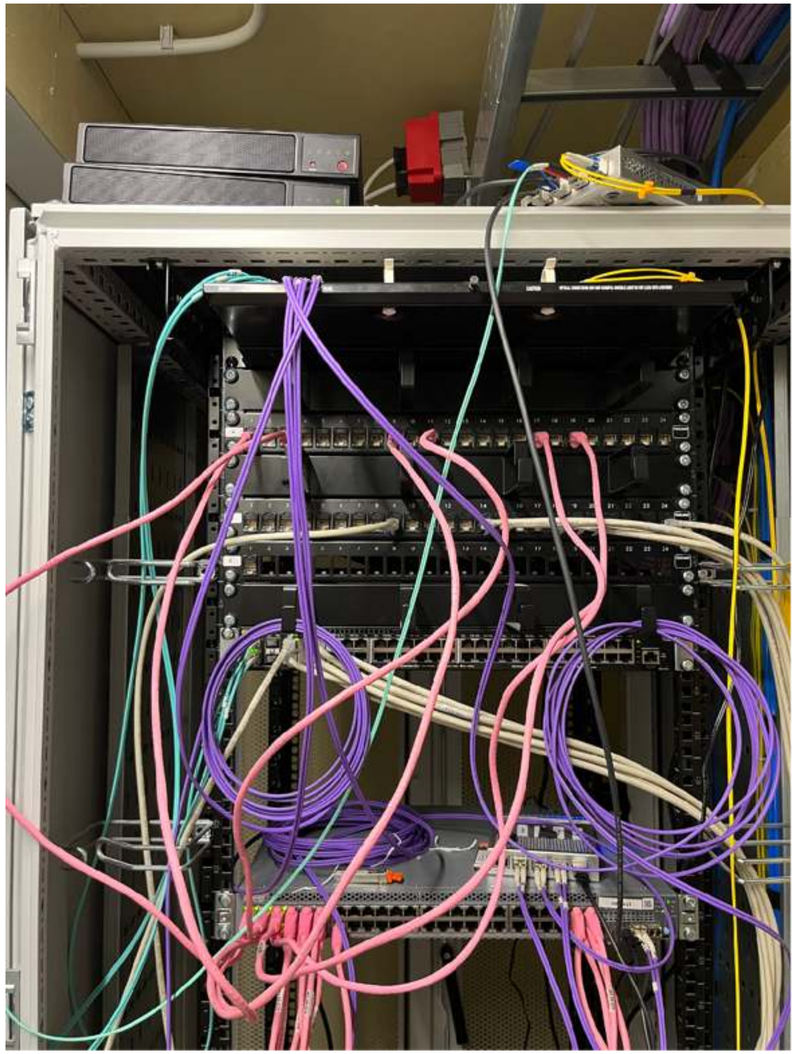
- BSSID-based geolocation
  - based on assumption that APs don’t move
  - issues with Google **disappeared around 2016**
- IP-based geolocation
  - Privately curated lists by many commercial parties
  - We publish a CSV list for Google
  - This is now **standardised** as **RFC 8805**
  - Many providers support it but you still **have to tell them**; RFC 9092 discovery is not very popular
  - We have a list of 7 geolocation providers to check prior every meeting; **two still need manual updates**



Amsterdam - Warsaw - Athens in one hour, 2014, colored

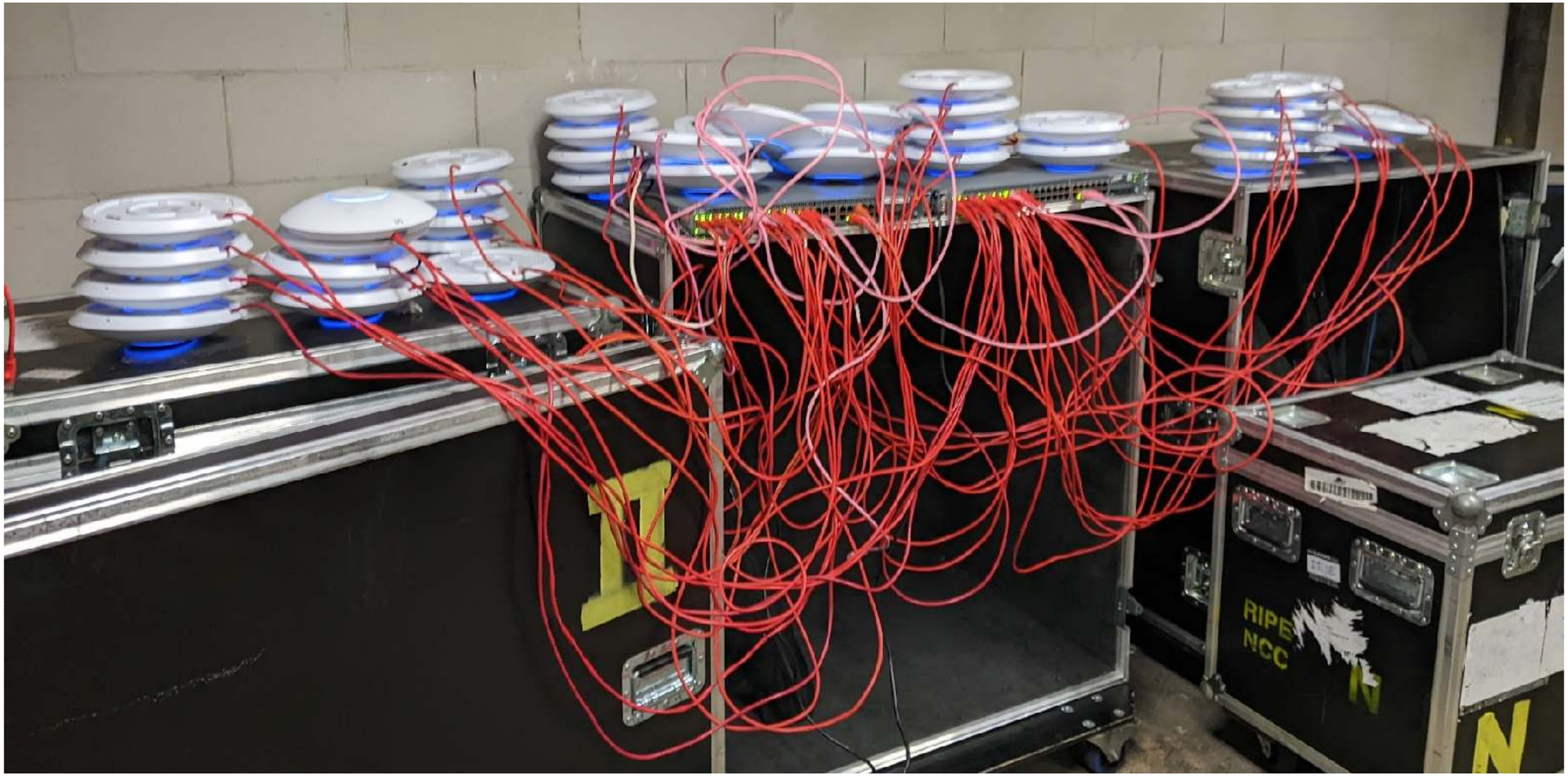
Physical network

- Two VM hosts running VMware vSphere 8.0
  - SuperMicro SuperServer E300-9D-8CN8TP
  - 25 VMs including routers, firewalls, DHCP servers, DNS resolvers, Wi-Fi controller
- Switches
  - Juniper EX2300 (48×GE PoE+ + 4×10GE SFP+)
  - Zyxel GS-1900-10HP (8×GE PoE+, 2×SFP, VLAN)
  - MikroTik CRS305-1G-4S+IN (4x10GE)
- Access Points
  - Unifi UAP AC (S)HD

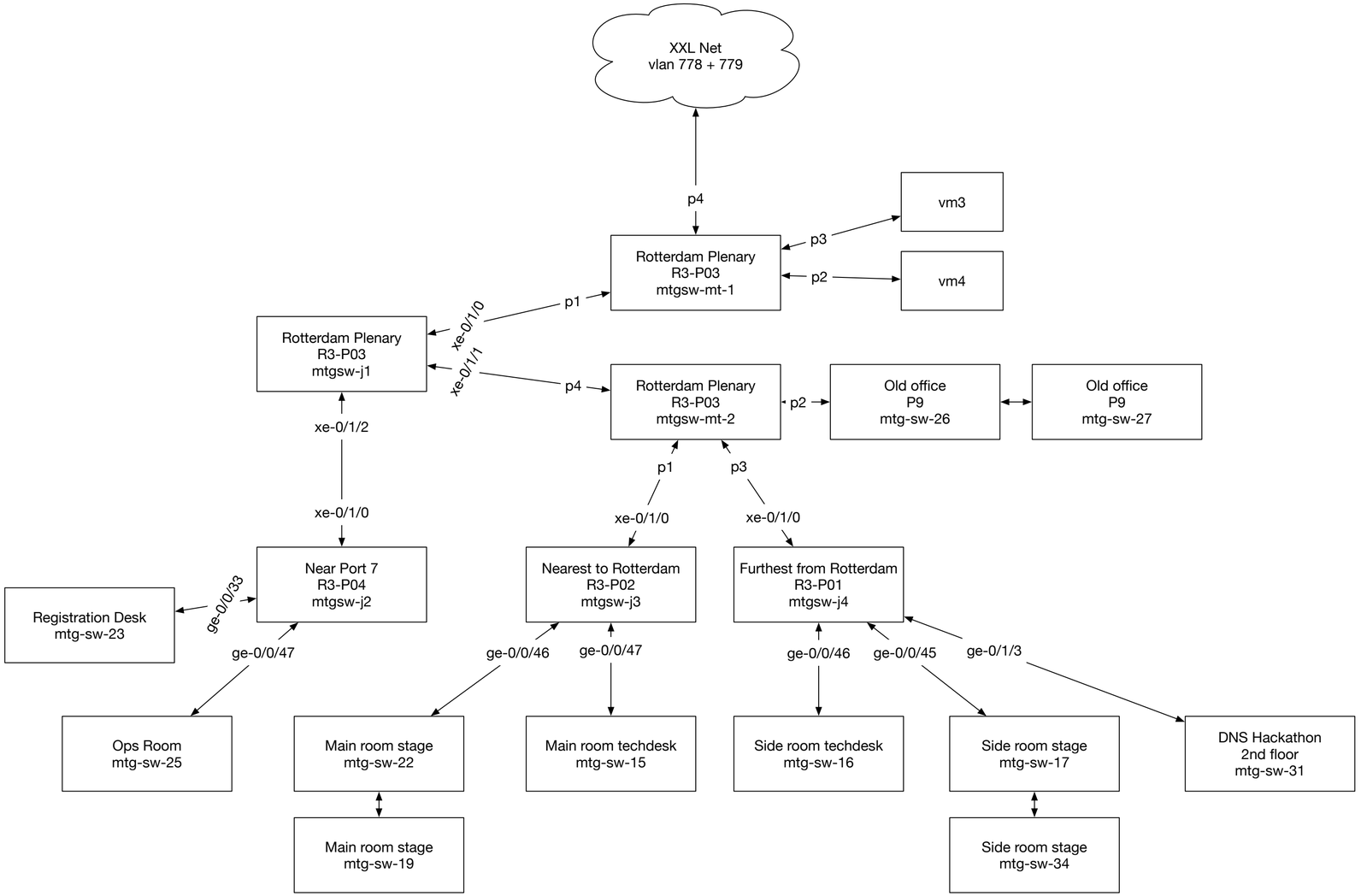


We bring a lot

Testing after covid break



Physical topology (RIPE 86)



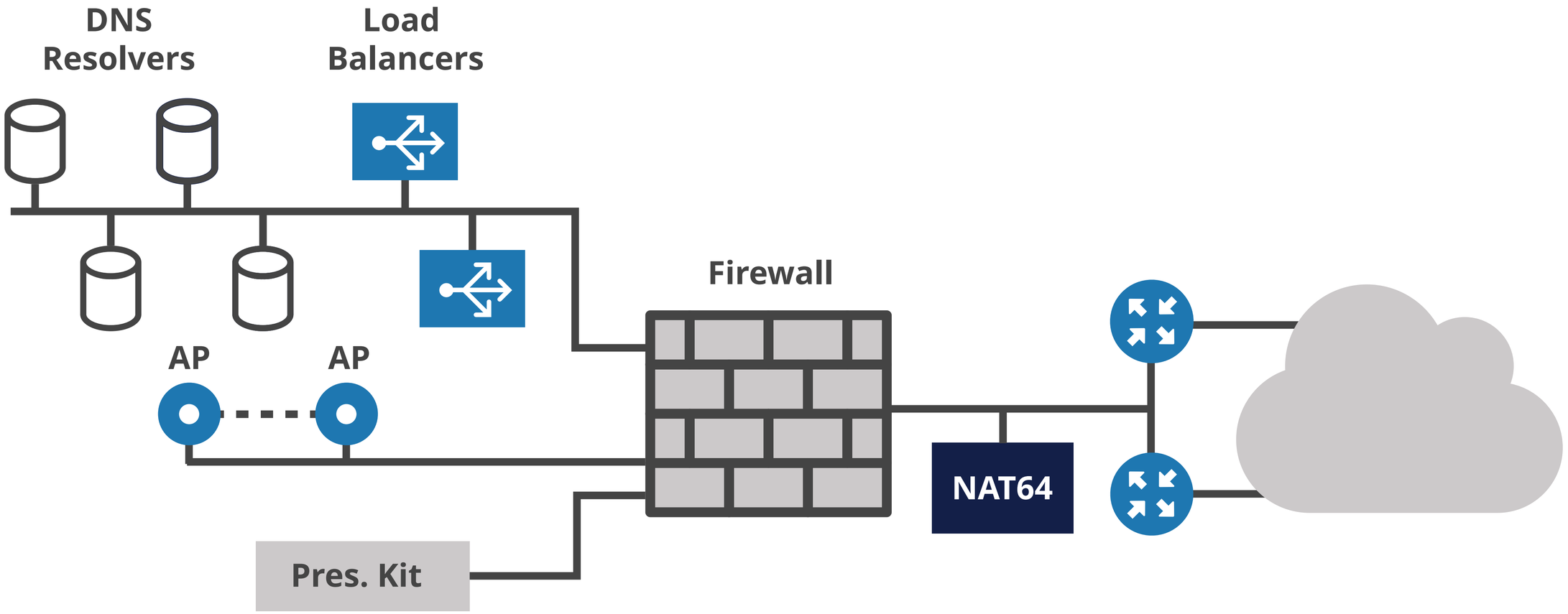
The whole network runs Open Source

- Edge routers running **BIRD**
- Firewall using **nftables**
- DNS resolver cluster of **Knot Resolver / BIND9**
- DNS load balancer running **keepalived**
- DHCP servers running **Kea**
- NAT64 using **Jool**
- Statistics collected using **collectd** + **InfluxDB** + **Grafana**
- Deployed using **Ansible**

Networks provided

- **Public** network
  - IPv6-mostly dual-stack
- **IPv6-only** network
  - then: NAT64+DNS64
  - now: pure IPv6-only
- **Legacy** network
  - dual-stack without IPv6-mostly signalling
- **Private** network
  - management interfaces
- **Service** network
  - A/V equipment and stenography
- Wi-Fi management
  - for APs and Zyxel switches
- **Meetecho** network
  - for video streaming supplier

Logical network topology



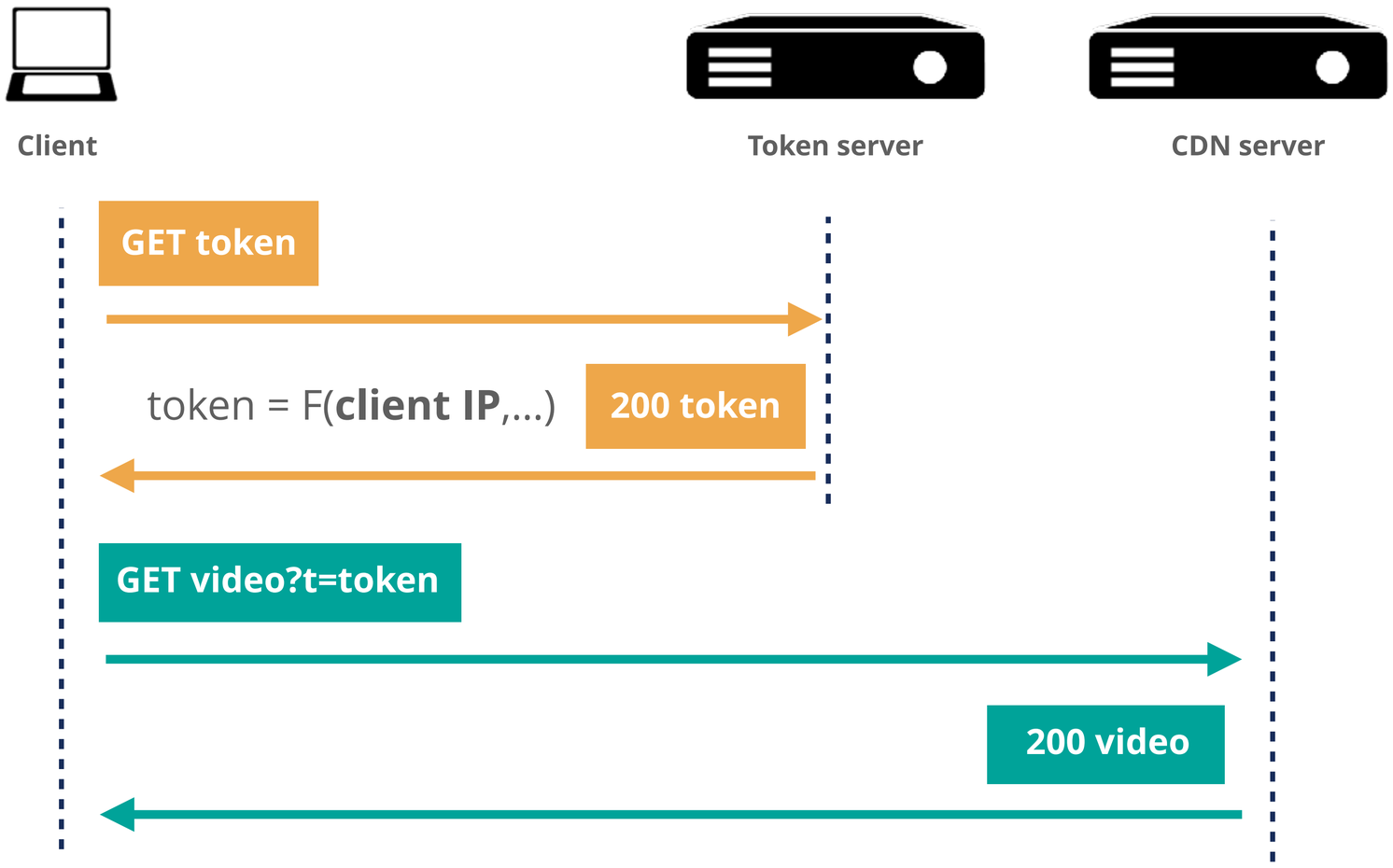
Routers on Oracle Linux 9

- Two OL9 VMs running BGP with the connectivity provider
- Receive **full BGP feed**, do RPKI **Route Origin Validation**
- Offer default route via OSPF
- **Problem:** high CPU load due to full routing table
  - not a **planned use case** for NetworkManager
  - worked around by **stopping NetworkManager after boot** :)
  - reported to RedHat by a paying customer
  - fixed in RHEL 9.4z

NAT64

- **Jool** on Debian Linux
- **pool6** = 64:ff9b::/96, **pool4** = 193.0.30.0/24
- BIRD offering **pool6** and **pool4** prefixes via OSPF
- **Problem:** some Video On Demand platforms (iVysilani.cz, NOS.nl,…) **fail to play over NAT64**
  - browser console shows **HTTP 403** error when accessing the CDN
  - so this is apparently an **application layer issue**

How some VOD platforms work



How does Jool allocate IPv4 addresses

- Address and port tuple is determined by **hashing some parts** of the IPv6 packet
  - by default: source IPv6, **destination IPv6**, **destination port**
  - hash collisions are resolved by a (slow) iterative process
- Global option **f-args** influence what is hashed
  - setting it to 8 (source IPv6 only) **resolves the issue with the VOD platforms**
  - but all sessions made by one host are causing collisions
- There's a branch of Jool with *Ondřej Caletka's hashing algorithm*
  - uses two hashes, one for choosing IPv4 address, other to choose port
  - no measurement data to prove it is indeed better, not merged

Firewall

- Oracle Linux 9 acting as a default gateway
- BIRD for internal OSPF
- **radvd** from *Git master* to support **PREF64** option for RAs
- **Problems:**
  - Slow throughput due to **extensive firewall logging to the console**
  - IPv6 default gateway **disappears after 6 seconds**, only on macOS
  - Lots of **ARP noise** in public segments

IPv6 neighbor advertisement RFC 4861

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |     Code      |          Checksum             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|R|S|O|                     Reserved                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                                                               |
+                       Target Address                          +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Options ...
+-+-+-+-+-+-+-+-+-+-+-+-
```

R - Router flag. When set, the R-bit indicates that the sender is a router. The R-bit is used by Neighbor Unreachability Detection to detect a **router that changes to a host**.

What makes Linux (un-)set R-flag?

- Turned out this to be the *(only)* feature of **per-interface forwarding sysctl** switch
- IPv6 forwarding is just a **global switch** on Linux
  - yet there are still per interface switches
- NetworkManager used to reset per-interface switch during interface setup
  - **fixed in version 1.44.0**

```
net.ipv6.conf.all.forwarding = 1
net.ipv6.conf.default.forwarding = 1
net.ipv6.conf.lo.forwarding = 1
net.ipv6.conf.eth0.forwarding = 0
net.ipv6.conf.eth1.forwarding = 0
```

Dealing with ARP noise

- Caused by omnipresent Internet-wide scans
- Kernel-space ARP implementation has **no negative cache**
- **arpd** to rescue!

```
# arpd -k -a2 eth0 eth1 eth2 eth3
```

  - part of iproute2
  - implements ARP in userspace
  - has **negative cache**
- **30 times less ARP** messages on an empty network
  - before: **250 pps**, 84 kbps
  - after: **8 pps**, 2.7 kbps

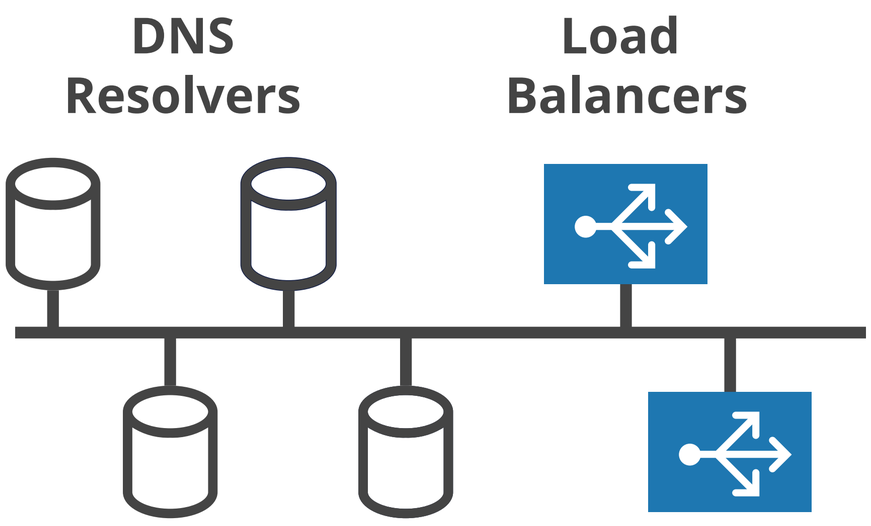
DHCP servers

- Two servers running Kea in **hot-standby** HA mode
- Directly connected to each VLAN, no relay agents
- Stateful DHCPv6 supported but not announced in RAs
- Avoiding addresses **ending with .0 or .255**

```
"pools": [
     { "pool": "193.0.24.32-193.0.24.254" },
     { "pool": "193.0.25.1-193.0.25.254" },
     { "pool": "193.0.26.1-193.0.26.254" },
     { "pool": "193.0.27.1-193.0.27.254" }
],
```

DNS Resolver Cluster

- Four **worker nodes**, two **load balancers**
- Service address **outside the subnet** on dummy interfaces
- Keepalived in **Direct Routing** mode
  - incoming traffic is bounced to a worker node
  - outgoing traffic goes directly
- Load balancers announce service addresses using OSPF
- Support for **DNS64**, **DoT**, **DoH**, **DDR**



DDR

Deutsche Demokratische Republik

Dance Dance Revolution

Double Data Rate

Discovery of Designated Resolvers (RFC 9462)

Discovery of Designated Resolvers

- A mechanism for DNS clients to use DNS records to discover a resolver's **encrypted DNS configuration**
- Supported by Windows, macOS and iOS
- Resolver queries special name `_dns.resolver.arpa IN SVCB`
- Gets list of encrypted DNS options
- Validates that **TLS certificate contains the IP address of the resolver advertised by RA or DHCP**

Getting TLS certificate for IP address

- **Not supported** (yet) by **Let's Encrypt**
  - RFC 8738 extends ACME protocol to support IP address validation
  - CA/B Forum Baseline Requirements **allow such validation**
- **Let's resort to *traditional* CAs for now**

Digicert: $ 1057 per year

- Probably would work
- The price is just insane

Order summary

Basic OV

1-year plan

Price details

Base price Primary URL x 1 year	$289.00 USD
Additional standard URLs 4 standard URLs x 1 year	$768.00 USD
Subtotal	$1,057.00 USD
Tax	$0.00 USD
Total	**$1,057.00 USD**

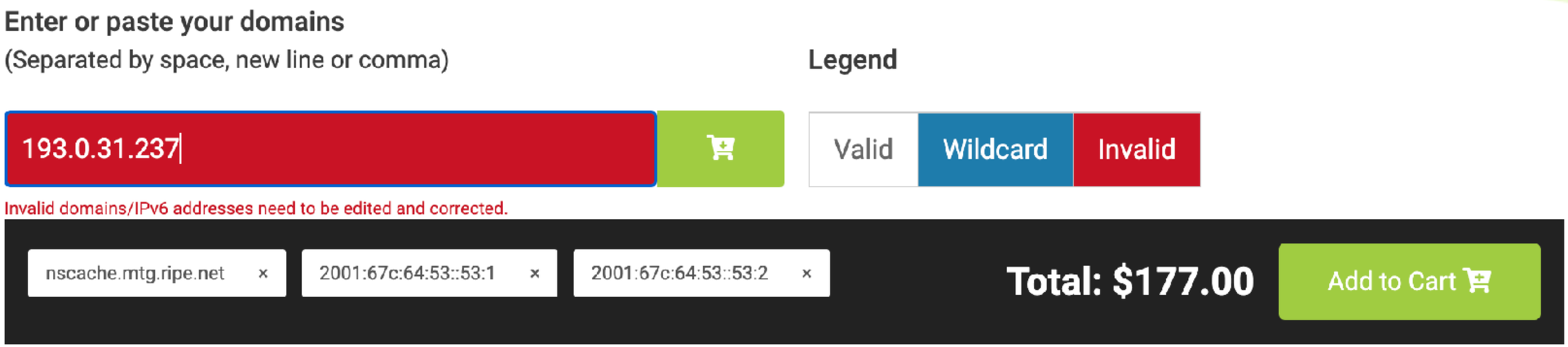
Sectigo: $ 149 per year

- Error -1 when trying to add IPv6 SAN
- Tech support:
  ”I understand that you want to add this to the SAN but unfortunately, if you add this the **order will not be processed further**. I would hence request you to kindly proceed further with the order **without adding IPV6** to the SAN.“

Additional Domains : 0	Quantity : 1
Total Domains:	5
Warranty:	$250,000
Re-issue:	Unlimited
Your Savings:	$300.00
Total:	**$149.00**

SSL.com: $ 177 per year

- IPv6 is supported
- IPv4 is causing troubles



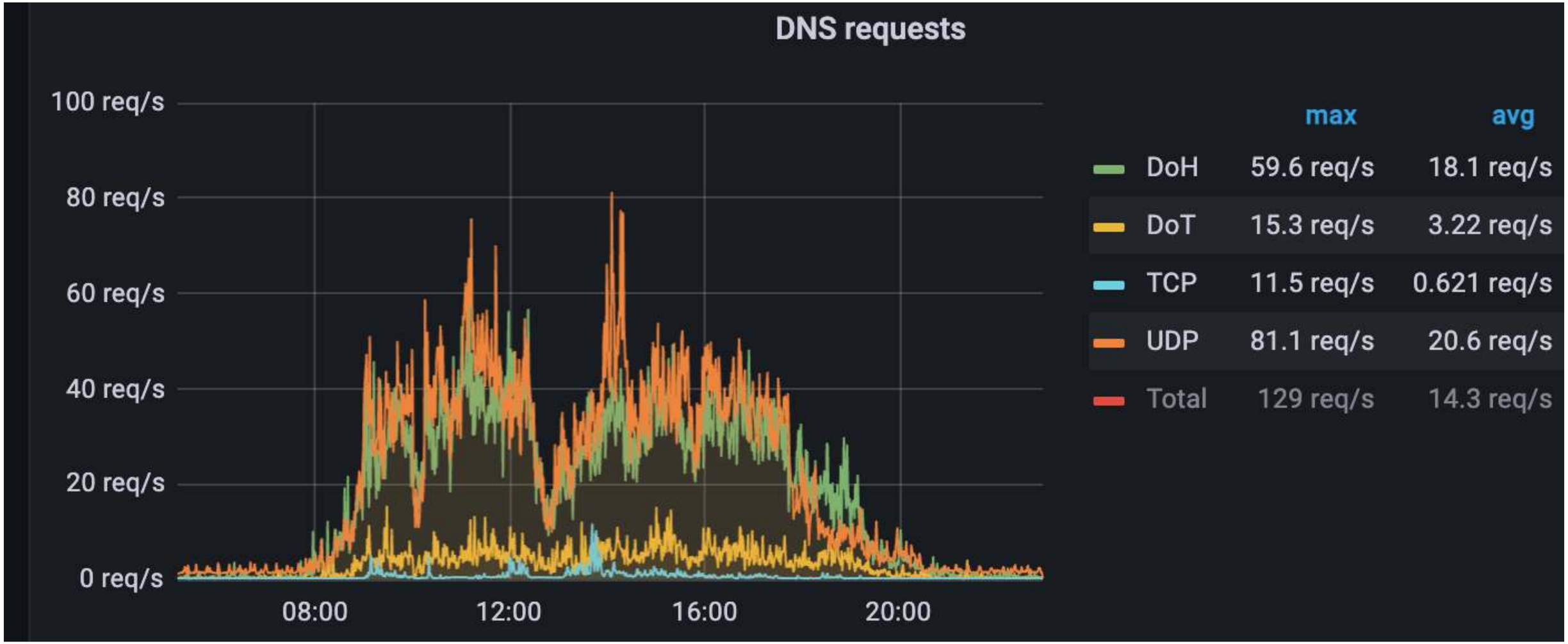
SSL.com: success at last

- We reduced number of IPv4 and IPv6 addresses to one each for cost reasons
- They are both high available anyway

```
X509v3 Subject Alternative Name:
    DNS:nscache.mtg.ripe.net
    IP Address:193.0.31.237
    IP Address:2001:67C:64:53:0:0:53:1
```

https://crt.sh/?id=10495720523

Increased DoH usage



Unexpected issue with DDR

- Only **github.com** and **duckduckgo.com**
- Only in **Safari** and on **iOS**
- Only in **IPv6-only network**
- Only **intermittently**
- Only with **our DNS resolvers**
- Only with **DDR triggered DoH**

A race condition in Knot DNS64 module

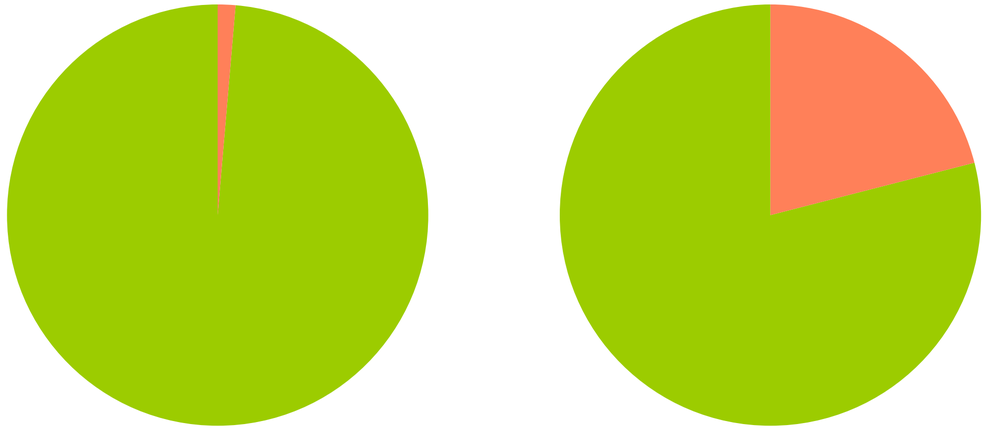
- When:
  - queried name is an apex name with A but no AAAA record
  - dns64 module is loaded
  - queried rrset nor the nsset of the zone is in cache
  - client is ~~using doh2 and~~ asking concurrently for A and AAAA record
- Then Knot Resolver sometimes returns referral in place of an answer
- Reported as Knot Resolver bug #905

Looking for a stand-in resolver

- Required features:
  - DNSSEC validation
  - DNS64 only for particular client subnets
  - DoT (for Android) and DoH (for macOS and Windows)
- The winner is… BIND 9.18

Wi-Fi

- Unifi controller running on Debian Linux
- **Manual channel configuration**, mostly **5 GHz only**
- Legacy eSSID names **changing every meeting**
- *Multicast and Broadcast control* **kills IPv6 NDP**
  - you have to **white-list MAC addresses** of all wired IPv6 hosts
  - unintended **RA-guard-like function**



RIPE 89 Wi-Fi stats

Future

- Redundant firewall with VRRP
  - without or with state synchronisation
- Redundant NAT64
  - probably without state synchronisation
- Wi-Fi hardware upgrade
  - so we can utilise the 10 GBE backbone

Questions ?

Ondrej@Caletka.cz
https://ondřej.caletka.cz
@oskar456@mastodon.social