2024 APRICOT

APNIC 57

BANGKOK, THAILAND

21 February – 1 March 2024

#apricot2024

IPv6-only Network Report

Or: how to turn "IPv6-mostly" into "IPv6-only"

Brian Candler

NSRC
Network Startup Resource Center

APRICOT 2024
APNIC 57

SSID "apricot-v6only"

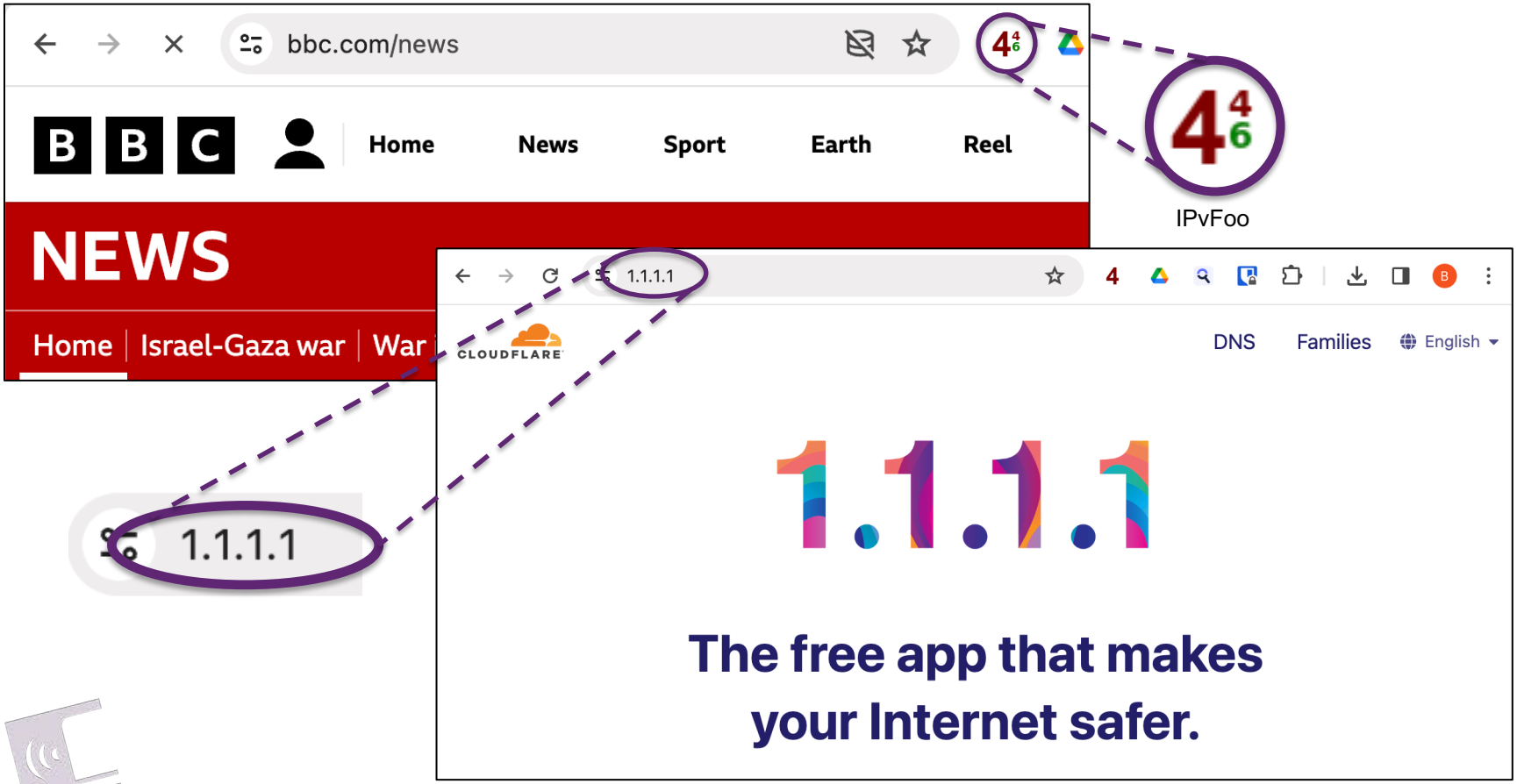
- How we built it
- What's different from previous conference v6only networks
- Issues found
- Usage stats

Brief detour: "IPv6-mostly"

- Relatively new mechanism for the graceful sunset of IPv4 in dual-stack networks
  - RFC 8925: DHCPv4 "IPv6-Only Preferred" (option 108)
  - RFC 8781: PREF64 in Router Advertisements
- If both features are present, the client declines an IPv4 address and enables an embedded CLAT (NAT46)
- Magic happens…

```
% ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8): 56 data bytes
64 bytes from 8.8.8.8: icmp_seq=0 ttl=57 time=34.383 ms
64 bytes from 8.8.8.8: icmp_seq=1 ttl=57 time=33.917 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=57 time=38.316 ms
```

```
09:00:48.499160 IP6 2001:df9:0:3:c88:a341:b8c:8ade > fd64::808:808:
ICMP6, echo request, id 12084, seq 0, length 64
09:00:48.533271 IP6 fd64::808:808 > 2001:df9:0:3:c88:a341:b8c:8ade:
ICMP6, echo reply, id 12084, seq 0, length 64
```


bbc.com/news
BBC
Home
News
Sport
Earth
Reel
NEWS
Home | Israel-Gaza war | War
IPvFoo
1.1.1.1
CLOUDFLARE
DNS
Families
English
1.1.1.1
The free app that makes your Internet safer.

Client view (macOS)

```
% ifconfig en0
en0: flags=88e3<UP,BROADCAST,SMART,RUNNING,NOARP,SIMPLEX,MULTICAST> mtu 1500
        options=6460<TSO4,TSO6,CHANNEL_IO,PARTIAL_CSUM,ZEROINVERT_CSUM>
        ether 60:3e:5f:81:98:c4
        inet6 fe80::14a5:e833:7d65:c178%en0 prefixlen 64 secured scopeid 0xe
        inet6 2001:df9:0:3:1424:5189:5756:bcaa prefixlen 64 autoconf secured
        inet6 2001:df9:0:3:7407:fd14:7b93:124b prefixlen 64 autoconf temporary
        inet 192.0.0.2 netmask 0xffffffff broadcast 192.0.0.2
        inet6 2001:df9:0:3:c88:a341:b8c:8ade prefixlen 64 clat46
        nat64 prefix fd64:: prefixlen 96
        nd6 options=201<PERFORMNUD,DAD>
```

This IPv4 address is entirely internal and does not leave the machine – *or so I thought!*

Building a better IPv6-only network

- by enabling the IPv6-mostly functionality

1. DHCPv4 server for IPv6-only

- Needed a DHCP server that responds to clients that request option 108, but *does not respond* to others
- ISC and KEA will always offer an IPv4 address from a pool
- Deployed a custom DHCP server in Go (coredhcp) and patched it to implement the right behavior
- Patches have now been merged upstream

https://github.com/coredhcp/coredhcp/pull/170
https://github.com/insomniacslk/dhcp/pull/524

```
server4:
    listen:
        - "%enp6s0"
        - ":10067"
    plugins:
        - server_id: 10.12.65.1
        - ipv6only: 24h
        - autoconfigure:

# Optionally act as stateless DHCPv6 server too
server6:
    listen:
        - "[ff02::1:2%enp6s0]"
        - "[ff05::1:3%enp6s0]"
        - ":10547"
    plugins:
        - server_id: LL 00:16:3e:a2:64:a4
        - dns: 2405:3340:e000::77:77 2001:df9:0:1::2
        - searchdomains: apricot.bknix.net
```

2. PREF64 in RAs

- Many router vendors only have it in very new firmware
  - e.g. Mikrotik added in RouterOS 7.8 (but no NAT64)
- Linux **radvd** does it, but not in any released version
  - Needed to compile from source
- You can't generate RAs on behalf of another router, so the Linux VM had to become the default gateway

```
interface enp6s0
{
    AdvSendAdvert on;

    MinRtrAdvInterval 240;
    MaxRtrAdvInterval 720;

    AdvManagedFlag off;
    # Optional: use stateless DHCPv6 as well
    AdvOtherConfigFlag on;
    AdvHomeAgentFlag off;

    prefix 2001:df9:0:3::/64
    {
        AdvOnLink on;
        AdvAutonomous on;
        AdvRouterAddr off;
    };
    nat64prefix fd64::/96 {
        AdvValidLifetime 1800;
    };

    RDNSS 2405:3340:e000::77:77 2001:df9:0:1::2
    {
        AdvRDNSSLifetime 1800;
    };

    DNSSL apricot.bknix.net
    {
        AdvDNSSLLifetime 1800;
    };
};
```

3. NAT64 (PLAT)

- Since the VM has to forward all the IPv6 traffic anyway, I decided to let it do the NAT64 as well
- Linux kernel module: "apt install jool-dkms jool-tools"
  - *or so I thought!*
- By default uses the single outside IPv4

```
modprobe jool
jool instance add --netfilter --pool6 fd64::/96
```

It works! Mostly…

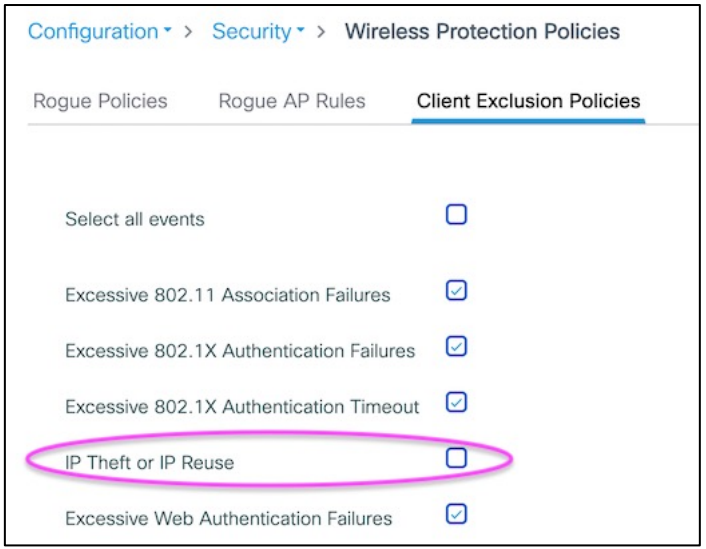
- Problem 1: multicast packets on wire with 192.0.0.2 source

```
16:25:42.710644 IP 192.0.0.2.56483 > 239.255.255.250.1900: UDP, length 176
16:25:43.715587 IP 192.0.0.2.56483 > 239.255.255.250.1900: UDP, length 176

17:06:57.893135 IP 192.0.0.2 > 239.255.255.250: igmp v2 report 239.255.255.250
17:07:07.952649 IP 192.0.0.2 > 224.0.0.2: igmp leave 239.255.255.250

22:34:43.010782 IP 192.0.0.2.5353 > 224.0.0.251.5353: 0 [7a] [24q] [1au] PTR
(QM)? lb._dns-sd._udp.local. PTR (QM)? _airport._tcp.local. PTR (QM)? …
```

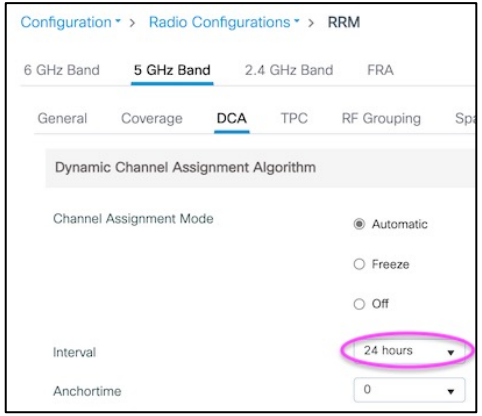
Cisco WLC security feature

```
Feb 25 10:16:56.836: %CLIENT_ORCH_LOG-5-ADD_TO_EXCLUSIONLIST_REASON: Chassis 1
R0/0: wncd: Client MAC: 3c22.fb13.c1cc with IP: 192.0.0.2 was added to
exclusion list, legit Client MAC: 603e.5f81.98c4, IP: 192.0.0.2, reason: IP
address theft
```

Solution:



More problems



- Randomly or regularly kicked off wireless
  - Increased Dynamic Channel Assignment interval from 10 mins to 24 hours. Better?? *
- Occasional fail to connect over UDP (SoftEther, Wireguard)
  - Resolved itself. Feels like NAT exhaustion? See next page
- An iOS banking app did not generate confirmation page
- But we didn't provide a channel for users to report issues

* Disable "IPv4 DHCP Required" option, at [Configuration > Tags & Profiles > Policy] > [SSID] - Advanced -> DHCP"

NDP expiry!

- NDP neighbors timed out for client's CLAT IPv6 address
- Version of jool in Ubuntu 22.04 repo is 4.1.7
- Fix? Install the latest (4.1.11) deb packages from github

v4.1.10

ydahhrk released this Jun 12, 2023 · 22 commits to main since this release v4.1.10

Improvements since 4.1.9:

- #382, #400: Clean up `skb->tstamp` during translation to prevent dropped packets.

Minor reproducible niggles

- traceroute shows only "*" for every hop
- macOS ssh client with -4 & hostname

```
% ssh -4 nsrc.org
ssh: connect to host nsrc.org port 22: Undefined error: 0
```

And yet:

```
% ssh 128.223.157.25
… works

% /opt/homebrew/bin/ssh -4 nsrc.org
… works
```


DHCP log data (to Friday 1.30pm)

- 142 unique MAC addresses seen in total *(64 today)*
- 115 (81%) of these requested option 108 *(60 today)*
- Of the remaining 27, *none* of them offered option 116 (disable stateless AutoConfigure: RFC 2563)
- These 27 requested DHCPv4 repeatedly
  - Median 53 times
  - One device tried 49,482 times over 2 days

How compatible is this?

- macOS, iOS, Android: all good
- Windows: no (only on mobile broadband cards)
- Linux: no (install & configure **clatd** by hand??)
- In an IPv6-*mostly* network, this is not an issue
- Wider compatibility if you use DNS64? But this fakes AAAA addresses even for option-108 supporting clients

More information

- https://labs.ripe.net/author/ondrej_caletka_1/deploying-ipv6-mostly-access-networks/ -- *Ondřej Caletka*
- https://ripe87.ripe.net/archives/video/1160/ -- *Jen Linkova*
- Monitor your own network to see how many devices include option 108 in their Parameter Request List

```
DHCP-Message (53), length 1: Discover
Parameter-Request (55), length 12:
    Subnet-Mask (1), Classless-Static-Route (121), Default-Gateway (3),
    Domain-Name (15), Unknown (108), URL (114), Unknown (119)
    Unknown (252), LDAP (95), Netbios-Name-Server (44), Netbios-Node (46)
```

Final Word

"most of us are ipv6 haters, but we're also pragmatic. ipv6 may suck caterpillar snot, but we have no alternative. so get over it."

(~2008)

2024 APRICOT

APNIC 57

BANGKOK, THAILAND

21 February – 1 March 2024

#apricot2024