IPv6 resolved IPv4 gateways

draft-ipv6-resolved-gateway

Remco van Mook, IETF 124, November 2025

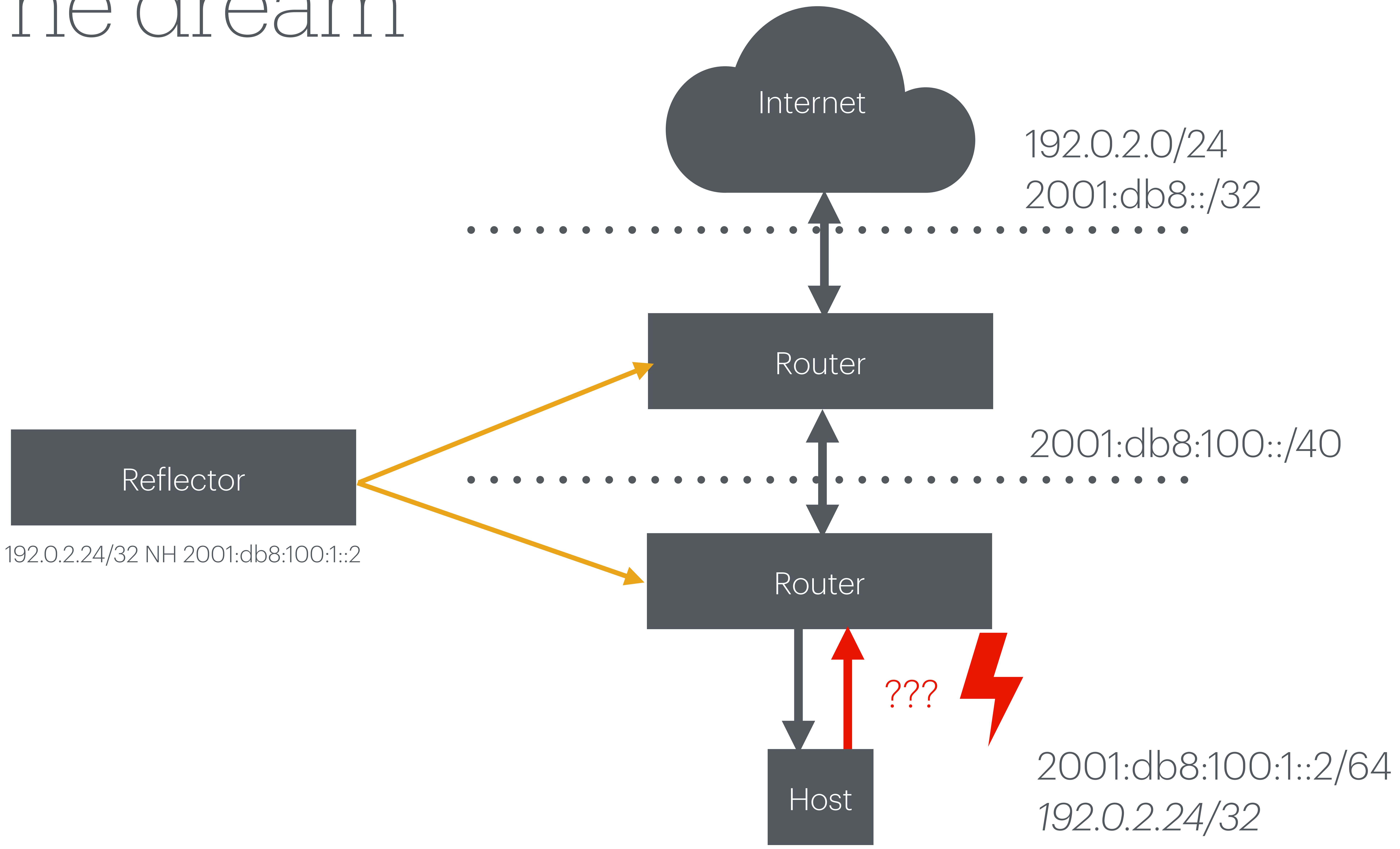
The basic premise

- The need to communicate over IPv4 *across the Internet* will be there for a good while
- **Talking IPv4 to the Internet *is not necessarily the same thing* as talking IPv4 locally**
- It will be *a very long time* before a router won't know what an IPv4 packet is
- **Discovery mechanisms (like ARP) are *a bad idea in multi tenant environments***
- A modern, orchestrated network knows where to send packets
- A layer 2 network knows how to carry IPv4 (no tunnels needed)

The dream

- Have a network design that solely focuses on IPv6 - **no IPv4 subnets** *(and no ARP)*
- Automatic/orchestrated addressing, prefix delegation
- Network design and routing fully based on IPv6
- Hosts can have an IPv4 address as a ‘tag’, an endpoint for software, sockets
- The network knows how to move IPv4 traffic - RFC8950 etc
- The challenge: ***Hosts don’t know how to send IPv4 traffic back to the Internet***

The dream


Internet
192.0.2.0/24
2001:db8::/32
Router
2001:db8:100::/40
Reflector
192.0.2.24/32 NH 2001:db8:100:1::2
Router
???
Host
2001:db8:100:1::2/64
192.0.2.24/32

The IPv4 default gateway IP

The least used, most important IPv4 address

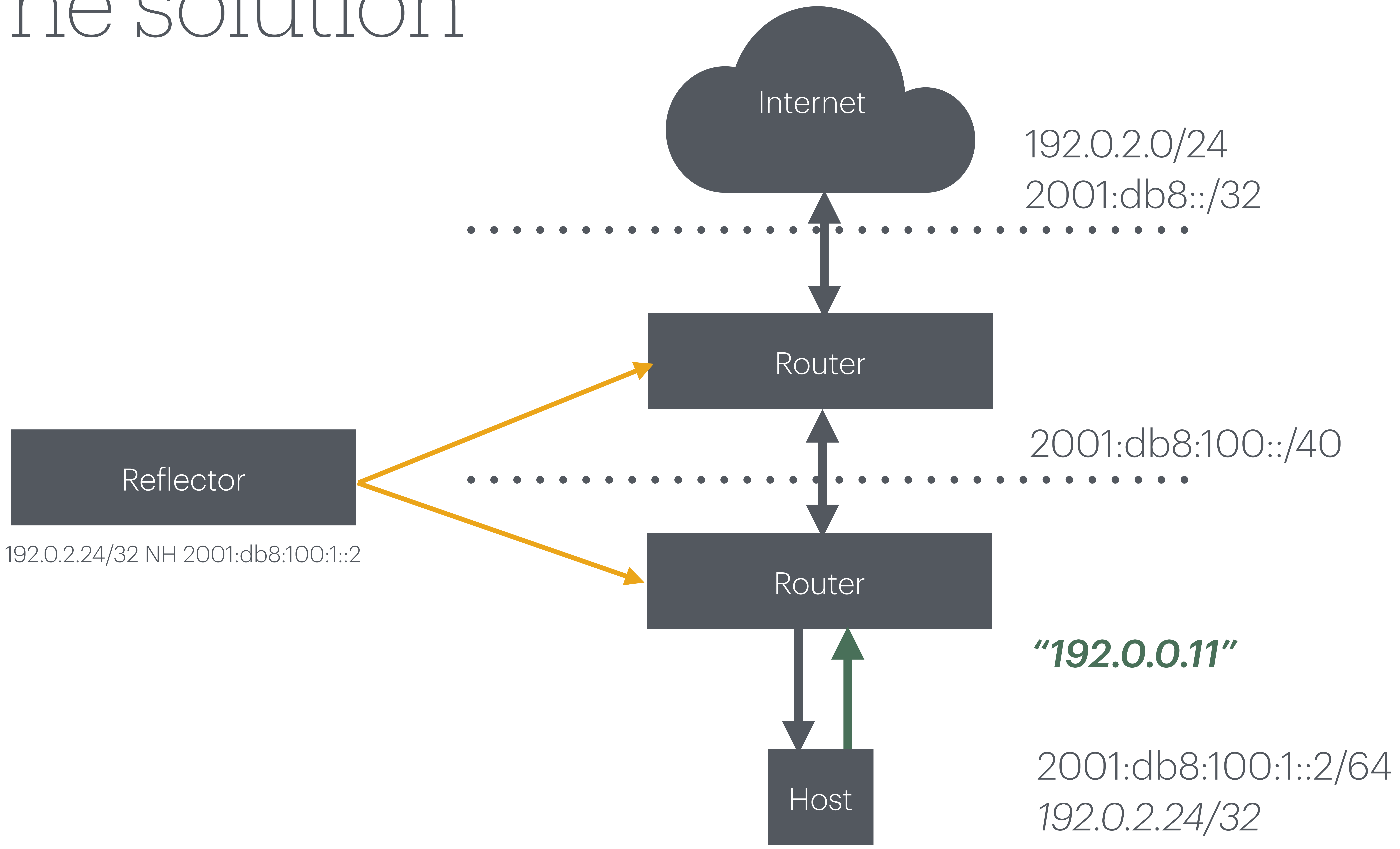
- In many cases, it's the only other device a host/CPE talks to on a public network
- The main function is as an _*identifier*_ to do a MAC Address lookup
- **The value of that identifier _*DOESN'T MATTER*_ -** as long as the host knows which interface to use

A better integrated approach?

- **Have a default gateway IPv4 address that means '*use the IPv6 default gateway L2*'**
- Instead of everyone building their own hacks, let's use a special address to signal
- So this can be built into OS network stacks natively to use the IPv6 neighbor cache
- **Use 192.0.0.11 from the IPv4 Special-Purpose Address Space**

  https://www.iana.org/assignments/iana-ipv4-special-registry/iana-ipv4-special-registry.xhtml

The solution



Off-link IPv4 default gateways

- This is undefined in current IPv4 standards, BCP, etc
- Only RFC coming close is RFC1027 - “Using ARP to Implement Transparent Subnet Gateways”
- The functionality is in wide use with various hosting providers (in a v4-only context)
- Tested on Windows 11, MacOS, Android, iOS, Linux, FreeBSD, ChromeOS
- Potential hackathon topic?

Backward compatibility

Example transitional method

- Using current DHCP server and clients:
  - Set a /32 IPv4 address
  - Set host route for 192.0.0.11 with DHCP
  - Set 192.0.0.11 as default gateway
  - Configure 192.0.0.11 as IP on router interfaces so ARP replies will be sent

Backward compatibility

Example dnsmasq config

```
port=0 # disable DNS
interface=wlan0
bind-dynamic
listen-address=192.0.0.11
dhcp-range=192.168.0.0,192.168.255.255,255.0.0.0,12h
dhcp-option=option:router,192.0.0.11
dhcp-option=option:netmask,255.255.255.255
dhcp-option=option:static-route,192.0.0.11,0.0.0.0
dhcp-option=option:dns-server,8.8.8.8
```

Needs Clarification/Discussion

- In the case of multiple IPv6 default GW, which one to use?
- Proposed clarification - the scope for selection is at the interface level, not the host
- Clear cookbook/configuration examples?
- ~~Update the name of the special purpose registry~~
- References to v4-via-v6 standards, automation?

Thoughts?

Current status - individual I-D, version 00

- Thanks to Tobias Fiebig, Warren Kumari, Jen Linkova and others for their feedback so far
- Working code in Linux user space, using rtnetlink to track the IPv6 gateway
- Aligns with ongoing IETF work in the routing space - v4-via-v6
- Github: remcovanmook/draft-ipv6-resolved-gateway
- remco.vanmook@gmail.com
- **Is intarea WG willing adopt this I-D?**