IPv6 Host OS Training

Windows and Linux

Terms and Conditions

All training content has the following terms and conditions for those that attend/participate in any HexaBuild and/or Partner training provided by HexaBuild.

- This digital file is provided for your use during and after this IPv6 class. HexaBuild wants you to have the information and have the ability to refer back to it in the future.
- HexaBuild has spent years developing the course training materials and we provide them to students electronically. In return, HexaBuild require students to preserve the confidentiality of the course materials so HexaBuild can continue to offer classes.
- Class attendees shall not share the electronic training materials with anyone else. Class attendees are not permitted to share the training class presentations, lecture information, or other class materials with others, or upload any of the training materials to public websites, and/or social websites, and/or non-restricted, and/or any private restricted websites.
- Class attendees are not permitted to copy, record, replay, or reproduce any of the course training materials or labs.
- Copyright and intellectual property of the course materials will be retained by HexaBuild. Class attendees agree to comply with all applicable federal, state, local, or international law or regulations (including without limitation any laws regarding copyright and intellectual property).

Outline Day 1

Day 1 - Morning

General IPv6 Protocol Review

Day 1 - Afternoon

Windows OS

Outline Day 2

Day 2 - Morning

Linux OS

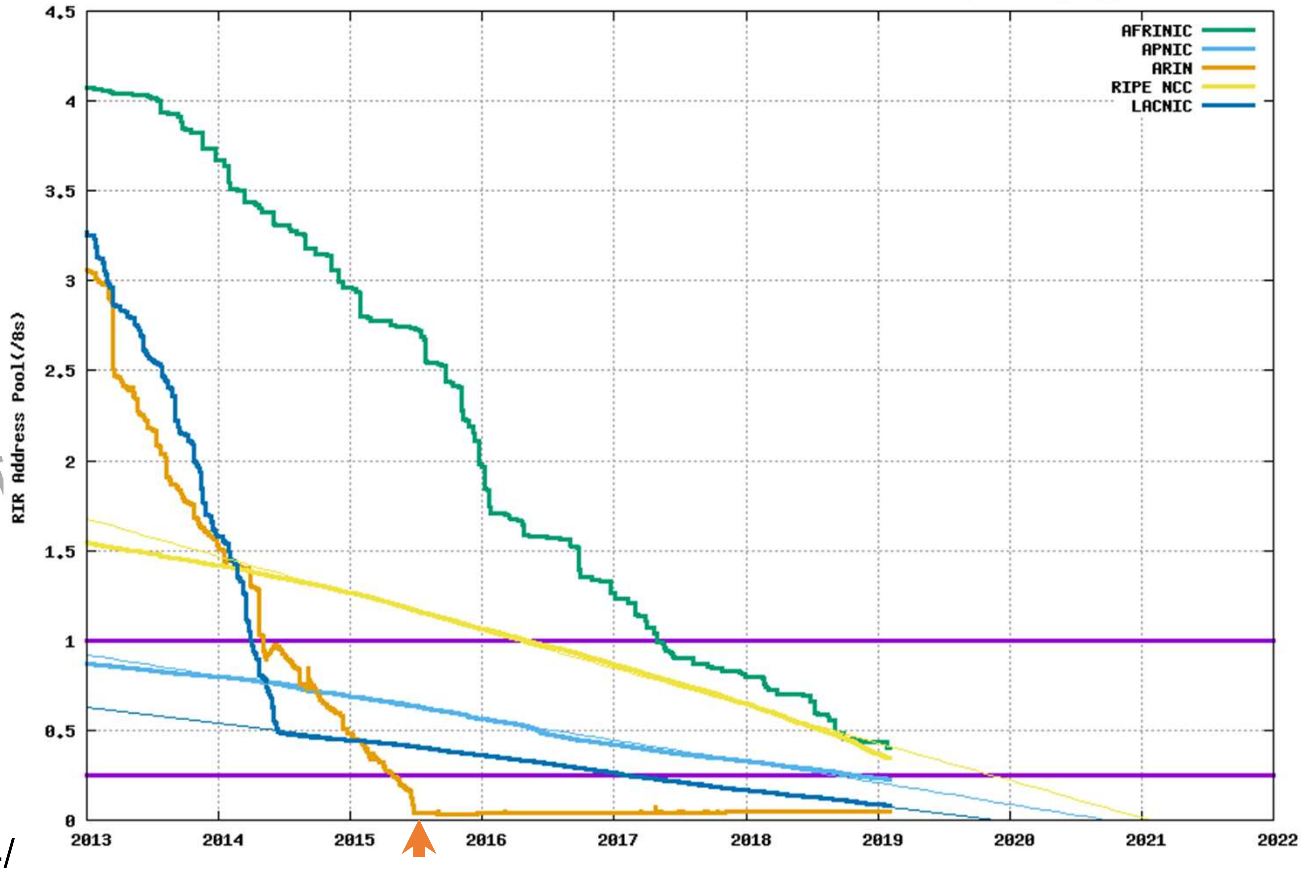
Day 2 - Afternoon

Lab
Q&A

General IPv6 Review

IPv4 Address Exhaustion

ARIN effectively ran out of IPv4 in 2015, by design. They chose to not reserve ANY IPv4 for future needs, in part to force the adoption of IPv6.



Source: http://www.potaroo.net/tools/ipv4/

Cisco Visual Networking Index - 2017

By The End of 2021:

- 4.6B Internet users
- 5.5B Mobile users
- 12B Mobile-ready devices
- 27.1B Networked devices
- 82% of all traffic will be video
- 95% of data center traffic from the cloud
- 20.6ZB per year in global data center traffic

https://www.cisco.com/c/en/us/solutions/service-provider/visual-networking-index-vni/index.html

Internet Protocol Version 6

- Next generation computer network protocol for use on the Internet & within private networks.
- IPv6 draft standard defined by the Internet Engineering Task Force (IETF) in 1995, now full Internet Standard (RFC 8200).
- IPv6 has had more than a decade to mature and is now ready for wide-scale deployment.
- IPv6 is designed to replace IPv4 but IPv6 is a different protocol, yet they can both coexist.
- Organizations that connect to the Internet need to prepare their systems to communicate using IPv6.

5 Stages of Grief for IPv6

Denial – IPv6 will never happen. We don't need IPv6, we've got plenty of IPv4 space.

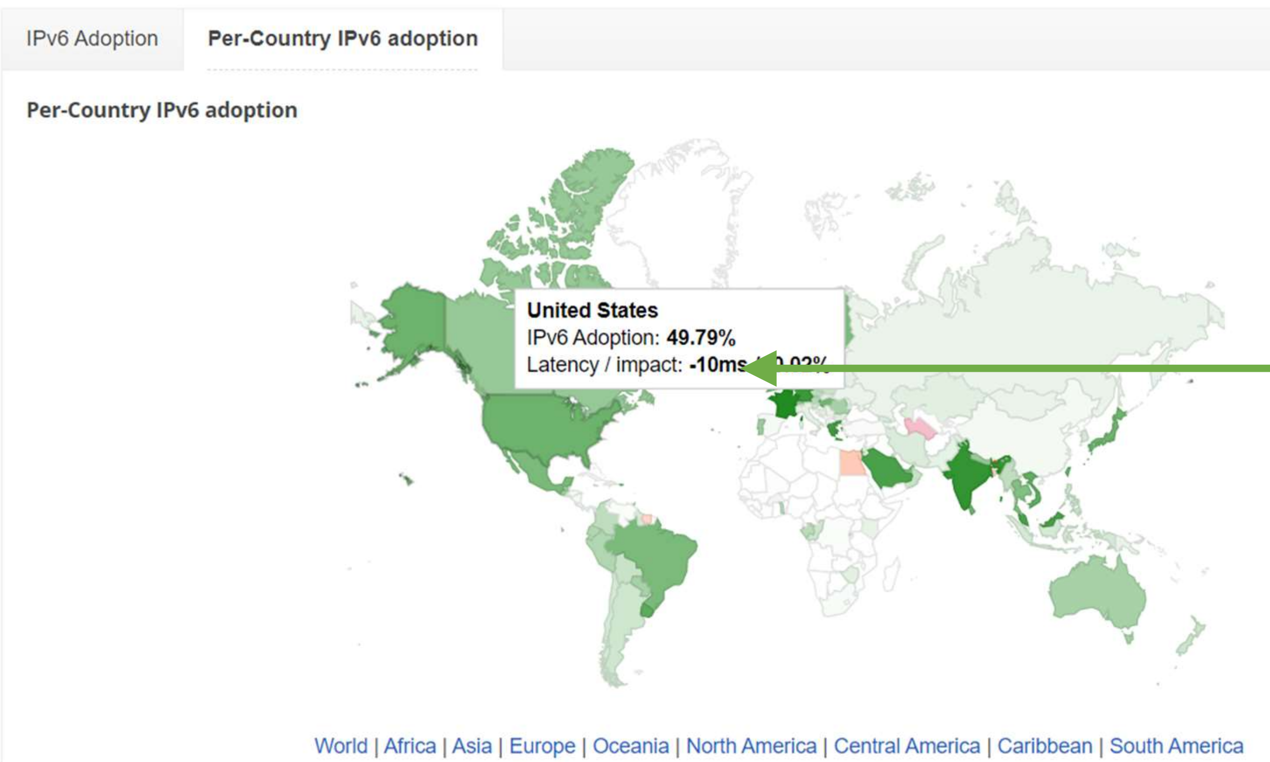
Anger – Why did they have to make the addresses so complicated?

Bargaining – If I go to IPv6 training then maybe the boss will quit bugging me about IPv6. I'll start on IPv6 when Google reaches 50% IPv6 traffic.

Depression – I have my work cut out for me to enable IPv6 on everything in our environment.

Acceptance – Maybe I should start at the Internet perimeter and start making some progress. Why didn't I deploy IPv6 sooner?

Where are we at?



eb 28

*Country data ranked by % of IPv6 connections from that country.

RANK	IPV6%	COUNTRY / REGION
1	100%	Bahrain
2	58.4%	Saudi Arabia
3	58%	Montserrat
4	55.2%	Uruguay
5	54.8%	Germany
6	54.6%	France
7	54.2%	Malaysia
8	52.1%	United States
9	51.4%	Belgium
10	50.9%	Greece

IPv6 is SIMILAR to IPv4

Both IP versions are:

Same 4-layer protocol model
Connectionless datagram delivery protocols
End-to-end routed protocols – support for unicast, multicast, anycast
Carried on many Layer-2 protocols
    Ethernet, PPP, SONET, Frame Relay, Fibre Channel, L2TP, FDDI, ARCNET
Supports popular TCP, UDP, SCTP, DCCP applications
DNS, DHCP, FTP, HTTP, etc. are similar
Same routing protocol & algorithms
ICMPv6 – similar
QoS (DiffServ, IntServ)

IPv6 is DIFFERENT that IPv4

IPv6:

IPv6 header has 128-bit source/destination addresses
IPv6 header has 20-bit Flow Label field
IPv6 does not use broadcast – multicast is used for all one-to-many communications
IPv6 hosts have multiple IPv6 addresses
Link-Local addresses (fe80::/10)
SLAAC
DHCPv6, Prefix Delegation
IPv6 has extension headers (e.g. IPsec AH and ESP)
Mobile IPv6 is much improved – but not really being used
IPv6 routers don't fragment packets - PMTUD

Comparing IPv4 and IPv6

IPv4:

192.0.2.1/24

2^{32}
Or 32-bit address space

We use NAT to hide our RFC 1918 private IPv4 addresses behind one or more public IPv4 addresses, we use a stateful firewall often at the same location as NAT

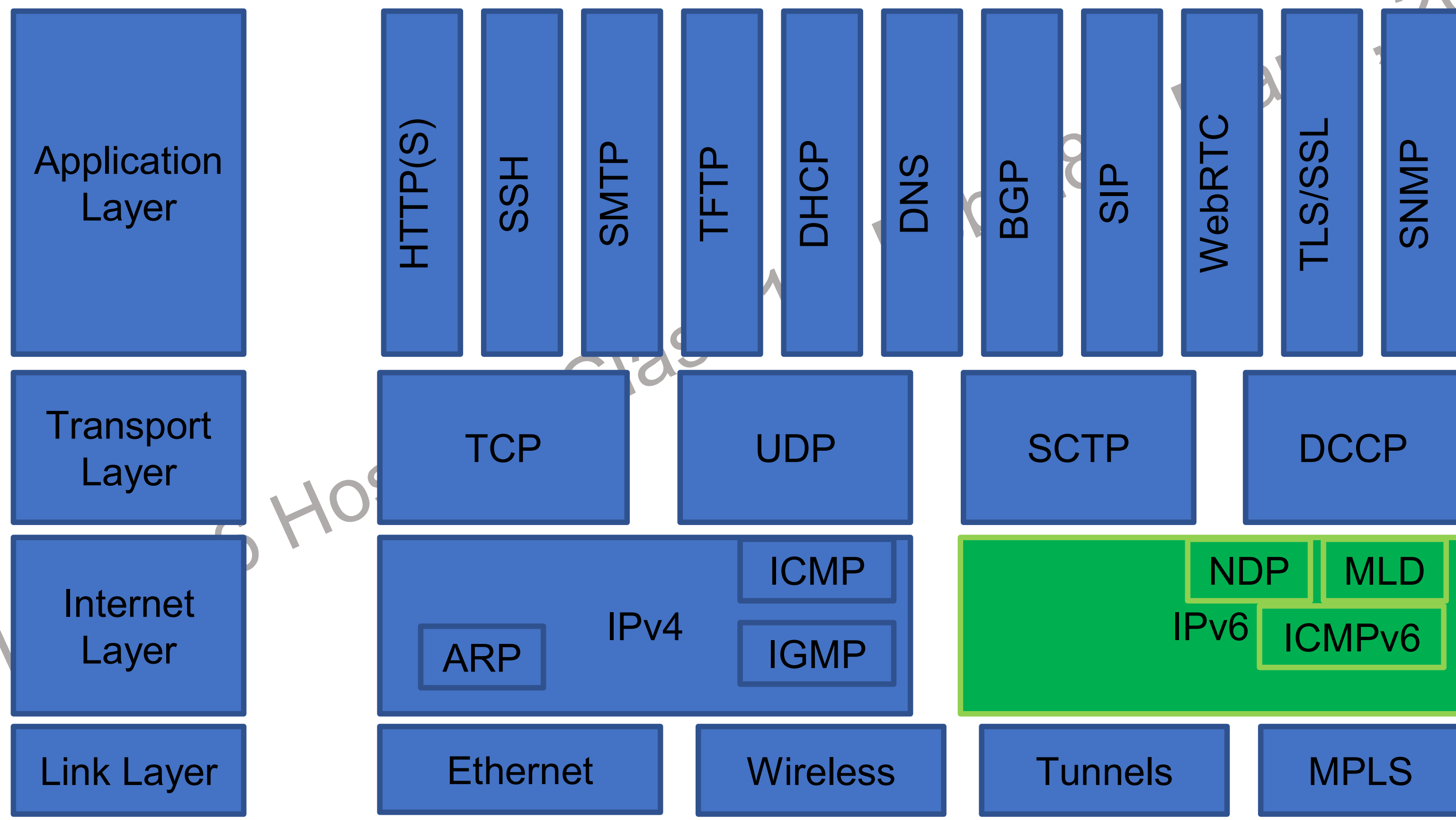
IPv6:

2001:db8:1234:abcd::1/64

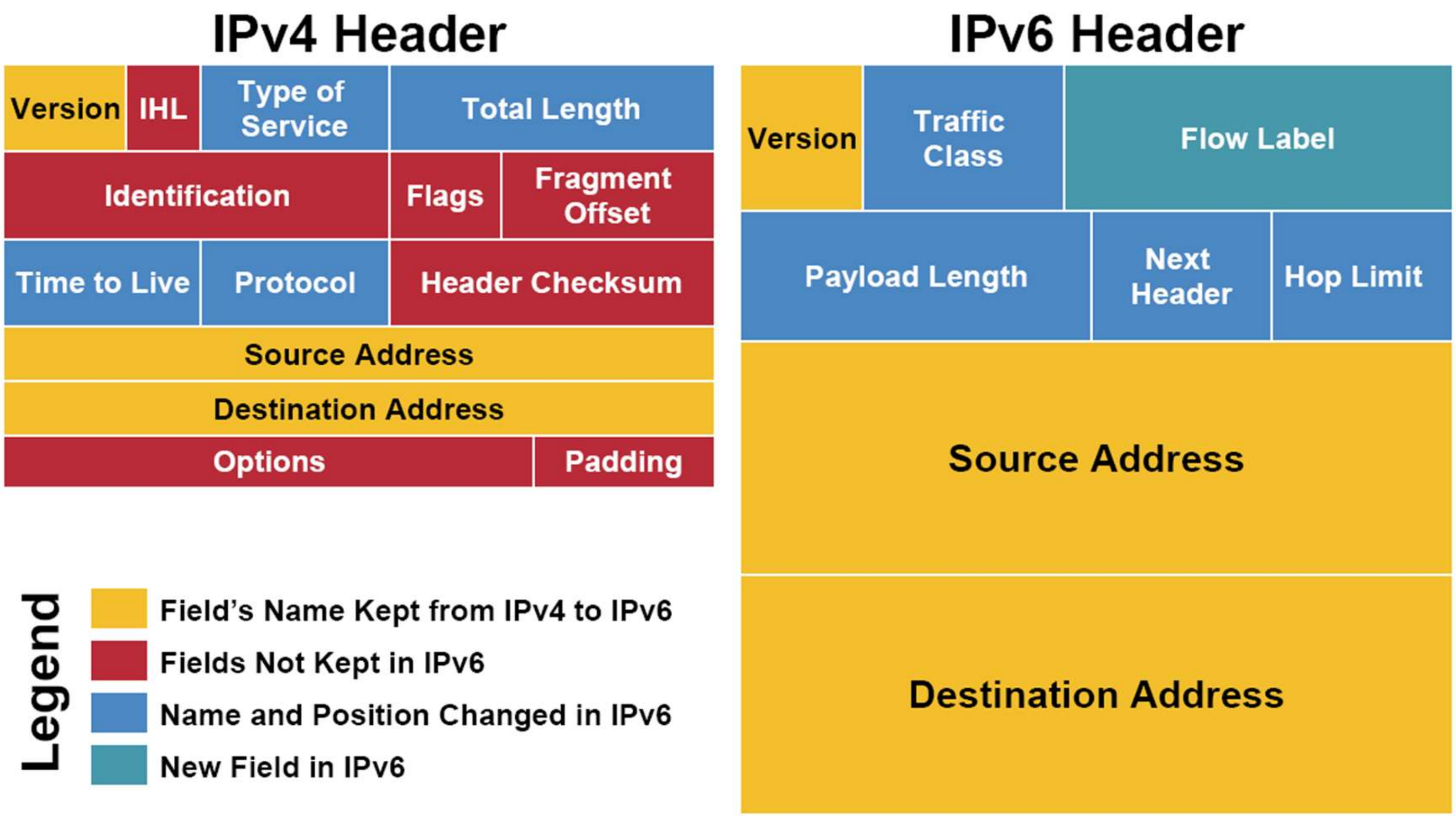
2^{128}
Or 128-bit address space

We use global IPv6 addresses so there is no need for NAT, but you will still use a stateful firewall to protect your network

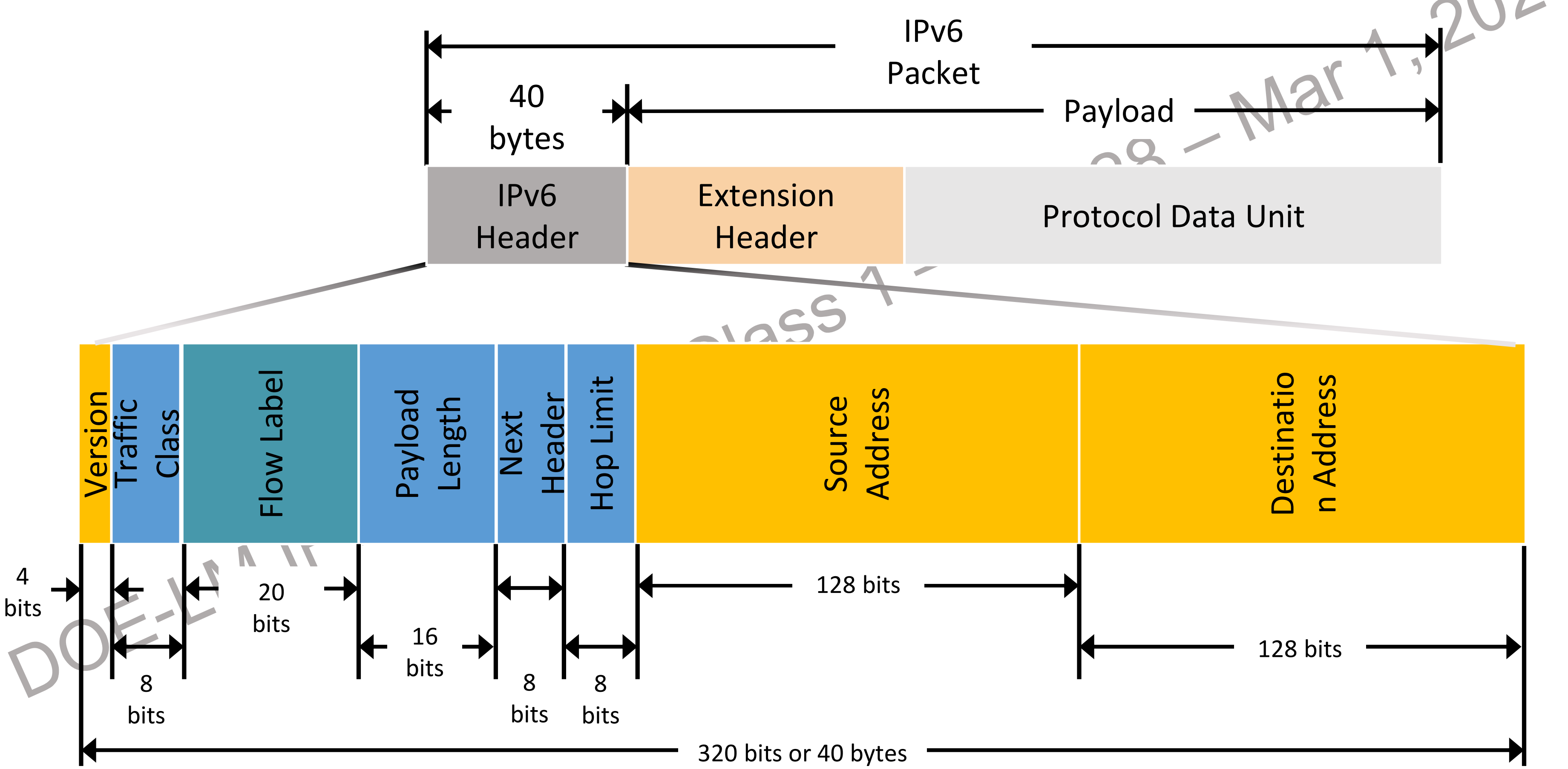
TCP/IP Protocol Stack



TCP/IP Protocol Headers Comparison



IPv6 Packet



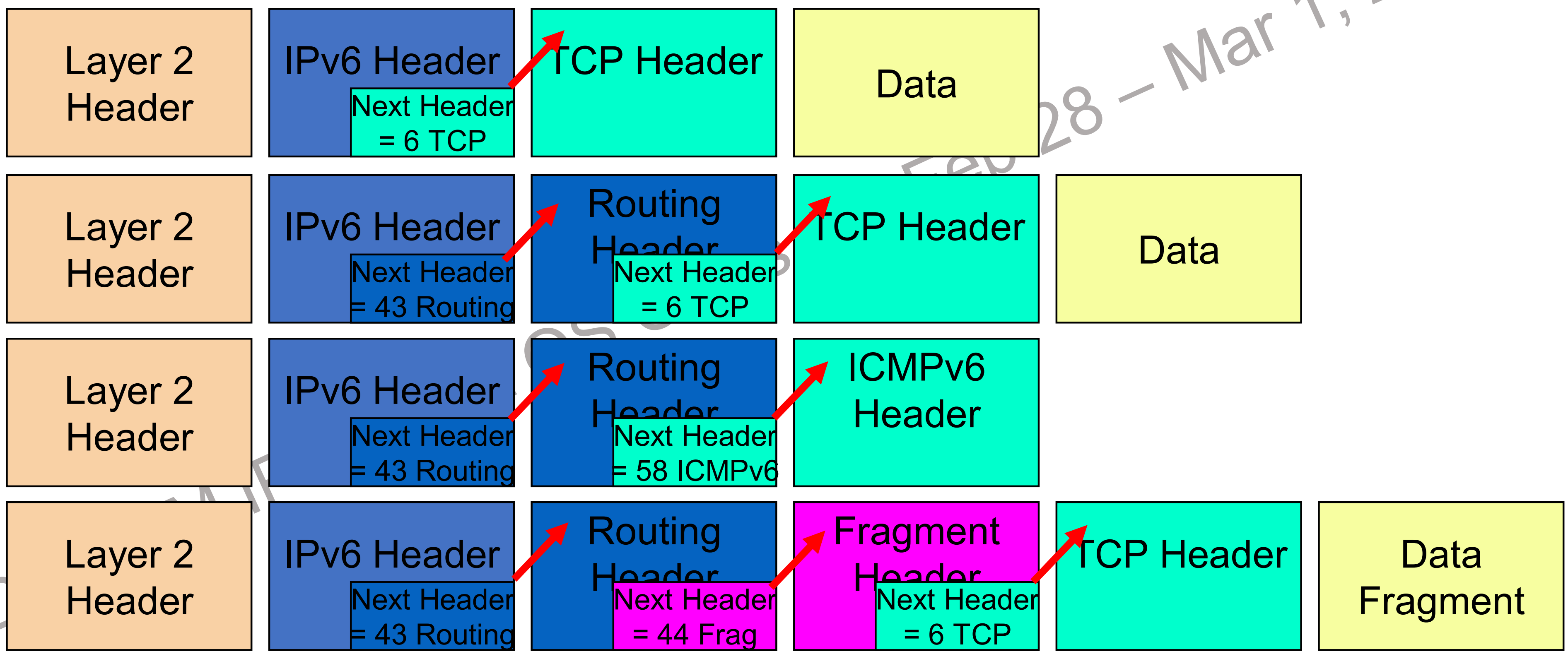
IPv6 Packet
40 bytes
Payload
IPv6 Header
Extension Header
Protocol Data Unit
Version
Traffic Class
Flow Label
Payload Length
Next Header
Hop Limit
Source Address
Destinatio n Address
4 bits
20 bits
8 bits
16 bits
8 bits
8 bits
128 bits
128 bits
320 bits or 40 bytes

Flow Label

What is a Flow Label?

Flow Label is a 20-bit field (new in IPv6) that labels all packets in a single flow
Flow Label is set by the source host (or first-hop router) and must not be changed in transit.
All packets belonging to the same stream must have the same flow label
Can be a 2/3/5-tuple
source/destination IP address, protocol type, and source/destination port numbers of transport protocol
Flow Label can help introduce entropy so that load balancing across link aggregation connections can operate more efficiently

IPv6 Extension Headers – RFC 8200



http://www.iana.org/assignments/ipv6-parameters/ipv6-parameters.xhtml

IPv6 Extension Header Numbers

0 - Hop-by-Hop Options, RFC 8200, Router Alert, RFC 2711

6 - Transmission Control Protocol (TCP), RFC 8200

17 - User Datagram Protocol (UDP), RFC 8200

41 - Generic Packet Tunneling in IPv6, RFC 2473

43 - Routing Options, RFC 8200 (RH0 – deprecated RFC 5095)

44 - Fragmentation Options, RFC 8200

50 - Encapsulating Security Payload (ESP), RFC 2406

IPv6 Extension Header Numbers

51 - Authentication Header (AH), RFC 2402

58 - ICMPv6, RFC 2463 (now RFC 4443)

59 - No next header (NULL), RFC 8200

60 - Destination Options, RFC 8200

88 - Enhanced IGRP (EIGRP) for IP version 6

89 - Open Shortest Path First version 3 (OSPFv3), RFC 5340

103 - PIM-SM, RFC 4601

Abundance of IPv6 Addresses

IPv4 is 32-bit address
2^{32} is
4,294,967,296
or about 4.3 billion addresses total (~3.7B useable)

IPv6 Increased Src/Dst addresses to 128 bits
2^{128} is
$3.4X10^{38}$ is
340,282,366,920,938,463,463,374,607,431,768,211,456
or ~340 undecillion addresses

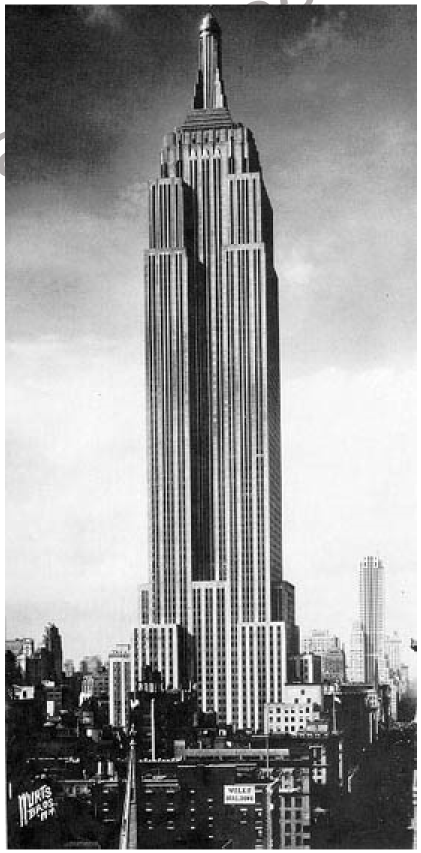
How Big Is the IPv6 Address Space?

If each IP address equaled one gram:
IPv4 would be 1/76th the weight of the Empire State Building
IPv6 would be 56.7 billion times the Earth's weight!

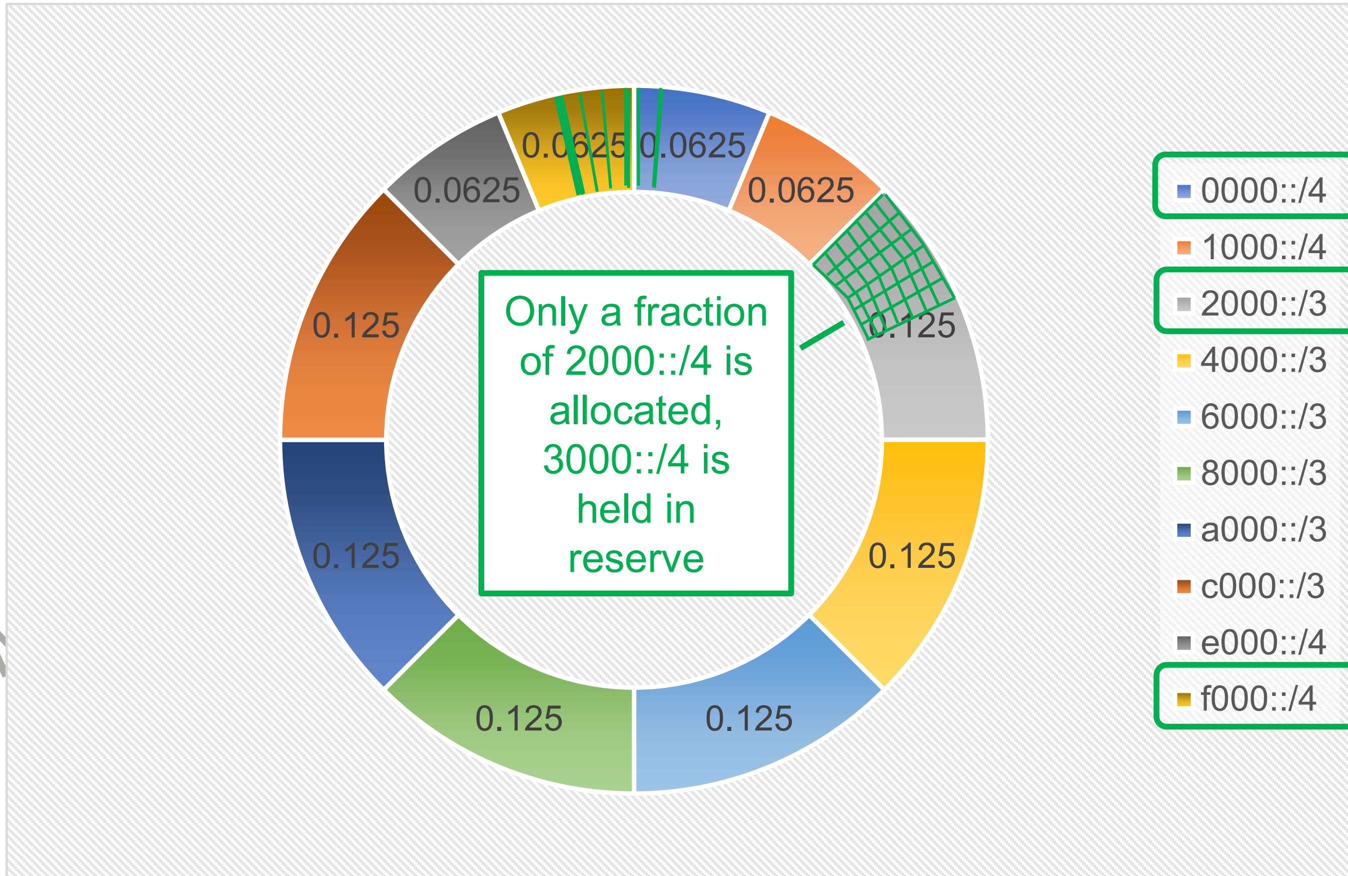
There are 1,246 IPv6 addresses per square meter of the area of the Milky Way galaxy

Every node on the IPv4 Internet can have its own IPv4 Internet 2^{64} times (the existing Internet squared!)

That ought to be enough!



How much IPv6 Address Space in Use?



IPv6 Addressing Notation

128 bits get converted into more readable form
0010 0000 0000 0001 1001 0000 1110 0000 0000 0000 0000 0011 0000 0000 0000 0000 / 0000 0000 0000 0000 0000 0000 0101 0000 0000 0000 0000 0000 0000 0000 0000 0000

Convert bits to hex, 8 “chunks” each with 4 hex digits

Also referred to as hextets, words, groups, quads

Fully-expanded format with all leading zeros shown
2001:90e0:0003:0000:0000:0050:0000:0000

IPv6 Addressing Notation Continued

2001:90e0:0003:0000:0000:0050:0000:0000
Reduce by removing leading zeros
2001:90e0:3:0:0:50:0:0

Use :: to consolidate multiple zeros – only once

Fully-compressed format
2001:90e0:3::50:0:0 <- prefer compressing the leftmost zeros
But you can also do – both are acceptable
2001:90e0:3:0:0:50::

Prefix format/notation
2001:90e0:3::/64

IPv6 Address Formatting

Store IPv6 addresses uncompressed – no zero compression and w/ leading zeros
2001:90e0:0003:0000:0000:0050:0000:0000

Prefix format/notation should be used for expressing a route

Lowercase letters of **a** through **f** should be used in addresses

Note: some operating systems and network equipment use uppercase **A** through **F** in addresses and prefix/routes. No idea why, just be aware

IPv4 addresses should just be expressed as IPv6 addresses via Mapped Addresses
::ffff:0:0/96

IPv6 Address Formatting Continued

When using an IPv6 address in an application it may require:
Square brackets around it, i.e.: [2001:db8:cafe:abcd:0001:0002:0003:0004]

A % to indicate the scope id (which interface to use), i.e.: 2001:db8:cafe::1%13

If a port is required, then brackets MUST be used, i.e.: [2001:db8:cafe::1]:8080

When in doubt, use brackets, but don't be surprised if things are not consistent

While it is possible to use IPv4 address notation for mapped addresses, it isn't guaranteed to always work so convert them to HEX

Hexadecimal Refresher

Decimal	Hex	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

Can You Find Something Odd?

2001:0db8:0011:0000:abcd:0120:0000:0000

1) 2001;db8;0011;0000;abcd;0120;0000;0000
2) 2001:db8:0011:0:abcd:120::0000
3) 2001:db8:11::abcd:12:0:0
4) 2001:db8:11:0:abcd:120:0:0
5) 2001:db8:11::abcd:120::
6) 2oo1:db8:11:o:abcd:o12o:o:o
7) 2001:db8:11:0:abcd:120::

Addressing: Format Prefix

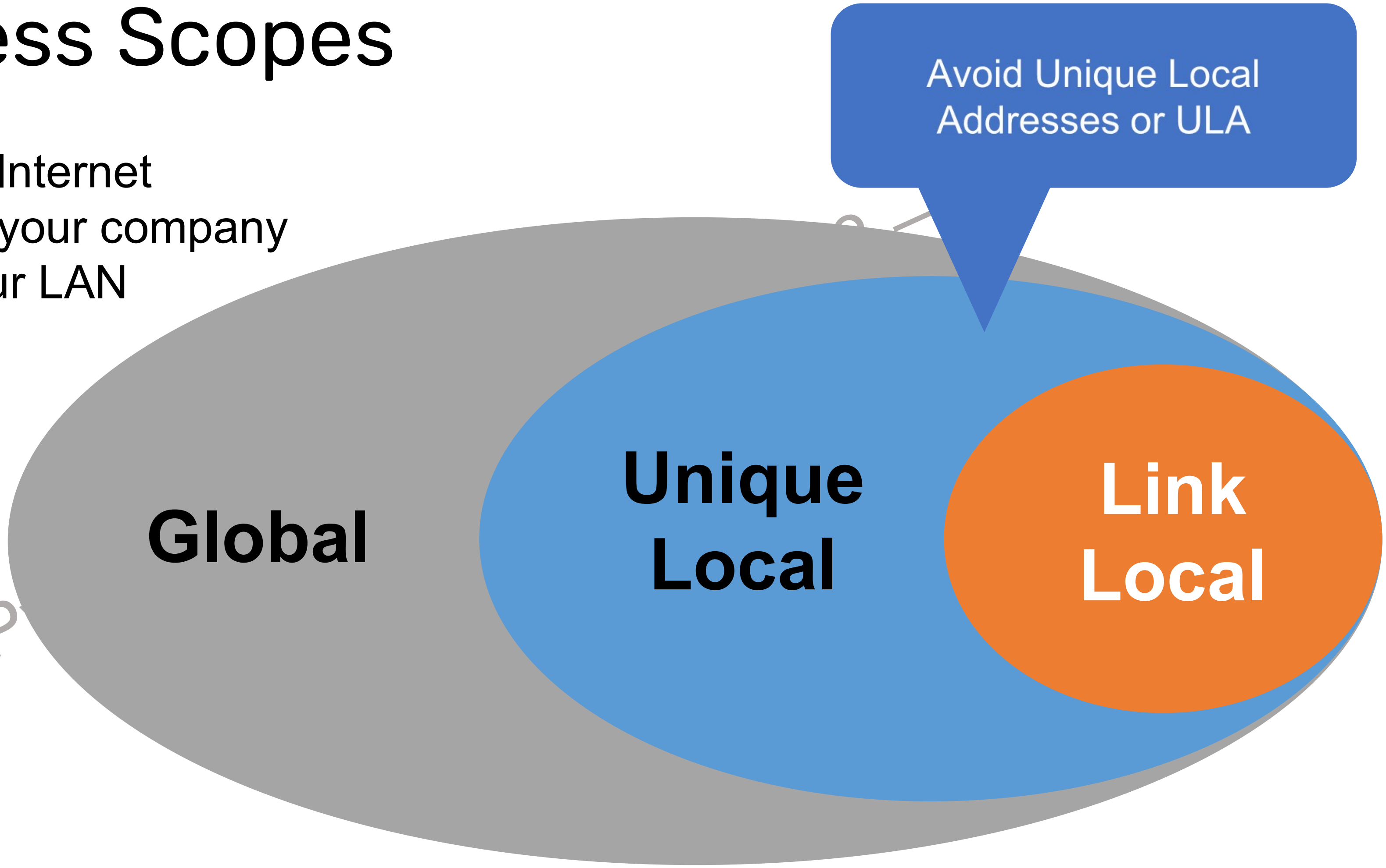
Reserved (::0/128)	0000 0000
Unassigned	0000 0001
Reserved for NSAP Allocation	0000 001
Reserved for IPX Allocation – later deprecated	0000 010
Unassigned	0000 011
Unassigned	0000 1
Unassigned	0001
Aggregatable Global Unicast Addresses (2000::/3)	001
Provider-Based Unicast Address	010
Unassigned	011
Rsvd for Neutral-Interconnect-Based Unicast Addrs	100
Unassigned	101

Addressing: Format Prefix

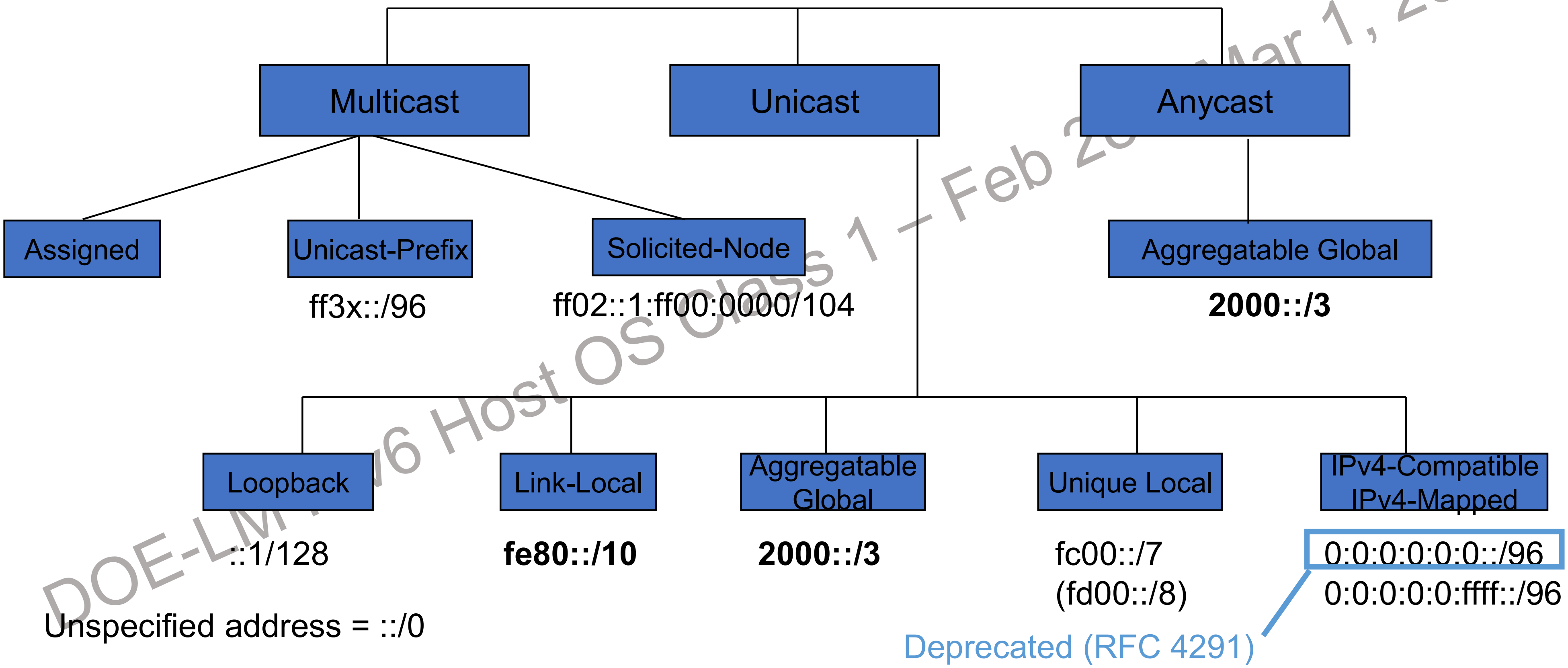
Unassigned	110
Unassigned	1110
Unassigned	1111 0
Unassigned	1111 10
Unique Local (fc00::/7 = fc00::/8 & fd00::/8)	1111 110
Unassigned	1111 1110 0
Link Local Use Addresses (fe80::/10)	1111 1110 10
Site Local Use Addresses (fec0::/10) - deprecated	1111 1110 11
Multicast Addresses (ff00::/8)	1111 1111

IPv6 Address Scopes

Global = whole Internet
Unique Local = your company
Link Local = your LAN



IPv6 Address Types and Scope



IPv4 Compatible/Mapped IPv6 Addresses

Addresses used for transition between IPv4 and IPv6
Both have first 80 bits of “0”s
IPv4-Mapped IPv6 Addresses
For IPv4-only-capable devices
192.168.12.34 gets mapped to
0000:0000:0000:0000:0000:**ffff**:192.168.12.34
IPv4-Compatable IPv6 Addresses (Deprecated)
192.168.12.34 gets mapped to
0000:0000:0000:0000:0000:**0000**:192.168.12.34

::ffff:c0a8:017b = ::ffff:192.168.1.123 = 192.168.1.123

RIR Global Unicast Addresses - 2021

2001:0000::/23	IANA
2001:0200::/23	APNIC
2001:0400::/23	**ARIN**
2001:0600::/23	RIPE NCC
2001:0800::/22	RIPE NCC
2001:0c00::/23	APNIC (2001:db8::/32)
2001:0e00::/23	APNIC
2001:1200::/23	LACNIC
2001:1400::/22	RIPE NCC
2001:1800::/23	**ARIN**
2001:1a00::/23	RIPE NCC
2001:1c00::/22	RIPE NCC

RIR Global Unicast Addresses

2001:2000::/19	RIPE NCC
2001:4000::/23	RIPE NCC
2001:4200::/23	AFRINIC
2001:4400::/23	APNIC
2001:4600::/23	RIPE NCC
2001:4800::/23	**ARIN**
2001:4a00::/23	RIPE NCC
2001:4c00::/23	RIPE NCC
2001:5000::/20	RIPE NCC
2001:8000::/19	APNIC
2001:8000::/19	APNIC
2001:a000::/20	APNIC

RIR Global Unicast Addresses

2001:b000::/20	APNIC
2002:0000::/16	6to4
2003:0000::/18	RIPE NCC
2400:0000::/12	APNIC
2600:0000::/12	**ARIN**
2610:0000::/23	**ARIN**
2620:0000::/23	**ARIN**
2630:0000::/12	**ARIN**
2800:0000::/12	LACNIC
2a00:0000::/12	RIPE NCC
2a10:0000::/12	RIPE NCC
2c00:0000::/12	AFRINIC

ARIN GUA IPv6 Prefix Allocation

If you are an enterprise with an IPv4 ARIN address block and an ASN
https://www.arin.net/policy/nrpm.html
https://www.arin.net/resources/request.html

Then you qualify for a /48 per “site”, rounded up to next nibble boundary of Provider Independent (PI) space

1 site = /48
≤12 sites = /44
≤ 192 sites = /40
≤ 3072 sites = /36
≤ 49152 sites = /32

Many SMB will be a /44 to /40 by default

American Registry for Internet Numbers

Additional IPv6 Addressing Thoughts

Typically break up IPv6 addresses on a hex-character boundary (nibble)

Enterprise Solution

- /44 allocation would be typical for 12 sites or less
- /48 for each campus/office – min. Internet route – allows high availability
- /56 for each building
- /64 for each VLAN and Pt-2-Pt link

IP Address Management Solutions are required

Excel spreadsheets are not a good plan

Get your staff to think/dream in hex – for nibble prefix, count by 4

Wasting a /64 – Let’s Do Some Math

Many people can’t seem to get past the fact that a /64 is considered “wasteful”

A single /64 can have 18 quintillion nodes or 18,446,744,073,709,551,616 nodes

If you had a network with 1M nodes, your efficiency is infinitesimally small

$$n = \frac{x}{1.8 \text{ x } 10^{19}} = \frac{1.0 \text{ x } 10^{6}}{1.8 \text{ x } 10^{19}} = 0.00000000000005421011$$

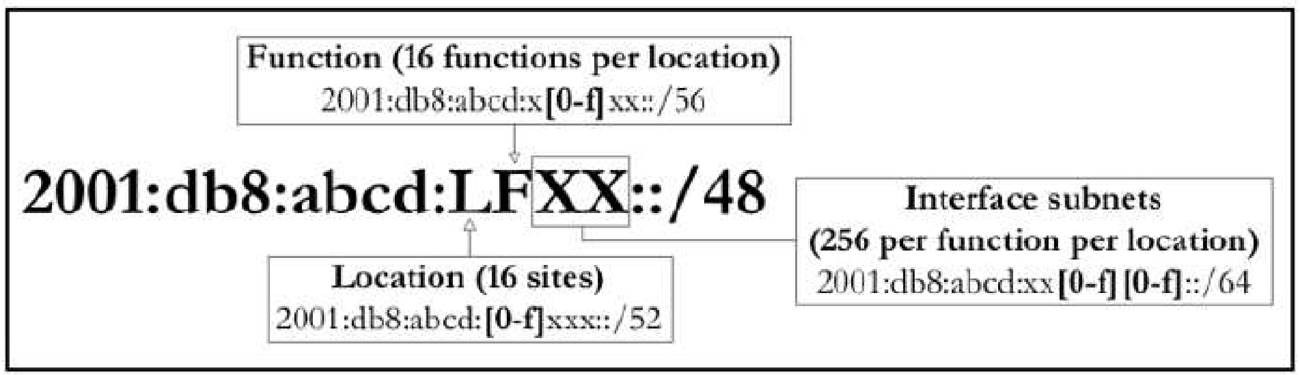
Whether you have 2 or 2M nodes, you are using only a small fraction of the /64

When the human population reaches 10B, each person could receive 1,844,674,407 /64s

https://community.infoblox.com/t5/IPv6-Center-of-Excellence/How-Many-IPv6-Nodes-Can-You-Have-on-a-LAN/ba-p/6092

IPv6 Address Simplicity

Abundance of IPv6 address prefixes means addressing for simplicity rather than scarcity
IPv6 addresses naturally relate to location/topology
IPv6 addressing and routing area BFFs
Location represented in high-order hex digits
Opportunity to assign function within locations



Source: IPv6 Address Planning: Designing an Address Plan for the Future, by Tom Coffeen

IPv6 Addressing Principles

Plan for number of **subnets** not number of host addresses (subnets are a /64)

Allocate or assign subnets in a way that leave additional in reserve for future use (sparse allocation)

Define a site in a way that best fits your organization's operational and administrative needs (a typical site allocation is a /48)

Though consistency within sites can increase operational efficiency, not every site must have the same plan internally (but start off that way – keep it simple)

http://www.cisco.com/en/US/docs/solutions/SBA/February2013/Cisco_SBA_BN_IPv6AddressingGuide-Feb2013.pdf

IPv6 Addressing Principles Continued

For instance, a remote office is unlikely to have the same hierarchical requirements (or function or location assignments) as a data center

Just because you have the bits to accommodate it doesn't mean that you need to use them to create hierarchy for its own sake

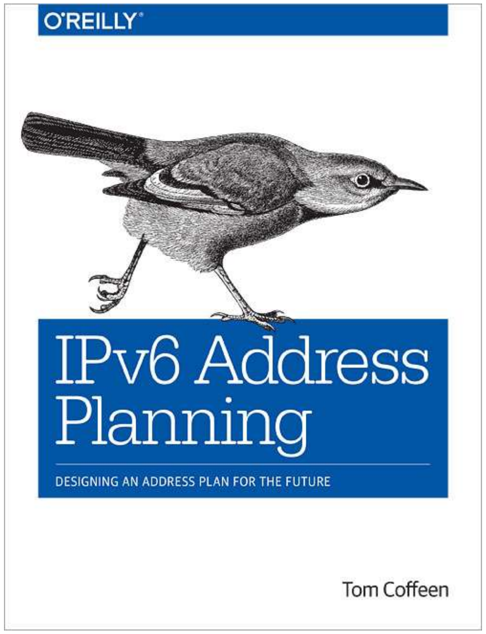
Try to make your plan as simple as possible with as few levels of hierarchy as absolutely necessary (but no fewer than that – ying & yang and inner Zen help!)

http://www.cisco.com/en/US/docs/solutions/SBA/February2013/Cisco_SBA_BN_IPv6AddressingGuide-Feb2013.pdf

IPv6 Address Planning Book

IPv6 Address Planning: Designing an Address Plan for the Future, by Tom Coffeen

Chapter 1 Where We’ve Been, Where We’re Going
Chapter 2 What You Need To Know About IPv6 Addressing
Chapter 3 Planning Your IPv6 Deployment
Chapter 4 IPv6 Subnetting
Chapter 5 IPv6 Address Planning Concepts
Chapter 6 Getting IPv6 Addresses
Chapter 7 Creating an IPv6 Addressing Plan
Chapter 8 Working with IPAM and DDI
Chapter 9 Managing Growth and Change
Chapter 10 Keeping Your IPv6 Addresses



http://shop.oreilly.com/product/0636920033622.do

Unique Local Addresses (ULA)

RFC 4193 - Unique Local IPv6 Unicast Addresses - fc00::/7

Unicast addresses for private use – not for use on the Global Internet
Unique to an organization to avoid address overlaps when organizations connect
fd00::/8 – then 40-bit Global ID is pseudo-random to avoid conflicts
fdxx:xxxx:xxxx:yyyy:abcd:abcd:abcd:abcd
ULA-Centralized – Registered ULA – no longer active – it sort of defeated the purpose of ULA!!!
    https://www.sixxs.net/tools/grh/ula/ - Even SixXS realized this was dumb
Global Unique Addresses (GUA) are preferred

https://community.infoblox.com/t5/IPv6-Center-of-Excellence/3-Ways-to-Ruin-Your-Future-Network-with-IPv6-Unique-Local/ba-p/6177

IPv6 Multicast Addresses

Flags Field: [0|R|P|T] (RFC 7371)

Bit 1 = Reserved must be zero
Bit 2 = Set to 1 if Embedded RP is used
Bit 3 = 1 if address is assigned based on network prefix
Bit 4 = Set to 0 if it is a well-known multicast address – Permanently assigned, Set to 1 if this is a non-permanently-assigned ("transient") – Temporarily assigned

FF	Flags	Scope	Group ID
8 Bits	4 & 4 Bits		112 Bits

IPv6 Multicast Addresses

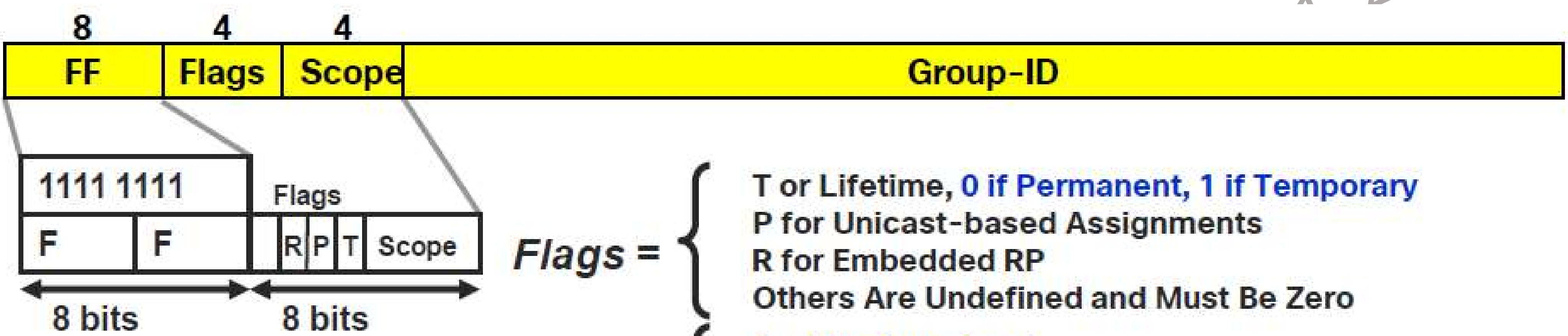
Scope Field: (RFC 4291, RFC 7346)

1 – Interface/Node-Local – ff01
2 – Link-Local – ff02
3 – Realm-Local – ff03 (RFC 7346)
4 – Admin-Local – ff04
5 – Site-Local – ff05
8 – Organization-Local – ff08
E – Global – ff0e

FF	Flags	Scope	Group ID
8 Bits	4 & 4 Bits		112 Bits

IPv6 Multicast Addresses

2023



Flags =
- T or Lifetime, 0 if Permanent, 1 if Temporary
- P for Unicast-based Assignments
- R for Embedded RP
- Others Are Undefined and Must Be Zero

Scope =
- 1 = interface-local
- 2 = link
- = admin-local
- 5 = site
- 8 = organization
- E = global
- 0, 3, F = reserved

4

Note: Other scopes (6, 7, 9-D) are unassigned but can be used

Source: Cisco Live BRKRST-2095

IPv6 Multicast Addresses

Permanently assigned multicast addresses

ff02:0:0:0:0:0:0:1 - All Nodes Address (ff02::1)
ff02:0:0:0:0:0:0:2 - All Routers Address (ff02::2)
ff02:0:0:0:0:0:0:5 - OSPFIGP (ff02::5)
ff02:0:0:0:0:0:0:6 - OSPFIGP DR (ff02::6)
ff02:0:0:0:0:0:0:9 - RIPng Routers (ff02::9)
ff02:0:0:0:0:0:0:A - EIGRPv6 Routers (ff02::A)
ff02:0:0:0:0:0:0:B - Mobile-Agents (ff02::B)
ff02:0:0:0:0:0:0:D - All PIM Routers (ff02::D)
ff02:0:0:0:0:0:0:12 – VRRP (ff02::12)

IPv6 Multicast Addresses

Permanently assigned multicast addresses

ff02:0:0:0:0:0:0:16 - All MLDv2 capable Routers (ff02::16)
ff02:0:0:0:0:0:1:2 - All-dhcp-agents (ff02::1:2)
ff02:0:0:0:0:0:1:3 - Link Local multicast name resolution (ff02::1:3)
ff02:0:0:0:0:1:ff00:0000/104 - Solicited-Node Address for NDP
ff02:0:0:0:0:2:ff00::/104 - Node Information Queries (NIQ)
ff05:0:0:0:0:0:0:2 - All Routers Address (ff05::2)
ff05:0:0:0:0:0:1:3 - All DHCP servers (ff05::1:3)

IPv6 Anycast Addresses

Anycast is the method of sending a packet to the topologically "nearest" node from a group of potential receivers all using the same address
Anycast is "one-to-one-of-many"
Anycast addresses violate the "laws of uniqueness" of IP addresses
Only routers and special servers should respond to an anycast address
IPv6 anycast addresses look the same as an IPv6 unicast address
Anycast is used for DNS root name servers
Subnet Router Anycast Addresses (RFC 4291)
Reserved Subnet Anycast Addresses (RFC 2526)

IPv6 Link-Local Addresses

These are the IPv6 addresses that every host has on its interfaces

When a host boots up it assigns a fe80::/10 IPv6 address to its interfaces, the last 64 bits are the IID

`fe80:0000:0000:0000:abcd:abcd:abcd:abcd`

You can think of these like APIPA (169.254.0.0/16)

They are only used for communication on the local link (local “broadcast” domain) – not used for off-link communications

The same link-local address can be used on a node’s multiple interfaces

A link-local address is a perfectly valid next-hop address

ICMPv6

ICMPv6 uses IPv6 extension header #58 (RFC 4443)

Type	Description
1	Destination Unreachable
2	Packet to Big
3	Time exceeded
4	Parameter problem
128	Echo Request
129	Echo Reply
130	Multicast Listener Query – sent to ff02::1 (all nodes)
131	Multicast Listener Report
132	Multicast Listener Done – sent to ff02::2 (all routers)

PING

MLD

ICMPv6

ICMPv6 uses IPv6 extension header #58 (RFC 4443)

133 Router Solicitation (RS) – sent to ff02::2 (all routers)
134 Router Advertisement (RA) – sent to ff02::1 (all nodes)
135 Neighbor Solicitation (NS) – sent to ff02:0:0:0:0:1:ff00::/104
136 Neighbor Advertisement (NA)
137 Redirect message

Router Redirection

ARP Replacement

IPv6 Router Advertisements

IPv6 uses Router Advertisements (RAs)
An RA is sent out every 200 seconds or in response to a Router Solicitation (RS)
An RA uses ICMPv6 Type 133 and an RS uses ICMPv6 Type 134

The RA is used to tell all the host who is the default gateway for the network
The RA us also used to tell the hosts how to provision an address (SLAAC or DHCPv6)
The RA can also provide DNS server information

You may not always have an RA, think a Data Center deployment, but otherwise, it is likely being used

Router Solicitation and Advertisements

Nodes send RSs (Type 133)
On bootup when they can't wait
200 seconds for the next RA

Source: fe80::/10
    Link-Local address of Node
Destination: ff02::2 (all routers)

Data: Query to send RA,
Source LL Addr

Routers send RAs (Type 134)
Every 200 seconds or
Responding to an RS message

Source: fe80::/10
    Link-Local address of Router
Destination: ff02::1 (all nodes)

Data: Options, subnet prefix,
lifetime, autoconfig flags (M&O bits)

https://community.infoblox.com/blogs/2014/06/16/why-you-must-use-icmpv6-router-advertisements-ras

RA Message Data

RA messages contain valuable information for nodes to pull themselves up by their bootstraps and get on the network

- Router Lifetime
- Reachable Time
- Retransmission Timer
- Source Link-Layer Address
- MTU size for the link
- Prefix Information
- Address Autoconfiguration Flag – A flag
- On-Link Flag – L flag
- Managed Address Configuration Flag - M flag
- Other Stateful Configuration Flag - O flag

RA Message Information

Address Autoconfiguration Flag – A-bit (AdvAutonomousFlag) – Indicates SLAAC
A = 0 = Node should not perform stateless address assignment
A = 1 = Node should perform stateless address assignment

On-Link Flag - L-bit (AdvOnLinkFlag) – Indicates Link-Status
L = 1 = Prefix is “on-link”, hosts can directly reach each other in same prefix without using local router
L = 0 = Hosts in same prefix may be on a different physical/virtual link, use the local router

RA Message Information

Managed Address Configuration Flag – M-bit (AdvManagedFlag) – Indicates DHCPv6 use
M = 0 = No managed address assignment
M = 1 = User managed address assignment with DHCPv6

Other Stateful Configuration Flag – O-bit (AdvOtherConfigFlag) – Indicates other info is available
O = 0 = No other information is available
O = 1 = Host should use DHCPv6 to get other config info

RA Message Parameters

A = 1, L = 1, M = 0, O = 0

End host use stateless address auto-configuration (SLAAC)

A = 1, L = 1, M = 0, O = 1

End hosts use SLAAC, but also use DHCPv6 stateless to learn DNS and other information DHCPv6 from local router

A = 0, L = 1, M = 1, O = 0

End hosts or subscriber devices should use stateful DHCPv6 for all information and other optional data

A = 0, L = 1, M = 0, O = 0

Servers using static address assignment

Renumbering Still Needs Work – RFC 5887

Site renumbering for IPv4 and IPv6 can be burdensome

Static addresses must be renumbered

DHCP/DHCPv6/SLAAC can help

Public IPv6 addresses must change, but IPv4 private addresses don't because they are NAT'ed

Preference to use PI IPv6 addresses rather than PA IPv6 addresses

Even static private IPv4 renumbering can still be difficult

Neighbor Solicitation and Advertisements

Nodes send NSs (Type 135)
When sending IPv6 packet to
Another node

Source: Unicast IPv6 Address
Destination: Solicited Node
Multicast Address
ff02::1:ffaa:bbcc

Data: Target link-layer address
Query: What is your link-layer address?

Routers send NAs (Type 136)
Responding to an NS message

Source: Unicast IPv6 Address
Destination: Unicast Address of
Requestor or ff02::1 (all nodes)

Data: R/S/O Flags, Target’s Link-layer
address
Response: Here is my IPv6 and link-
layer address.

Solicited Node Multicast Address

Multicast addresses used for Neighbor Discovery
Destination IPv6 address in ICMPv6 NS message
    ff02::1:ff00::/104
    ff02:0:0:0:0:1:ffXX:XXXX
    Last 24-bits of queried IPv6 address is copied into last bits of solicited node multicast address

Destination MAC address of NS packet uses IPv6 multicast address
    33-33-00-00-00-00 to 33-33-FF-FF-FF-FF
    Last 32-bits of destination IPv6 address is copied into MAC address
2001:db8:1000:2000:1234:5678:9abc:1234
ff02::1:ffbc:1234
33-33-FF-BC-12-34

Duplicate Address Detection – DAD Rules

Must be performed by all nodes

Performed with both stateless and stateful autoconfiguration

Performed before assigning a unicast address to an interface

Performed on interface initialization

Not performed for anycast addresses

Link must be multicast capable

New address is called "tentative" as long as duplicate address detection takes place

DAD Process

1. Interface joins all-nodes multicast group

2. Interface joins solicited-node multicast group

3. Node sends one NS with
   Target address = tentative IP address
   Source address = unspecified (::)
   Destination address = tentative solicited-node address

DAD Resolution

If address already exists, the particular node sends a NA with

Target address = tentative IP address

Destination address = tentative solicited-node address

If soliciting node receives NA with target address set to the tentative IP address, the address must be duplicate

The host does not configure or use the address

DAD Issues

What if you were to ping your own link-local from yourself?

Would that trigger DAD detection?

What if a node on the local network used that same technique to take you offline?

For statically assigned hosts, you may not need the DAD process to function.

Some devices now allow disabling of DAD

Highly unlikely to be a conflict when using a dynamic addressing method

Stateless Address Autoconfiguration (SLAAC)

RFC 4862, IPv6 Stateless Address Autoconfiguration

If the Address Autoconfiguration Flag (“A” flag) = 1

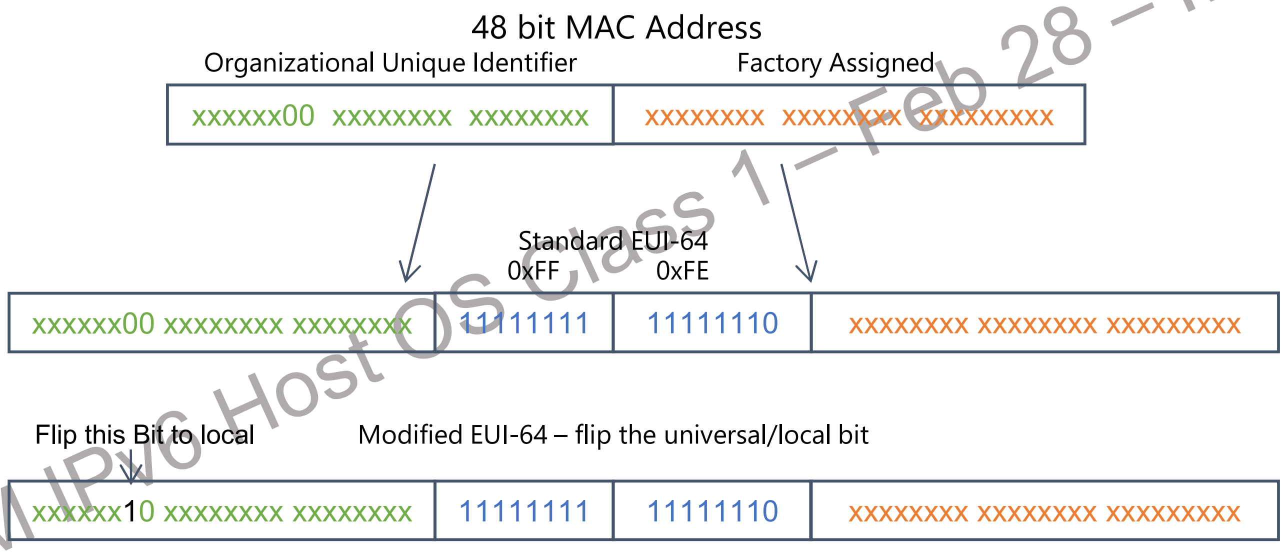
If the Managed Address Configuration Flag (“M” flag) = 0

If the Other Stateful Configuration Flag (“O” flag) = 0

Then host then uses the prefix information from the RA for the network portion of the node’s IPv6 address

- Statelessly configures its own Interface Identifier (Interface ID)
- Using either EUI-64, privacy, or temporary addressing

EUI-64 Interface ID



SLAAC Cisco Router Configuration Example

```
ipv6 unicast-routing
ipv6 multicast-routing
ipv6 cef
!
interface GigabitEthernet 0/2
 description Dual-Protocol, RA A=1,L=1,M=0,O=0
 ip address 192.168.123.1 255.255.255.0
 ipv6 address 2001:db8:1234:5678::1/64
! Or
 ipv6 address 2001:db8:1234:5678::/64 eui-64
!
show ipv6 interface
show ipv6 neighbors
show ipv6 routers
```

Recap IPv6 Host Startup

Host sends ICMPv6 type 135 NS for DAD of its link-local address

If this succeeds (no ICMPv6 type 136 NA reply) then it sends an ICMPv6 type 133 RS message to all routers (ff02::2)

The router sends the ICMPv6 type 134 RA message to all nodes (ff02::1)

The host interprets the A/M/O/L bits and proceeds with address configuration (SLAAC or DHCPv6)

IPv6 Addressing Methods

Servers/appliances in data centers or DMZ:
- Statically assigned
- Stateful DHCPv6 – very rare cases and only by approval
- SLAAC – never allowed

Host based systems on internal Networks:
- Stateful DHCPv6
- Stateless DHCPv6
- Full Statically assigned – must be obtained from appropriate authority (define the authority(ies)). DNS and other critical items are manually assigned
- SLAAC – only if no other means is supported.
- Privacy addresses are not allowed at all.

Other IPv6 Features

IPv6 requires every network link be capable of minimum MTU of 1280 bytes

IPv6 routers don't fragment packets
- Hosts perform their own Path MTU Discovery
- MTU path discovery uses ICMP "packet too big" error messages

Provider selection (based on policy, performance, cost, …)

Host mobility (route to current location)

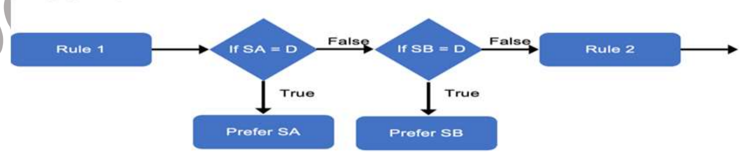
Auto-readdressing (route to new address)
- (Use IPv6's routing extension header)

IPv6 Address Selection

RFC 6724 Pairwise Sorting

Source Address Selection

1. Prefer same address
2. Prefer appropriate scope
3. Avoid deprecated addresses
4. Prefer home addresses
5. Prefer outgoing interface
6. Prefer matching label
7. Prefer privacy addresses
8. Use longest matching prefix

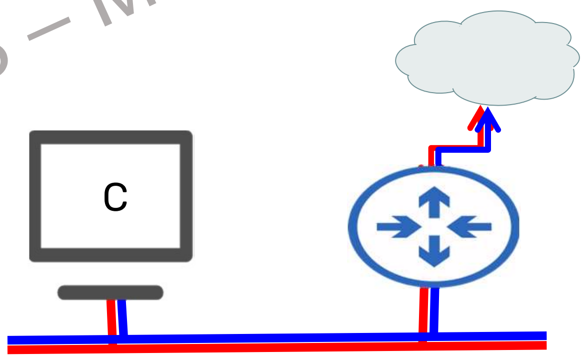


Destination Address Selection

1. Avoid unusable destinations
2. Prefer matching scope
3. Avoid deprecated addresses
4. Prefer home addresses
5. Prefer matching label
6. Prefer higher precedence
7. Prefer native transport
8. Prefer smaller scope
9. Use longest matching prefix
10. Otherwise, leave the order unchanged

RFC 6724 – Source Address Selection

- IPv6 over IPv4, but which IPv6 first..
- Scope = Local, ULA, Global
- State = Preferred over Deprecated
- Interface = Assigned vs. another
- Privacy = Temporary over Public

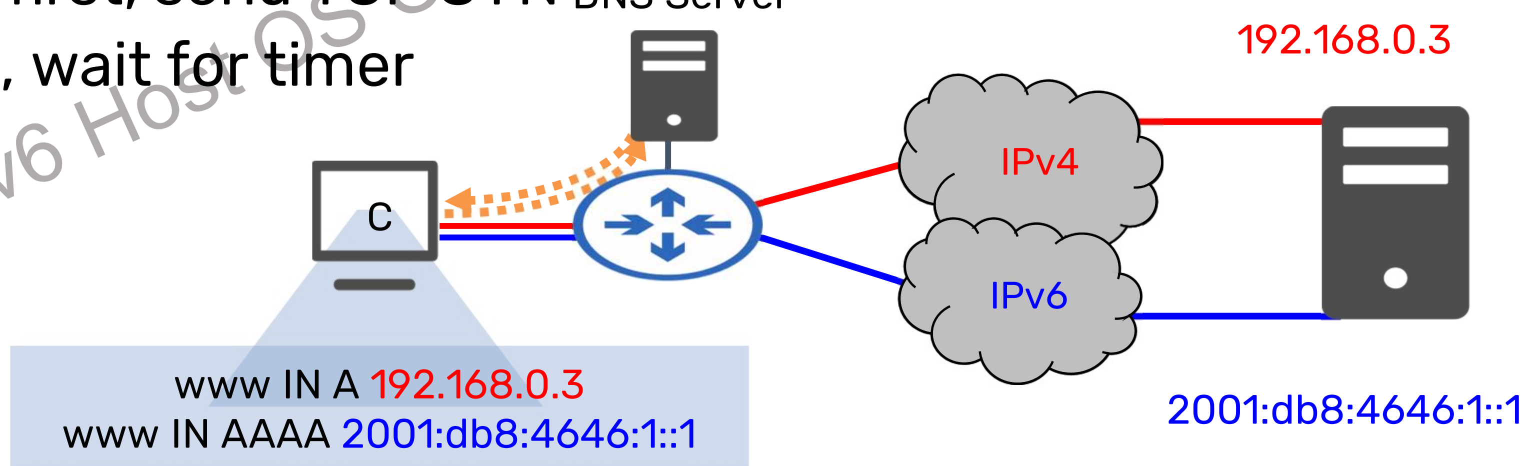


Public	Preferred	2001:db8:4646:1:4f02:8a34:bead:a136	Idx
Temporary	Preferred	2001:db8:4646:1:bd86:ea49:41f1:39c1	Idx
Link	Preferred	fe80::4f02:8a34:bead:a136	

Dual Stack OS Consideratior

RFC ~~6555~~ 8305
Happy Eyeballs

- In a dual stack case, an application can:
  - Query DNS (AAAA/A) over **IPv6** then **IPv4**
- Resolution delay (50ms) IPv6 head start
  - AAAA first, send TCP SYN
  - A first, wait for timer



Address, Which Address?

- Link Local (fe80::/10) is required for any device with IPv6 enabled
- At least 2 IPv6 addresses per interface for global connectivity
- Majority of access layer devices will have LL as their DfGw

Host	Addresses	Router	Addresses
Ethernet	B8:E8:56:1A:2B:3C	Ethernet	00:00:0C:3A:8B:18
IPv6 Link Local	fe80::bae8:56ff:fe1a:2b3c	IPv6 Link Local	fe80::0200:0cff:fe3a:8b18
IPv6 Global	2001:db8:1:46:1b2:c:3:4e5	IPv6 Global	2001:db8:1:46::1
Default Gwy.	fe80::0200:0cff:fe3a:8b18	RA Prefix	2001:db8:1:46::/64

Transition and Translation

IPv6 Transition Techniques

Dual Stack

Tunnel/Encapsulation
- Configured Tunnels
- Automatic Tunnels (Dynamic Tunneling)
    - 6to4
    - ISATAP
    - Teredo
    - Tunnel Broker with TSP

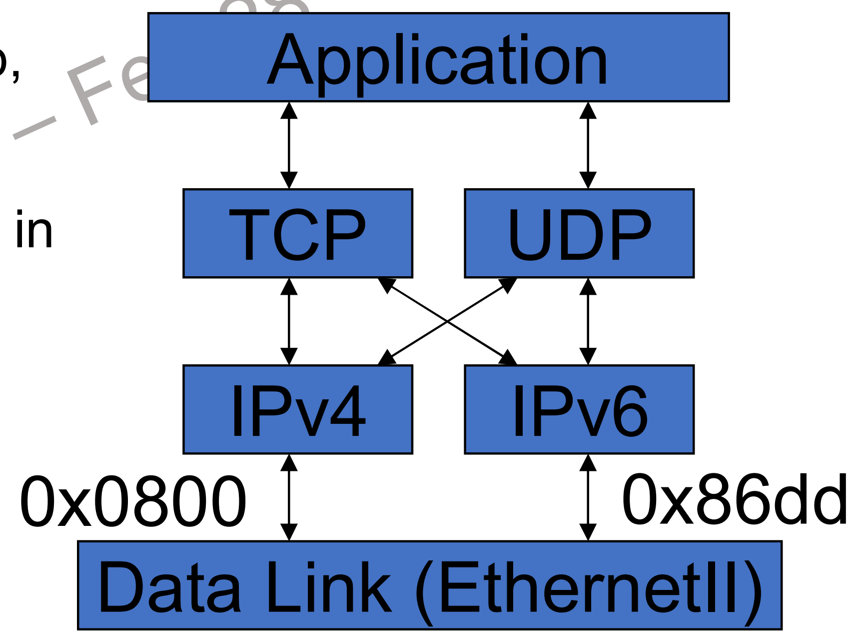
Translation, Application Layer Gateways
- Proxy
- NAT-PT (deprecated), NAT64/DNS64, NPT

Dual-Stack Model

Dual-Stack Architecture – RFC 1933

Choice of the IP version is based on name lookup, application or operating system preference

IPv4 and IPv6 packets flow in Ethernet like “ships in the night”



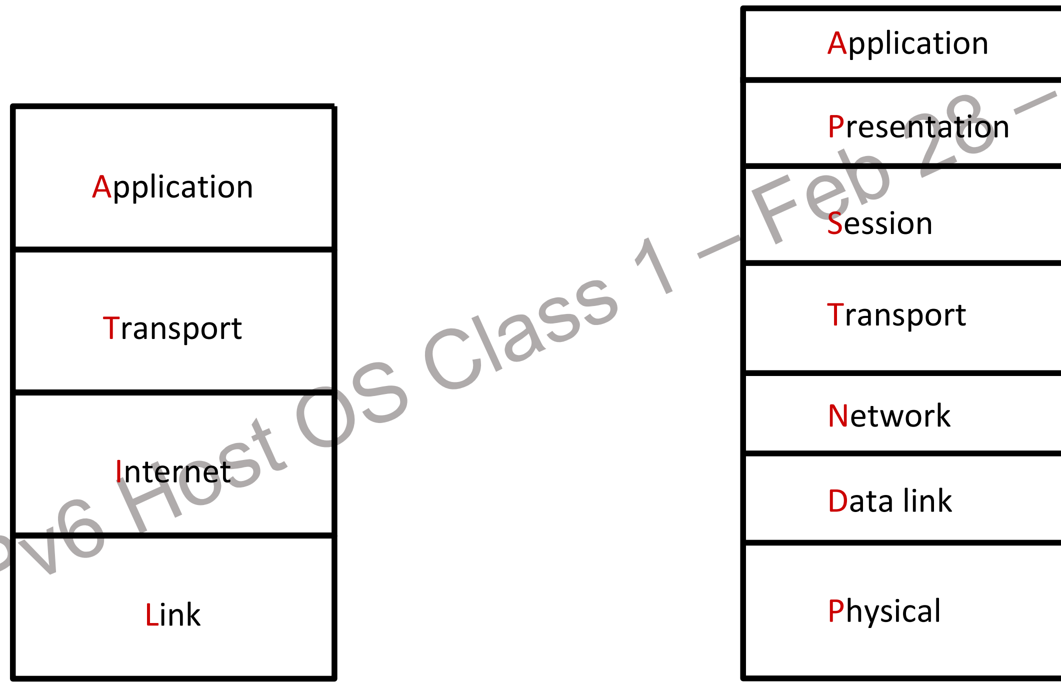
Node-to-Node Communication

For two nodes to communicate they must support one common protocol

An IPv4-only node cannot communicate with an IPv6-only node

	IPv4-Only	Dual Protocol	IPv6-Only
IPv4-Only	Yes (IPv4)	Yes (IPv4)	No
Dual Protocol	Yes (IPv4)	Yes (IPv6, IPv4)	Yes (IPv6)
IPv6-Only	No	Yes (IPv6)	Yes (IPv6)

TCP/IP Verses OSI Model

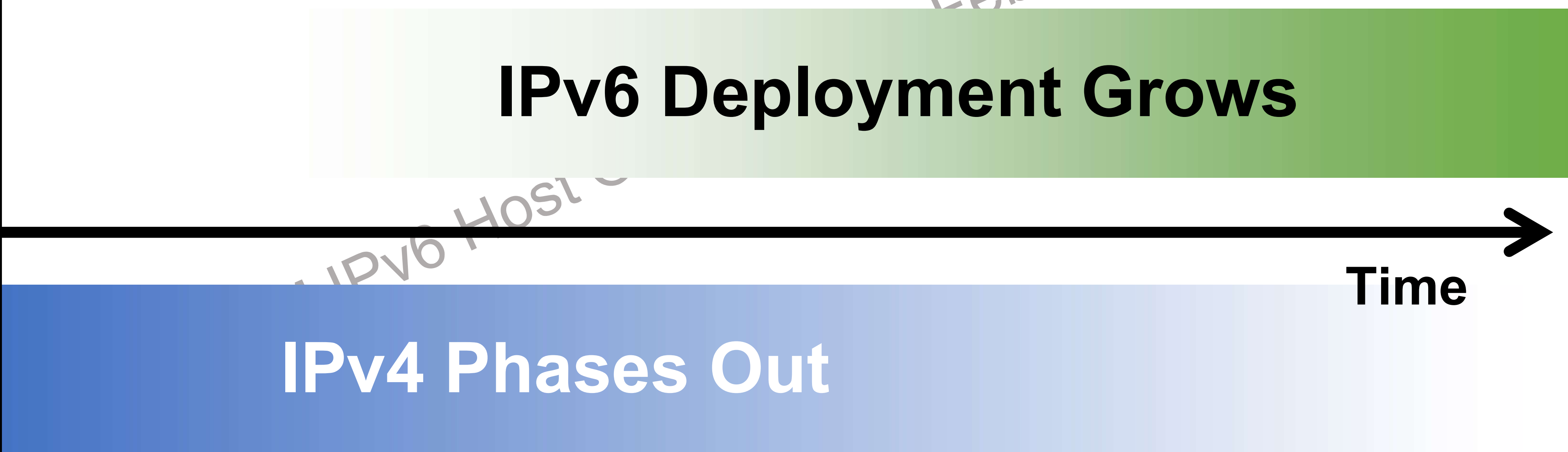


TCP/IP Model

OSI Model

Dual-Stack Transition

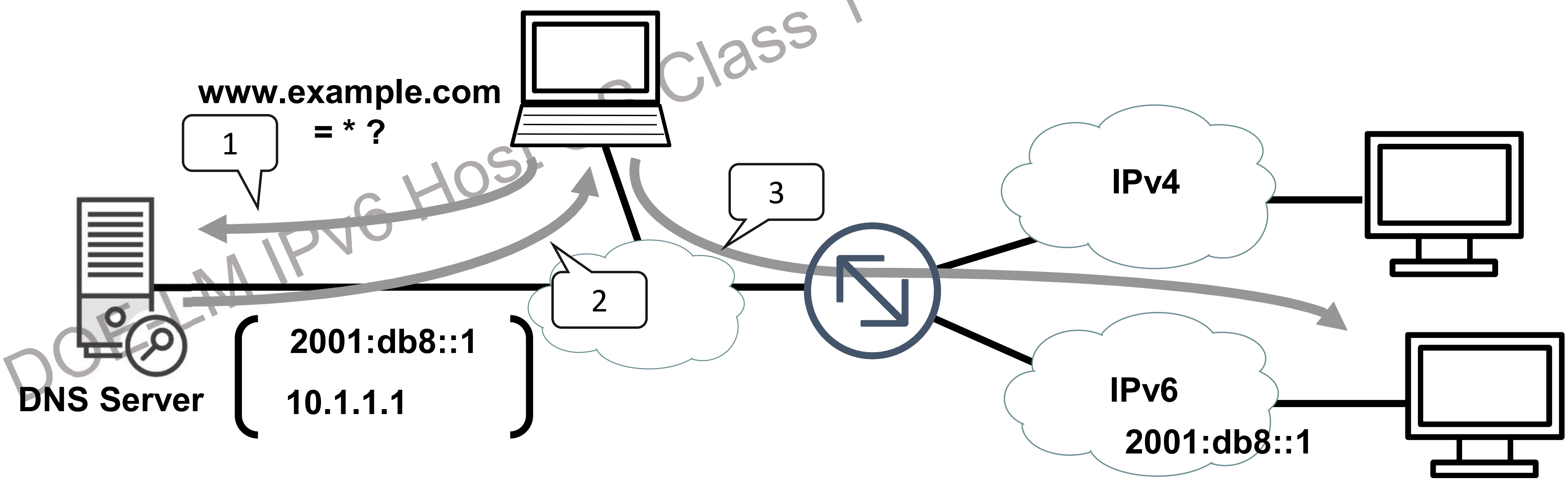
Organizations use IPv4 today and will add IPv6 as a separate protocol, run them in parallel for many years, and after many years, start to disable IPv4



Dual-Stack Behavior

In a Dual Stack Case, an Application that:

- Is IPv4 and IPv6-enabled
- Asks the DNS for all types of addresses
- Chooses one address and, for example, connects to the IPv6 address



Simple Cisco IOS Dual-Stack Router Example

```
ipv6 unicast-routing
ipv6 multicast-routing
ipv6 cef
!
interface Loopback0
 ip address 200.100.1.3 255.255.255.255
 ipv6 address 2001:db8:10:10::10/128
!
interface Ethernet 0
 ip address 192.168.100.1 255.255.255.0
 ipv6 address 2001:db8:1:1::1/64
!
ipv6 route ::/0 2001:db8:1:1::100
```

Simple JunOS Dual-Stack Router Example

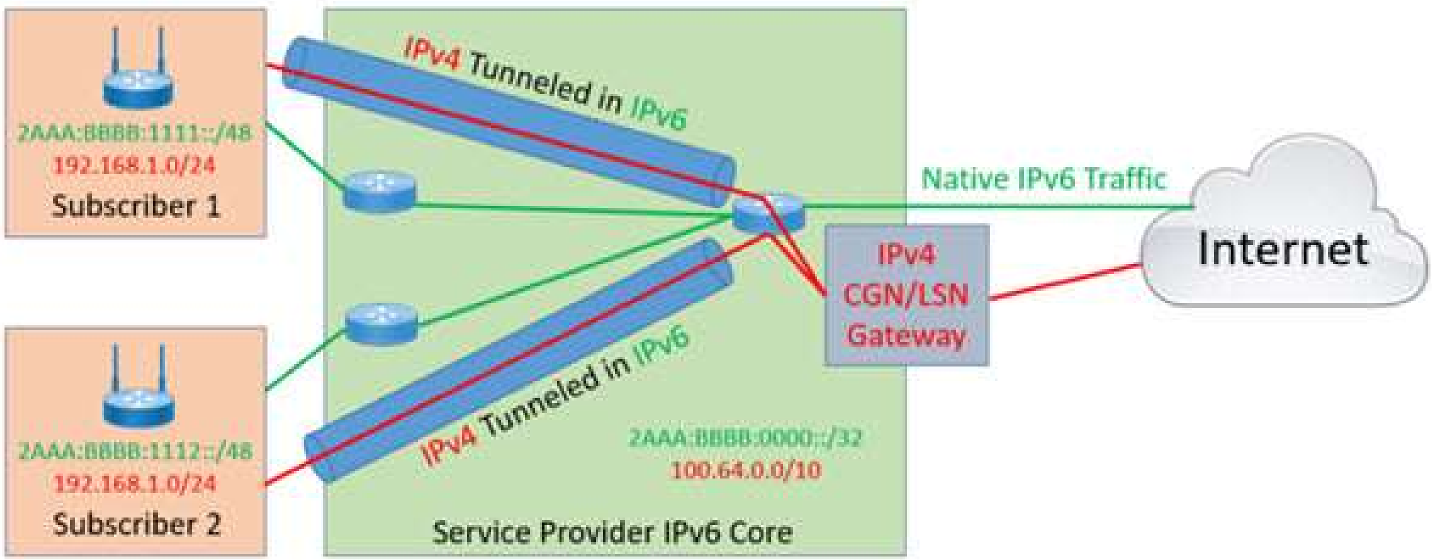
```
interfaces {
    ge-0/0/0 {
        unit 0 {
            family inet {
                address 192.168.4.2/24;
            }
            family iso {
                mtu 1492;
            }
            family inet6 {
                address 2001:db8:4::2/64;
            }
        }
    }
    ge-0/0/1 {
        unit 0 {
            family inet {
address 192.168.64.1/24;
            }
            family iso {
                mtu 1492;
            }
            family inet6 {
                address
2001:db8:64::1/64;
            }
        }
    }
}
```

Simple JunOS Dual-Stack Router Example Cont

```
protocols {
    router-advertisement {
        interface ge-0/0/0.0 {
            prefix 2001:db8:4::/64;
        }
        interface ge-0/0/1.0 {
            prefix 2001:db8:64::/64;
        }
        interface ge-0/0/3.0 {
            prefix 2001:db8:5::/64;
        }
    }
```

IPv4 As A Service (v4aaS)

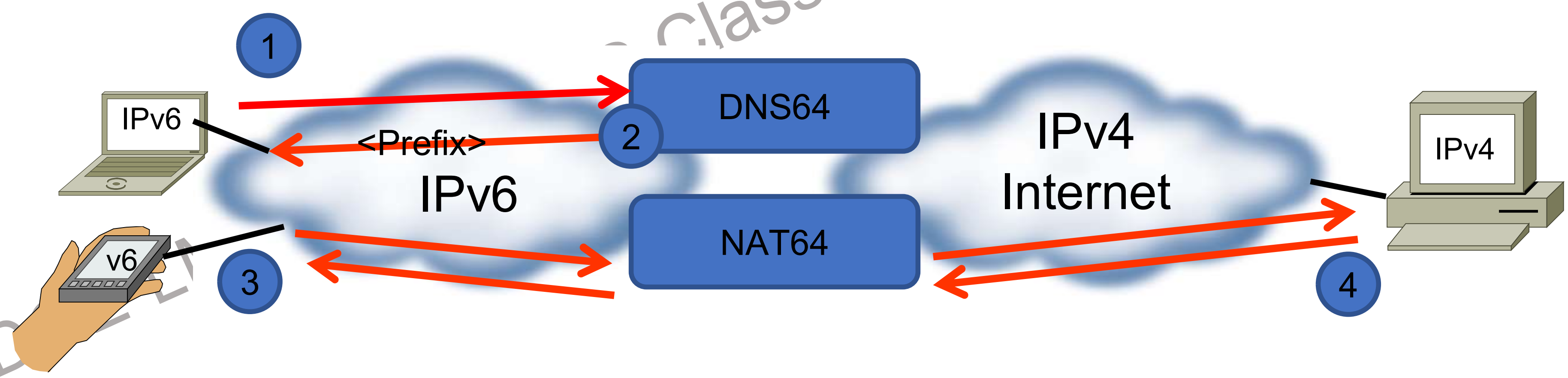
IPv4 packets are tunneled over an IPv6-only core network through an edge gateway so NAT44 is only performed once



https://community.infoblox.com/t5/IPv6-Center-of-Excellence/IPv4-as-a-Service-IPv4aaS-Part-1/ba-p/8279
https://community.infoblox.com/t5/IPv6-Center-of-Excellence/Methods-of-Providing-IPv4-as-a-Service-IPv4aaS-Part-2/ba-p/8744
https://community.infoblox.com/t5/IPv6-CoE-Blog/Even-More-Methods-of-Providing-IPv4-as-a-Service-IPv4aaS-Part-3/ba-p/8986

NAT64 and DNS64

Translation that uses NAT and DNS in combination
Uses an IPv6 prefix to map the 32-bit IPv4 DNS responses into an IPv6 address
DNS64 (RFC 6147) and NAT64 (RFC 6146) “appear” to share state information about connections but they do not directly share state information



http://www.networkworld.com/article/2231256/cisco-subnet/testing-nat64-and-dns64.html

NAT64/DNS64 Drawbacks

NAT64 only works for applications that start with a DNS lookup

Breaks with applications that embed an IPv4 address in the URL (embedded literals) (e.g. http://192.168.1.1)

Breaks for domains that have IPv4-only DNSSEC signed domains

~15% of applications break with IPv6 native or break with NAT64
- Spotify, WhatsApp, Skype, SIP, RTSP, H.323, XMPP peer to peer
- http://tinyurl.com/nat64-breakage

NetFlix blocks 6in4 tunnels to reduce geounblocking

NAT64Check2

NAT64Check is an ISOC-developed tool to validate dual-protocol and NAT64 reachability of a web site.

Browse to the nat64check site, enter the URL that you want it to test
https://www.nat64check.org/

https://www.internetsociety.org/blog/2019/01/nat64check-version-2-is-launched/

NAT64/DNS64 Systems

Systems that can perform NAT64 and DNS64
- Internet Systems Consortium (ISC) Address Family Transition Router (AFTR)
- Viagenie Ecdysis LiveCD
- Microsoft Forefront UAG SP1 Direct Access

Systems that can perform DNS64
- ISC BIND 9.8
- Infoblox
- Secure64
- Trick or Treat Daemon (TOTD)

DNS64/NAT64 Services
- TREX (Tampere Region Exchange – Finland)
- LITNET (Lithuania)

NAT64/DNS64 Systems

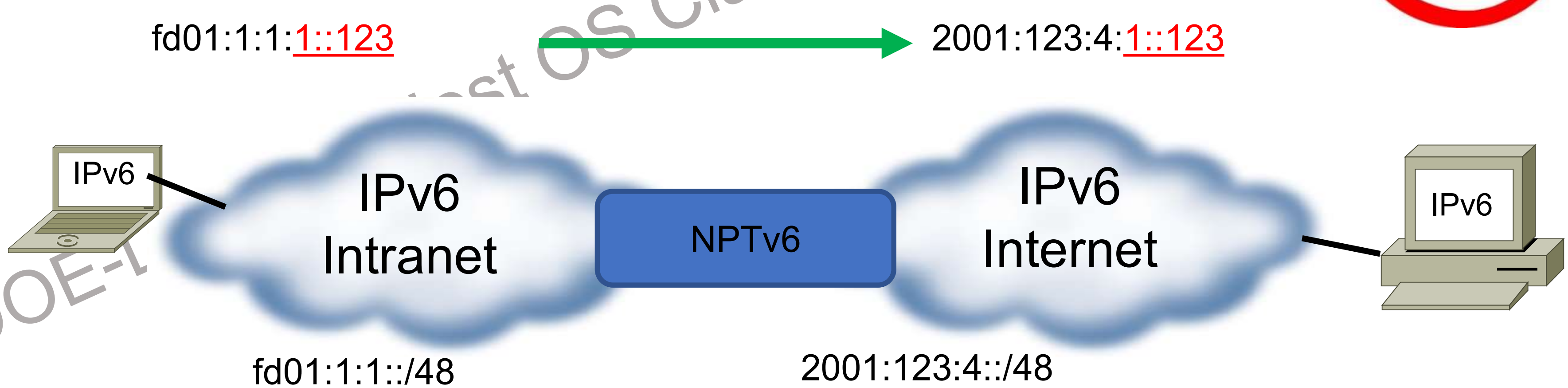
Systems that can perform Stateless NAT64

- Cisco Routers (IOS-XE and CGSE with IOS-XR)
  - Stateless NAT64, IOS XE 3.2S, IOS XR 3.9.3
  - Stateful NAT64, IOS XE 3.4S, IOS XR 4.1.2
- Juniper Routers, MX-series 3D, Juniper SSG 6.2.0R7
- Brocade ServerIron ADX
- A10 Networks AX Series Application Delivery Controller
- F5 Networks BIG-IP LTM
- Fortinet FortiGate
- Citrix NetScaler 9.2
- Cisco ACE (with optional NAT64 license)
- TAYGA (Linux)
- OpenBSD 5.1 (PF packet filter)
- http://ipv6.lt/nat64_en.php (public NAT64 service LITNET)

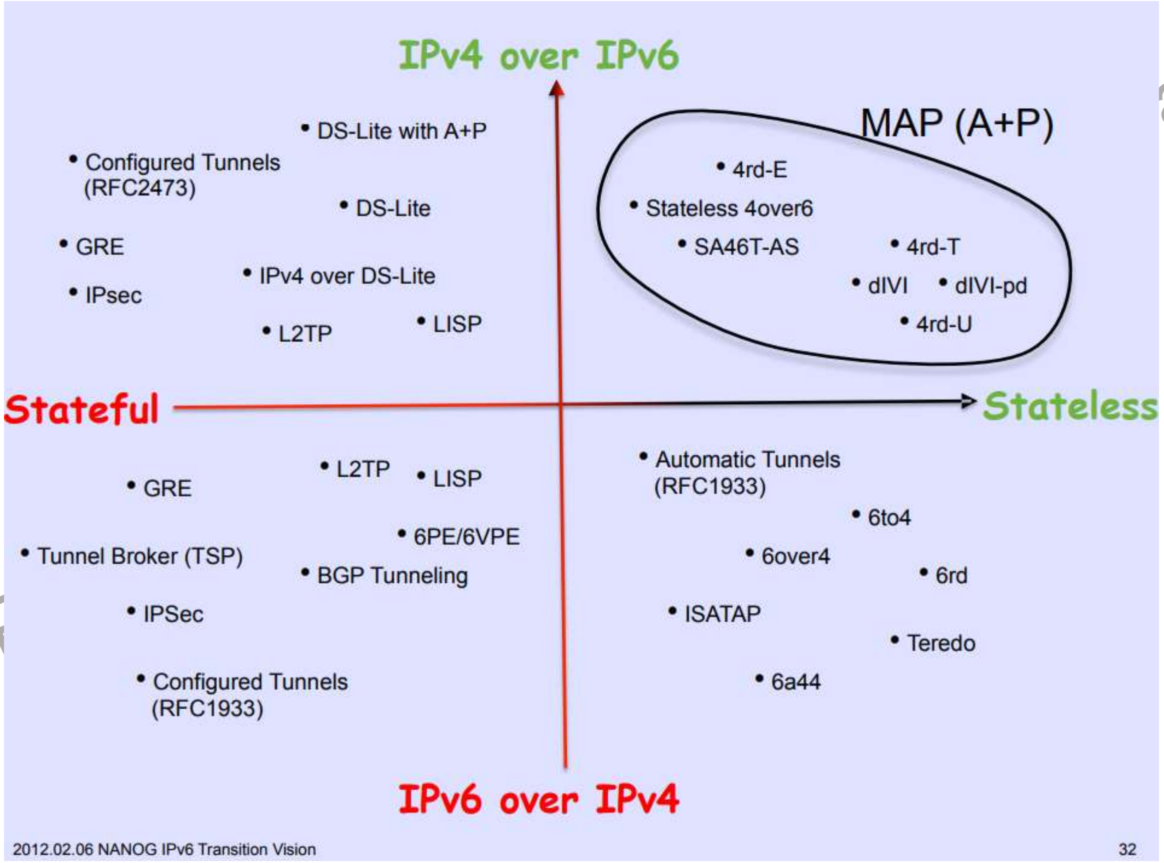
NAPT/NPT/NPTv6 (NAT66)

Network Prefix Translation (NPT) (RFC 6296)

Translates source address from one prefix to another prefix
Rest of IPv6 address remains
1:1 mapping – stateless
Few products support this today



IPv6 Transition Universe



Source: NANOG54 Randy Bush - https://www.nanog.org/meetings/nanog54/presentations/Monday/Bush.pdf

Comparison of IPv6-Only

	6RD	Softwires v2	NAT444	DS-Lite	Lw4o6	NAT64	464XLAT	MAP-E	MAP-T
Tunel/Translation (X)	T 6in4	T 6in4	X	T 4in6	T 4in6	X	X	T 4in6	X
Dual-stack LAN	YES	YES	optional	YES	YES	NO	YES	YES	YES
IPv4 Multicast	YES	YES	YES	NO	NO	NO	NO	NO	NO
Access Network	IPv4	IPv4	IPv4 /dual	IPv6	IPv6	IPv6	IPv6	IPv6	IPv6
Overhead	20 bytes	40 bytes	-	40 bytes	40 bytes	20 bytes	20 bytes	40 bytes	20 bytes
Impact in IPv6 addressing plan	YES	NO	NO	NO	NO	NO	NO	YES	YES
CPE Update	YES	YES	optional	YES	YES	YES	YES	YES	YES
NAT44/NAPT	CPE	CPE	CPE + CGN	CGN	CPE	CPE	CPE	CPE	CPE
46/64 Translation	-	-	-	-	-	ISP	ISP +/or CPE	-	CPE + ISP
Translation at ISP with or w/o state	-	-	with	-	-	with	with	w/o	w/o
Scalability	High	Medium	Medium	Medium	High	High	High	High	High
Performance	High	Low	Low	Low	High	Medium	High	High	High
ALGs	NO	NO	YES	YES	NO	YES	YES	YES	YES
Any Protocol or only-TCP/UDP/ICMP	YES	YES	YES	YES	YES	NO	NO	NO	NO
Sharing IPv4 Ports	NO	NO	YES	YES	YES	NO	NO	YES	YES
IPv6 Aggregation	NO	NO	optional	YES	YES	YES	YES	YES	YES
IPv4 Mesh	YES	YES	YES	NO	NO	NO	NO	YES	YES
IPv6 Mesh	YES	NO	optional	YES	YES	YES	YES	YES	YES
Impacts on logging	NO	NO	YES	YES	NO	YES	YES	NO	NO
HA simplicity	High	Low	Low	Low	High	Medium	High	High	High
DPI simplicity	Low	Low	High	Low	Low	High	High	Low	High
Support in cellular	NO	NO	YES	NO	NO	YES	YES	NO	NO
Support in CPEs	YES	YES	YES	YES	YES	YES	YES	YES	YES
	15.5	12.5	10.5	9.5	15	12	14	13	13.5

Source: https://www.slideshare.net/apnic/tutorial-ipv6only-transition-with-demo

IPv6-Only

This is not easily achievable because most of the content on the Internet is IPv4-only. This would require NAT64/DNS64 to access but only applications that use DNS may work.

There are many systems in environments that are IPv4-only

- Game consoles (PS3, Xbox 360/Live, Wii)
- Printers, Tivo, BlueRay, IP cameras
- UPSs, KVMs, iLO
- Windows XP/Win2k3 use IPv4 for DNS queries

RFC 6586 - Experiences from an IPv6-Only Network

RFC 7404 - Using Only Link-Local Addressing inside an IPv6 Network

Benefits of IPv6-Only

- Reduced OPEX costs by running only a single IP protocol
- IPv6 addressing (operations) is simpler
  - No NAT makes everything better
- Reduced dependence on increasingly expensive IPv4 addresses
  - If you know you are going to need more IPv4, buy it now
  - Sell your public IPv4 at the peak price
- No need to purchase and maintain CGN/LSN systems
- In some cases IPv6 performs better than IPv4

IPv6-Only Environments

- Facebook data centers
  https://code.fb.com/production-engineering/legacy-support-on-ipv6-only-infra/
- LinkedIn data centers
  https://engineering.linkedin.com/blog/topic/ipv6
- Microsoft
  http://www.rmv6tf.org/wp-content/uploads/2017/04/02-MicrosoftIT_IPv6-NA-IPv6-Summit-2017_VMcKillop-min.pdf
  https://labs.ripe.net/Members/mirjam/microsoft-making-progress-towards-ipv6-only
- Cisco – Building 23
  http://www.rmv6tf.org/wp-content/uploads/2017/04/06-IPV6-Summit-Cisco-min.pdf

IPv6-Only RA

IPv6 Router Advertisement IPv6-Only Flag

https://tools.ietf.org/html/draft-ietf-6man-ipv6only-flag-05

Will need implementation support in RA-Guard, otherwise someone could set this proposed S flag on an IPv4 only network and kill ALL network traffic

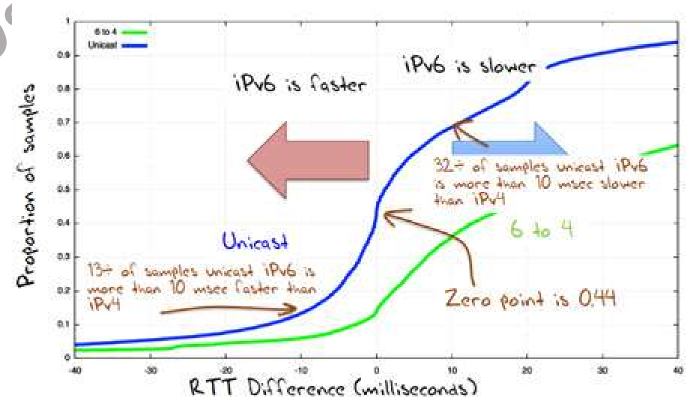
Keep an eye on this one, my guess is that it will be move to a full RFC, it is currently in draft status

I expect major OS's to implement supporting it within 6mo to a year from when it is accepted

Is IPv6 Faster Than IPv4?

- There are now several studies analyzing if IPv6 is faster than IPv4.
  - Google's 2010 paper titled "Evaluating IPv6 Adoption in the Internet"
- Geoff Huston of APNIC at NANOG 66
  - 6to4 and Teredo are responsible for most of the connection failures
  - He concluded that native IPv6 can be as-fast as IPv4

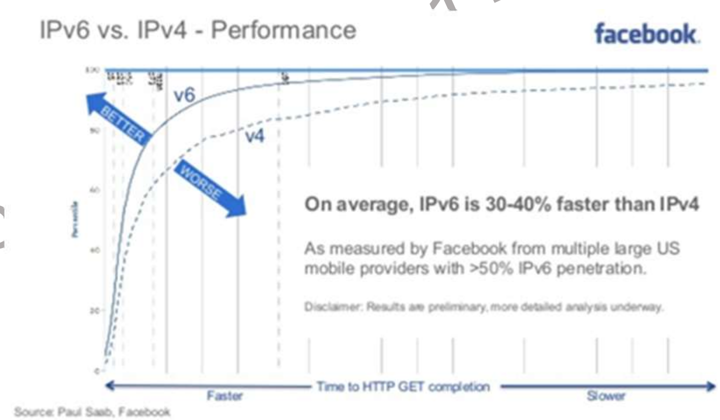
2015/6 RTT Data CDF



Is IPv6 Faster Than IPv4?

- Paul Saab at Facebook has shows data from Mobile Proxygen that shows IPv6 is faster for them.

"Facebook says it has seen users' News Feeds loading 20 percent to 40 percent faster on mobile devices using IPv6".



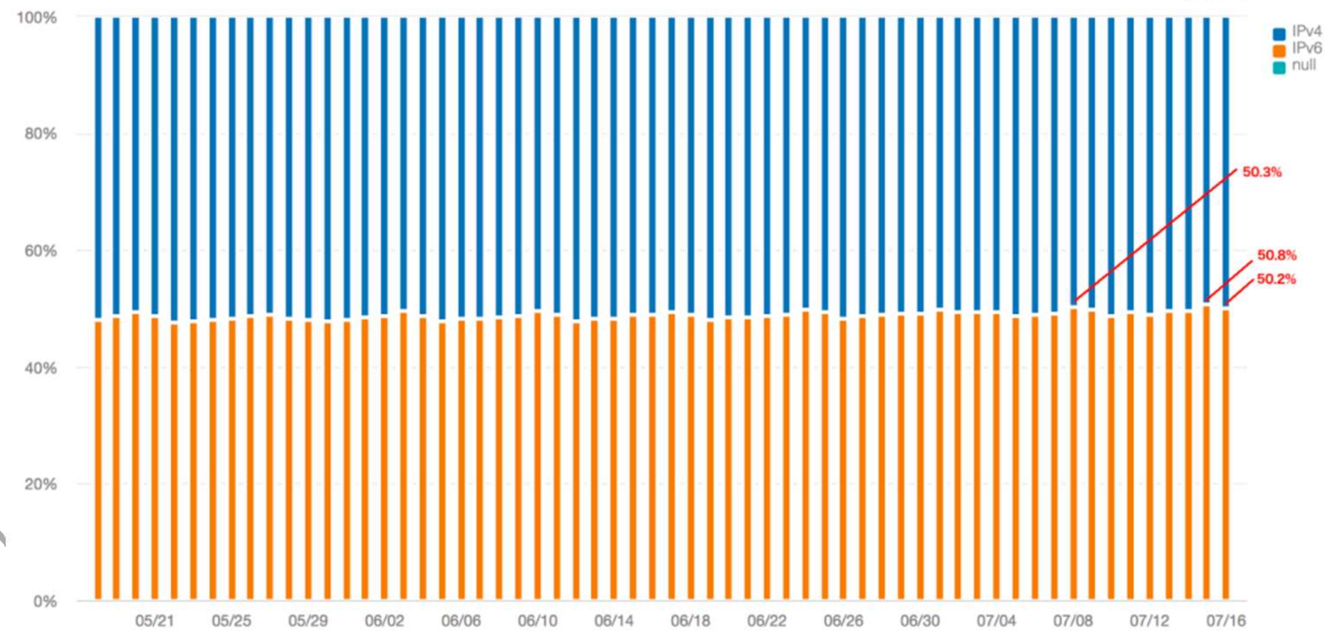
- Hurricane Electric Global IPv6 Deployment Progress Report

"Percentage of IPv6 rDNS Nameservers where IPv6 is as fast or faster than IPv4 (within 1ms): 74.9%"

Adoption Status

LinkedIn IPv6 Traffic is Greater Than 50%

- LinkedIn IPv6 traffic passed 50%



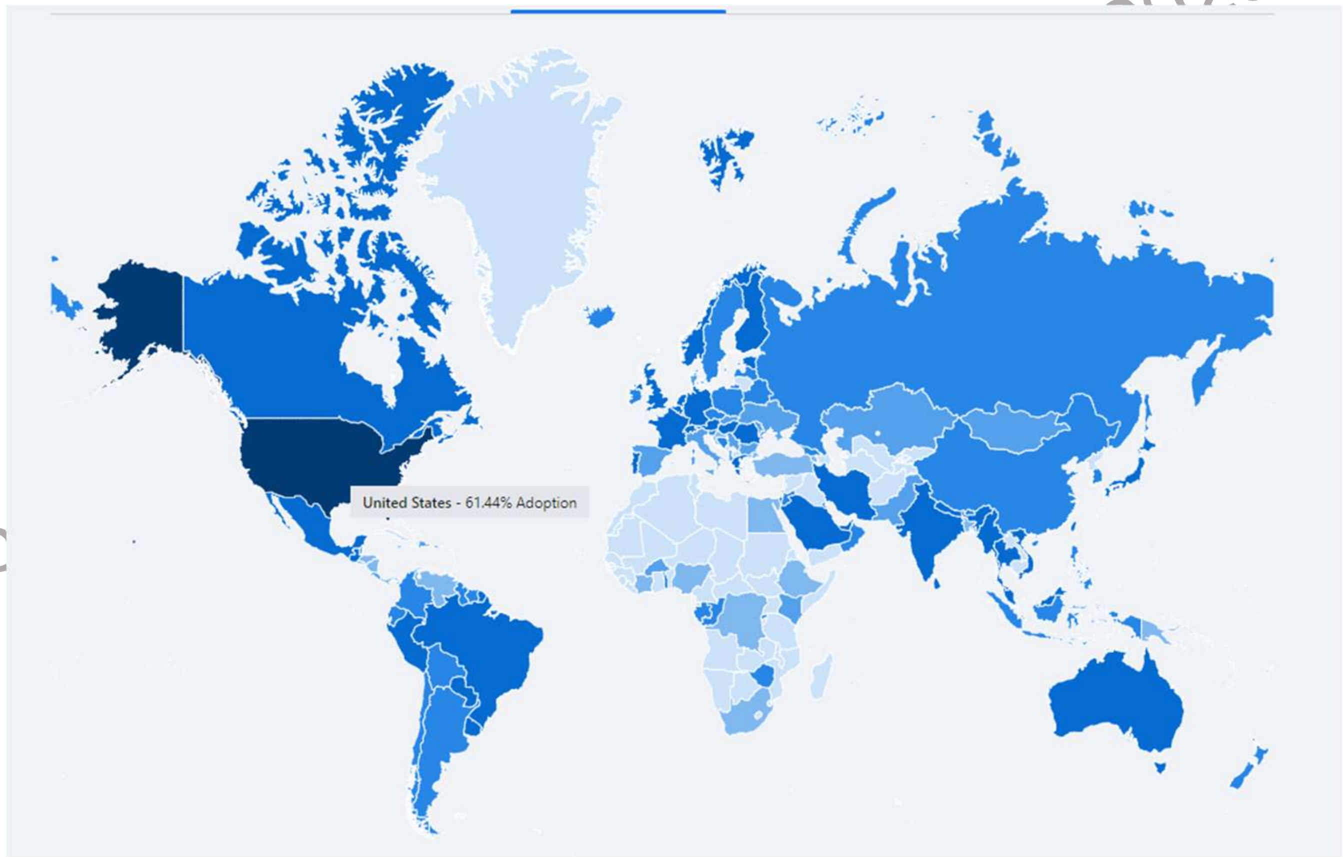
- Percentage of page views over IPv6 for LinkedIn over the last two months
  https://engineering.linkedin.com/blog/2017/07/linkedin-passes-ipv6-milestone

Facebook Traffic is Greater Than 50%

- Passed 50% IPv6 traffic on May 26, 2018

Source: https://code.fb.com/connectivity/how-ipv6-deployment-is-growing-in-u-s-and-other-countries/

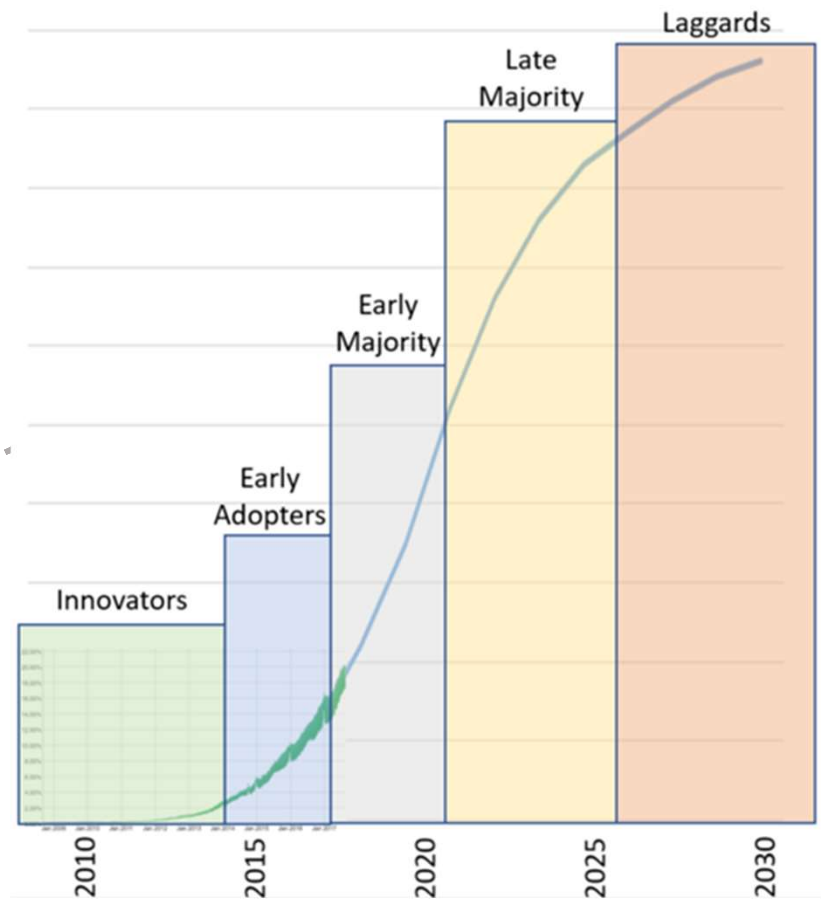
Facebook Traffic



Source: https://www.facebook.com/ipv6/?tab=ipv6

IPv6 Adoption Curve

Organizations adopt IPv6 based on the "diffusion of innovations"



https://community.infoblox.com/t5/IPv6-CoE-Blog/Where-Are-You-On-The-IPv6-Adoption-Curve/ba-p/11116
https://community.infoblox.com/t5/IPv6-CoE-Blog/Life-as-an-IPv6-Technology-Laggard/ba-p/11989

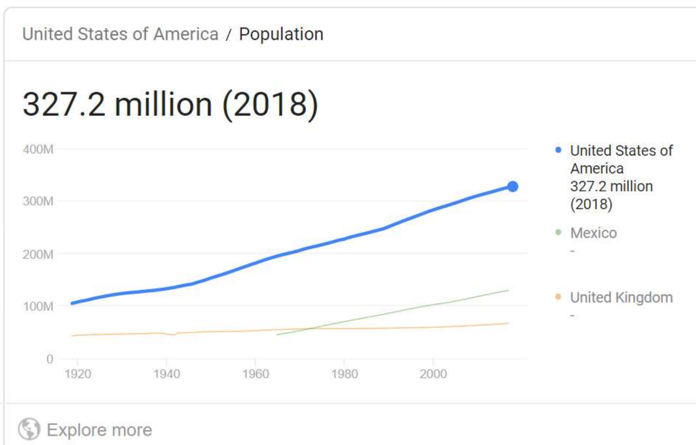
North America 2019

Akamai Experience the Edge

RANK	IPV6%	COUNTRY
1	61.9%	India
2	56.6%	Saint Barthelemy
3	47.3%	Belgium
4	45.3%	United States

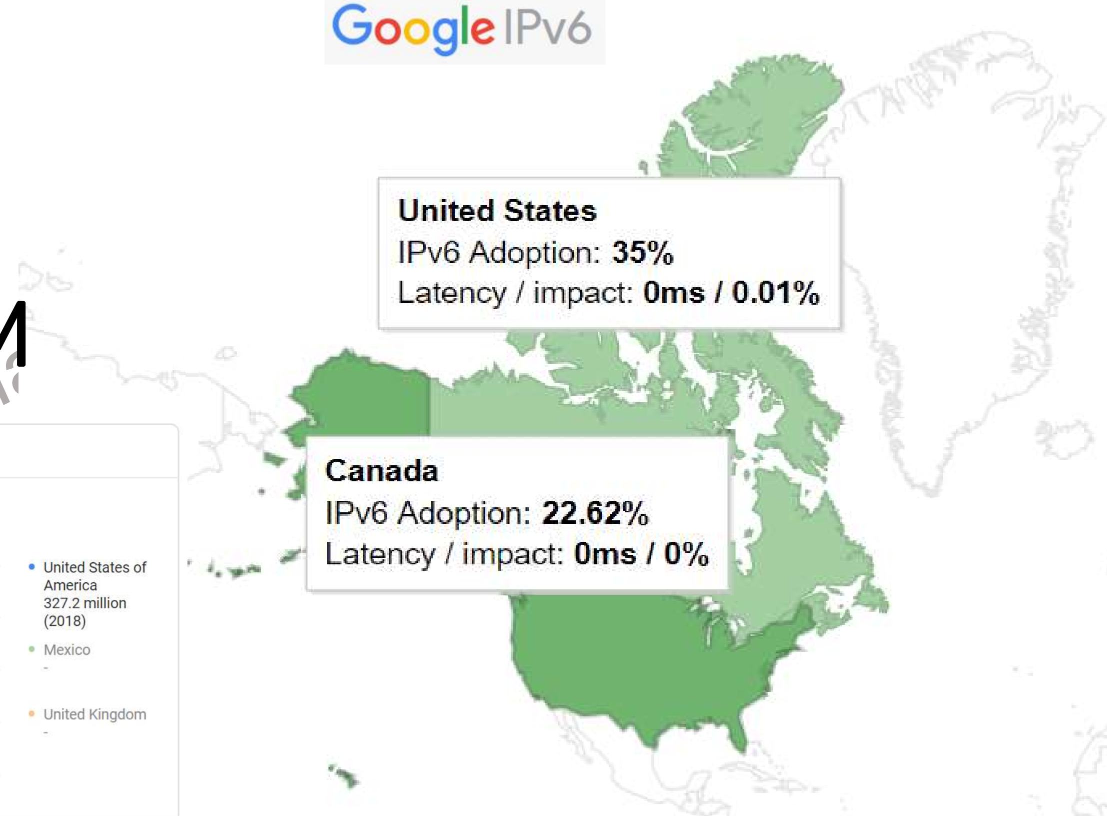
148M

Google



Sources include: United States Census Bureau, World Bank, ONS UK
Feedback

Google IPv6



Worldwide - 2023

Akamai
Experience the Edge

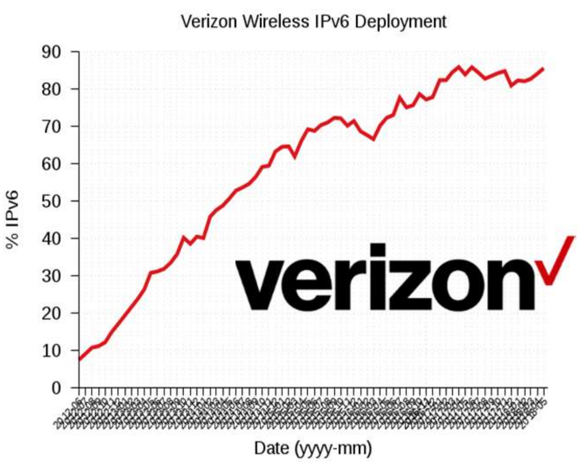
*Country data ranked by % of IPv6 connections from that country.

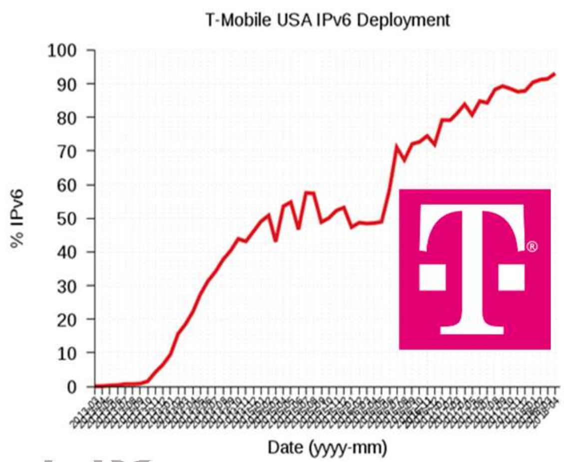
RANK	IPV6%	COUNTRY / REGION
1	100%	Bahrain
2	58.4%	Saudi Arabia
3	58%	Montserrat
4	55.2%	Uruguay
5	54.8%	Germany
6	54.6%	France
7	54.2%	Malaysia
8	52.1%	United States
9	51.4%	Belgium
10	50.9%	Greece

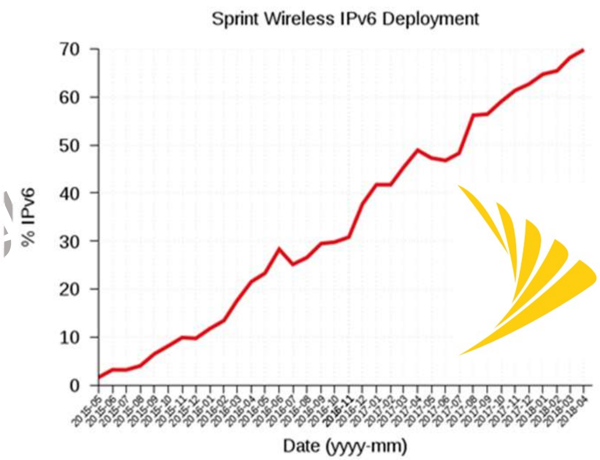
Content and Cloud Providers

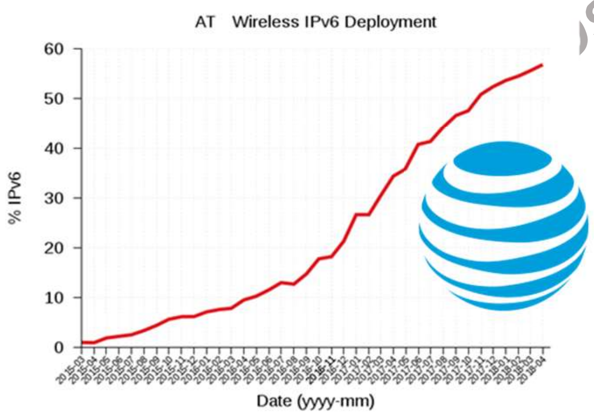
- Akamai – Experience the Edge
- Cloudflare
- Google Cloud
- aws
- DigitalOcean
- IBM Cloud
- Limelight Networks
- Azure
- fastly
- Oracle Cloud *Soon

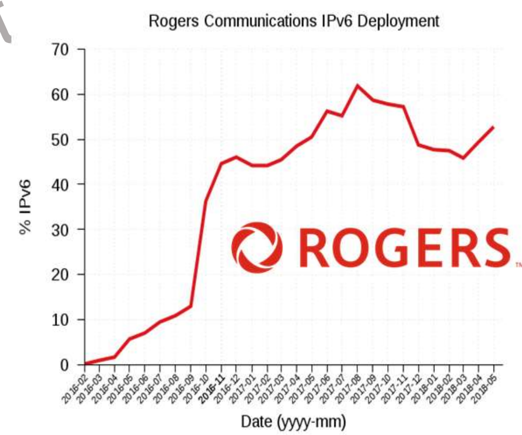
Mobile and Cable

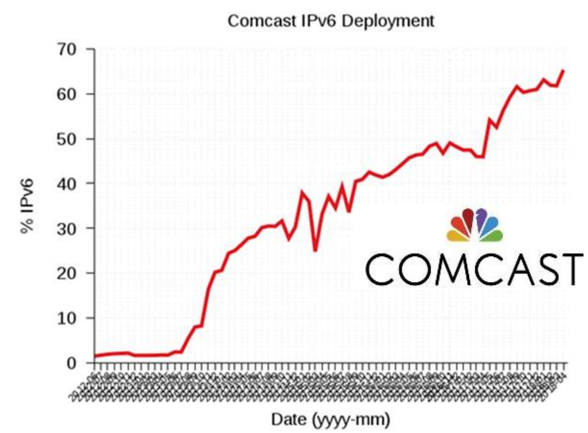








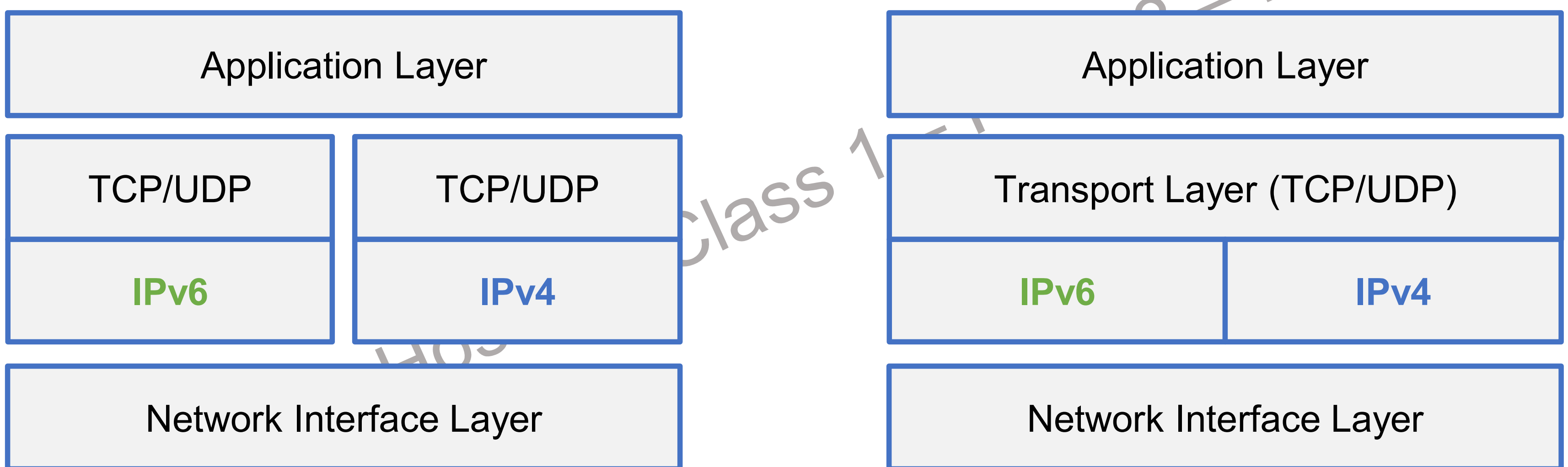




Windows Networking Stack

Windows Networking

Since Windows Vista and Server 2008R2 there has been a new OS networking stack



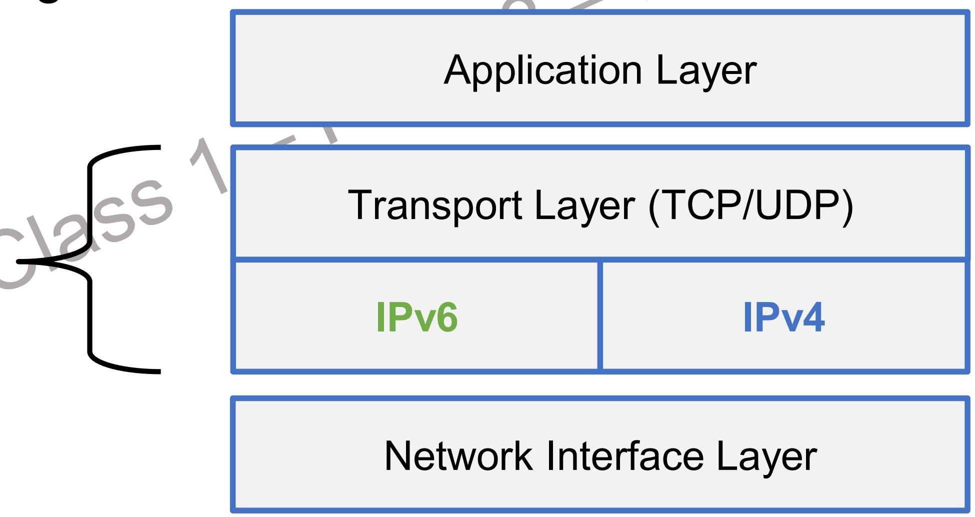
Older TCP/IP Networking Stack (Dual-Stack)
Prior to Vista and Server 2008R2

New TCP/IP Networking Stack (Dual IP Layer)
Vista, Win7/8/10 Server 2012/R2/2016/2019

Windows IPv6 Networking

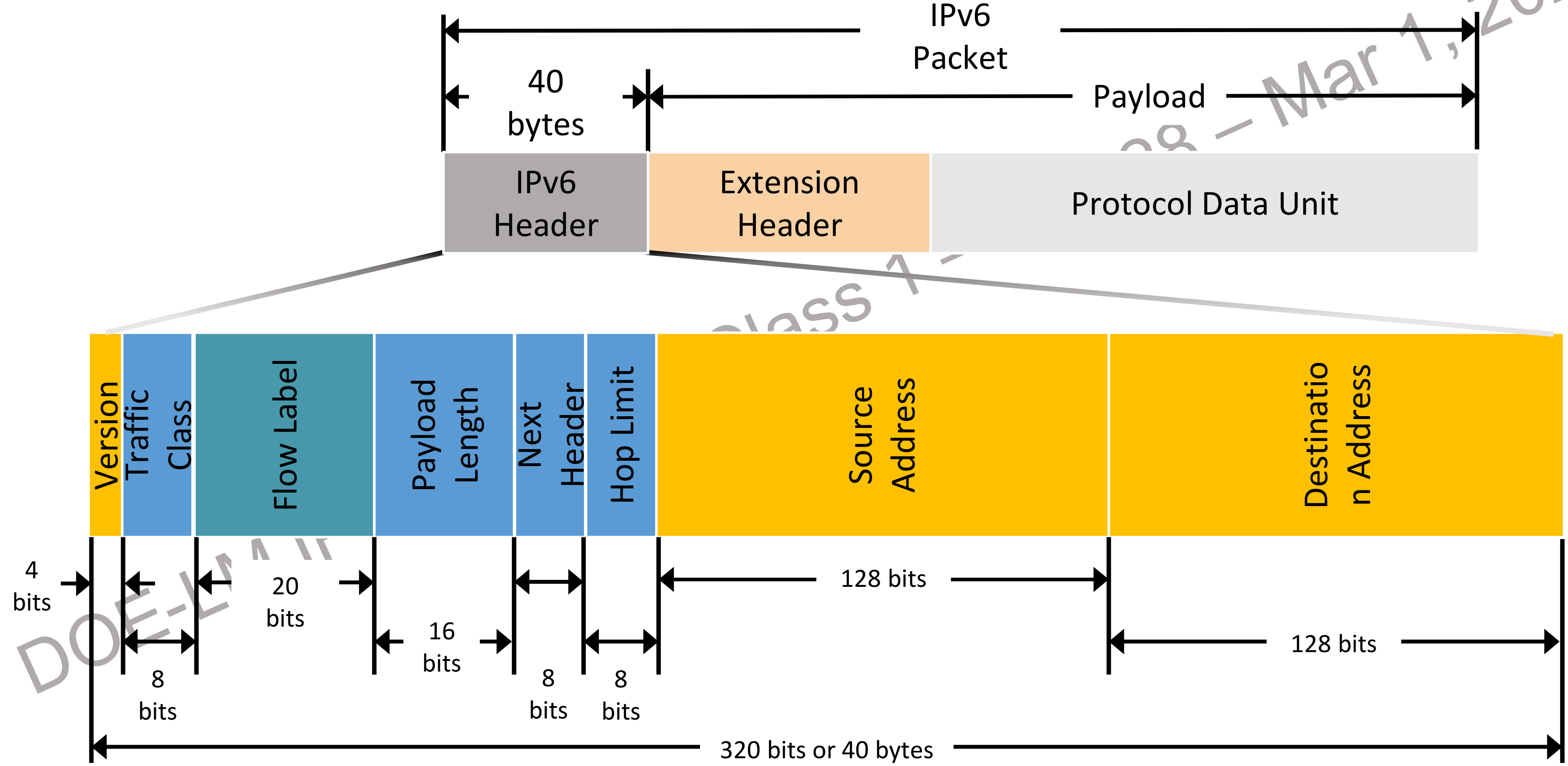
Because TCP/UDP straddle both IPv6 and IPv4 you cannot remove either IPv4 or IPv6 from the networking stack nor “disable” them

It is a dual IP stack!



New TCP/IP Networking Stack (Dual IP Layer)

IPv6 Packet



IPv6
Packet
40
bytes
Payload
IPv6
Header
Extension
Header
Protocol Data Unit
Version
Traffic
Class
Flow Label
Payload
Length
Next
Header
Hop Limit
Source
Address
Destinatio
n Address
4
bits
8
bits
20
bits
16
bits
8
bits
8
bits
128 bits
128 bits
320 bits or 40 bytes

Unicast Address Types

Unicast types

Global unicast addresses (2000::/3 – RFC 4291 & RFC 3587) – it is only a fraction of the available global address space

Link-local addresses (fe80::/64 – RFC 4007)

Special addresses (unspecified is :: & loopback is ::1 – RFC 5156)

Unique local addresses or ULA (fc00::/7 – RFC 4193)

Transition addresses (6to4, ISATAP, Teredo)

Most common unicast IPv6 addresses you will see are Global and Link-local

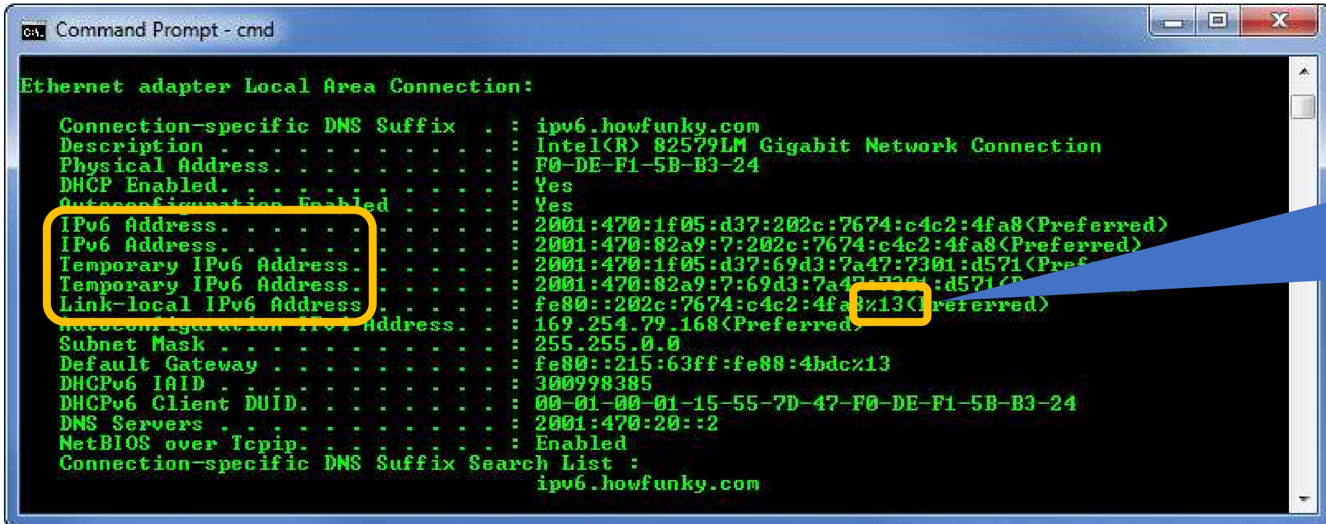
You will see ULA, those addresses typically start with fd00::/8 and are intended to function like RFC 1918 IPv4 addresses

In Use Global Unicast Address

Per IANA

Global unicast addresses are currently assigned from **2000::/3**
Most of the GUA space is held in reserve, specifically:
0::/4
1000::/4 - hold
4000::/3 – 4000::/4 and 5000::/4 – for future growth
6000::/3 – 6000::/4 and 7000::/4 – for future growth
8000::/3 – 8000::/4 and 9000::/4 – for future growth
a000::/3 – a000::/4 and b000::/4 – for future growth
c000::/3 – c000::/4 and d000::/4 – for future growth
e000::/4 -
f000::/4 – fc00::/7 is ULA, fe80::/10 is Link-local, ff00::/8 is Multicast

Windows IPv6 unicast information



Link-local will have a Zone ID (also called Scope ID) represented by the %<ID> after the IPv6 Address

“IPv6 Addresses” are unicast and global

“Temporary IPv6 Addresses” are unicast and global

“Link-local IPv6 Address” is unicast and local

Scope or Zone ID – Why?

Link-local addresses are “locally” significant only

So how do you ping or forward a packet to a link-local address if you have multiple interfaces up and running with IPv6?

Each OS has a way to identify an interface – either a number or a string

Simply add that identifier to the link-local address to specify which interface to use
fe80:202c:7674:c4c2:4fa8**%13**

The interface ID is 13

Think of it as an extension of the routing table

Why Do We Need a Zone ID?

- Ambiguity exists when using link-local addresses
  - Particularly when a host has 2 or more interfaces
  - Link local is required and of the same scope
- **Zone ID** is used to link neighbors table to a specific interface
  - ping fe80::1:2:3:4%12

fe80::cd87:5dd6:cf39:dd08 %12

fe80::80d4:29c9:2b3c:a0e2 %13

IPv6 Address Options in Windows

By default Windows will do:

- Temporary Address (RFC 4941) – used for outbound connections
- Stateful autoconfiguration (DHCPv6)
- Permanent (randomly generated) – doesn't do EUI-64 – known as Privacy

You can optionally set Windows to do:

- Manual (statically assigned)
- EUI-64 (RFC 4291) – turn off Privacy
- Turn off Temporary Addresses
- Or combinations of the above depending on OS setting

Windows IPv6 unicast commands

PowerShell:

```
Get-NetIPAddress -AddressFamily IPv6 -InterfaceIndex <#>
Get-NetIPConfiguration -InterfaceIndex <#>
New-NetIPAddress -InterfaceIndex <#> -IPAddress 2001:0db8:cafe:0010::1 -
PrefixLength 64 -DefaultGateway 2001:0db8:cafe:0010::254
Set-NetIPAddress -InterfaceIndex <#> -IPAddress 2001:0db8:cafe:0010::2 -
PrefixLength 64
Remove-NetIPAddress -InterfaceIndex <#> -Confirm:$false
```

Command line:

```
ipconfig /all
netsh interface ipv6 show addresses <#>
```

IPv6 Interface id options

The last 64 bits of a unicast address is the Interface ID:

- Interface ID is a fixed length and is the host portion of the IPv6 address

Interface ID options:

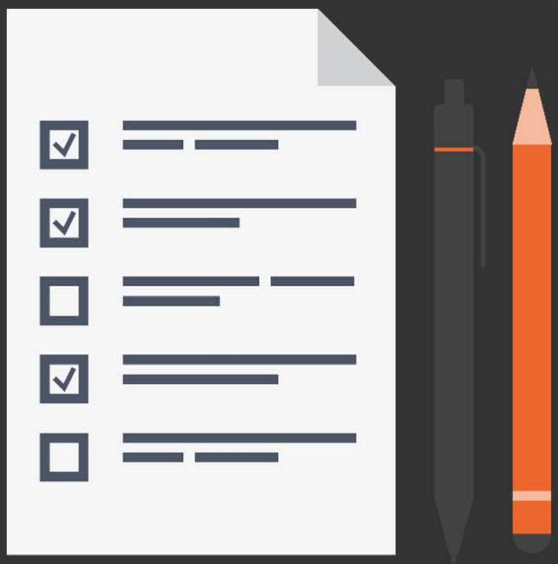
- EUI-64 (RFC 4291) – stateless autoconfiguration
- Random Address (RFC 4941) – stateless autoconfiguration

Default

- Stateful autoconfiguration (DHCPv6)
- Manual (statically assigned)

IPv6 Unicast Configuration Demo

- Lab #1 & 2
- Connect your system to the lab
- Connect to your GCP EVE-NG VM
- Goal: Basic IPv6 commands and set up of Host OS



Windows IPv6 multicast address types

Multicast addresses have the first 8 bits set to 1:

So multicast addresses start with **ff** (ff00::/8)
Multicast also has a 4 bit flag field and 4 bit scope field

Solicited-node address (ff02::1:ff00:0/104)

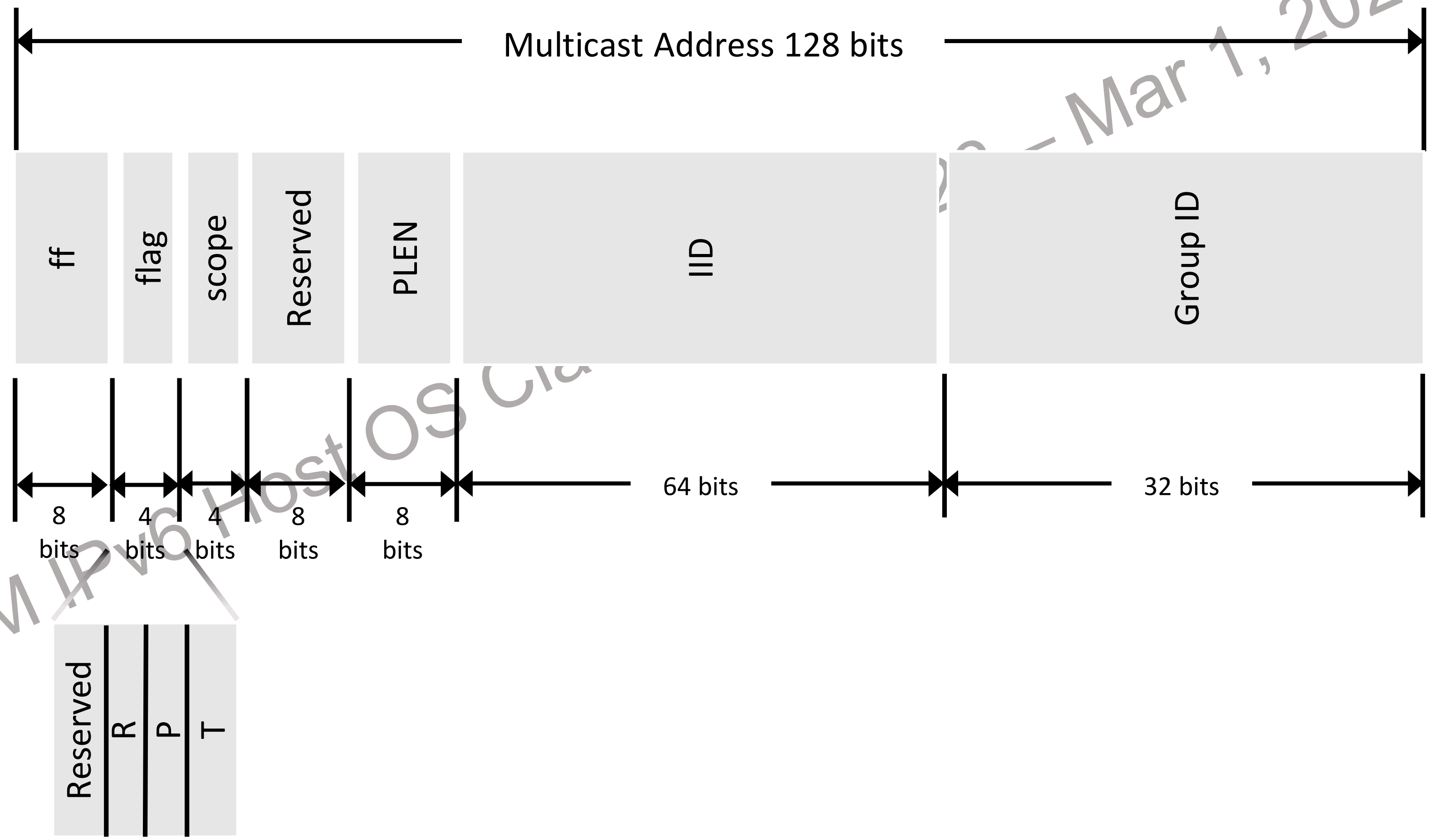
This address is special and built for every IPv6 address you have on a host

Some example multicast addresses:

ff01::1 – interface-local scope all-nodes
ff02::1 – link-local all-nodes

IPv6 multicast Packet

IPv6 Multicast Fields

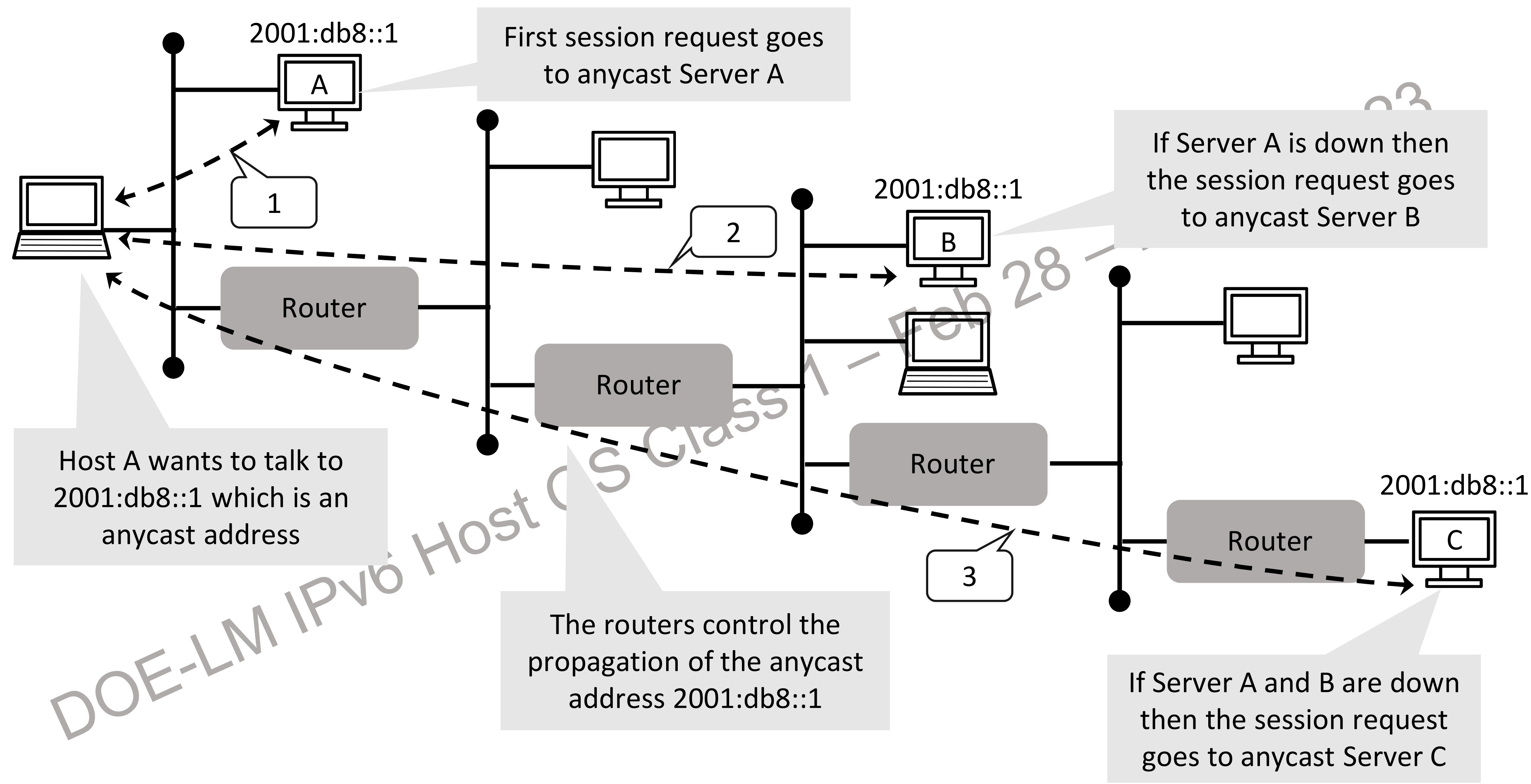


IPv6 Anycast

Anycast addresses are basically special use case Unicast addresses

You can't tell they are Anycast unless you see how they function in a network

Designed to make a Unicast address highly available


2001:db8::1
A
First session request goes to anycast Server A
1
2
2001:db8::1
B
If Server A is down then the session request goes to anycast Server B
Router
Router
Router
Router
2001:db8::1
C
3
Host A wants to talk to 2001:db8::1 which is an anycast address
The routers control the propagation of the anycast address 2001:db8::1
If Server A and B are down then the session request goes to anycast Server C
DOE-LM IPv6 Host OS Class 1 – Feb 28 – Mar 1, 2023

Windows IPv6 Address Configuration

Permanent IPv6 addresses

SLAAC:
- EUI-64 or Random

DHCPv6:
- Stateful
- Stateless

IPv6 Autoconfiguration

Manual

IPv6 Autoconfiguration

There are multiple ways to do autoconfiguration:

- Stateless (SLAAC) – A flag
- Stateful (DHCPv6 stateful) – M and O flags
- Both (DHCPv6 stateless) – A and O flags

Router Advertisements include flags (A, M and O) that determine what the host will do:

- Autonomous flag (A flag)
- Managed Address Configuration flag (M flag)
- Other Stateful Configuration flag (O flag)

There are other combos – avoid them

IPv6 interface ID

Interface ID options:

- EUI-64 (RFC 4291) – stateless autoconfiguration
- Random Address (RFC 4941) – stateless autoconfiguration
- Stateful autoconfiguration (DHCPv6)
- Manual (statically assigned)

Windows will generate Random Interface IDs by default (not EUI-64)

Permanent IPv6 Address

A permanent IPv6 address is a:

- Global Unicast or ULA that is assigned to the interface either via autoconfiguration (SLAAC or DHCPv6) or
- manually that doesn't change for as long as the interface is active (enabled)
- Potentially an ISATAP address (DirectAccess use cases)

A permanent IPv6 address is registered in:

- DNS (like for Active Directory joined hosts)

Temporary IPv6 Address

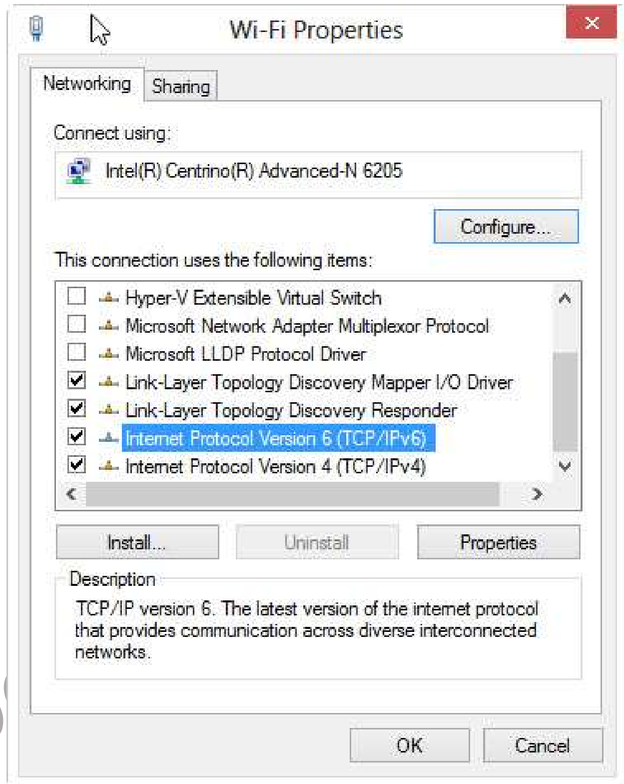
A temporary IPv6 address:

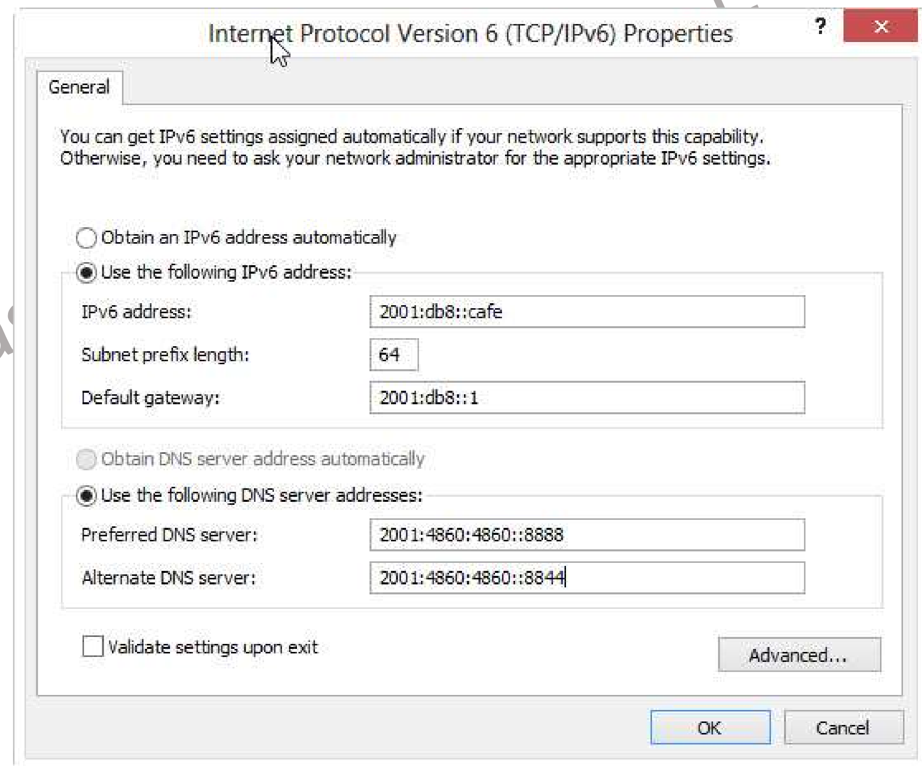
- Is a Global Unicast or ULA that uses a random interface ID in order to build an IPv6 address that will have a limited lifetime and is designed to provide more privacy for a client
- Can change while the interface is active
- There can be more than one temporary IPv6 address active on an interface at the same time

A temporary IPv6 address:

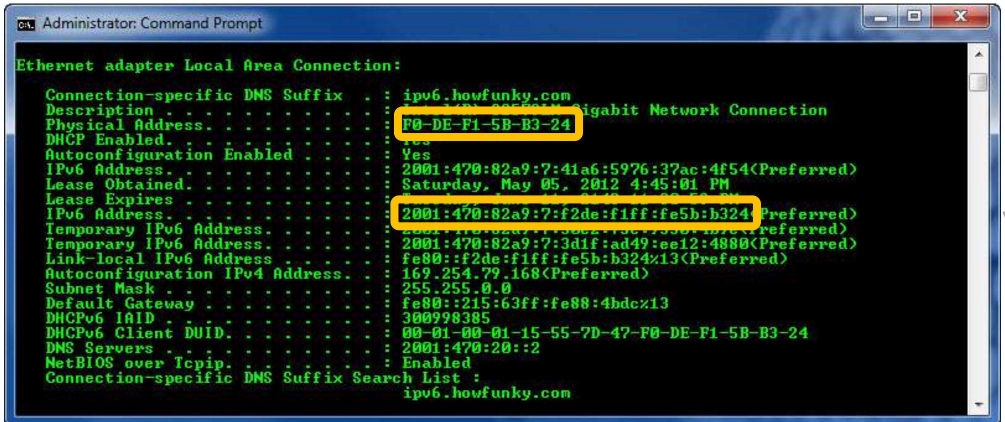
- Is used for outbound sessions (like browsing website content)
- Will not be registered in DDNS

Windows IPv6 manual configuration





Windows IPv6 Addressing



Look for the FF:FE in the address, then it is likely EUI-64

2001:470:82a9:7:f2de:f1ff:fe5b:b324

MAC address: F0-DE-F1-5B-B3-24

Take MAC and split it, stuff in FF:FE and set the 7th bit to “local”

Windows IPv6 Interface ID Commands

PowerShell:

```
Set-NetIPv6Protocol -RandomizeIdentifiers Disabled
Set-NetIPv6Protocol -RandomizeIdentifiers Enabled
```

Command line:

```
netsh interface ipv6 set global randomizeidentifiers=disabled store=active
netsh interface ipv6 set global randomizeidentifiers=disabled store=persistent
```

Have you noticed anything odd yet?

Stateless Address Autoconfiguration (SLAAC)

IPv6 hosts can self provision an IPv6 address

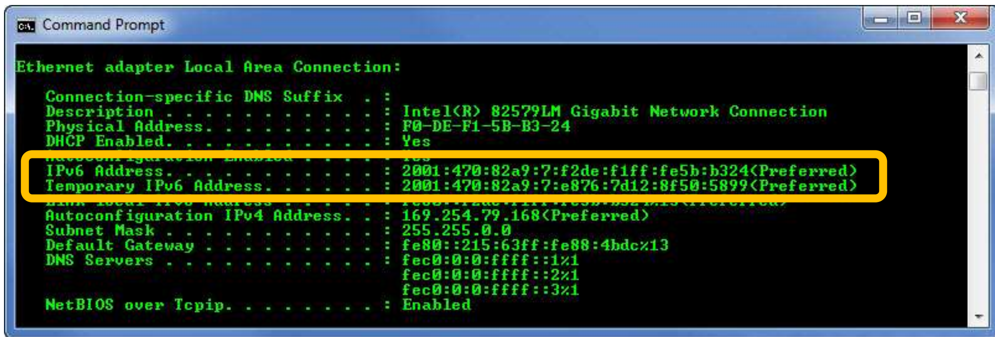
A host uses router discovery to do:

- Router Solicitation messages
- Router Advertisement messages (RA's are sent via ICMPv6)
- Host gets information like on-link prefixes and other routers

SLAAC requires the prefix be a /64:

- If the network prefix is larger the host cannot do EUI-64 or privacy
- This breaks the ability of the host to do SLAAC or to self provision

SLAAC Only Pre Windows10



2001:470:82a9:7::/64 is the prefix

Notice there are no global DNS Servers – if older

Notice there is no FQDN information

This typically indicates SLAAC – the host lacks DNS server info

Also indicates poorly implemented IPv6

SLAAC Windows 10

Windows 10 post Creators Update:

Supports RFC 6106 – Also called RDNSS (Recursive DNS Server)

Accept RA's with DNS info

Will still prefer the DNS server information from DHCPv6 over RDNSS

You can still manually assign a DNS server which will be preferred over DHCPv6 and RDNSS

Things get interesting with VPN

DHCPv6 – RFC 3315

DHCPv6 supports both stateful and stateless address configuration for IPv6 hosts

What is Stateful?

- The DHCPv6 server is assigning the IPv6 address
- The RA has the M and O flag set (A flag may or may not be set – typically don't set)
- All options except default gateway come from DHCPv6 scope

What is Stateless?

- The DHCPv6 server is used for options only (like DNS or NTP)
- The RA has the O flag set and the A flag

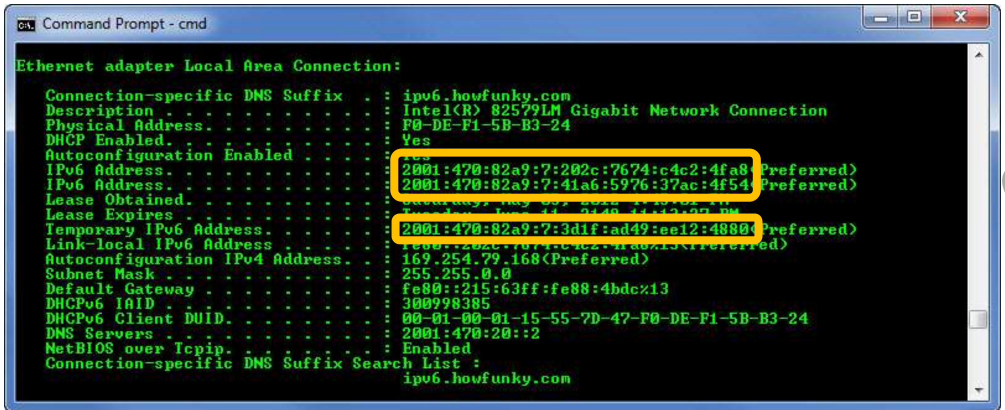
DHCPv6 – RFC 3315

The typical flag settings combination:

- SLAAC: **A** flag = 1, M flag = 0, O flag = 0
- DHCPv6 Stateful: A flag = 0, **M** flag = 1, **O** flag = 1
- DHCPv6 Stateless: **A** flag = 1, M flag = 0, **O** flag = 1

For all situations a host link-local address is configured automatically (exception maybe routers)

DHCPv6 w/ A, O and M Flags



2001:470:82a9:7:41a6:5976:37ac:4f54 is from DHCPv6

2001:470:82a9:7:202c:7674:c4c2:4fa8 is from SLAAC

2001:470:82a9:7:3d1f:ad49:ee12:4880 is from SLAAC

Windows specific Autoconfiguration behavior

By default Windows generates a random interface ID for:

- Autoconfiguration addresses (permanent & temporary)
- It does this for Link-local, Global Unicast and Unique Local Addresses
- In other words – it doesn't do EUI-64

Windows specific Autoconfiguration behavior

Because the host randomly built an interface ID:

It starts using it right away
It doesn't wait for duplicate address detection (DAD)
This is called Optimistic DAD

Autoconfiguration is used to build:

link-local
and can be used for global and ULA (depending on the flag combinations A/M/O)

Windows specific Autoconfiguration behavior

If you are using a Windows Server as an IPv6 Router*:

Windows always sets the **A** flag

This means that the Windows Server is doing an RA and is enabling SLAAC

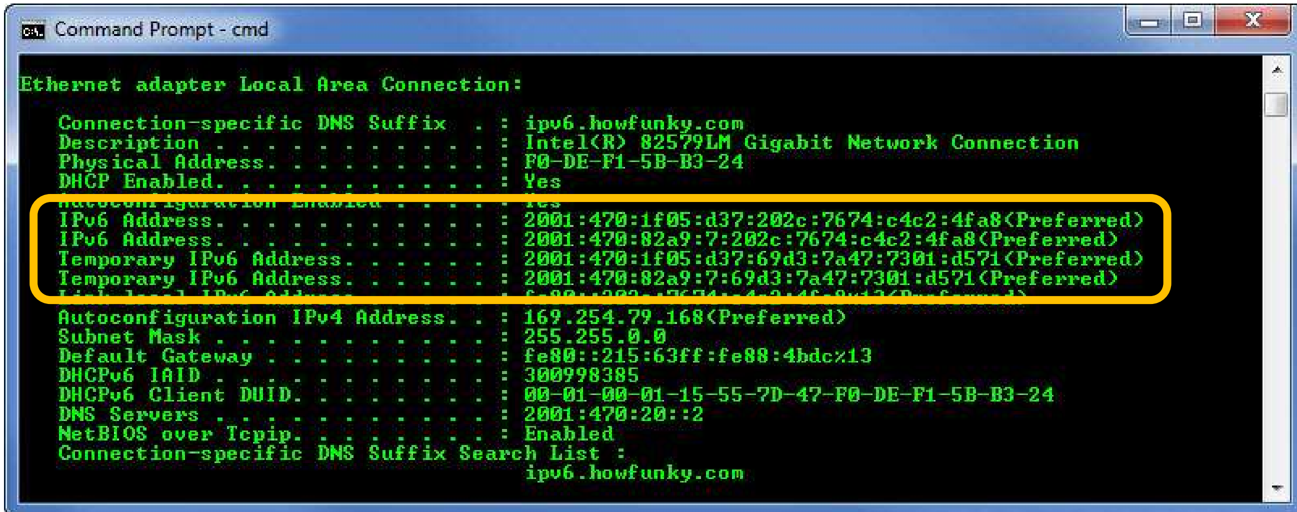
By default all hosts on that subnet will build out a random permanent address and random temporary address

Windows Server 2008 and Window Vista do not attempt stateful DHCPv6 if no RA advertisement are received

Windows specific Autoconfiguration behavior

* We don't recommend running a Windows Server as a router, but just know the behavior to expect from the server if you do use it that way

SLAAC + DHCPv6 w/ O flag



2001:470:1f05:d37::/64 gets a permanent and temporary

2001:470:82a9:7::/64 gets a permanent and temporary

The router only has 2001:470:82a9:7::/64 as an address

IP Address Management (IPAM)

It is normal for a single host to have 3 or more IPv6 addresses

Each /64 subnet can have 18 million trillion addresses

You don't want to manage a spreadsheet with 18 million trillion fields for 128 bit long addresses in HEX – network or hosts!

If you don't want to do that use an IPAM solution!

Bluecat Networks, BT Diamond, Cisco Prime Network Registrar, Infoblox, Other…

IPv4 and Equivalent IPv6 properties

IPv4	IPv6
Multicast (224.0.0.0/4)	Multicast (ff00::/8)
Broadcast	Replaced with Multicast groups
APIPA (169.254.0.0/16)	Link-local (fe80::/64)
Public Address Space	Global Unicast Space
Loopback (127.0.0.1)	Loopback (::1)
Unspecified (0.0.0.0)	Unspecified (::)
Dotted decimal	Colon HEX format
RFC 1918 (10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16)	ULA (fd00::/8)

ICMPv4 vs. ICMPv6 properties

ICMPv4	ICMPv6
Destination Unreachable – Network Unreachable (Type 3, Code 0)	Destination Unreachable – No Route to Destination (Type 1, Code 0)
Destination Unreachable – Host Unreachable (Type 3, Code 1)	Destination Unreachable – Address Unreachable (Type 1, Code 3)
Destination Unreachable – Protocol Unreachable (Type 3, Code 2)	Parameter Problem – Unrecognized Next Header Type Encountered (Type 4, Code 1)
Destination Unreachable – Port Unreachable (Type 3, Code 3)	Destination Unreachable – Port Unreachable (Type 1, Code 4)
Destination Unreachable – Fragmentation Needed and DF Set (Type 3, Code 4)	Packet Too Big (Type 2, Code 0)

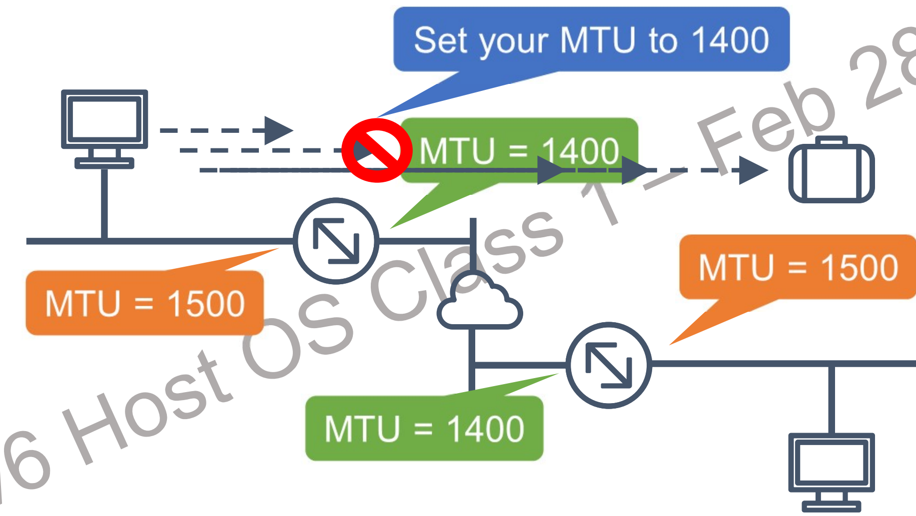
Table 5-3 Chapter 5 – ICMPv6 from Understanding IPv6, 3rd Ed. By Joseph Davies, Microsoft Press

ICMPv4 vs. ICMPv6 properties

ICMPv4	ICMPv6
Destination Unreachable – Communication with Destination Host Administratively Prohibited (Type 3, Code 10)	Destination Unreachable – Communication with Destination Administratively Prohibited (Type 1, Code 1)
Source Quench (Type 4, Code 0)	This message is not present in IPv6.
Redirect (Type 5, Code 0)	Neighbor Discovery Redirect message (Type 137, Code 0)
Time Exceeded – TTL Exceeded in Transit (Type 11, Code 0)	Time Exceeded – Hope Limit Exceeded in Transit (Type 3, Code 0)
Time Exceeded – Fragment Reassembly Time Exceeded (Type 11, Code 1)	Time Exceeded – Fragment Reassembly Time Exceeded (Type 3, Code 1)
Parameter Problem (Type 12, Code 0)	Parameter Problem (Type 4, Code 0 or Code 2)

Table 5-3 Chapter 5 – ICMPv6 from Understanding IPv6, 3rd Ed. By Joseph Davies, Microsoft Press

Why is Path MTU so important in IPv6?



Important ICMPv6 Things to Know

Path MTU must be allowed or IPv6 will break

This means you need to allow ICMPv6 traffic through your network

RFC 4890 outlines firewall rules for ICMPv6
https://tools.ietf.org/html/rfc4890

Must Permit ICMPv6 Traffic

Destination Unreachable (Type 1 – all codes)

Packet Too Big (Type 2)

Time Exceeded (Type 3 – code 0 only)

Parameter Problem (Type 4 – code 1 and 2 only)

Echo Request (Type 128)

Echo Reply (Type 129)

Disabling IPv6 in Windows

You need to modify the registry using:

GPO
Registry
PowerShell
netsh

The registry key is:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\tcpip6\Parameters\DisabledComponents

Registry key values

Registry Value (type DWORD) for "DisabledComponents"

Bit 7	Default value is 0 or off. If set to 1 then it disables all IP-HTTPS interfaces.
Bit 6	Default value is 0 or off. It is reserved.
Bit 5	Default value is 0 or off. If set to 1 then it sets the default prefix policy table to prefer IPv4 over IPv6.
Bit 4	Default value is 0 or off. If set to 1 then it disables all LAN and PPP interfaces but NOT tunnel based interfaces like 6to4, ISATAP, Teredo and IP-HTTPS.
Bit 3	Default value is 0 or off. If set to 1 then it disables all Teredo interfaces.
Bit 2	Default value is 0 or off. If set to 1 then it disables all ISATAP interfaces.
Bit 1	Default value is 0 or off. If set to 1 then it disable all 6to4 interfaces.
Bit 0	Default value is 0 or off. If set to 1 then it disables all the tunnel interfaces which include 6to4, ISATAP, Teredo and IP-HTTPS.

Window Prefix Policy Table

Windows Prefix Policy Table

Windows maintains a Prefix Policy Table:

That controls IPv6 and IPv4 behavior

It is based on RFC 6724 (RFC 3484 is the older version)

Using PowerShell you can display the current table:

```
Get-NetPrefixPolicy
```

Using netsh to modify the table:

```
netsh interface ipv6 show prefixpolicies
netsh interface ipv6 add prefixpolicy <prefix> <precedence> <label>
netsh interface ipv6 set prefixpolicy <prefix> <precedence> <label>
netsh interface ipv6 remove prefixpolicy <prefix> <precedence> <label>
```

RFC 6724 to 3484 Table Difference

2023

IPv6 Prefix Range	RFC 3484 Precedence	RFC 6724 Precedence
3ffe::/16	40	1
fec0::/10	40	1
fc00::/7	40	3
::/96	20	1
::ffff:0:0/96	10	5

Prefix Policy Table Management

Use the following netsh command to update an RFC 3484 host to RFC 6724:

```
netsh int ipv6 add prefixpolicy 3ffe::/16 1 12 store=persistent
netsh int ipv6 add prefixpolicy fec0::/10 1 11 store=persistent
netsh int ipv6 add prefixpolicy fc00::/8 4 13 store=persistent
netsh int ipv6 add prefixpolicy fd00::/8 3 14 store=persistent
netsh int ipv6 add prefixpolicy ::/96 1 3 store=persistent
netsh int ipv6 add prefixpolicy ::ffff:0:0/96 35 4 store=persistent
```

Example Prefix Policy Table

```
PS C:\> Get-NetPrefixPolicy
```

Prefix	Precedence	Label
3ffe::/16	1	12
fec0::/10	1	11
::/96	1	3
fc00::/7	3	13
2001::/32	5	5
2002::/16	30	2
::ffff:0:0/96	35	4
::/0	40	1
::1/128	50	0

ULA

IPv4

Protocol Selection Process

Windows Protocol selection

By default Windows prefers IPv6

You can modify this behavior by changing the prefix policy table

It checks if IPv6 is working using NCSI

This check happens each time Windows connects to a network
Wireless join / Wired connection
It will recheck in 30 days if the interface stays connected that long (servers and workstations are the common situation for that)
NCSI can be set to a custom value

NCSI = Network Connectivity Status Indicator

Windows protocol selection considerations

If you use ULA

If you have IPv4 (dual-stack) then all hosts will prefer IPv4 due to the preference settings in RFC 6724

If you want to use ULA and prefer IPv6

Change the prefix policy table
Or turn off IPv4
Or assign an GUA in addition to the ULA

Auto Address Configuration

IPv6 Automatic Address Method

Stateless Address Auto Configuration (SLAAC):

Very common for in wireless environments
Often used in home or smaller network situations

Stateful (DHCPv6 stateful):

Matches how DHCP is utilized today in IPv4
Requires the correct RA flags be set

Both (DHCPv6 stateless):

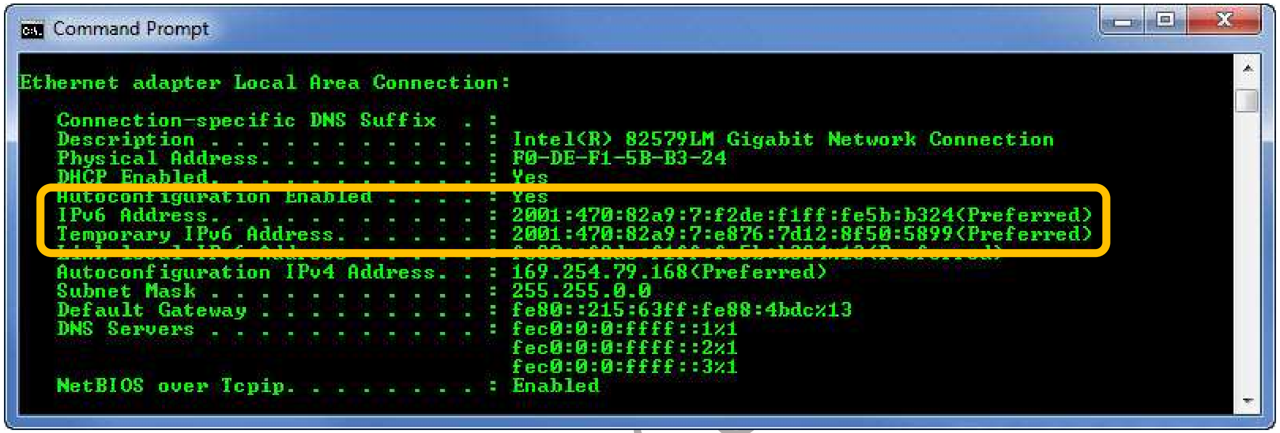
Will be more common for wireless environments, home and smaller networks
DHCPv6 server is providing simple information like DNS servers
Host still uses SLAAC to set up an address automatically

IPv6 Manual Method

You can always assign a static address to the host, even in IPv6

A pseudo method that essentially functions like the manual method is to use static reservations in DHCPv6

SLAAC Only



2001:470:82a9:7::/64 is the prefix

Notice there are no global DNS Servers

Notice there is no FQDN information

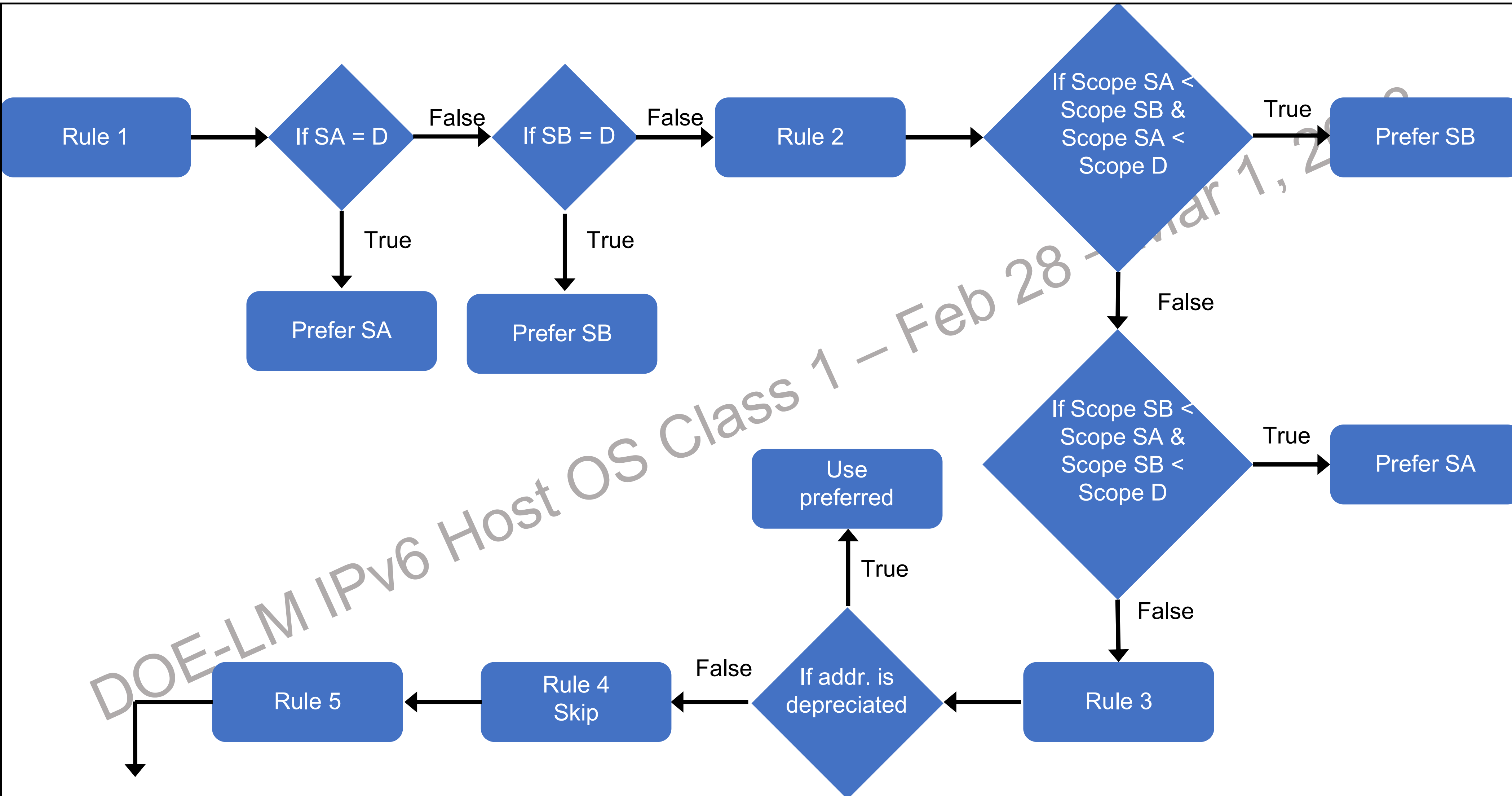
Source Address Selection

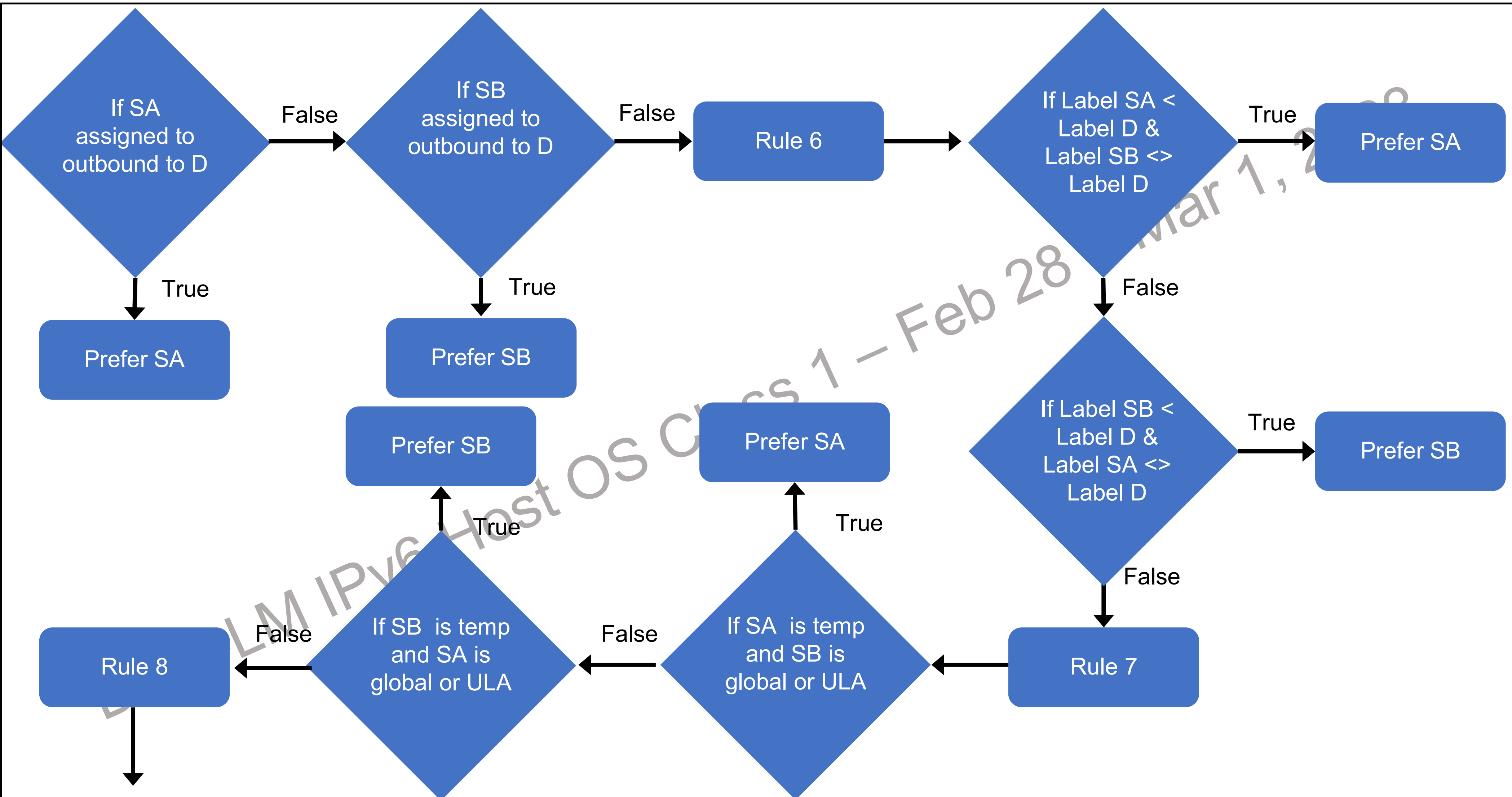
Default Address Selection for IPv6

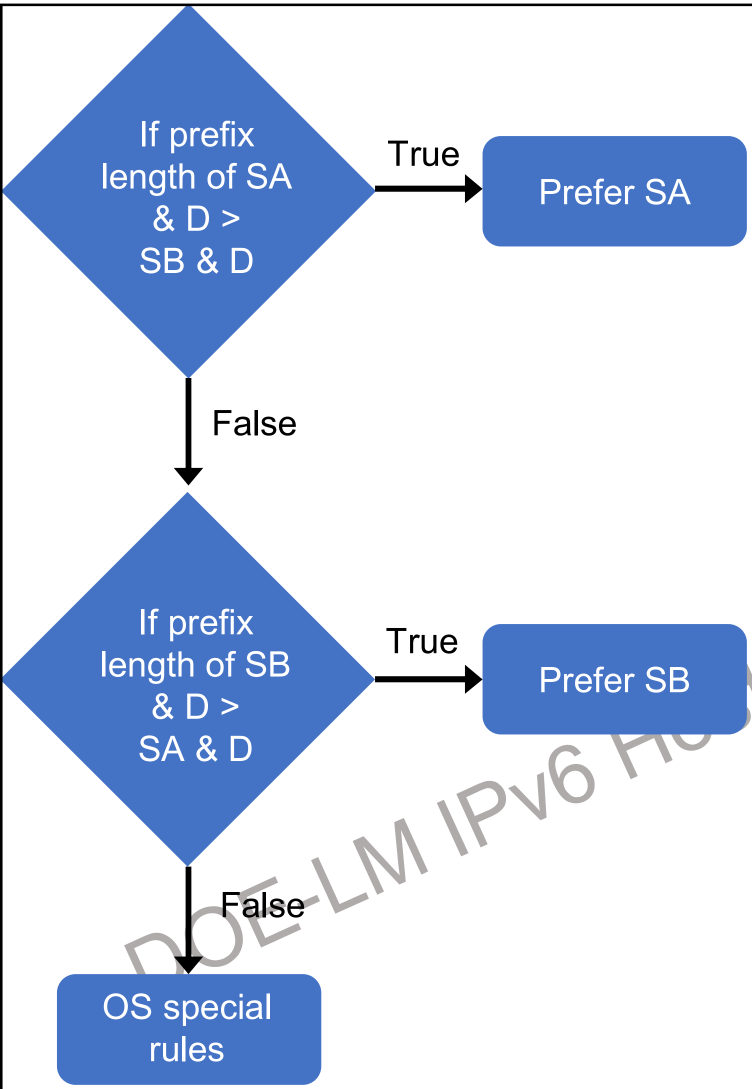
This process is defined in RFC 6724

Not all OS versions conform to RFC 6724 – it was released in September 2012

Older RFC was 3484 – many OS versions still use that RFC


Rule 1
If SA = D
False
If SB = D
False
Rule 2
If Scope SA < Scope SB & Scope SA < Scope D
True
Prefer SB
True
Prefer SA
True
Prefer SB
False
If Scope SB < Scope SA & Scope SB < Scope D
True
Prefer SA
Use preferred
True
False
Rule 3
If addr. is depreciated
False
Rule 4 Skip
Rule 5


If SA assigned to outbound to D
False
True
Prefer SA
If SB assigned to outbound to D
False
True
Prefer SB
Rule 6
If Label SA < Label D & Label SB <> Label D
True
Prefer SA
False
If Label SB < Label D & Label SA <> Label D
True
Prefer SB
False
Rule 7
If SA is temp and SB is global or ULA
True
Prefer SA
False
If SB is temp and SA is global or ULA
True
Prefer SB
False
Rule 8


If prefix length of SA & D > SB & D
True
Prefer SA
False
If prefix length of SB & D > SA & D
True
Prefer SB
False
OS special rules

Transition Technologies in Windows

Windows Transition Technologies

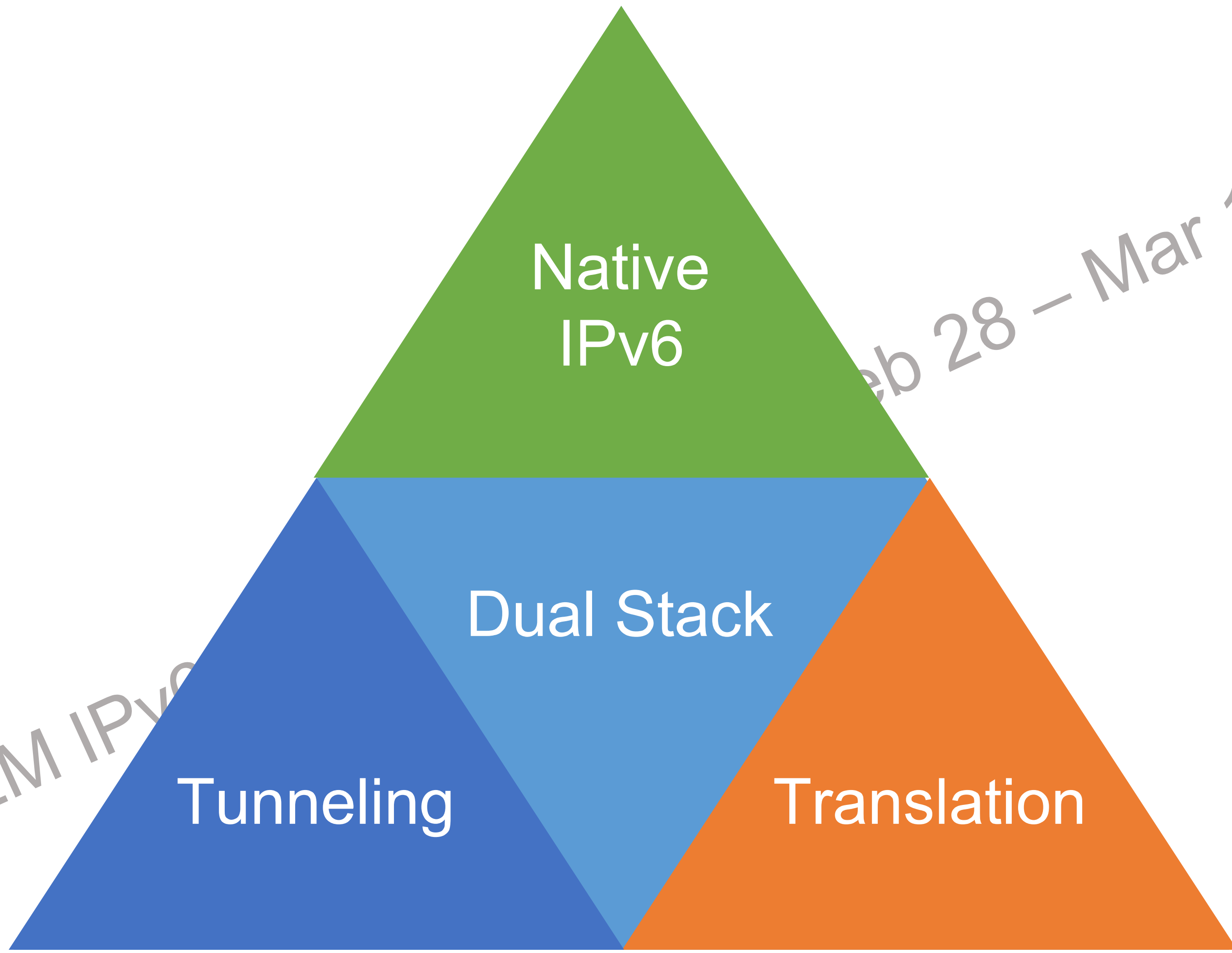
Windows was designed to leverage several methods of transition technologies

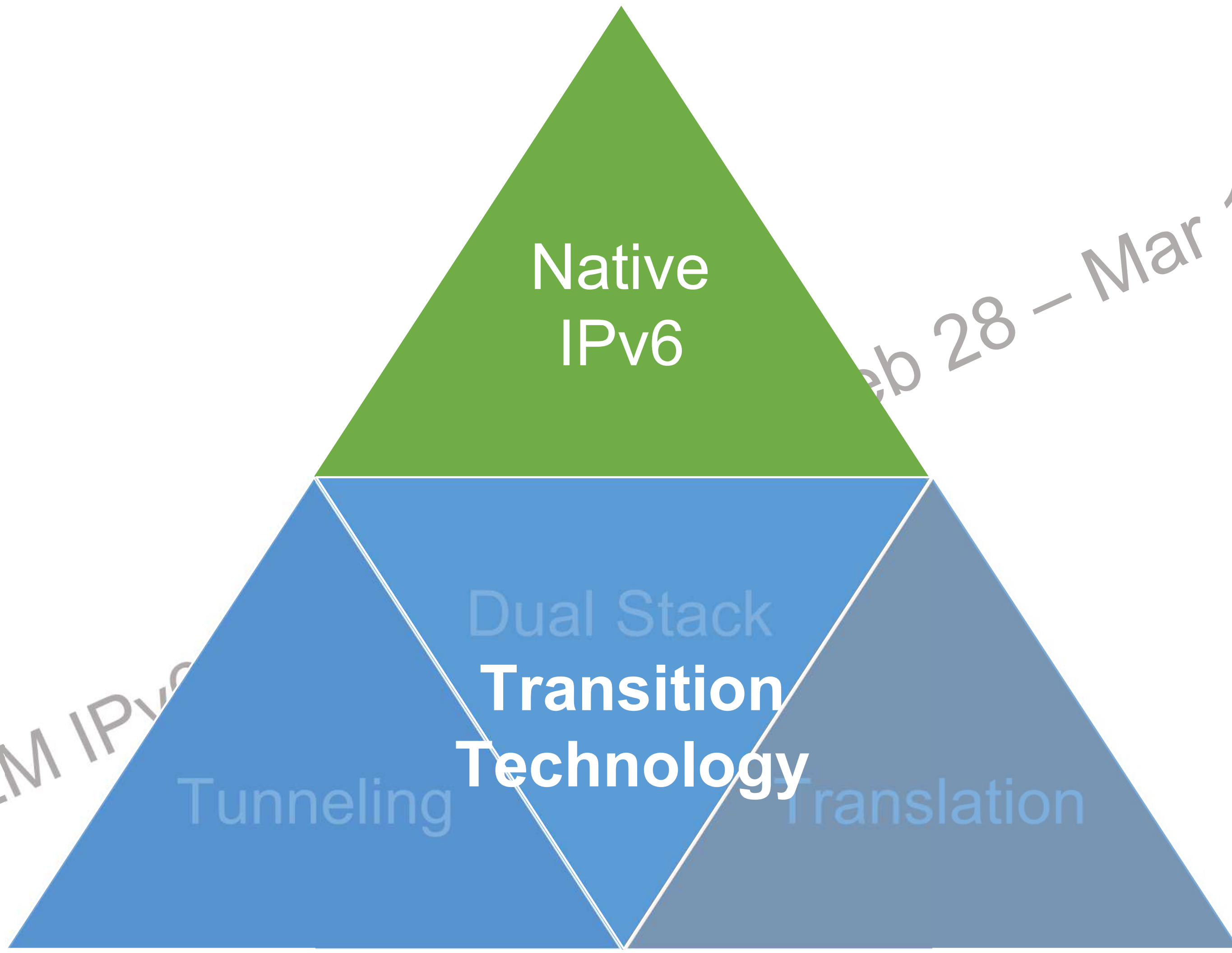
Microsoft invented several transition methods to speed up the adoption of IPv6

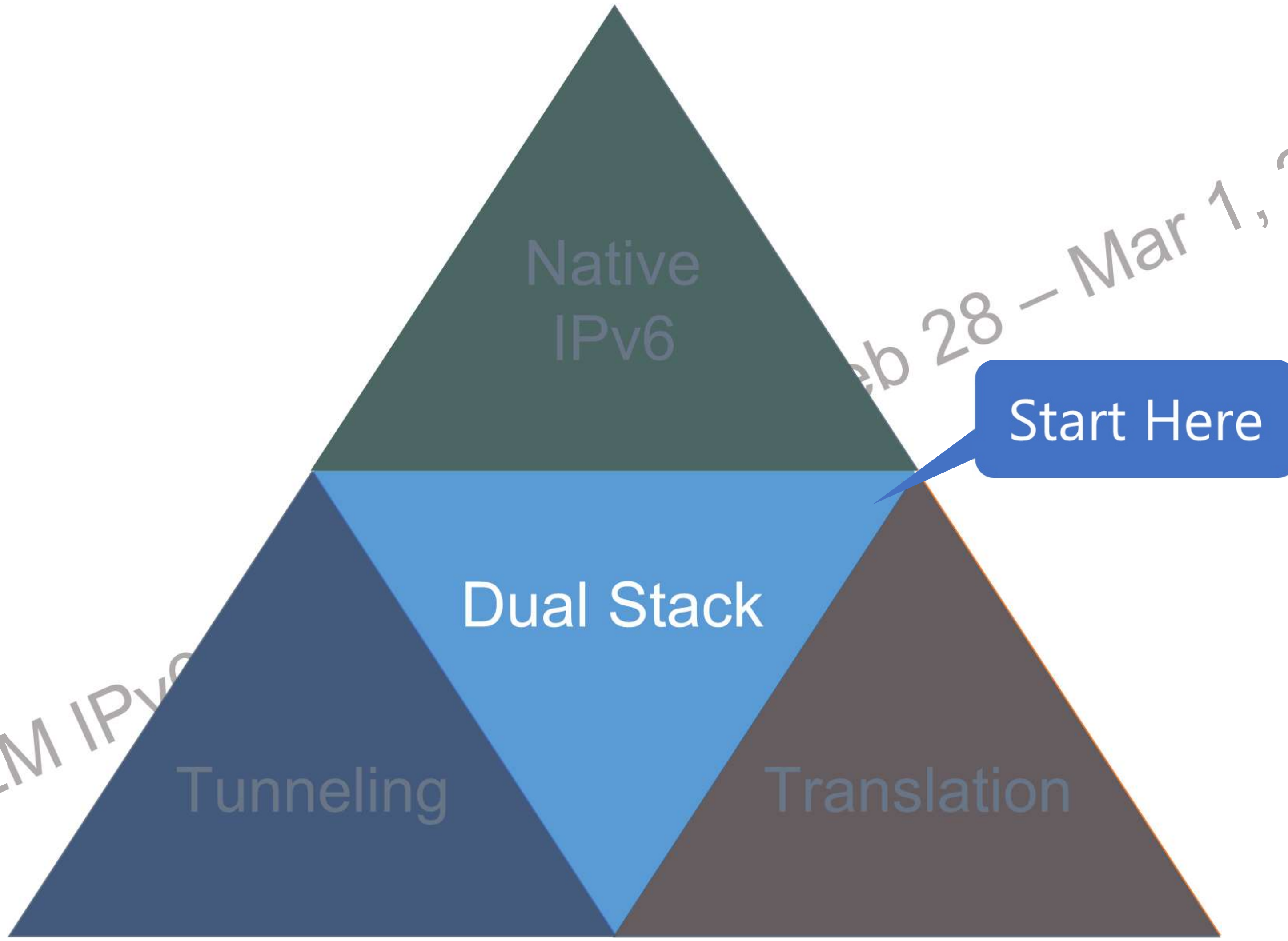
There are many other transition methods

The goal of a transition technology is to help in the migration of IPv4 to IPv6

Remember, the end goal is native IPv6!!!


Native
IPv6
Dual Stack
Tunneling
Translation


Native
IPv6
Dual Stack
Transition
Technology
Tunneling
Translation


Native
IPv6
Start Here
Dual Stack
Tunneling
Translation

Dual Stack

You run IPv6 and IPv4 side by side
You let the host decide which protocol to use
For routing the two protocols operate like ships in the night (for the most part)
You will consume more resources to run dual stack
You have to understand how your applications behave

Dual Stack

Windows 8 and Windows Server 2012 have partial RFC 6555/8305 implemented (readiness update for Windows 7 and Server 2008R2) - NCSI

All applications benefit from Windows 8 and Windows Server 2012 having partial RFC 6555/8305 behavior enabled

Windows uses RFC 6724 and NCSI

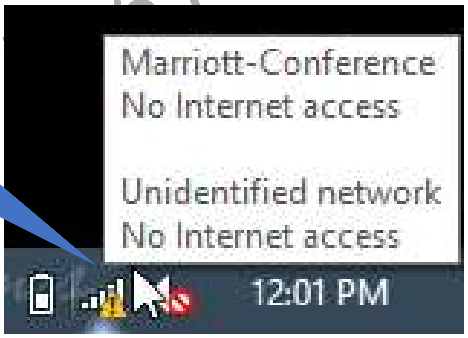
Dual stack is a transition method – the eventual goal is only IPv6

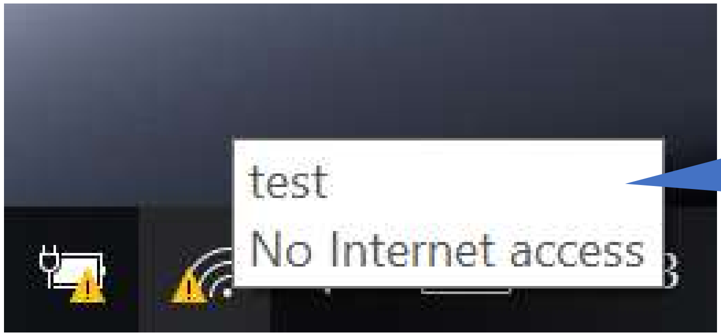
Network Connectivity Status Indicator

Windows 7 and Server 2008R2 w/ Readiness Update to current release uses NCSI - http://support.microsoft.com/kb/2750841

NCSI tests connectivity via a DNS lookup to an IPv6 specific URL: http://ipv6.msftncsi.com/ncsi.txt

Windows 7/8





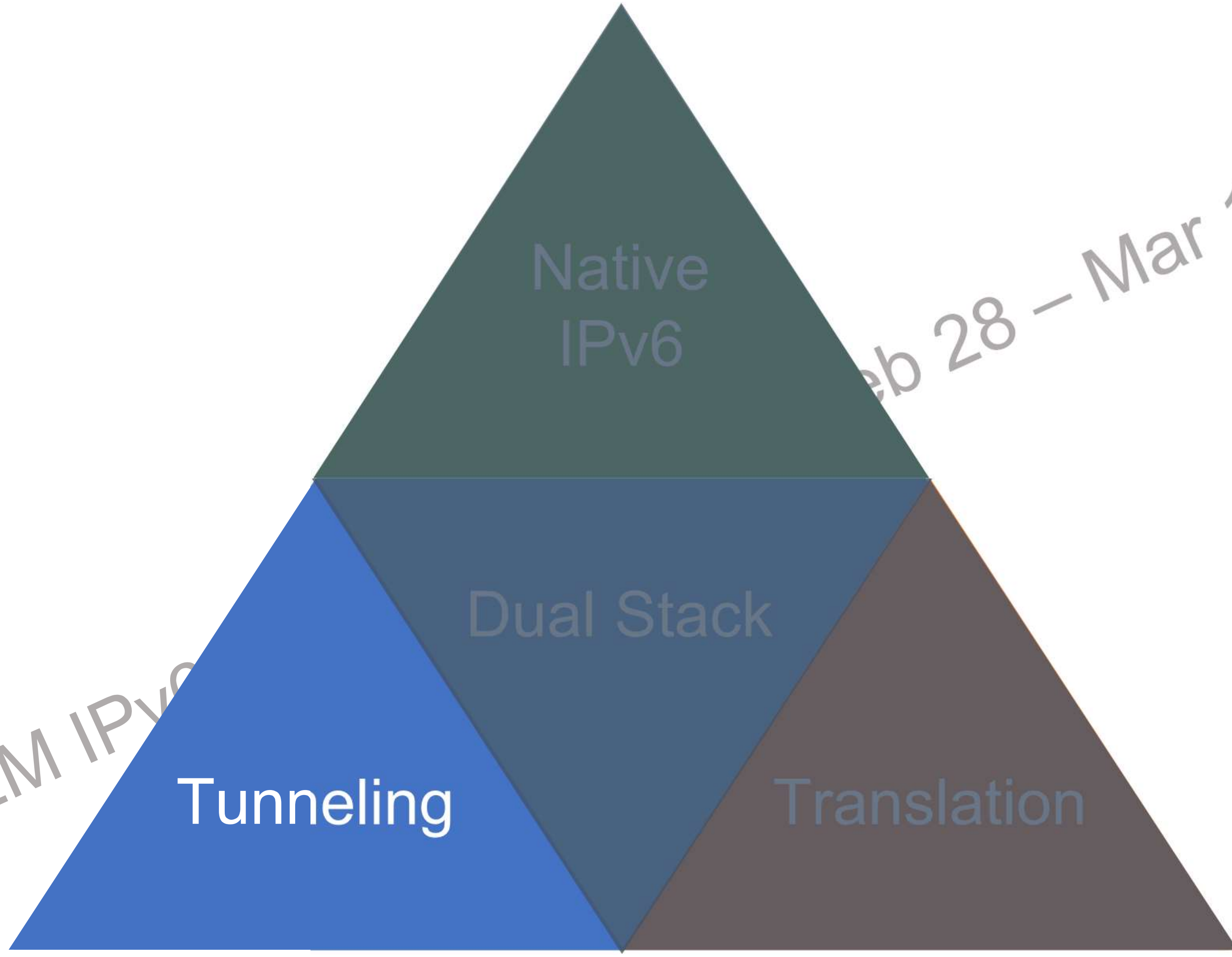
Windows 10

Network Connectivity Status Indicator

Every time a network configuration event occurs (meaning that something has changed in the network configuration), the NCSI process performs several tests to check for IPv6 connectivity

If IPv6 access can't be established NCSI fails back to IPv4

The results of the test are cached for 30 days



Native IPv6
Dual Stack
Tunneling
Translation

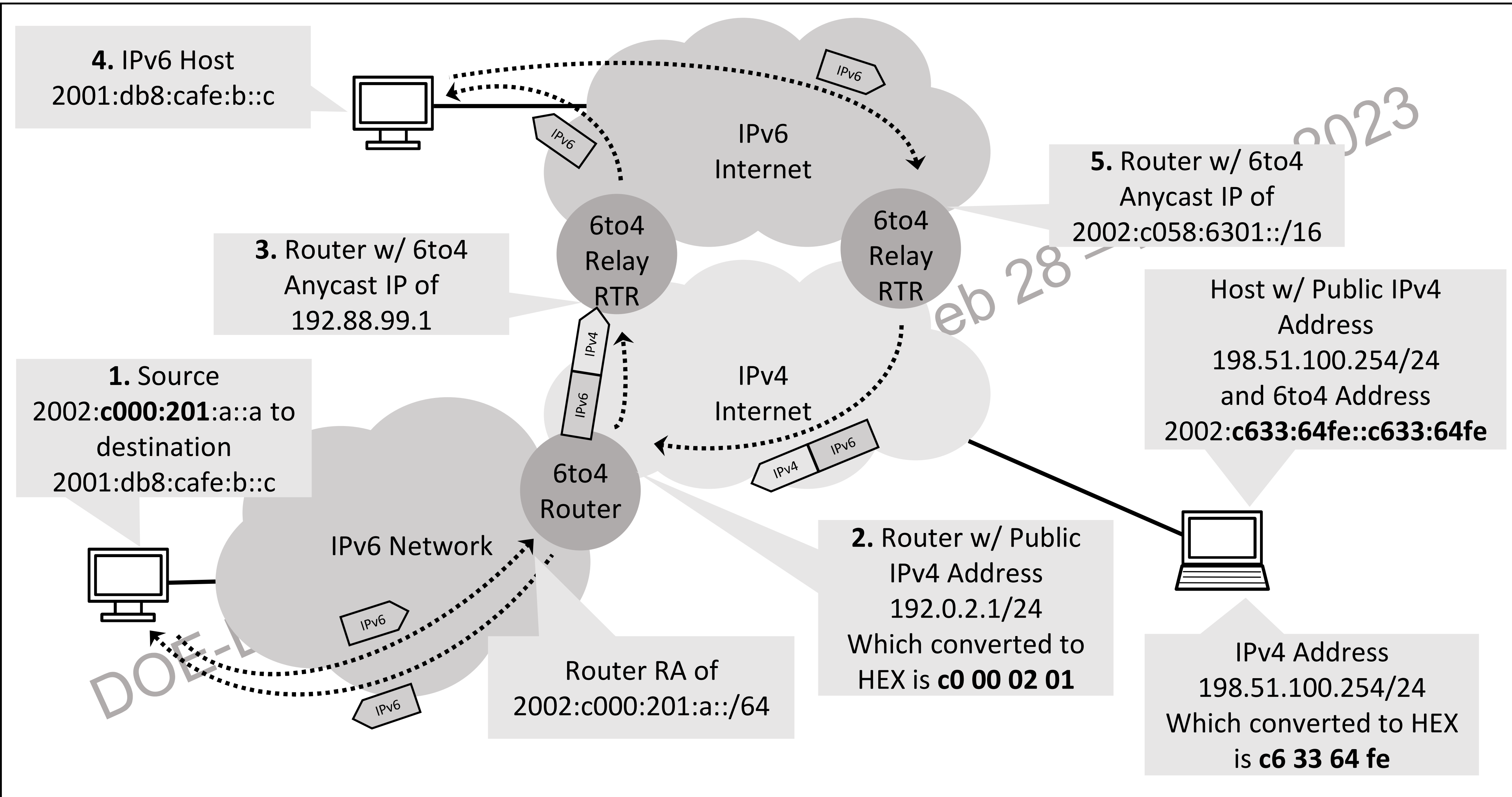
Tunneling

In Windows there are three main tunneling transition technologies

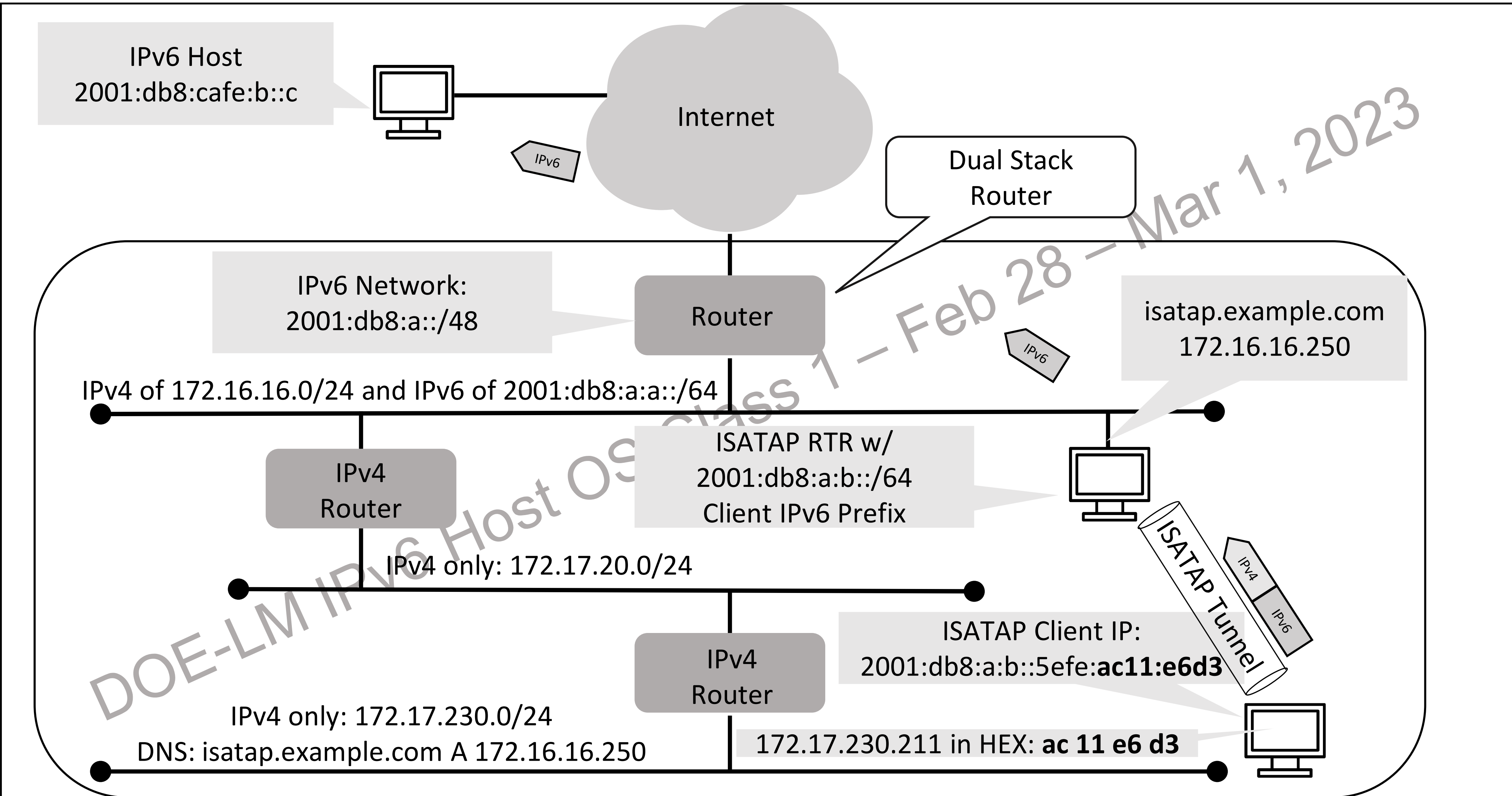
- 6to4
- ISATAP
- Teredo

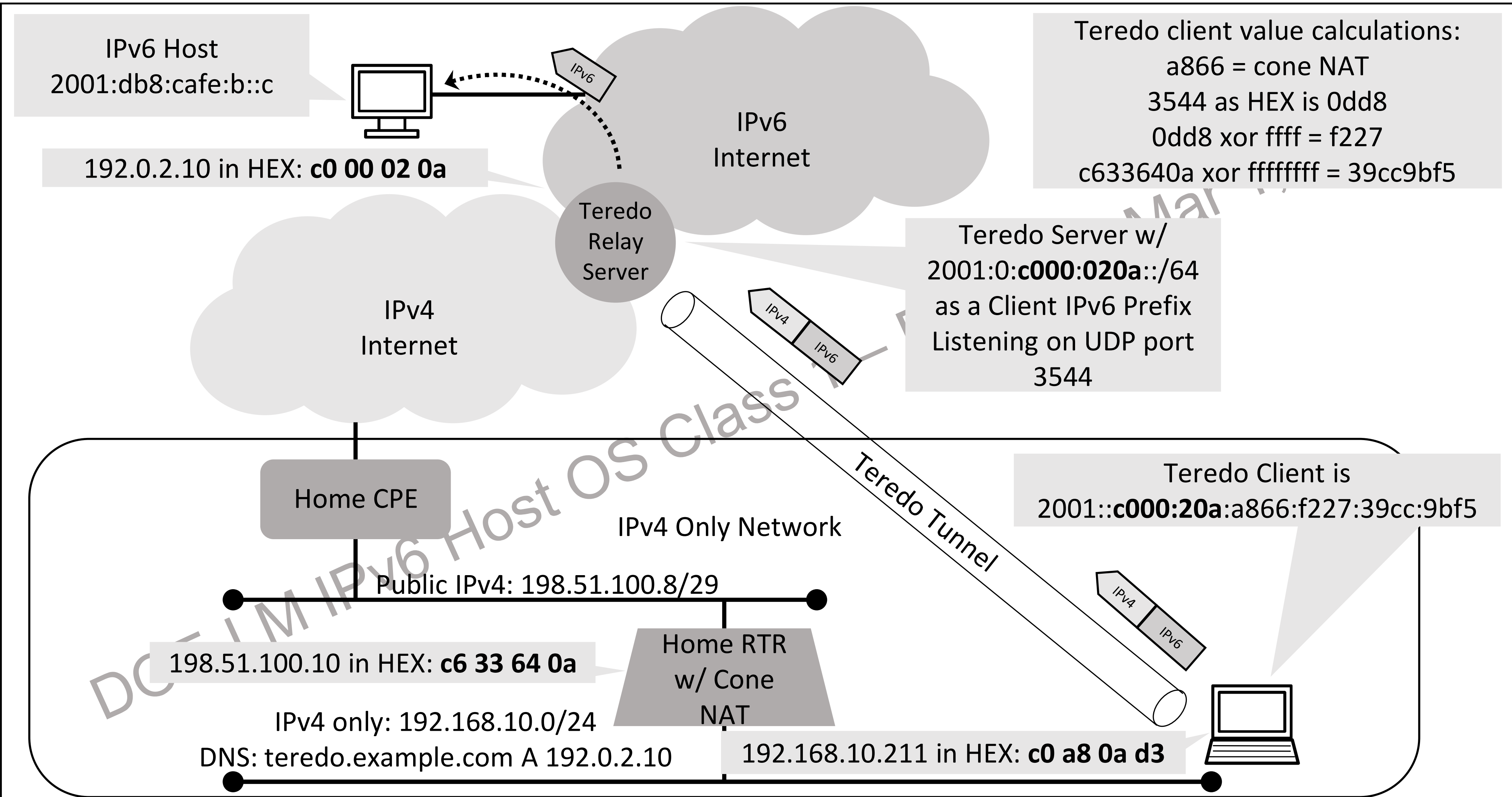
All of these enable IPv6 to IPv6 communication, they just use IPv4 as a transport

We recommend turning off all the transition tunneling technologies to avoid problems troubleshooting IPv6



4. IPv6 Host
2001:db8:cafe:b::c
IPv6
IPv6
IPv6 Internet
5. Router w/ 6to4
Anycast IP of
2002:c058:6301::/16
6to4 Relay RTR
6to4 Relay RTR
3. Router w/ 6to4
Anycast IP of
192.88.99.1
Host w/ Public IPv4
Address
198.51.100.254/24
and 6to4 Address
2002:**c633:64fe::c633:64fe**
1. Source
2002:**c000:201**:a::a to
destination
2001:db8:cafe:b::c
IPv4
IPv6
IPv4 Internet
IPv4
IPv6
6to4 Router
IPv6 Network
2. Router w/ Public
IPv4 Address
192.0.2.1/24
Which converted to
HEX is **c0 00 02 01**
IPv6
IPv6
Router RA of
2002:c000:201:a::/64
IPv4 Address
198.51.100.254/24
Which converted to HEX
is **c6 33 64 fe**


IPv6 Host
2001:db8:cafe:b::c
Internet
IPv6
Dual Stack
Router
IPv6 Network:
2001:db8:a::/48
Router
IPv6
isatap.example.com
172.16.16.250
IPv4 of 172.16.16.0/24 and IPv6 of 2001:db8:a:a::/64
IPv4
Router
ISATAP RTR w/
2001:db8:a:b::/64
Client IPv6 Prefix
ISATAP Tunnel
IPv4
IPv6
IPv4 only: 172.17.20.0/24
ISATAP Client IP:
2001:db8:a:b::5efe:**ac11:e6d3**
IPv4
Router
IPv4 only: 172.17.230.0/24
DNS: isatap.example.com A 172.16.16.250
172.17.230.211 in HEX: **ac 11 e6 d3**



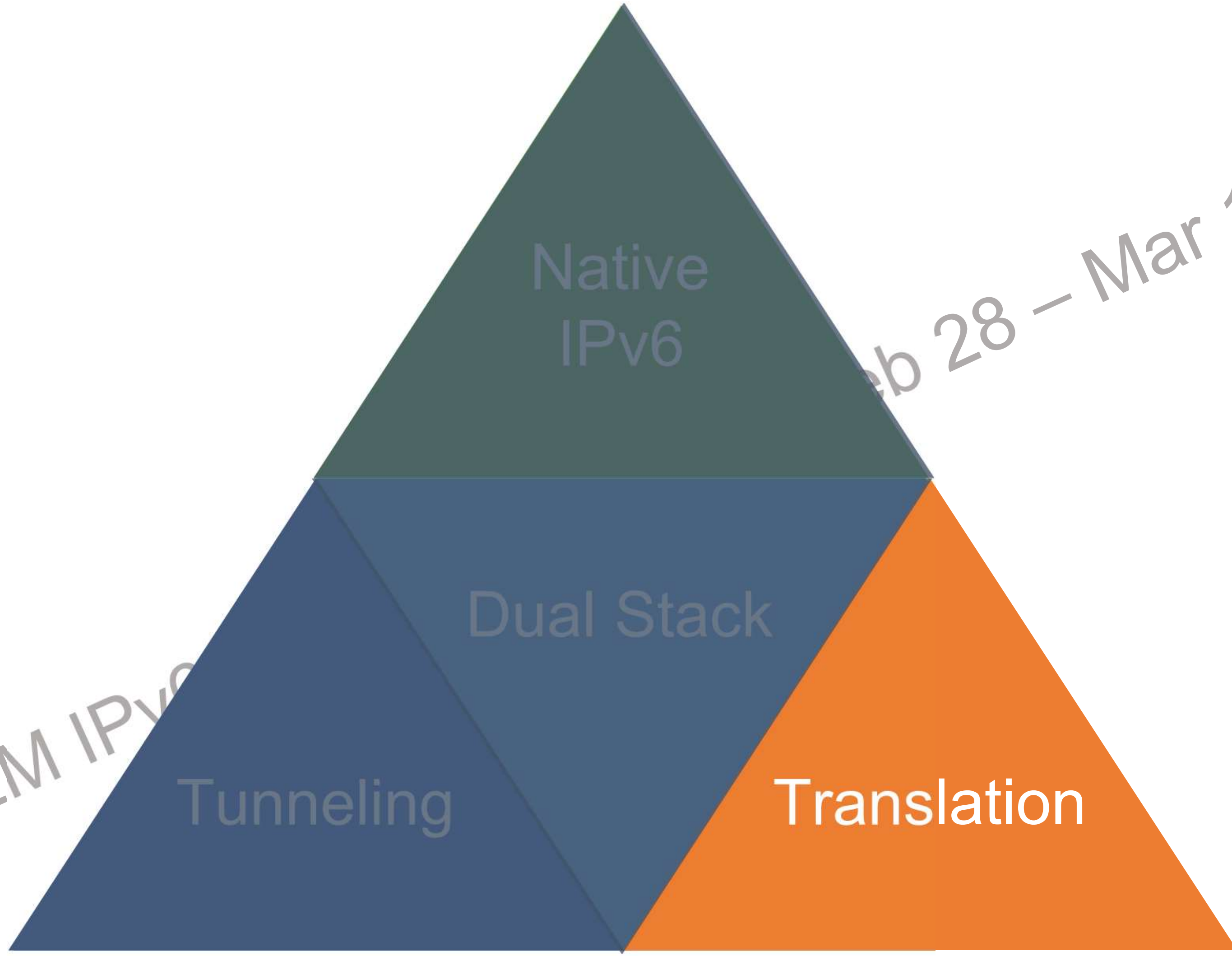
IPv6 Host
2001:db8:cafe:b::c
IPv6
IPv6
Internet
Teredo client value calculations:
a866 = cone NAT
3544 as HEX is 0dd8
0dd8 xor ffff = f227
c633640a xor ffffffff = 39cc9bf5
192.0.2.10 in HEX: **c0 00 02 0a**
Teredo
Relay
Server
Teredo Server w/
2001:0:**c000:020a**::/64
as a Client IPv6 Prefix
Listening on UDP port
3544
IPv4
Internet
IPv4
IPv6
Teredo Tunnel
Home CPE
IPv4 Only Network
Teredo Client is
2001::**c000:20a**:a866:f227:39cc:9bf5
Public IPv4: 198.51.100.8/29
IPv4
IPv6
198.51.100.10 in HEX: **c6 33 64 0a**
Home RTR
w/ Cone
NAT
IPv4 only: 192.168.10.0/24
DNS: teredo.example.com A 192.0.2.10
192.168.10.211 in HEX: **c0 a8 0a d3**

Tunneling

So why do I recommend turning off all the transition tunneling technologies?

Unless you have a specific design build around them they introduce unpredictable behavior

Still leave IPv6 enabled, just disable the transition tunneling, that way when it is time to deploy IPv6 you don't have to touch the hosts again


Native
IPv6
Dual Stack
Tunneling
Translation

Translation

In Windows there is one main translation technology

NAT64/DNS64

Additional Enterprise translation technologies to know

SLB64 & NPTv6

These do IPv6 to IPv4 translation (think of them as a proxy)

NAT64/DNS64 and SLB64

NAT64/DNS64

Windows Server 2008 R2 with Forefront UAG is capable of doing NAT64/DNS64

Windows Server 2012/R2+ is capable of doing NAT64/DNS64 (DirectAccess)

NAT64 allows an IPv6 host to access an IPv4 host through a NAT64 devices that proxies the session

DNS64 builds synthetic AAAA records for an IPv4 A record if no IPv6 AAAA record already exists

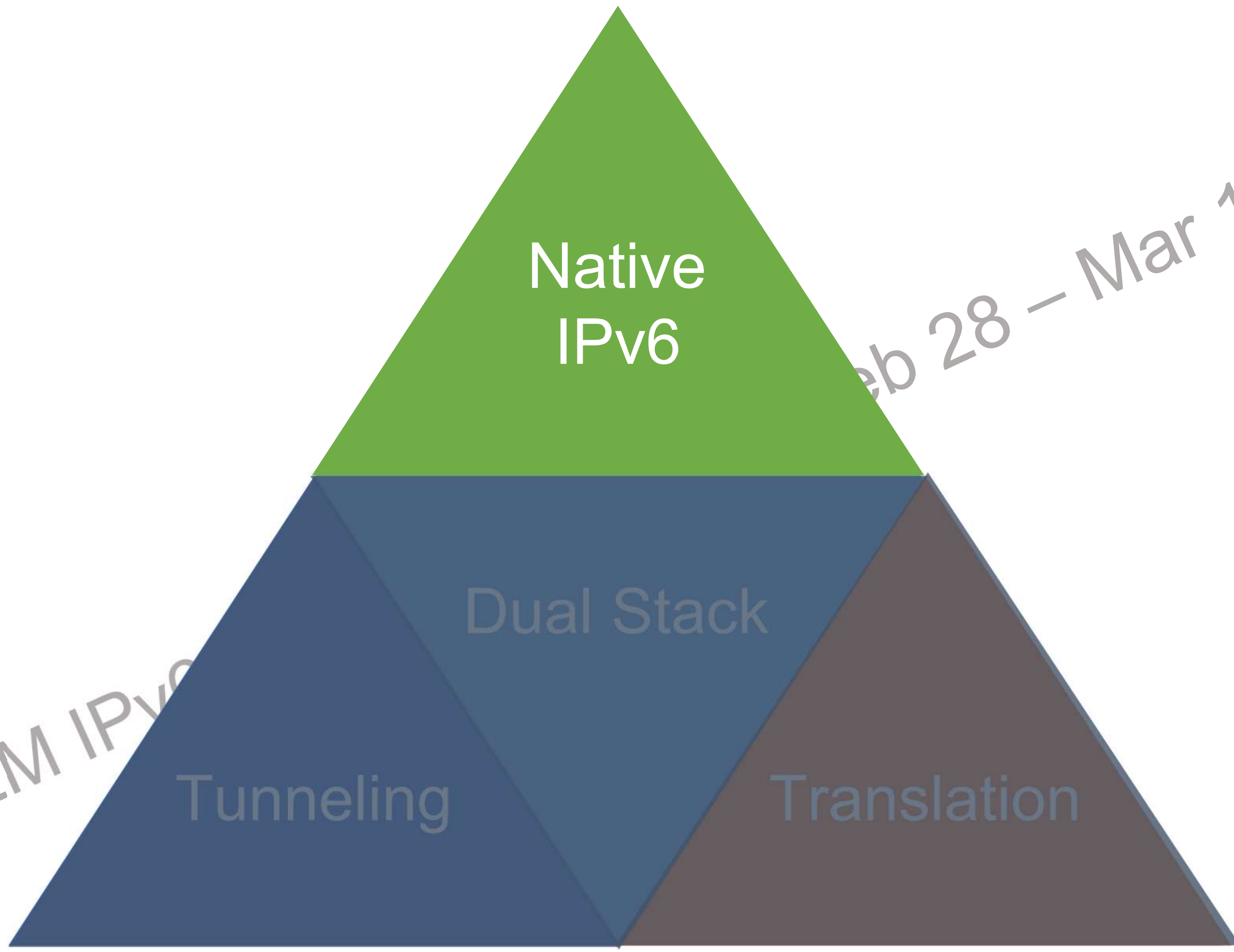
NAT64/DNS64 – Common Deployment

Run DNS64 on a commercial solution like Infoblox or Bluecat

Run NAT64 on a Cisco router or Palo Alto Networks firewall

It is possible to run it all in Windows. For a lab, that is fine, but do not use it for production


Future
Past


Native
IPv6
Dual Stack
Tunneling
Translation

Migration

You want to move away from

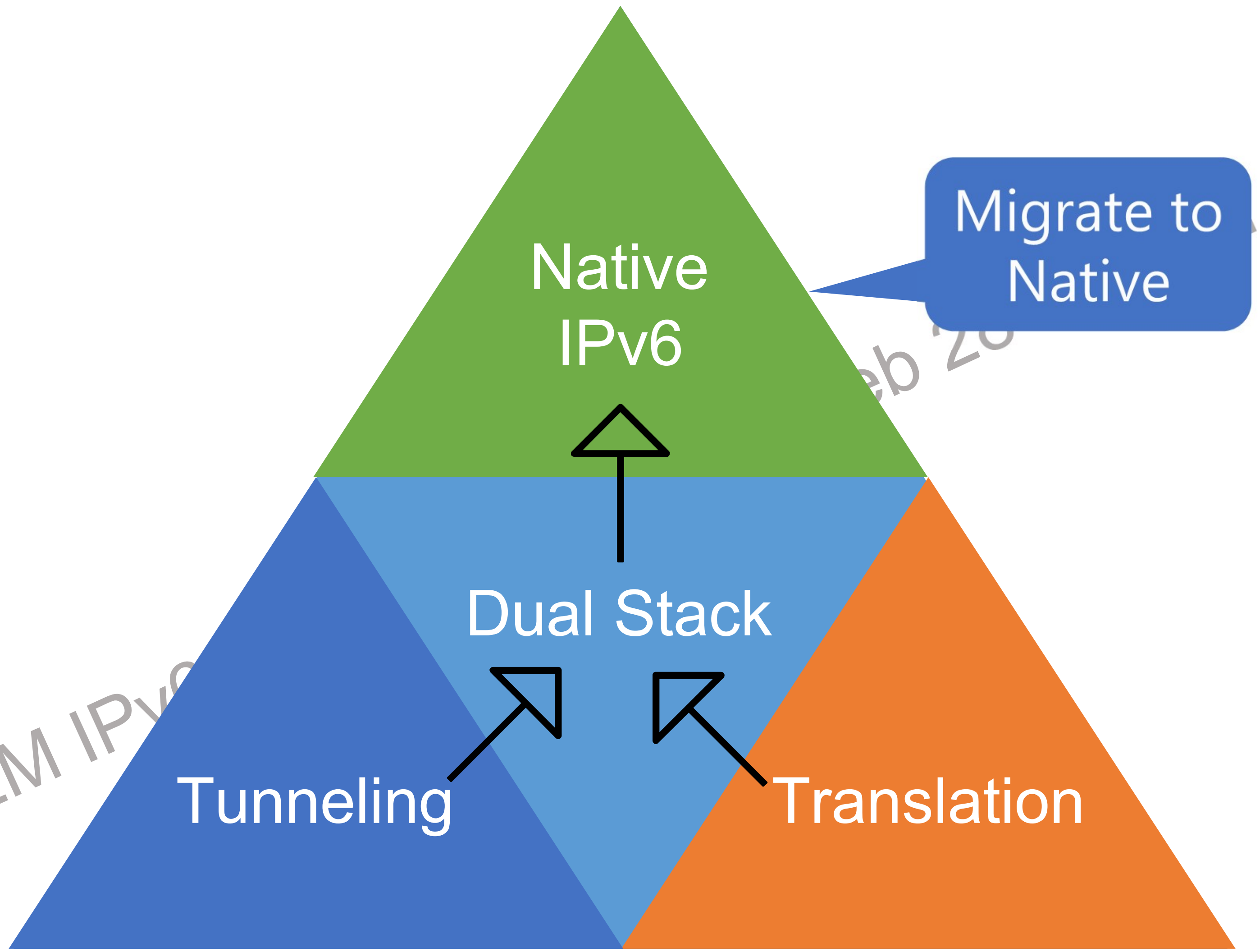
- Tunneling
- Translation

To Dual-Stack

Eventually you will want to move from Dual-Stack to:

Native

Translation (as needed) for older IPv4 only hosts



Native
IPv6
Migrate to
Native
Dual Stack
Tunneling
Translation

Disabling Transition Technologies

Turning off 6to4

PowerShell for Windows Server 2012R2, 2012, 8.1, 8

```
Set-Net6to4Configuration -State Disabled
```

Netsh for Windows Server 2008 R2, Windows Server 2008, Windows 7, Windows Vista

```
Netsh interface 6to4 set state disabled
```

Registry Key

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\tcpip6\Parameters\DisabledComponents registry value to 0x2 (DWORD)

(and then restart the computer)

Turning off 6to4

Windows Server 2012R2, 2012 and 2008 R2 Windows 8.1, 8 and 7

Set the 6to4 State Group Policy setting in Computer Configuration|Policies|Administrative Templates|Network|TCP/IP Settings|IPv6 Transition Technologies to Disabled

Windows Server 2016, 2019 and Windows 10 both with creators updates have 6to4 disabled

Turning off ISATAP

PowerShell for Windows Server 2012R2, 2012, 8.1, 8

```
Set-NetIsatapConfiguration -State Disabled
```

Netsh for Windows Server 2008 R2, Windows Server 2008, Windows 7, Windows Vista

```
Netsh interface isatap set state disabled
```

Registry Key

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\tcpip6\Parameters\DisabledComponents registry value to 0x4 (DWORD)

(and then restart the computer)

Turning off ISATAP

Windows Server 2012R2, 2012 and 2008 R2 Windows 8.1, 8 and 7

Set the ISATAP State Group Policy setting in Computer Configuration|Policies|Administrative Templates|Network|TCP/IP Settings|IPv6 Transition Technologies to Disabled

Windows Server 2016, 2019 and Windows 10 both with creators updates have ISATAP disabled

Turning off Teredo

PowerShell for Windows Server 2012R2, 2012, 8.1, 8

```
Set-NetTeredoConfiguration -Type Disabled
```

Netsh for Windows Server 2008 R2, Windows Server 2008, Windows 7, Windows Vista

```
Netsh interface teredo set state type = Disabled
```

Teredo client on managed networks (AD is present) is disabled by default

Turning off Teredo

Windows Server 2012R2, 2012 and 2008 R2 Windows 8.1, 8 and 7

Set the Teredo State Group Policy setting in Computer Configuration|Policies|Administrative Templates|Network|TCP/IP Settings|IPv6 Transition Technologies

Windows Server 2016, 2019 and Windows 10 both with creators updates have Teredo disabled

DHCPv6 - Windows

DHCPv6 in Windows

Once the DHCP service is installed it can function for both DHCP and DHCPv6

The main difference is you will need new values for DHCPv6 reservations

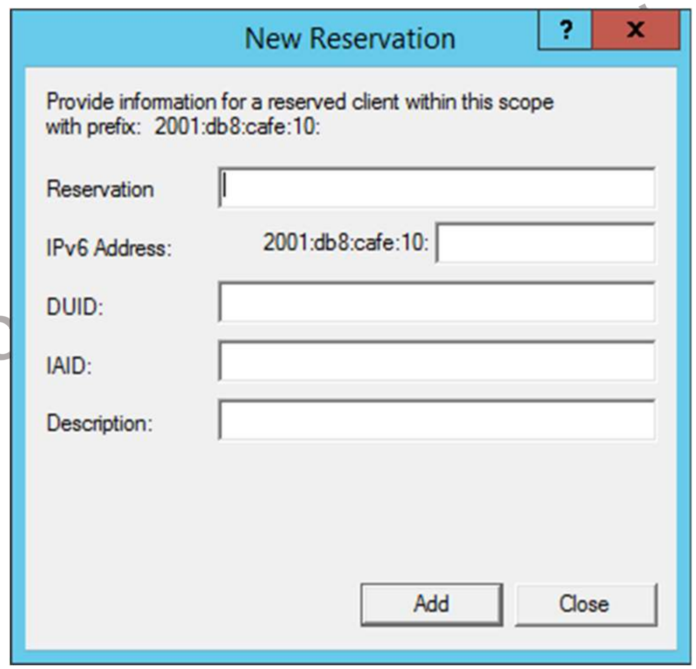
The option codes are different – they do NOT carry over from IPv4

DHCPv6 Reservation Parameters

You will need to following information from the host

DUID value

IAID value



DHCPv6 Problems

Sysprep images or duplicate DUID values will occur

Difficult to determine DUID and IAID values from hosts – they are generated at initial boot/runtime

Only knowing the MAC address will NOT allow you to build a reservation

Cisco IPv6 SLAAC + DHCPv6 w/ O flag config

```
interface Vlan7
 description - work - ipv6 address 2001:470:82a9:7::/64 and 2001:470:1f05:d37::/64
 ipv6 address 2001:470:1F05:D37:2::1/64
 ipv6 enable
 ipv6 nd prefix 2001:470:1F05:D37::/64 300 300
 ipv6 nd prefix 2001:470:82A9:7::/64 300 300
 ipv6 nd other-config-flag
 ipv6 dhcp server HE
```

Setting the O flag

Cisco IPv6 DHCPv6 w/ M & O flag config

```
interface Vlan7
 description - work - ipv6 address 2001:470:82A9:7::1/64
 ipv6 address 2001:470:82A9:7::1/64
 ipv6 enable
 ipv6 nd prefix 2001:470:82A9:7::/64 300 300
 ipv6 nd managed-config-flag
 ipv6 nd other-config-flag
 ipv6 dhcp server HE
```

Setting the M & O flag

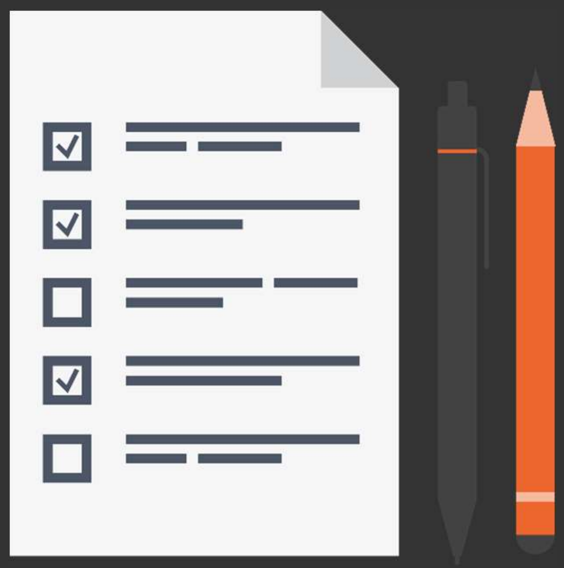
Cisco IPv6 SLAAC config

```
interface Vlan7
 description - work - ipv6 address 2001:470:82a9:7::/64
 ipv6 address 2001:470:82A9:7::1/64
 ipv6 nd prefix 2001:470:82A9:7::/64 300 300
 ipv6 enable
```

By default the A flag is enabled

Windows IPv6 DHCPv6 Demo

- Lab #3
- Windows Server 2016R2
- Check binding for service
- Build a IPv6 scope
- What does the host OS get?



Best Practices for Windows

Windows Logging Best Practices

Logging may have issues with AAAA records

You should store addresses in binary format (avoid integer or stings)
You should normalize the IPv6 address format (expand the address, no compression)

You many have to aggregate logs

Depending on your logging, you many have two different files for the same protocol (i.e. port 80)
You will have to aggregate the logs for business analytics

Splunk supports IPv6

Many deployments use Splunk which does the expected behavior and regular expression processing

Deployment

Start with forgiving applications

DNS and SMTP vs. Web Services

You need to understand

Dual Stack behavior

Transition technology behavior

Impact on your hardware and software

Remember – IPv6 is used FIRST so mistakes have immediate impact!

Do the heavy lifting in the lab – don't try out IPv6 in your production network!

Windows IPv6 Caveats and Notes

- Windows will use 6to4 if it has a public IPv4 address
- Make sure to add your IPv6 Prefixes to AD Sites and Services/Sites/Subnets
- Older Windows does NOT support RFC 6106/8106 (RDNSS)
- Windows has a native DHCPv6 client
- Windows 10 currently prefers the DHCPv6 supplied DNS over the RDNSS one, per Section 5.3.1 of the RFC 8106

Exchange and IPv6

Exchange 2007 and 2010 will prefer IPv6 for same subnet traffic

Even if you only have static IPv4 addresses configured
You likely didn't even notice unless you disable IPv6

Exchange Edge Transport role is a special IPv6 case

If you have it running in a DMZ that uses Public IPv4 addresses it will attempt to use 6to4
This means if someone has a AAAA record published for their MX in DNS the Exchange server will FIRST use 6to4 to attempt delivery
This behavior is likely NOT desirable – disable 6to4 on the Exchange Server

Exchange and IPv6

If IPv6 is enabled and is being routed properly make sure your advanced firewall is ON!

Make sure to test reachability from all networks prior to turning up IPv6 for client access networks

Test all Exchange services for both IPv4 and IPv6

If you are using IPv4 Load Balancers make sure you do the same for IPv6 or you might get strange behavior in clients

What Does Microsoft Recommend?

Microsoft no longer tests their software with IPv4 ONLY networks, the expect dual stack

Microsoft has standardized on dual stack support

There are only three products that have been delayed in broad IPv6 support – all are no longer current products

Forefront TMG, Lync, Windows Phone 7

Impacts of disabling IPv6

Recent anecdotal information about Microsoft support cases indicates customers disabling IPv6 and then having operational problems has grown

Make sure you test the planned configuration with IPv6 disabled in a lab

Do NOT randomly turn it off in your production environment – bad things can happen!

Troubleshooting IPv6 in Windows

Windows DNS Problems

Applications may have issues with AAAA records

You have to test your application to see if it supports AAAA records or if it fails

You also should test if hosts can fail back and forth between IPv4 and IPv6

Getaddrinfo()

Will responded with an ordered list – apps will walk the list until it can connect

http://msdn.microsoft.com/en-us/library/windows/desktop/ms738520%28v=vs.85%29.aspx

Not all hosts require an AAAA record

Decide if you want AD DDNS for host registration, decide if you want a host to respond on IPv6, etc.

GUA permanent, ULA and ISATAP will register in DDNS

Windows and NCSI

Network Connection Status Indicator

Determines if you are connected to the Internet

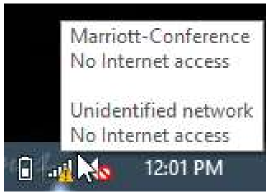
Used to detect IPv6 connectivity

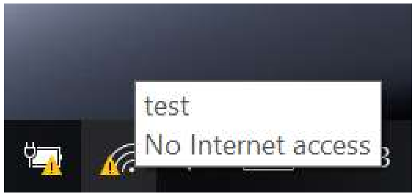
http://ipv6.msftncsi.com/ncsi.txt

```
Get-NCSIPolicyConfiguration
Set-NCSIPolicyConfiguration –CorporateWebsiteProbeURL http://ipv6.ncsi.example.com –
PolicyStore Howfunky
Reset-NCSIPolicyConfiguration –PolicyStore Howfunky
```

Network Connectivity Status Indicator

Windows 7 and Server 2008R2 w/ Readiness Update to current release uses NCSI
http://support.microsoft.com/kb/2750841
Every time a network configuration event occurs (meaning that something has changed in the network status), NCSI performs several tests
If IPv6 access can't be established, NCSI uses IPv4
Value is cached for 30 days





Windows IPv6 Name Resolution

Windows IPv6 DNS

IPv6 DNS

An IPv6 address record type in DNS is AAAA

The PTR zone is IP6.ARPA.

Why is it AAAA?

An A record in IPv4 represents 32 bits

IPv6 has 128 bits so 4 A's make it 128 (4x32)

IPv6 Reverse DNS

Examples:

2001:470:1f05:d37:202c:7674:c4c2:4fa8
2001:470:1f05:d37::1

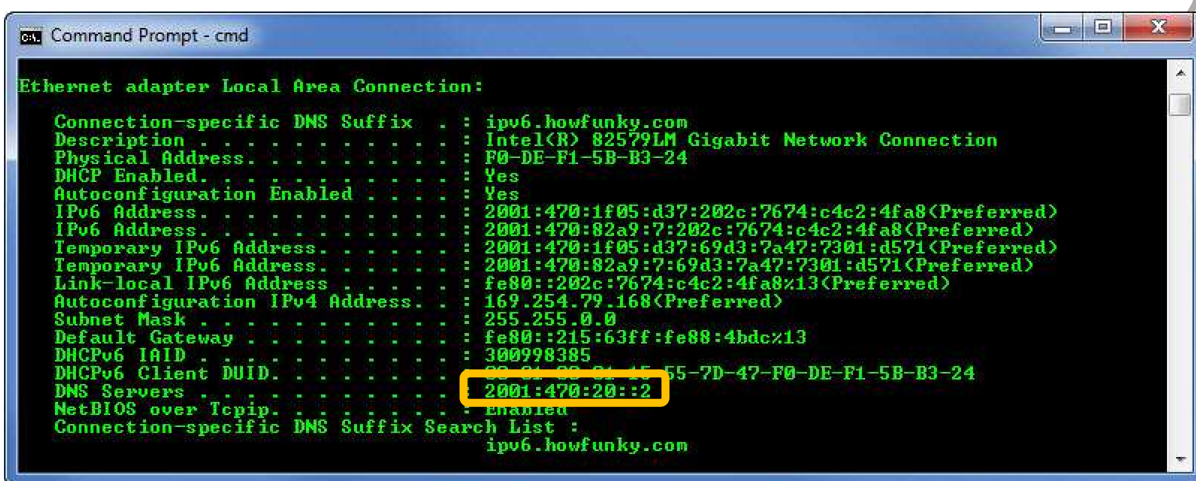
The reverse entry in IP6.ARPA. looks like:

8.a.f.4.2.c.4.c.4.7.6.7.c.2.0.2.7.3.d.0.5.0.f.1.0.7.4.0.1.0.0.2.IP6.ARPA.
1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.7.3.d.0.5.0.f.1.0.7.4.0.1.0.0.2.IP6.ARPA.

Every nibble must be included so:

Zero compression has to be expanded back out
Leading zeros must be added back in

How Does a Windows Client get DNS Servers?



DNS Servers are obtained via DHCPv6 w/ O or M flag

If there is no DHCPv6 then:

the DNS is manually configured or via RDNSS

Windows DNS Behavior

A permanent IPv6 address is registered by DDNS for an Active Directory joined hosts (server or client)

Every interface of a host that has a permanent IPv6 address will be registered

Windows will prefer to use an IPv6 DNS name caching resolvers

Windows DNS Behavior

Windows can leverage DNS64/NAT64 natively in Server 2012 R2

Windows will register Global Unicast and ULA (ISATAP if they are in use)

Window will not register temporary, 6to4, Teredo, loopback and link-local

Exception is DirectAccess which will register Teredo or IP-HTTPS with DDNS

Other IPv6 Name Resolution Methods

Windows can also use Link Local Multicast Name Resolution (LLMNR), mDNS, and WS-Discovery

Similar to Apple's mDNS, it is a Zeroconf multicast solution – RFC 4795

WS-Discovery and SOAP-over-UDP https://learn.microsoft.com/en-us/windows/win32/wsdapi/ws-discovery-specification-compliance

Windows also uses IPv6 Literals

Created because Windows cannot use colons in a UNC pathname

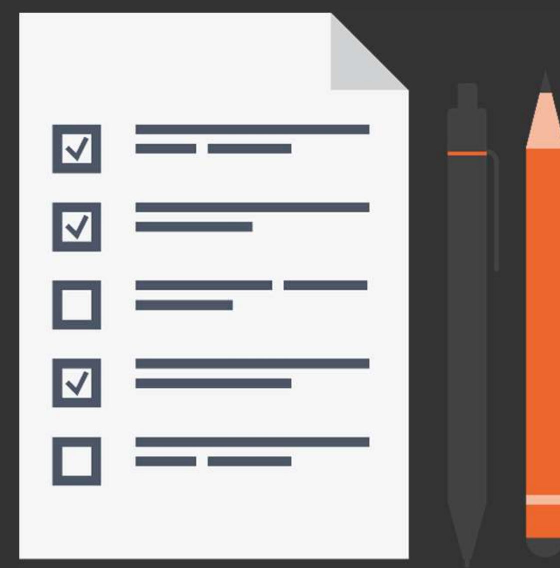
Replace the ":" with a "-" and append ipv6-literal.net to the end

Name Resolution Policy Table (NRPT)

Used by DirectAccess and DNSSEC

Windows IPv6 DNS Demo

- Windows Server 2016R2
- Check binding for service
- Build a AAAA record
- Does automatic PTR work?
- What host address gets entered?



Sample DNS PowerShell cmdlets

DNS Client cmdlets

```
Get-DnsClientServerAddress
Set-DnsClientServerAddress
Resolve-DnsName
```

DNS Server Role and cmdlets

```
Add-WindowsFeature DNS
Get-WindowsFeature DNS
```

```
Add-DnsServerPrimaryZone
Add-DnsServerResourceRecordAAAA
Remove-DnsServerResourceRecord
Get-DnsServerGlobalQueryBlockList
Get-DnsServerRootHint
Import-DnsServerRootHint
Add-DnsServerRootHint
Remove-DnsServerRootHint
Get-DnsServerSetting
Test-DnsServer
```

Windows DNS Name Resolution with IPv6

Sample PowerShell cmdlets:

```
Resolve-DnsName -Name www.cav6tf.org -Type AAAA -Server 2001:4860:4860::8888
```

```
$rn = Resolve-DnsName -Name www.cav6tf.org -Type AAAA
Find-NetRoute -RemoteIPAddress $rn.ip6address
```

Windows IPv6 Literals

Windows IPv6 Literals

Used as a hostname resolvers, specifically for UNC pathnames because they do not support “:”

Replace “:” with a “-” and append ipv6-literal.net

2001:470:0:0:11a::403e:a5c5 becomes

2001-470-0-0-11a--403e-a5c5.ipv6-literal.net

Replace “%” with “s” with link-local addresses

fe80::6085:be82:8442:9b54%4 becomes

fe80--6085-be82-8442-9b54s4.ipv6-literal.net

Windows IPv6 Literals Continued

Microsoft used to have ipv6-literal.net registered – they let it go (no idea why) so this is just an internal OS function for the OS to resolve IPv6 to UNC pathnames

So, effectively ipv6-literal.net is a special domain name space that is local to the OS

DOE-LM IPv6 Host OS Class 1 - Feb 28 - Mar 1, 2023

Windows IPv6 LLMNR and mDNS

Windows Link-local Multicast Name Resolution

LLMNR is used for IPv4 and IPv6 peer-to-peer

Multicast address ff02::1:3 with UDP port 5355 is used for LLMNR

Similar to mDNS (ff02::fb) from Apple – Microsoft's Zeroconf standard implementation

Windows mDNS

mDNS is used for IPv4 and IPv6 peer-to-peer

Multicast address ff02::fb

RFC 6762 from Apple, think bonjour (used to be called rendezvous) that allows AirPlay, AirDrop, and AirPrint

Apple OSX Networking

OSX Networking

Since Mac OS X v10.2 (Jaguar – May 2002) Apple has had IPv6 support (in some form) in the OS

If you are using older versions of OSX you will likely have unpredictable behavior even among sub-releases of a major release. There was a lot of work being done and the regression testing was not as good as it should have been.

IPv6 support was variable until 10.6.7 (Snow Leopard – August 2009)

Essentially the first “usable” and “predictable” OS version with IPv6

OSX IPv6 Address Configuration

DHCPv6 client support was added in 10.7 (Lion – July, 2011)

IPv6 privacy addresses are enabled by default in 10.7

Automatically (SLAAC & DHCPv6), *Manually*, and *Link-local only* configurations are supported (System Preferences > Network > Advanced > TCP/IP > Configure IPv6)

OSX Manual IPv6 Configuration

Choose Apple menu > System Preferences, and then click Network

If the Network Preference is locked, click on the lock icon and enter your Admin password to make further changes

Choose the network service you want to use with IPv6, such as Ethernet or AirPort.

Click Advanced, and then click TCP/IP

Click on the Configure IPv6 pop-up menu (typically set to Automatically) and select Manually

Enter the IPv6 address, router address, and prefix length you received from your network administrator or Internet service provider. Your router address may be referred to as your gateway address by some ISPs

Reference: https://support.apple.com/guide/mac-help/use-ipv6-on-mac-mchlp2499/mac

OSX Disabling IPv6

- Open System Preferences and click the Network icon.
- Select the network you are using - It will usually be labeled Wi-Fi.
- Click the Advanced.. button.
- Click the TCP/IP tab.
- Select the Configure IPv6 drop-down menu and set it to Link-local only.
- Click the OK button.

Reference: https://support.blindsidenetworks.com/hc/en-us/articles/360050168052-How-to-disable-IPV6

Mac OS X Server

Mac OS X Server shares many components with Mac OS X however it has unique IPv6 issues. These issues are documented in the Network Services Administration guide for Leopard

Services that support IPv6 (DNS, firewall, POP, IMAP, SMTP, AFP, HTTP)
There is no support in the Server Admin GUI for IPv6

Reference: https://wikispaces.psu.edu/display/ipv6/IPv6+on+OS+X

Mac OS X Server continued

The FTP daemon on OS X server does not support IPv6 (see Apple bug 3060157). Specifically, it does not support RFC 2428, which defined IPv6 extensions for IPv6. Despite this, the FTP daemon on OS X Server will bind to an IPv6 socket. So an IPv6-enabled FTP client will connect to the FTP daemon but will be unable to perform basic tasks. (see Apple bug 3060157).
The graphical firewall rule editor in Server Admin can only create rules for IPv4. If you manually create IPv6 rules using i6fw(8), those rules will be deleted the next time the graphical firewall rule editor is launched.
The graphical DNS editor does not allow AAAA records or ip6.arpa PTR records. The named daemon does bind to an IPv6 socket on 10.5 Server, however. On 10.4 Server, it only listens on IPv4.

Reference: https://wikispaces.psu.edu/display/ipv6/IPv6+on+OS+X

OSX Multicast

Apple utilizes mDNS (RFC 6762) for name resolution for AirPlay, AirPrint and other Bonjour services

mDNS will work for both IPv4 and IPv6 but like all modern OSs, OSX will prefer IPv6 – this can cause issues for IPv4 only Bonjour proxy services

Details at http://en.wikipedia.org/wiki/Multicast_DNS

References

Test IPv6 - https://test-ipv6.com/broken.html

DOD - https://www.hpc.mil/index.php/2013-08-29-16-03-23/networking-overview/2013-10-03-17-24-38/ipv6-knowledge-base-ip-transport/enabling-ipv6-in-mac-os-x

Troubleshooting IPv6 in OSX

OSX IPv6 Network Troubleshooting

OSX uses RFC 6555/8305 in the core OS

This means ALL applications use this behavior
It can be incredibly hard to know what is going on for some applications
You will need a packet sniffer in some cases to figure it out

OSX will use both IPv6 and IPv4

Due to Happy Eyeballs it will prefer IPv6
It will test continuously and flop back and forth
It is focused on user experience, not easy of troubleshooting!
Some applications ignore the getaddrinfo() sequence order

OSX IPv6 mDNS Troubleshooting

OSX prefers IPv6 for mDNS

This means it will try IPv6 multicast for mDNS (ff02::fb) process first
There can be a long fallback timeout
It will cache entries – so once it works with IPv4 multicast (224.0.0.251) it will stay on IPv4

OSX mDNS will impact wireless networks

Because wireless is a broadcast media
mDNS is multicast – keep the number of hosts to a minimum
Tune your wireless network accordingly

Linux Networking

Linux Networking

Since Kernel version 2.6.12 (2005) Linux has had IPv6 support, in some form, in the OS

If you are using older versions of Linux you will likely have unpredictable behavior even among sub-releases of a major release. Don't be surprised if it doesn't work at all or you get strange behavior.

IPv6 support is variable between distributions

Pay attention to distribution notes (Debian vs. Redhat vs. others)
Recent distributions should universally have IPv6 support, they vary in commands to implement (same issue w/ general networking commands)

Linux IPv6 Address Configuration

DHCPv6 client support is mixed for different versions – your mileage will vary

Edit /etc/dhcp/dhclient.conf to modify behavior or turn off DHCPv6 client

IPv6 temporary addresses are enabled by default in Ubuntu server and client in 12.04 and 14.04 LTS

SLAAC and manual address is supported

Linux IPv6 Address Configuration

The move to **systemd** on Linux has changed the network interface names. You no longer have the traditional eth0, eth1, wlan0, etc.

VMware has a bug and will give random long numbers to interfaces: enoXXXX or ensXXXX

Older systems may use GUA as Default Gateway

Linux IPv6 Basics

Ubuntu – check for an IPv6 address:

```
ip -6 addr show dev eth0
cat /proc/net/if_inet6
```

Ubuntu – check for IPv6 temporary addresses:

```
sudo sysctl –a | grep tempaddr
```

Ubuntu – check for your IPv6 address

```
Ifconfig eth0 | grep "inet6 addr:"
```

Ubuntu – check for IPv6 neighbors:

```
ip -6 neigh show
```

Linux Manual IPv6 Address Configuration

Ubuntu – set up a manual IPv6 address:

```
sudo vi /etc/network/interfaces
iface eth0 inet6 static
address 2001:db8:<prefix>::ff
netmask 64
up ip -6 route add 2001:db8:<prefix>::<rtr> dev eth0
down ip -6 route del 2001:db8:<prefix>::<rtr> dev eth0
gateway 2001:db8:<prefix>::<rtr>
```

Notice the default gateway is static and using the global unicast IPv6 address

Linux IPv6 Prefix Behavior

Linux still just uses route to manage everything:

```
Route -A inet6 -n
```

Linux will use the link-local IPv6 address as the next hop with SLAAC and DHCPv6 (RA)

If you set up things Manually double check your default gateway (consider HA for the gateway)

Linux IPv6 Host Behavior Configuration

Edit the /etc/sysctl.conf file to change host behavior:

```
#turn on IPv6 forwarding (not routing per say)
net.ipv6.conf.all.forwarding=1
#turn off auto configuration (SLAAC)
net.ipv6.conf.all.autoconf=0
#turn off RA learning - don’t recommend
net.ipv6.conf.all.accept_ra=1
```

For manual configuration use the above settings

Linux IPv6 Behavior Continued

Ubuntu - Turning off privacy addressing:

```
# Disable IPv6 Privacy Extensions
net.ipv6.conf.all.use_tempaddr = 0
net.ipv6.conf.default.use_tempaddr = 0
```

Turning it back on:

```
# Enable IPv6 Privacy Extensions
net.ipv6.conf.all.use_tempaddr = 2
net.ipv6.conf.default.use_tempaddr = 2
```

Linux Disabling IPv6

Ubuntu - you can disable IPv6 by editing:

```
sudo vi /etc/sysctl.conf

#disable IPv6
net.ipv6.conf.all.disable_ipv6 = 1
net.ipv6.conf.default.disable_ipv6 = 1
net.ipv6.conf.lo.disable_ipv6 = 1
```

Restart the network interfaces

```
sudo sysctl -p
sudo ifdown eth0
/etc/init.d/networking restart
sudo ifup eth0
```

Linux Server w/ Tunnelbroker

Connect directly to Tunnelbroker.net as long as it has a public IPv4 address and access to the Internet

```
sudo vi /etc/network/interfaces

auto he-ipv6
iface he-ipv6 inet6 v4tunnel
     endpoint <public IPv4 address of Linux host>
     address  2001:470:<prefix from HE>::2
     netmask  64
     up ip -6 route add default dev he-ipv6
     down ip -6 route del default dev he-ipv6
sudo ifup he-ipv6
```

Ubuntu Linux Server as an IPv6 Router

By default Ubuntu Server is NOT a router so edit:

`/etc/sysctl.conf`

Uncomment the follow line:

`net.ipv6.conf.default.forwarding=1`

You can then install radvd:

`Sudo aptitude install radvd`

Edit the radvd file, but you need to know the prefix information prior to editing the file

Ubuntu Linux Server radvd setup

Edit the radvd file if it exists, if not, you will have to create it and then add (sample config info):

```
sudo vi /etc/radvd.conf
interface eth0
{
  AdvSendAdvert on;
  prefix 2001:db8:a:b::/64
  {
  AdvOnLink on;
  AdvAutonomous on;
  };
};
```

Ubuntu Linux Server Run radvd

After the edits you have to restart the service:

```
sudo etc/init.d/radvd restart
```

Caution:

If you do this the interface will start doing RA's for the prefix it is configured for per interface – make sure this is the behavior you actually want

Ubuntu Linux Server Miredo Service

It is possible to run a Teredo service on Linux:

```
sudo aptitude install miredo
```

This will create a Teredo tunnel (IPv6 over UDP IPv4):

You will see an IPv6 address with a prefix of 2001::/32

Linux Multicast

Multicast is fully supported, and you can use it to discovery information

```
ping6 ff02::1%eth0
```

You can ping a GUA and ask for a link-local

```
ping6 -I <GUA> ff02::1
```

Details at http://en.wikipedia.org/wiki/Multicast_DNS

Linux Commands for IPv6 Address

Basic commands

```
ping6
traceroute6
route –A inet6
netstat -nr -6
dig @2001:4860:4860::8888 www.cav6tf.org AAAA
ssh root@fe80::<lower64>%eth0
ssh root@<prefix:address>
```

How to reference:

http://tldp.org/HOWTO/html_single/Linux+IPv6-HOWTO/
https://wiki.kubuntu.org/IPv6

Linux Prefix Policy Table

It is possible to modify the prefix policy table

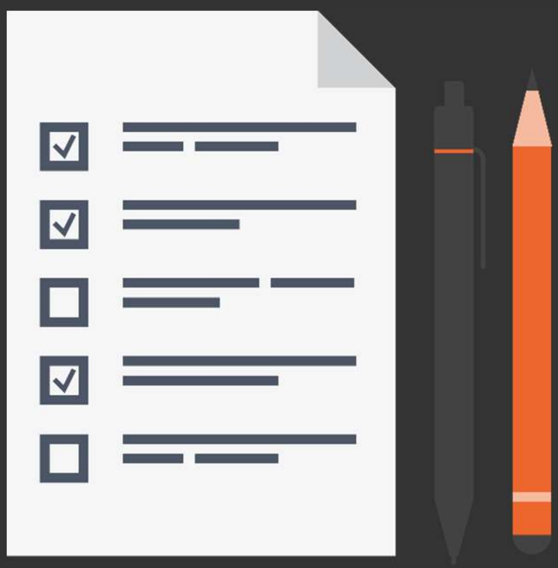
/etc/gai.conf

The default labels:

```
#label ::1/128       0
#label ::/0          1
#label 2002::/16     2
#label ::/96         3
#label ::ffff:0:0/96 4
#label fec0::/10     5
#label fc00::/7      6
#label 2001:0::/32   7
```

IPv6 Linux Server Configuration

- Basic Ubuntu Server Setup
- Check IPv6 configuration
- Confirm IPv6 works



CentOS Networking

CentOS IPv6 Address Configuration

DHCPv6 client support is mixed for different versions – your mileage will vary

Edit /etc/dhcp/dhclient.conf to modify behavior or turn off DHCPv6 client

IPv6 addresses in CentOS are enabled by default in server and client but utilize EUI-64, no temporary

SLAAC and manual address is supported

CentOS IPv6 Basics

CentOS – check your version supports IPv6

```
/etc/sysconfig/network-scripts/network-functions-ipv6
```

CentOS – check for an IPv6 address:

```
ip -6 addr show dev eth0
cat /proc/net/if_inet6
```

CentOS IPv6 Address Configuration

CentOS – turn on IPv6:

```
cat /etc/sysconfig/network
NETWORKING=yes
NETWORKING_IPV6=yes
HOSTNAME=YOURHOSTNAME
GATEWAY=YOURGATEWAY
```

Notice the default gateway is static and using the global unicast IPv6 address

Restart your interfaces:

```
/etc/init.d/network restart
services networks restart
```

CentOS IPv6 Manual Configuration

CentOS – edit:

```
cat /etc/sysconfig/network-scripts/ifcfg-eth<x>
IPV6INIT=YES
IPV6_AUTOCONF=NO
IPV6ADDR=<prefix>/64
```

You can also add it directly:

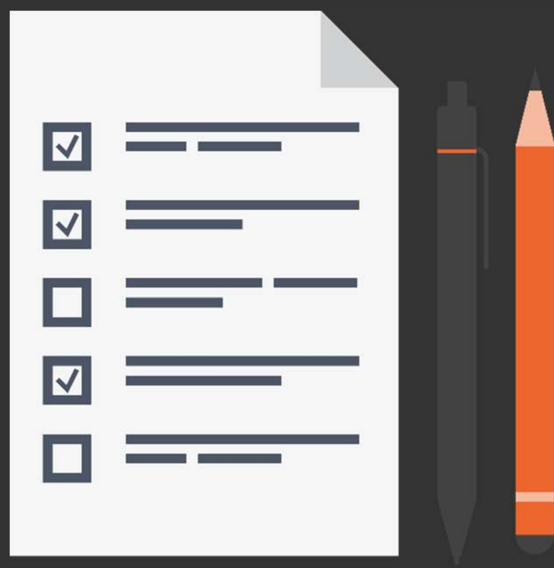
```
ifconfig eth0 inet6 add <prefix>/64
route -A inet6 add default gw <prefix>
```

Or:

```
ip -f inet6 addr add <prefix>/64 dev eth0
ip -f inet6 route add default via <prefix>
```

IPv6 Linux Server Configuration Demo

- Basic CentOS Server Setup
- Check IPv6 configuration
- Confirm IPv6 works



Troubleshooting IPv6 in UNIX

Linux IPv6 Network Troubleshooting

Linux uses RFC 6555/8305 in the core OS and browser

This means ALL applications use this behavior
It can be incredibly hard to know what is going on for some applications
You will need a packet sniffer in some cases to figure it out

Linux will use both IPv6 and IPv4

Due to Happy Eyeballs it will prefer IPv6
It will test continuously and flop back and forth
It is focused on user experience, not easy of troubleshooting!
Some applications ignore the getaddrinfo() sequence order

Linux Problems

Limited routing table size

4096 currently

It caches all connections in the routing table

This can cause performance problems
Lots of garbage collection for more than 512 entries
Therefore lots of churn

IP -6 route show

This can be really slow – duplicate entries, too many adjoining entries

Linux Problems Continued

If you change the options, it may require a reboot

options ipv6 disable = 1

Confirm you have turned off SLAAC before you start troubleshooting

options ipv6 autoconf = 0

Check to see if you have multiple GUA on eth0

Avoid this – causes application and monitoring issues. It is fine to have link-local and global unicast but having an additional ULA or GUA puts you at the mercy of RFC 6724 or 3484 and knowing which the OS actually uses!

Linux Application Problems

Curl

You have to use brackets in the address [2001:db8:a:b::1] or it will fail

You need a current version of curl – there are a lot of bugs still, they are fixing them

MySQL 5.6 and newer

Older versions do NOT support IPv6 for either client or server connections

MTAs will require DNS and AAAA records

You need to understand the application use cases but things like SMTP will attempt to use whatever DNS tells it so if you publish AAAA records then IPv6 needs to work or it will fail

Linux DNS Problems

Applications may have issues with AAAA records

You have to test your application to see if it supports AAAA records or if it fails

You also should test if hosts can fail back and forth between IPv4 and IPv6

Getaddrinfo(3)

Will responded with an ordered list – apps will walk the list until it can connect

http://linux.die.net/man/3/getaddrinfo

Not all hosts require an AAAA record

Decide if you want DDNS for host registration, decide if you want a host to respond on IPv6, etc.

Linux Regex Problems

String matching will likely fail

"2001:db8:a::1" != "2001:0db8:000a:0000:0000:0000:0000:0001"

Formatting will impact you

Different application methods do it slightly differently, Java InetAddress vs glibc for instance

There is no one right way with regex

There are many ways to use regex with IPv6

Beware code that uses a colon ":" as a delimiter to split an IP address from a host port number!!!

Remember that an IPv6 address may be passed in [ ] so you have to account for that in your regex (or what URI/URL string you return)

Mobile Linux Problems

Android has no IPv6 DHCP client!

At least right now – this may change

Android suffers from the same RFC 6555/8305 issues

But behavior is different on cellular vs. Wi-Fi

Android apps may not process getaddrinfo(3)

Some apps are IPv4 only and will ignore all IPv6 entries

Specific UNIX Lab Considerations

SLAAC vs. Static Assignments

SLAAC may work for a quick lab or temporary configuration, otherwise, don't use it

SLAAC will change it's IPv6 address if the NIC's change, even for the same host, not ideal

For a permanent lab configuration use Manual static address configuration

IPv6 Address Assignments

Do no do IPv4 to IPv6 address embedding

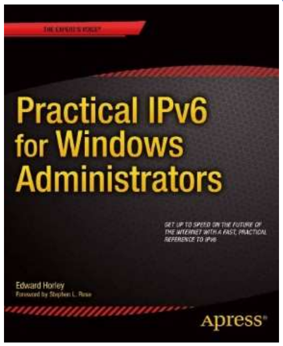
There is no reason to encode IPv4 address information into an IPv6 address unless the IPv4 address is a public one – otherwise just don't do it

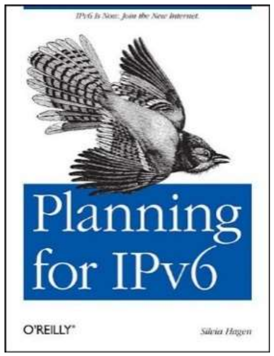
This method breaks when you don't have IPv4 anymore – like an IPv6 only network

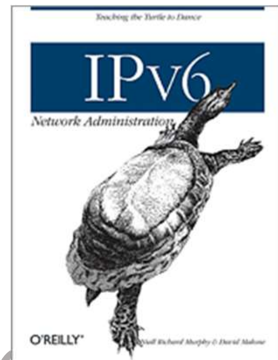
Q&A

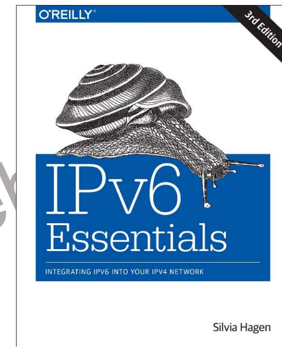
Resources

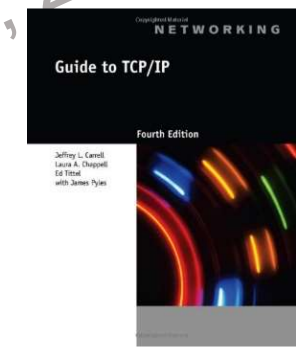
IPv6 Technical Resources

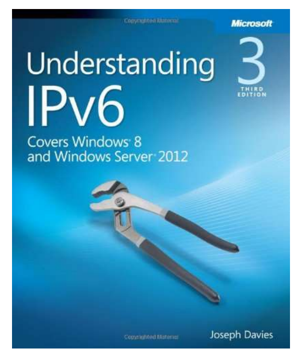


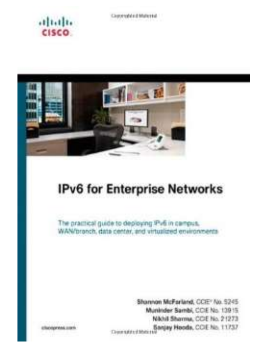




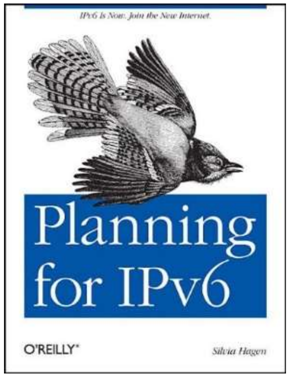


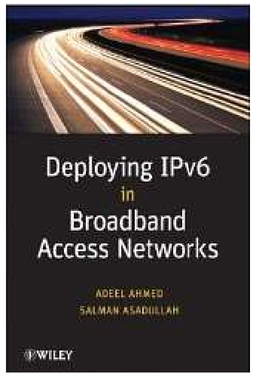


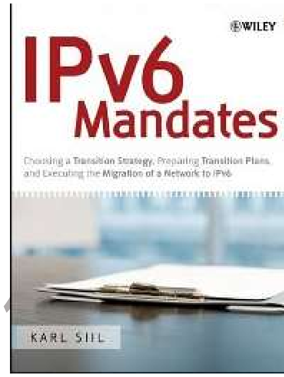


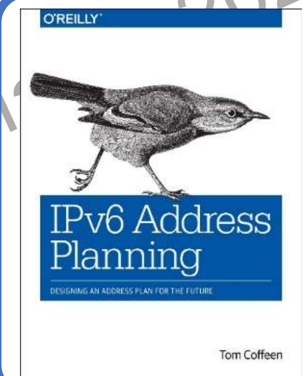


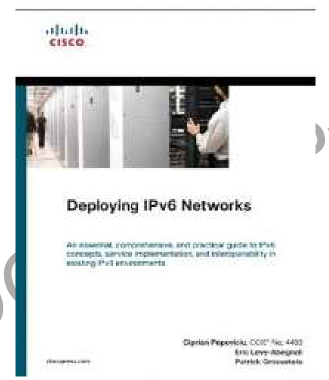
IPv6 Deployment Resources

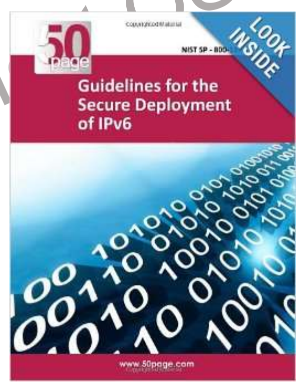


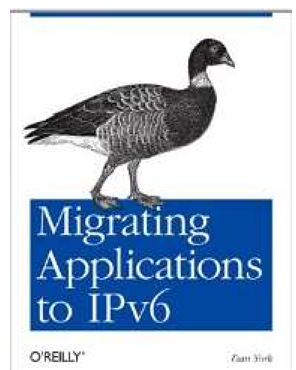


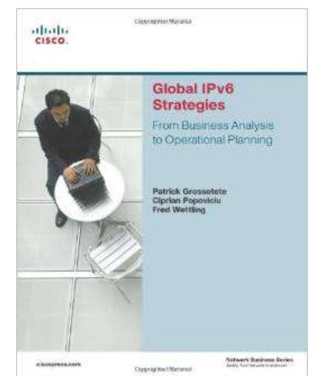




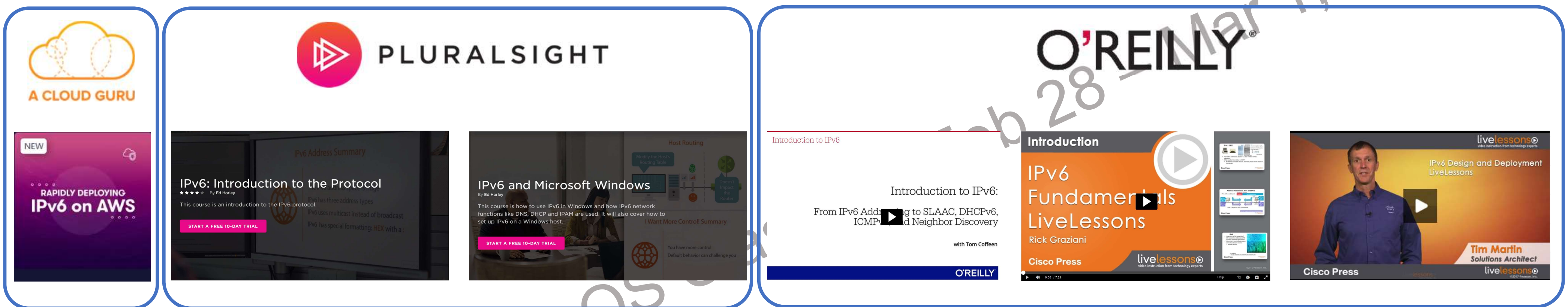








IPv6 Online Training Resources



[IPv6 on AWS - https://acloud.guru/learn/aws-ipv6](https://acloud.guru/learn/aws-ipv6)

[IPv6: Introduction to the Protocol - https://www.pluralsight.com/courses/ipv6-introduction-to-protocol](https://www.pluralsight.com/courses/ipv6-introduction-to-protocol)

[IPv6 and Microsoft Windows - https://www.pluralsight.com/courses/ipv6-microsoft-windows](https://www.pluralsight.com/courses/ipv6-microsoft-windows)

[Introduction to IPv6 - https://player.oreilly.com/videos/9781771375269](https://player.oreilly.com/videos/9781771375269)

[IPv6 Fundamentals LiveLessons - https://player.oreilly.com/videos/9781587204579](https://player.oreilly.com/videos/9781587204579)

[IPv6 Design and Deployment - https://www.oreilly.com/library/view/ipv6-design-and/9780134655529/](https://www.oreilly.com/library/view/ipv6-design-and/9780134655529/)

IPv6 Book Reference

- *IPv6 Fundamentals: A Straightforward Approach to Understanding IPv6 2nd Edition* by Rick Graziani (Cisco Press, 2017)
- *IPv6 Address Planning* by Tom Coffeen (O'Reilly Media, 2014)
- *IPv6 Essentials, Third Edition* by Silva Hagen (O'Reilly Media, 2014)
- *Practical IPv6 for Windows Administrators* by Ed Horley (Apress, 2013)
- *Understanding IPv6 Third Edition* by Joseph Davies (Microsoft Press, 2012)
- *IPv6 in Enterprise Networks* by Shannon McFarland, Muninder Sambi, Nikhil Sharma, and Sanjay Hooda (Cisco Press, 2011)
- *Planning for IPv6* by Silvia Hagen (O'Reilly Media, 2011)
- *DNS and BIND on IPv6* by Cricket Liu (O'Reilly Media, 2011)
- *Day One: Exploring IPv6* by Chris Grundermann (Juniper Networking Technologies Series, 2011)
- *IPv6 Network Administration* by Niall Richard Murphy and David Malone (O'Reilly Press, 2009)
- *IPv6 Security* by Scott Hogg and Eric Vyncke (Cisco Press, 2008)
- *IPv6 Essentials, Second Edition* by Silva Hagen (O'Reilly Media, 2006)
- *Deploying IPv6 Networks* by Ciprian Popoviciu, Eric Levy-Abegnoli, Patrick Grossetete (Cisco Press, 2006)
- *Running IPv6* by Iljitsch van Beijnum (Apress, 2005) (an older book but a great reference)
- *Global IPv6 Strategies: From Business Analysis to Operational Planning* by Patrick Grossetete, Ciprian Popoviciu, and Fred Wettling (Cisco Press, 2004)

Other IPv6 Reference

Infoblox IPv6 Center of Excellence - https://blogs.infoblox.com/category/ipv6-coe/

Packet Pushers IPv6 Buzz Podcast - https://packetpushers.net/series/ipv6-buzz/

Wikipedia IPv6 page - https://en.wikipedia.org/wiki/IPv6

IPv6 Forum

Exams

IPv6 Forum Certifications Expertise & Skills Logo Programs

http://www.ipv6forum.com/

Thank You

Survey – it is short!

Survey URL:
https://www.surveymonkey.com/r/MZPLRDY