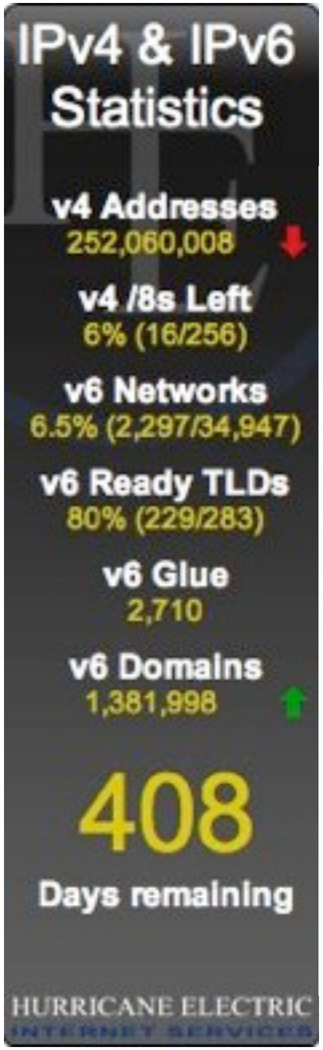
An Introduction to IPv6

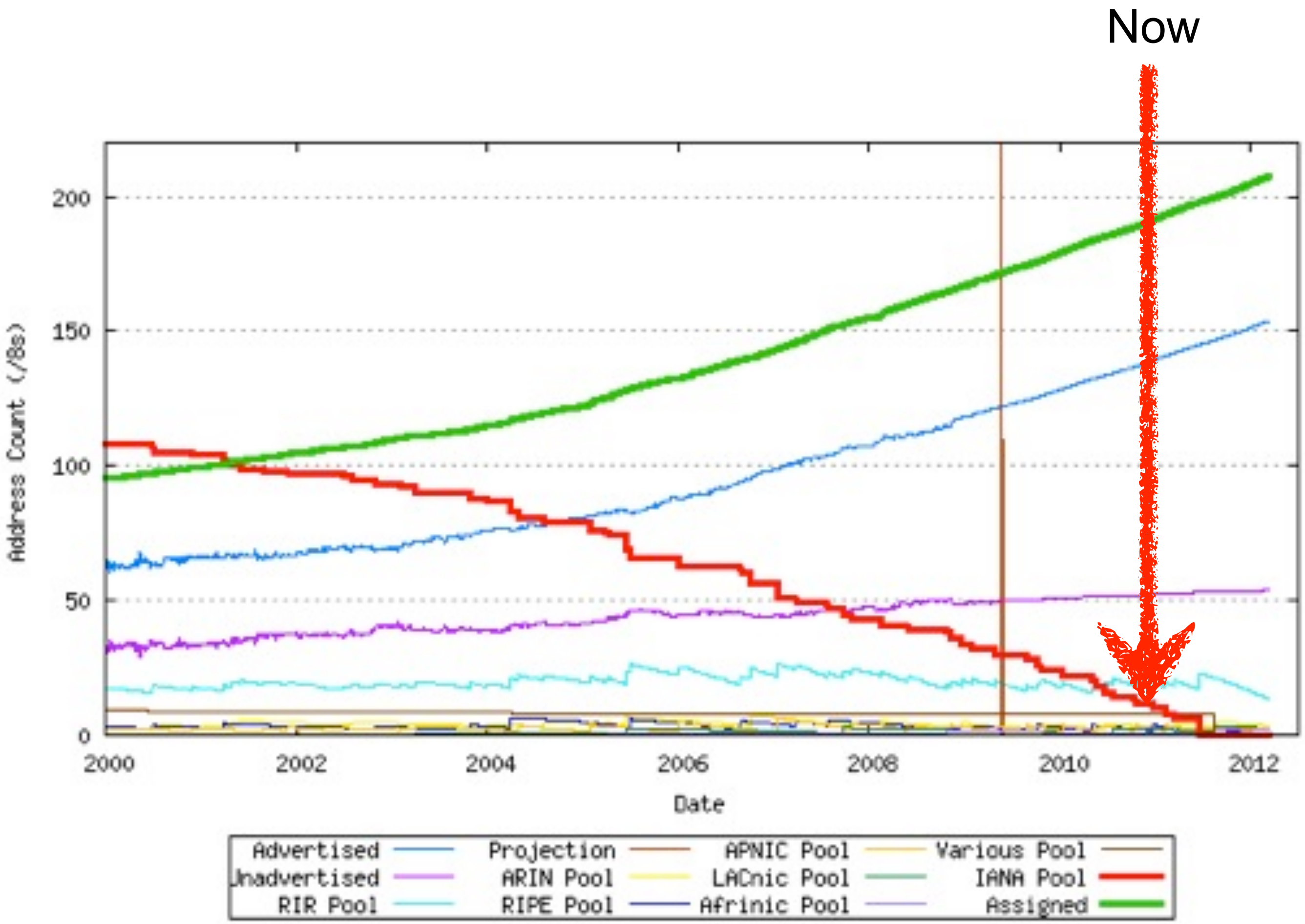
Owen DeLong
owend@he.net

Acknowledgements

- Special thanks for:
  - Content and graphics:
    - Mukom Akong Tamon (AfriNIC)
    - NIshal Goburdhan (AfriNIC)
  - Research, Data, and graphics
    - Geoff Huston (APNIC)
  - Inviting me to present this
    - Srinivas Chendi (APNIC)
  - Attending
    - All of you

Motivation:



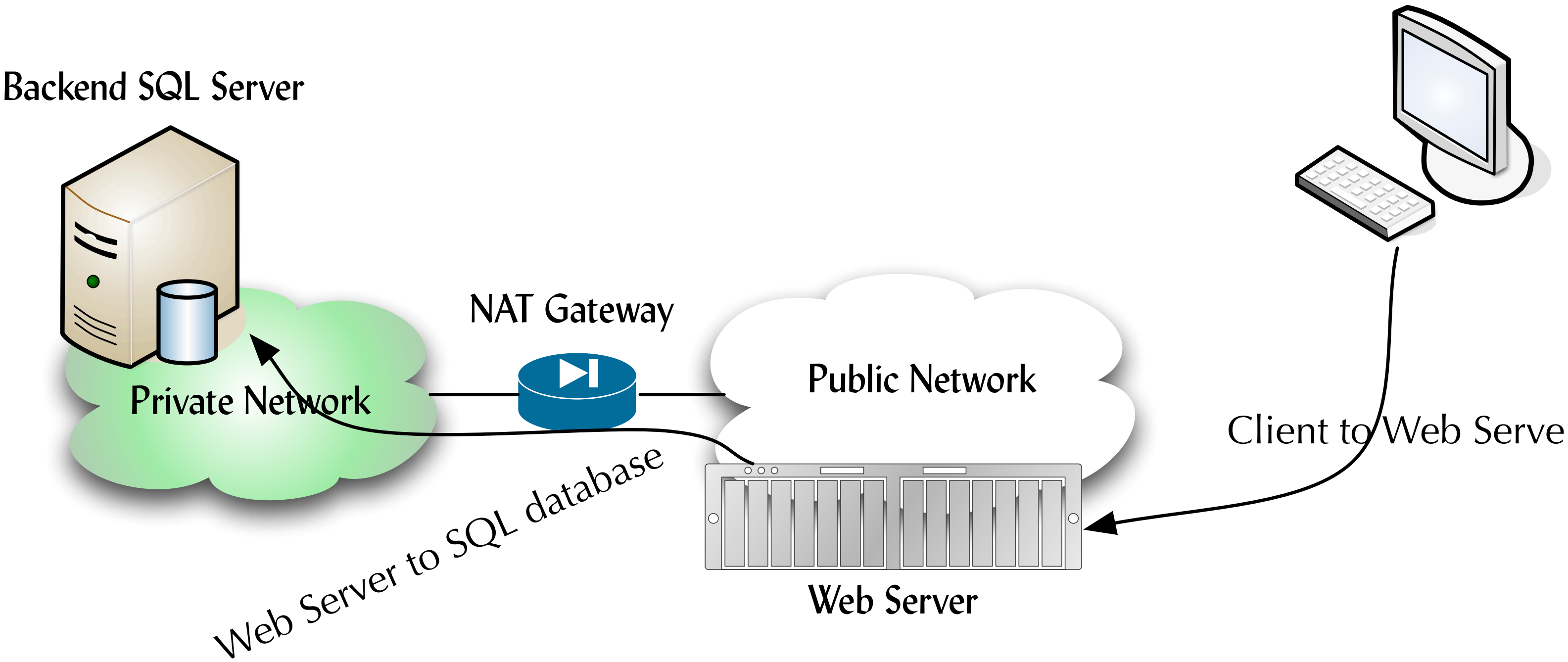


Monday, August 23, 2010

Logistics

- Location of rest rooms
- Please turn off your mobile phone(s)
- Please ask questions
  - Please don’t be afraid to speak up
  - No matter your English skills, I assure you they are better than my skills at your native language.

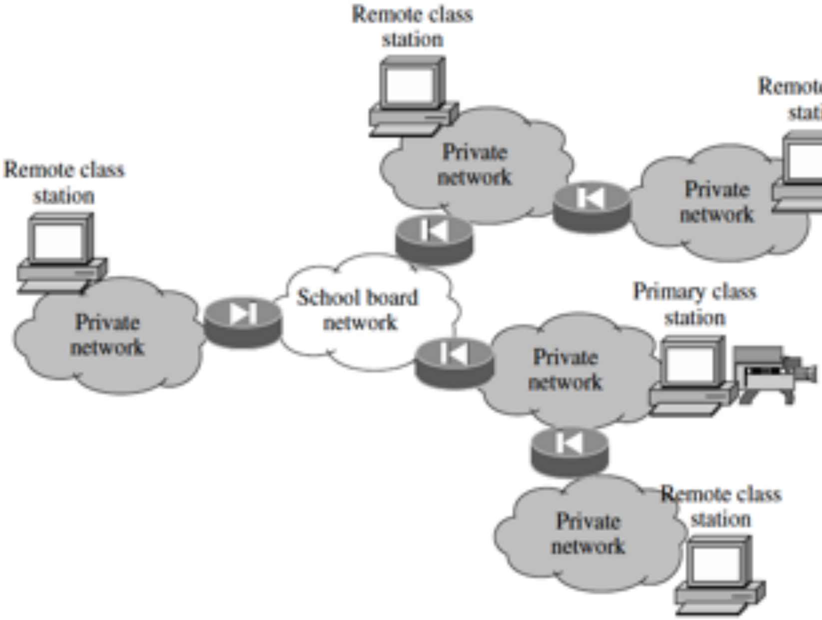
Why do we need IPv6? (1. NAT breaks IPSec)



- This works fine until you wan to secure the connection with IPSec
- The NAT device modifies the pakcet header, breaking IPSec

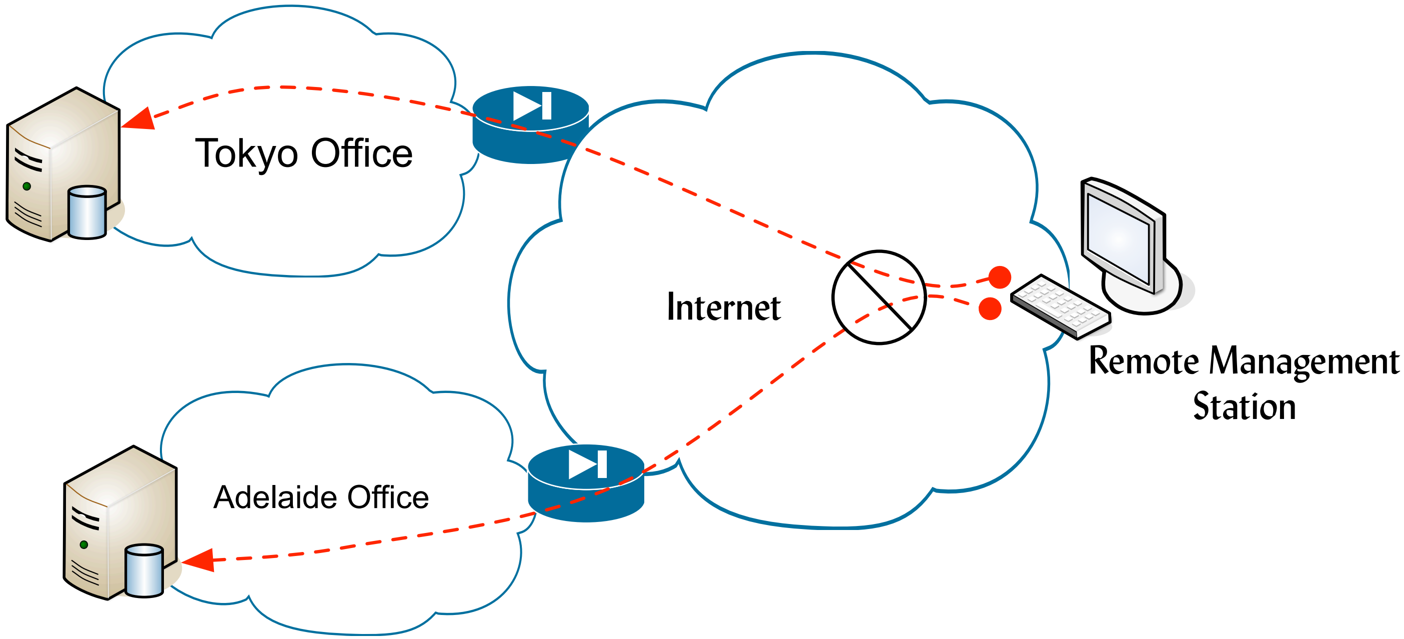
Monday, August 23, 2010

Why do we need IPv6?
(2. NAT Complicates Communications)



- Making this work (through NAT) is a nightmare.
- It's can be even worse for VOIP

Why do we need IPv6
(3. NAT breaks remote management)



- Branch offices behind NAT are not easily managed from remote locations

Why do we need IPv6 (4. Other IPv4 Limitations)

- Only 3.2 billion global unicast addresses
- Figure at least 5 IP addresses per person
- World Population 6.5 Billion
- Also need addresses for servers, provider infrastructure, etc.
- IPSec implemented as an afterthought mostly at L4

Why do we need IPv6? (5. Other Implications of NAT)

- Single points of failure -- NAT Gateway keeps state
- Additional hardware resources to maintain state tables/translation tables
- Breaks end-to-end model [p2p, voip, etc.] and hinders incoming connections to inside hosts
- Significant overhead/application bloat dedicated to working around NAT
- Causes problems for audit and abuse identification
- Complicates network troubleshooting and event correlation

Why IPv6?

- 2^96 times as much address space
- No NAT
- IPSec built into the layer 3 protocool
- Path MTU discovery built in and required to work (don’t block ICMP6)
- Multiple addresses per interface, easier and less disruptive renumbering
- Very large subnets -- no more need to count hosts

Why a new protocol?

- There simply aren't enough addresses in IPv4
- Restore the end-to-end model of communication enabling significant innovation
  - Smart grids
  - IP Telephony & IPTV
  - Smart devices (TVs, refrigerators, etc.)
  - "Internet of Things"
  - Internet access on planes, trains, and automobiles
- Growing demand for addresses to reach more people
- While we're at it, fix some (not all) of the issues with IPv4

Why a new protocol? (The main reason)

IPv4 Address Space Consumption January 2010

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

Allocated to RIR | IANA Free Pool | Other Uses

Why a new protocol?
(The main reason)

IPv4 Address Space Consumption — Current

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

Allocated to RIR | IANA Free Pool | Other Uses

Allocated to RIR in 2010

Applications that require IPv6 (not possible in IPv4)

- Green Tech - Give all energy consuming devices an address and manage/monitor remotely
- Global IP Telephony - No intermediate servers
- Give all users the ability to host services
  - Video streams
  - Personal web site or blog
  - Host games (Console or PC games)
- Natural Disaster warning systems
- Improved access to EMS, Medical telepresence

Part 1 – Nuts and Bolts

- First, we'll teach you about all the little pieces that make IPv6 possible
- Addresses
- Nomenclature
- Address Structure
- Address Scopes and Purposes
- Etc.

IPv6 -- The basics
1. Address Notation

- 8 groups of 4 hex digits (16 bits) separated by colons (:)
- Rules for shortening:
  - Drop any leading zeroes in a group
  - Replace ONE set of consecutive 0 groups with ::
- Example: 2001:0001:0000:0000:00A1:0CC0:01AB:397A

2001:0001:0000:0000:00A1:0CC0:01AB:397A

2001:1:0:0:A1:CC0:1AB:397A

2001:1::A1:CC0:1AB:397A

IPv6 -- The Basics
2. Types of IPv6 Addresses

Type	Sub-Types	Notes
Unicast	Global Unicast	Currently 2000::/3
	Protocol Use	Uses such as 6to4 , Teredo
	Unspecified	Default route or unknown address (::/0)
	Loopback	::1/128
	Link Local	fe80::/16 for use only on local link
Multicast	Used to implement all ‘broadcast-like’ behavior	
Anycast	Globally Unique address shared by multiple hosts	

IPv6 -- The Basics
3. Address Scopes

Type	Notes
Global	Routed throughout the entire IPv6 Internet
Link Local	•Can be used only by nodes on same link •Never forwarded by routers •Prefix fe80::/10 •IPv6 hosts automatically generate one per interface •Can also be manually configured •Requires ZoneID (%interface) for disambiguation
Unique Local	•Can be used with any site or group of sites on agreement •ULA Random (fd00::/8) High probability of uniqueness •fc00::/8 Reserved for future use (coordinated ULA?)

IPv6 -- The basics

Anatomy of a Global Unicast address

3 bits	9 bits	20 bits	16 bits	16 bits	64 bits
001	IANA to RIR	RIR to ISP	ISP to End Site	Net	Interface ID
001	IANA to RIR	RIR to End Site		Net	Interface ID
3 bits	9 bits	36 bits		16 bits	64 bits

- Every end site gets a /48
- Global Unicast currently being allocated from 2000::/3
  - Top: Provider assigned
  - Bottom: Provider Independent

IPv6 -- The basics

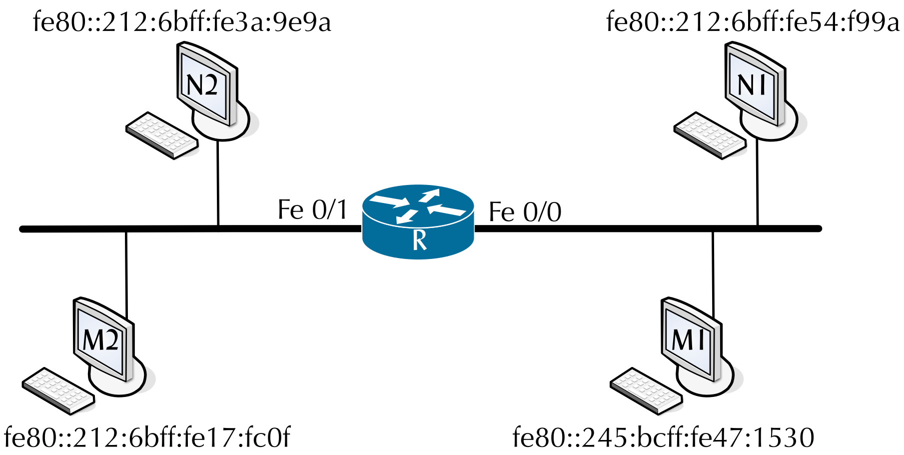
Anatomy of a Link Local address

16 bits	48 bits	64 bits
fe80	0	Interface ID

- Always fe80::/64, re-used on every link
- Never forwarded off-link (link scope)
- Must be present for interface to participate in IPv6, Automatically configured
- ZoneID used to disambiguate different links

IPv6 -- The Basics

Resolving Link Local Scope Ambiguity with Zone ID



- Recall: every link has link local address in fe80::/64
- So out which interface should the router send out a packet to fe80/64?
- ZoneIDs (or scopeIDs) disambiguate by specifying the interface to which the address belongs.
- Specified as [RFC4007]: address%zoneID
- ZoneID could be the interface number or other identifier.
- e.g ping fe80::245:bcff:fe47:1530%fxp0 [on a FreeBSD system]

IPv6 -- The basics

Anatomy of Unique Local Addresses [RFC4193]

8 bits	40 bits	16 bits	64 bits
fdXX	Global ID	SubnetID	Interface ID

- Like RFC1918 but lower chance of collision
- Prefix fc00::/7 + "L flag" (bit 8) indicates whether prefix is locally assigned [1] or globally assigned [0]. (Global currently deprecated)
- Scope is global, but, filtered by most routers and ignored by most eBGP peers
- fd00::/8 Global IDs generated at random (RFC4198)
- Uniqueness is NOT guaranteed but highly likely

IPv6 -- The basics

Anatomy of a 6to4 Transition Address

16 bits	32 bits	16 bits	64 bits
2002	WWXX:YYZZ IPv4 Address	SubnetID	Interface ID

- WWXX:YYZZ is the hex form of a public IPv4 address
- Used to create global prefixes and hosts for v6 capable sites on the IPv4 internet
- Better in theory than practice
- Example address: 192.159.10.200/32 (IPv4) becomes
  2002:c09f:0ac8::/48 (65,536 IPv6 subnets for every IPv4 host)

IPv6 -- The basics

Anatomy of an IPv4-Mapped Transition Address

80 bits	16 bits	32 bits
0	ffff	IPv4 Address

- Usually represented as ::ffff:w.x.y.z where w.x.y.z is a normal address
- Internally represents an IPv4 node to an IPv6 node
- Never used on the wire
- Simplifies software development for dual-stack (treat all connections as IPv6)

IPv6 -- The basics
ISATAP Transition Addresses

64 bits	32 bits	32 bits
Prefix	0000:5efe	Private IPv4 Address

- 64 bit prefix (fe80:: for link local, others depend on ISATAP server being used)
- Patented by SRI, License free
- Most widespread implementation of ISATAP is Teredo
- Brittle and hard to troubleshoot
- A poor substitute for native IPv6 connectivity or better tunnel choices

IPv6 -- The basics

Anatomy of a Multicast Address

8 bits	4 bits	4 bits	112 bits
ff	Flags	Scope	GroupID

- Flags determine whether the address is transient, based on a unicast prefix, or embeds an RP address
- ff00::/8
- Permanent groupIDs are independent of scope while transient groupIDs are only relevant within their scope.

Flag and Scope Bits in Multicast Addresses

The Flag Field	
Bit Position	Description
Bit 3	Undefined
Bit 2 (R flag)	Rendezvous Point address is embedded (1) or not (0)
Bit 1 (P flag)	Address is based on a unicast prefix (1) or not (0)
Bit 0 (T flag)	Address is IANA defined (0) or transient (1)

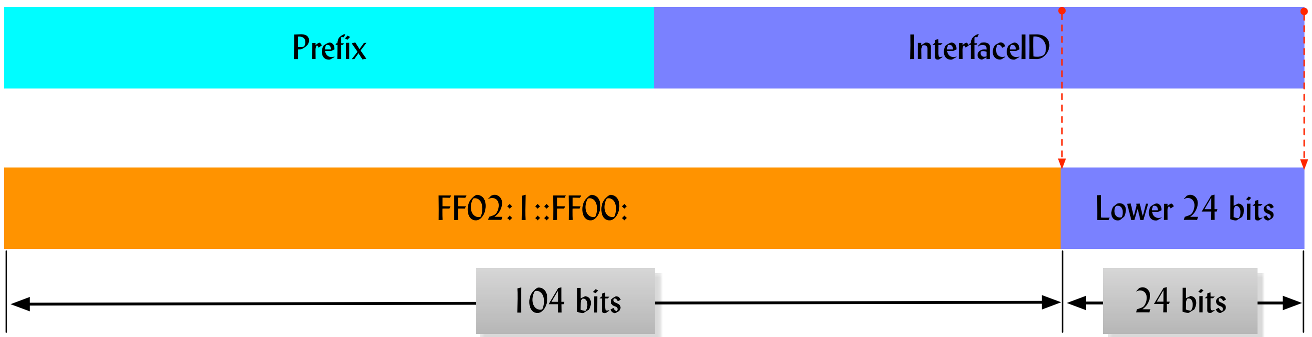
The Scope Field		
Binary	Hex	Scope
0001	1	Interface
0010	2	Link
0100	4	Admin
0101	5	Site
1000	8	Organisation
1110	E	Global
Others		Unassigned or Reserved

Some Reserved/Well Known Multicast Groups

Some Well-Known/Reserved Multicast Groups		
Address	Scope	Description
FF01::1	1=Interface	All nodes on the interface
FF02::1	2=Link	All nodes on the link
FF01::2	1=Interface	All routers on the interface
FF02::2	2=Link	All routers on the link
FF05::2	5=site	All routers in the site
FF02::5	2=Link	All OSPFv3 routers
FF02::6	2=Link	OSPFv3 designated routers
FF02::A	2=Link	All EIGRPv6 routers
FF02::D	2=Link	All PIM routers
FF02::1:FFXX:XXXX	2=Link	Solicited-node address

Solicited-Node Multicast Address

- Used as destination for Neighbor Solicitation
- Constructed from prefix ff02::1:ff00:0/104 and the last 24 bits of the IPv6 address
- Every node will listen to and respond to its solicited node address
- ALlows link-layer address resolution to not disturb (most) unconcerned nodes.



IPv6 – The Basics
Address Type Rehash

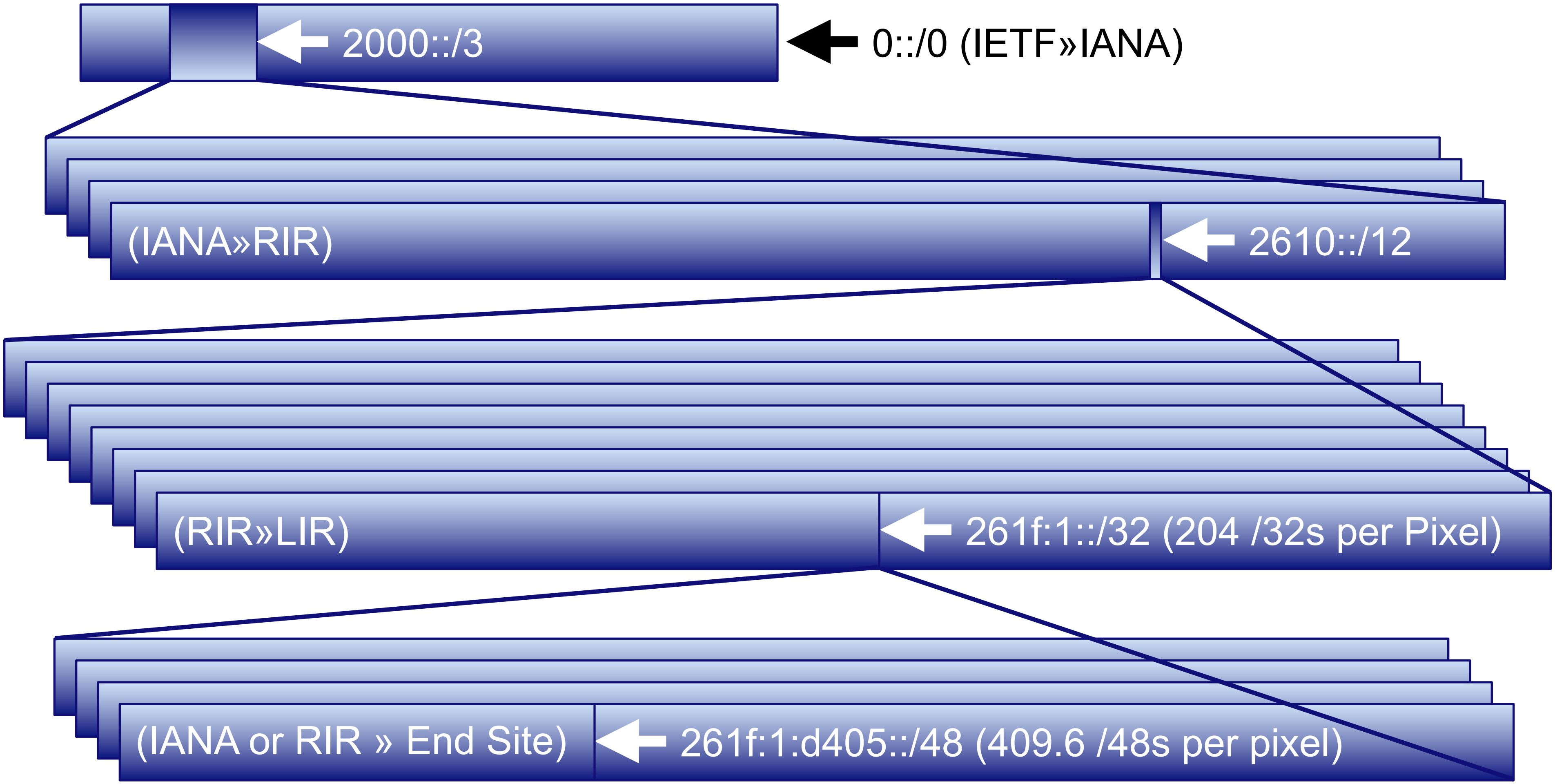
Summary of IPv6 Address Types

TYPE	STRUCTURE							
	16 BITS	16 BITS	16 BITS	16 BITS	16 BITS	16 BITS	16 BITS	16 BITS
Global Unicast	Global ID			Subnet ID	Interface ID			
ULA-Reserved	fcxx	0		Subnet ID	Interface ID			
ULA-Local	fdxx	0		Subnet ID	Interface ID			
Link-Local	fe80	0			Interface ID			
IPv4 Mapped	0					ffff	<IPv4 Address>	
6to4	2002	<IPv4 Address>		Subnet ID	Interface ID			
ISATAP	<64bit v6 Prefix>				0	5efe	<IPv4 Address>	
Unspecified	0							
Loopback	0							1
Multicast	ff<LS>	Multicast GroupID						

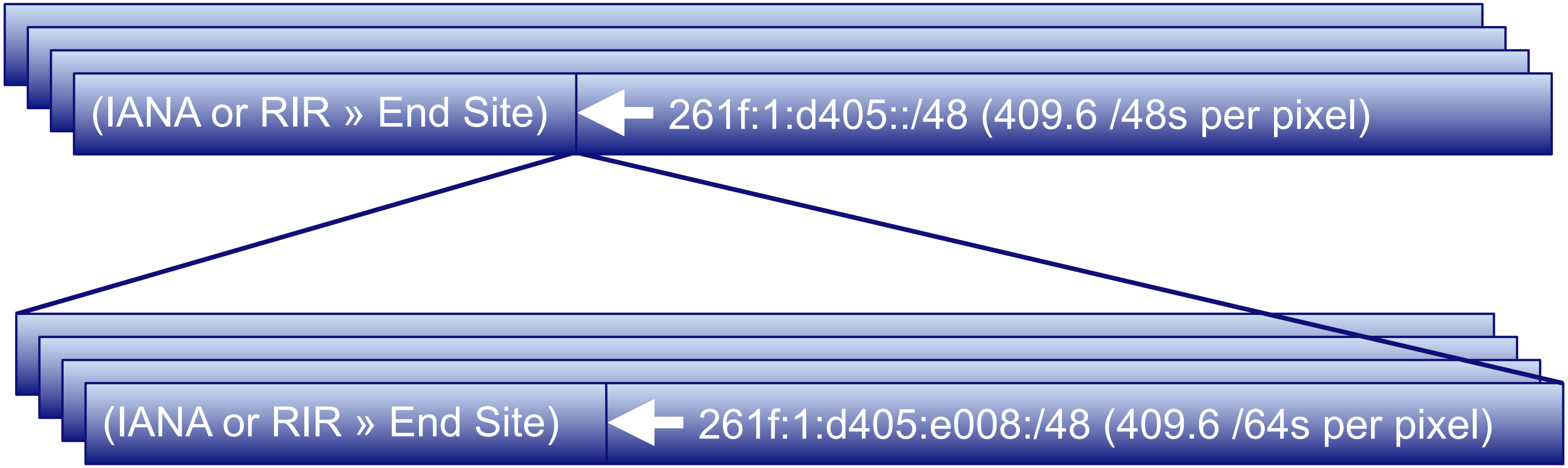
IPv6 – The Basics
Address Type Rehash

Summary of IPv6 Address Types		
Type	Range	Application
Global Unicast	2000::/3	Normal host to host communications
Link-local	fe80::/10	Connected-link communications
Anycast	2000::/3	One to Any (Any=topologically closest)
Unique-local	fc00::/7	Normal host to host communications within a site
IPv4-mapped		Represent an IPv4 address in IPv6 format
6to4	2002::/3	Automatic transition mechanism
ISATAP		Intra-site transition mechanism
Unspecified	::/0	When node doesn't have an address
Loopback	::1/128	Used for intra-node communications
Multicast	ff00::/8	One to Many

IPv6 -- The basics
How Global Unicast is Allocated



IPv6 -- The basics
How Global Unicast is Allocated



- The Numbers:
  - 8 /3s, one of which is in use
  - 512 /12 allocations to RIRs in first /3 (6 used so far)
  - 1,048,576 LIR /32s in each RIR /12
  - 65,536 /48 Assignments in each /32

IPv6 -- The Basics
Global Unicast in perspective

- The Numbers (cont.)
  - The first /12 assigned to each RIR can support 68,719,476,736 /48 End Sites
  - There are 506 /12s remaining if that's not enough for any particular region.
  - Many ISPs will require more than a /32, but, even if we figure a /28 for every ISP on average, that's still enough addresses for 65,536 ISPs in each RIR region without exhausting their first /12. (There are currently fewer than 30,000 BGP speaking ISPs worldwide)
  - In short... There is more than enough address space for liberal assignments under current and any likely policy.

IPv6 -- Not your father's IP

IPv6 -- Address Planning Don't oversimplify too much!

- There are lots of people saying "ISPs get /32s, end sites get /48s."
- That's an unfortunate oversimplification.
- ISPs get AT LEAST a /32 and can get whatever larger allocation they can justify.
- End sites should get at least a /48 and should be given whatever larger assignment they can justify.

IPv6 -- Address Planning Methodology

- Don't start with a /32 and figure out how to make your needs fit within it.
- Start by analyzing your needs and apply for a prefix that will meet those needs.
- In your analysis, it's worth while to try and align allocation units to nibble boundaries. A nibble boundary is a single hex digit, or, a number 2^n such that n is a multiple of 4. (e.g. 16, 256, 1024, 4096, 16384, 65536...)

IPv6 Address Planning Analysis

- Start with the number of end sites served by your largest POP. Figure a /48 for each. Round up to the a nibble boundary. (if it’s 3,000 end sites, round up to 4096, for example... a /36 per POP.
- Next, calculate the number of POPs you will have. Include existing POPs and likely expansion for several years. Round that up to a nibble boundary, too. (140 POPs, round up to 256).

IPv6 Address Planning Analysis

- Now that you have an address size for each POP (4096 = 12 bits in our example) and a number of POPs (256 = 8 bits in our example), you know that you need a total of POP*nPOPs /48s for your network (4096*256=1,048,576 or 12+8=20 bits).
- 48 bits - 20 bits is 28 bits, so, you actually need a /28 to properly number your network.
- You probably could squeeze this into a /32, but, why complicate your life unnecessarily?

IPv6 Address Planning
Apply for your addresses

- Now that you know what size block you need, the next step is to contact your friendly neighborhood RIR (Regional Internet Registry) and apply.
- Most RIRs provide either an email-based template or a web-based template for you to fill out to get addresses.
- If you are a single-homed end-user, you usually should get your addresses from your upstream rather than an RIR.

IPv6 Address Planning
The bad news

- The addressing methodology I described above may not be consistent with RIR policy in all regions (yet).
- This means you might have to negotiate to a smaller block.
- All RIRs have an open policy process, so, you can submit a proposal to enable this kind of allocation, but, that may not help you immediately.

IPv6 Address Planning The good news

- Having things on nibble boundaries is convenient, but, not necessary.
  - ip6.arpa DNS delegations
  - Human Factors
  - Routing Table management
  - Prefix lists
- The techniques that follow should work either way.

IP Address Planning
Carving it up

- For the most part, you've already done this.
- Take the number you came up with for the nPOPs round-up and convert that to a number of bits (256 = 8 bits in our example).
- Now, take what the RIR gave you (/28 in our example) and add that number to the above number (28+8 = 36) and that's what you need for each POP (a /36 in our example).

IPv6 Address Planning Carving it up

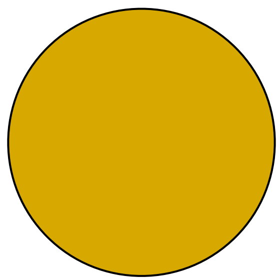
- Now let's give address segments to our POPs.
- First, let's reserve the first /48 for our infrastructure. Let's use 2000:db80 - 2000:db8f as our example /28.
- Since each POP gets a /36, that means we have 2 hex digits that designate a particular POP.
- Unfortunately, in our example, that will be the last digit of the second group and the first digit of the third group.

IPv6 Addressing
Carving it up

- Strategy
  - Sequential Allocation
    - Advantage: Simple, easy to follow
    - Advantage: POP Numbers correspond to addresses
    - DisAdvantage: Complicates unexpected growth
  - Allocation by Bisection
    - Advantage: Simplifies growth
    - Advantage: Greatest probability of Aggregation
    - Disadvantage: “Math is hard. Let’s go shopping!”

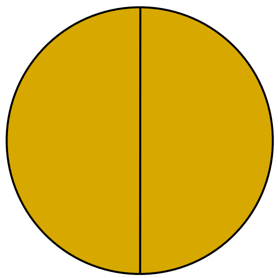
IPv6 Addressing Allocation by Bisection

- Bisection? What does THAT mean?
- Simple... It means to cut up the pieces by taking the largest remaining piece and cutting in half until you have the number of pieces you need.
- Imagine cutting up a pie into 8 pieces...



IPv6 Addressing Allocation by Bisection

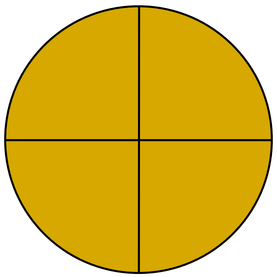
- Bisection? What does THAT mean?
- Simple... It means to cut up the pieces by taking the largest remaining piece and cutting in half until you have the number of pieces you need.
- Imagine cutting up a pie into 8 pieces...



First, we cut it in half...

IPv6 Addressing Allocation by Bisection

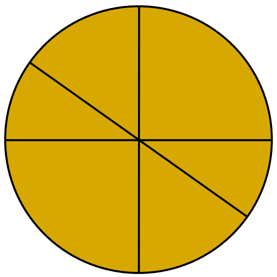
- Bisection? What does THAT mean?
- Simple... It means to cut up the pieces by taking the largest remaining piece and cutting in half until you have the number of pieces you need.
- Imagine cutting up a pie into 8 pieces...



Then we cut it in half again

IPv6 Addressing Allocation by Bisection

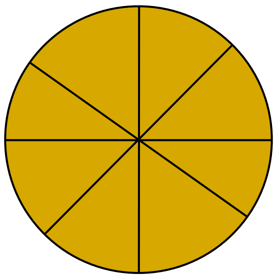
- Bisection? What does THAT mean?
- Simple... It means to cut up the pieces by taking the largest remaining piece and cutting in half until you have the number of pieces you need.
- Imagine cutting up a pie into 8 pieces...



Then Again

IPv6 Addressing Allocation by Bisection

- Bisection? What does THAT mean?
- Simple... It means to cut up the pieces by taking the largest remaining piece and cutting in half until you have the number of pieces you need.
- Imagine cutting up a pie into 8 pieces...



And finally a fourth cut

IPv6 Addressing Allocation by Bisection

- It's a similar process for IPv6 addresses.
  - Let's start with our 2001:db80::/28 prefix.
  - We've already allocated 2001:db80:0000::/48
  - Our available space is now 2001:db80:0001:: to 2001:db8f:ffff:ffff:ffff:ffff:ffff:ffff. Cutting that in half we get 2001:db88:0000::/36 as our first POP address.
  - That leaves the largest chunk at 2001:db88:1000:: to 2001:db8f:ffff:ffff:ffff:ffff:ffff:ffff. Cutting that in half, we get 2001:db8c:0000::/36 as our next POP

IPv6 Address Planning Allocation by Bisection

- After repeating this for 19 POP allocations, we have a table that looks like this:

Infrsastructure	2001:db80:0000:/48	POP1	2001:db88:0000::/36
POP12	2001:db80:8000::/36	POP13	2001:db88:8000::/36
POP8	2001:db81:0000::/36	POP9	2001:db89:0000::/36
POP4	2001:db82:0000::/36	POP5	2001:db8a:0000::/36
POP14	2001:db83:0000::/36	POP15	2001:db8b:0000::/36
POP2	2001:db84:0000::/36	POP3	2001:db8c:0000::/36
POP16	2001:db84:8000::/36	POP17	2001:db8c:8000::/36
POP10	2001:db85:0000::/36	POP11	2001:db8d:0000::/36
POP6	2001:db86:0000::/36	POP7	2001:db8e:0000::/36
POP18	2001:db87:0000::/36	POP18	2001:db8f:0000::/36

IPv6 Address Planning Allocation by Bisection

- After repeating this for 19 POP allocations, we have a table that looks like this:

Infrsastructure	2001:db80:0000:/48	POP1	2001:db88:0000::/36
POP12	2001:db80:8000::/36	POP13	2001:db88:8000::/36
POP8	2001:db81:0000::/36	POP9	2001:db89:0000::/36
POP4	2001:db82:0000::/36	POP5	2001:db8a:0000::/36
POP14	2001:db83:0000::/36	POP15	2001:db8b:0000::/36
POP2	2001:db84:0000::/36	POP3	2001:db8c:0000::/36
POP16	2001:db84:8000::/36	POP17	2001:db8c:8000::/36
POP10	2001:db85:0000::/36	POP11	2001:db8d:0000::/36
POP6	2001:db86:0000::/36	POP7	2001:db8e:0000::/36
POP18	2001:db87:0000::/36	POP18	2001:db8f:0000::/36

IPv6 Address Planning Allocation by Bisection

- After repeating this for 19 POP allocations, we have a table that looks like this:

Infrsastructure	2001:db80:0000:/48	POP1	2001:db88:0000::/36
POP12	2001:db80:8000::/36	POP13	2001:db88:8000::/36
POP8	2001:db81:0000::/36	POP9	2001:db89:0000::/36
POP4	2001:db82:0000::/36	POP5	2001:db8a:0000::/36
POP14	2001:db83:0000::/36	POP15	2001:db8b:0000::/36
POP2	2001:db84:0000::/36	POP3	2001:db8c:0000::/36
POP16	2001:db84:8000::/36	POP17	2001:db8c:8000::/36
POP10	2001:db85:0000::/36	POP11	2001:db8d:0000::/36
POP6	2001:db86:0000::/36	POP7	2001:db8e:0000::/36
POP18	2001:db87:0000::/36	POP18	2001:db8f:0000::/36

IPv6 Address Planning Allocation by Bisection

- Notice how by doing that, most of the /36s we created have 15 more /36s before they run into allocated space and all have at least 7.
- Notice also that if any POPs get larger than we expect, we can expand them to /35s,

  /34s, /33s, and most all the way to a /32 without having to renumber.
- By default, at /36, each pop has room for 4096 /48 customers. End sites that need more than a /48 should be extremely rare*.

IPv6 Address Planning Allocation by Bisection

- *End Site means a single customer location, not a single customer. Many customers may need more than a /48, but, with 65,536 /64 subnets available, even the largest building should be addressable within a /48.

TAKE A BREAK

- It's come time to accommodate that basic reality... Humans don't do well sitting in one place for extended periods of time listening to the same person blather on and on no matter how interesting the topic.
- EVERYONE stand up.
- EVERYONE leave the room for at least the next 10 minutes.
- EVERYONE try to be back within 20 minutes, please. There's still lots to cover.

Quick Interactive Review

Quick Interactive Review

- OK... Let’s see what you remember from the first half:

Quick Interactive Review

- OK... Let’s see what you remember from the first half:
- Plan a small ISP:
  - 300 End sites per POP
  - 12 POPs

Quick Interactive Review

- OK... Let's see what you remember from the first half:
- Plan a small ISP:
  - 300 End sites per POP
  - 12 POPs
- Plan a medium ISP:
  - 500 End sites per POP
  - 150 POPs

Quick Interactive Review

- OK... Let’s see what you remember from the first half:
- Plan a small ISP:
  - 300 End sites per POP
  - 12 POPs
- Plan a medium ISP:
  - 500 End sites per POP
  - 150 POPs
- Plan a very large ISP:
  - 1000 End sites per POP
  - 800 POPs

Part 2 – Making it work

- Now that we've covered the nuts and bolts, let's talk about how they go together into subassemblies and assemblies to make a working network.

IPv6 Deployment
Native connectivity

- Native connectivity is, by far, the simplest and cleanest way to deploy IPv6.
- Just like IPv4 (mostly) but with bigger addresses that look different.
- Example:

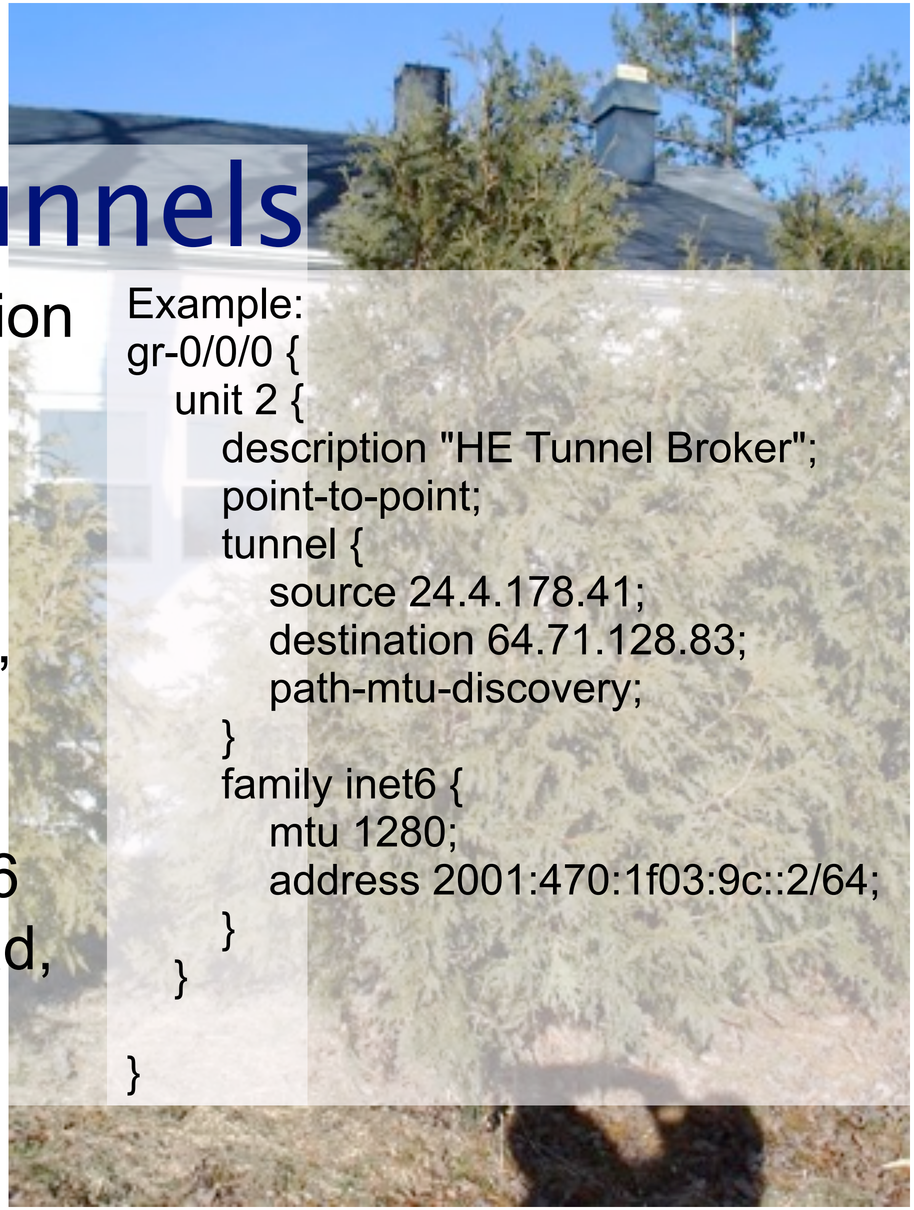
```
DEVICE=eth0
ONBOOT=yes
MTU=1280
IPADDR=192.159.10.2
NETMASK=255.255.255.0
GATEWAY=192.159.10.254

IPV6INIT=yes
IPV6ADDR=2620:0:930::0200:1/64
IPV6_DEFAULTGW=2620:0:930::1
IPV6_AUTOCONF=no
```

IPv6 Deployment Using Tunnels

- Slightly more complicated than native connectivity
- MTU problems
- Lower MTU reduces performance on some platforms. (Significantly on some versions of Windows)
- Harder to Configure or harder to Troubleshoot (depending on tunnel solution chosen)

IPv6 Tunneling
6in4 or GRE Tunnels

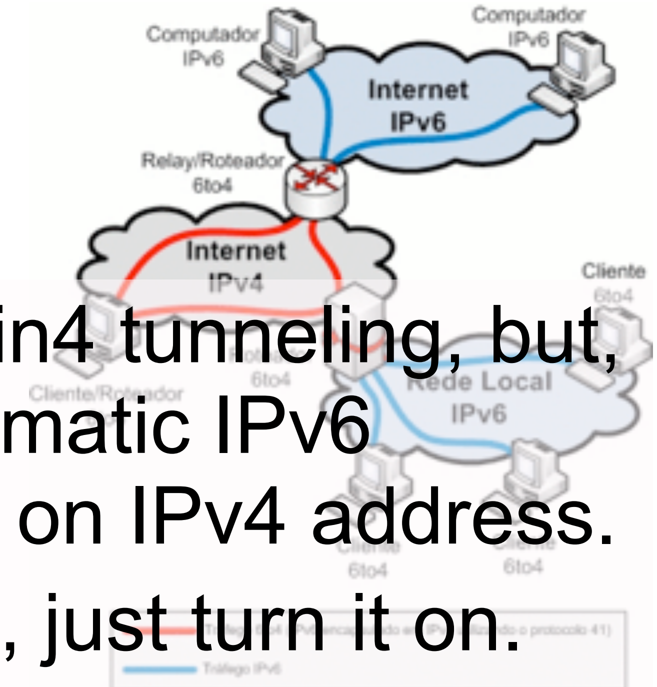
- Require manual configuration
- Direct point to point tunnel between two known (and configured) IPv4 nodes
- 6in4 is very similar to GRE, but, only IPv6 inside IPv4 payload, protocol 41.
- GRE can support both IPv6 and IPv4 in an IPv4 payload, protocol 47.



Example:

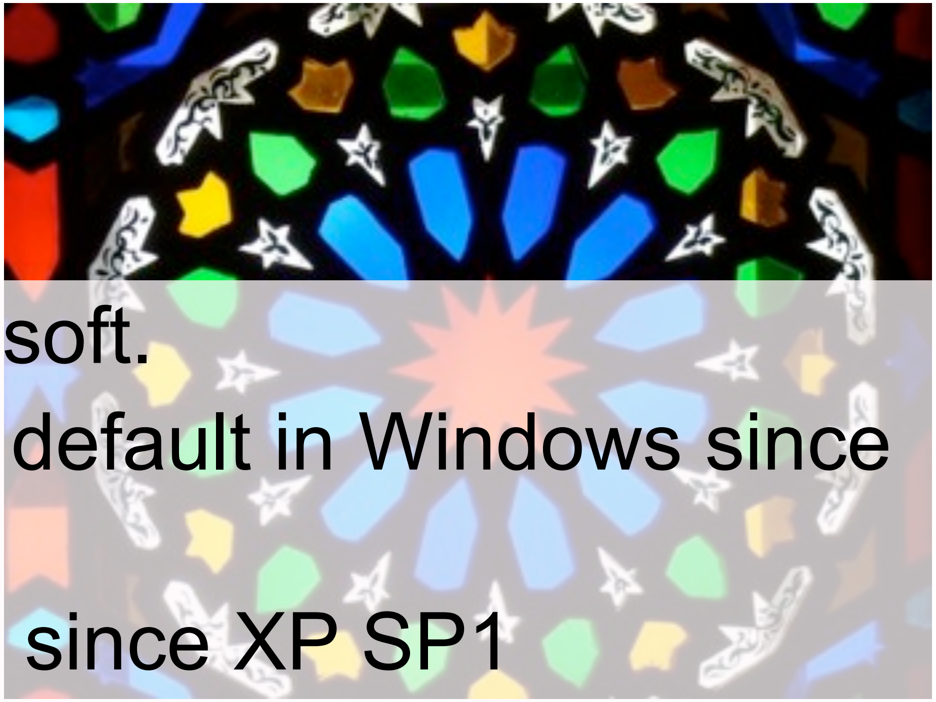
```
gr-0/0/0 {
    unit 2 {
        description "HE Tunnel Broker";
        point-to-point;
        tunnel {
            source 24.4.178.41;
            destination 64.71.128.83;
            path-mtu-discovery;
        }
        family inet6 {
            mtu 1280;
            address 2001:470:1f03:9c::2/64;
        }
    }

}
```

IPv6 Tunneling
6to4 Autotunneling



- Also uses protocol 41 like 6in4 tunneling, but, uses IPv4 anycast and automatic IPv6 address construction based on IPv4 address.
- Nothing to configure, per se, just turn it on.
- Non-deterministic.
- Changes can occur anywhere in the network.
- No control over which 6to4 gateway you use.
- Hard to troubleshoot
- Easy to deploy (when it works)

IPv6 tunneling Teredo



- Developed by Microsoft.
- Automatically on by default in Windows since Vista
- Present in Windows since XP SP1
- Works through NAT
- Even Microsoft calls it a “Last Resort” for IPv6 connectivity. Unless you absolutely need it, disable it and block it at your firewall.

IPv6 Routing
RIPng

- Just like RIPv2 for IPv4
- Don’t do this
- Same reasons as RIPv2.
- Don’t do this
- It’s just a bad idea.
- Really, Don’t use RIP.

IPv6 Routing
OSPF

- OSPF3 can handle both IPv6 and IPv4 (most vendors)
- OSPF2 (common version) is only IPv4.
- OSPF3 available on most routers, but, requires configuration as OSPF3, not just "OSPF".
- Can run OSPF2 for IPv4 and OSPF3 simultaneously on most routers
- Largely like IPv4 OSPF2 configuration

IPv6 Routing
OSPFv3 differences from OSPFv2

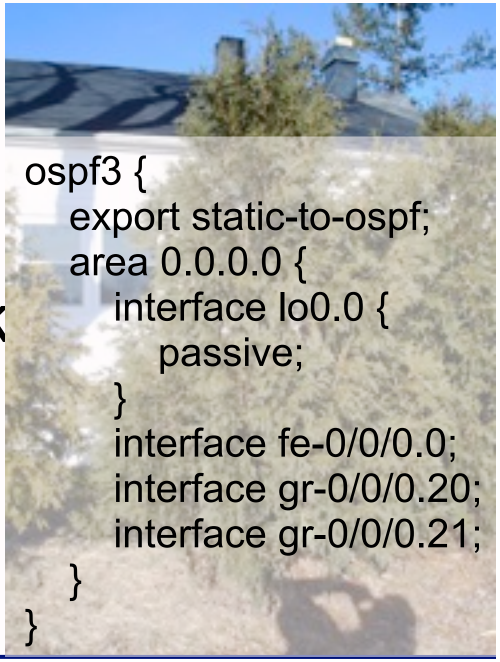
- Router and Network LSA don't carry IP addresses, new type 9 LSA instead.
- Relies on AH and ESP instead of having authentication in OSPF protocol
- Advertises all prefixes on the interface
- LSAs now have Flooding Scope
- Supports multiple instances per link - only routers in the same instance form adjacencies
- For all neighbor communications, all packets are sourced from link-local addresses with the exception of virtual links.

IPv6 Routing OSPFv3 LSA types

LSA Type	Common Name	Description	Flooding Scope
1	Router LSA	Describes a router’s link states and costs of its links to one area.	Area
2	Network LSA	Generated by a DR to describe the aggregated link state and costs for all routers attached to an area	Area
3	Inter-Area Prefix LSA for ABRs	Originated by ABRs to describe interarea networks to routers in other areas.	Area
4	Inter-Area Router LSA for ASBRs	Originated by ASBRs to advertise the ASBR location.	Area
5	Autonomous System External LSA	Originated by an ASBR to describe networks learned from other protocols (redistributed routes).	Autonomous System
8	Link LSA	Advertises link-local address and prefix(es) of a router to all other routers on the link as well as option information. Sent only if more than one router is present on a link.	Link
9	Inter-Area Prefix LSA for ABRs	Performs one of two functions Associates a list of IPv6 prefixes with a transit network by pointing to a Network LSA Associates a list of IPv6 prefixes with a router by pointing to a Router LSA	Area

IPv6 Routing OSPFv3 Configuration

- Still requires a 32 bit Router ID
  - Either configure a loopback interface with an IPv4 address or
  - Set the router-id in the appropriate block
  - Note: router-id does not require an IPv4 address and OSPF3 can work without IPv4.
- Example: Juniper SRX-100



```
ospf3 {
    export static-to-ospf;
    area 0.0.0.0 {
        interface lo0.0 {
            passive;
        }
        interface fe-0/0/0.0;
        interface gr-0/0/0.20;
        interface gr-0/0/0.21;
    }
}
```

IPv6 Routing
eBGP

- Nearly identical to BGP for IPv4
- Different vendors have different ways of differentiating routing tables
  - Cisco: address-family ipv6
    - Misconfigurations usually silently ignore your intent
  - Juniper: family inet6
    - Misconfigurations usually result in an error message

IPv6 Routing eBGP

- Configuration Example:
  - Juniper SRX-100

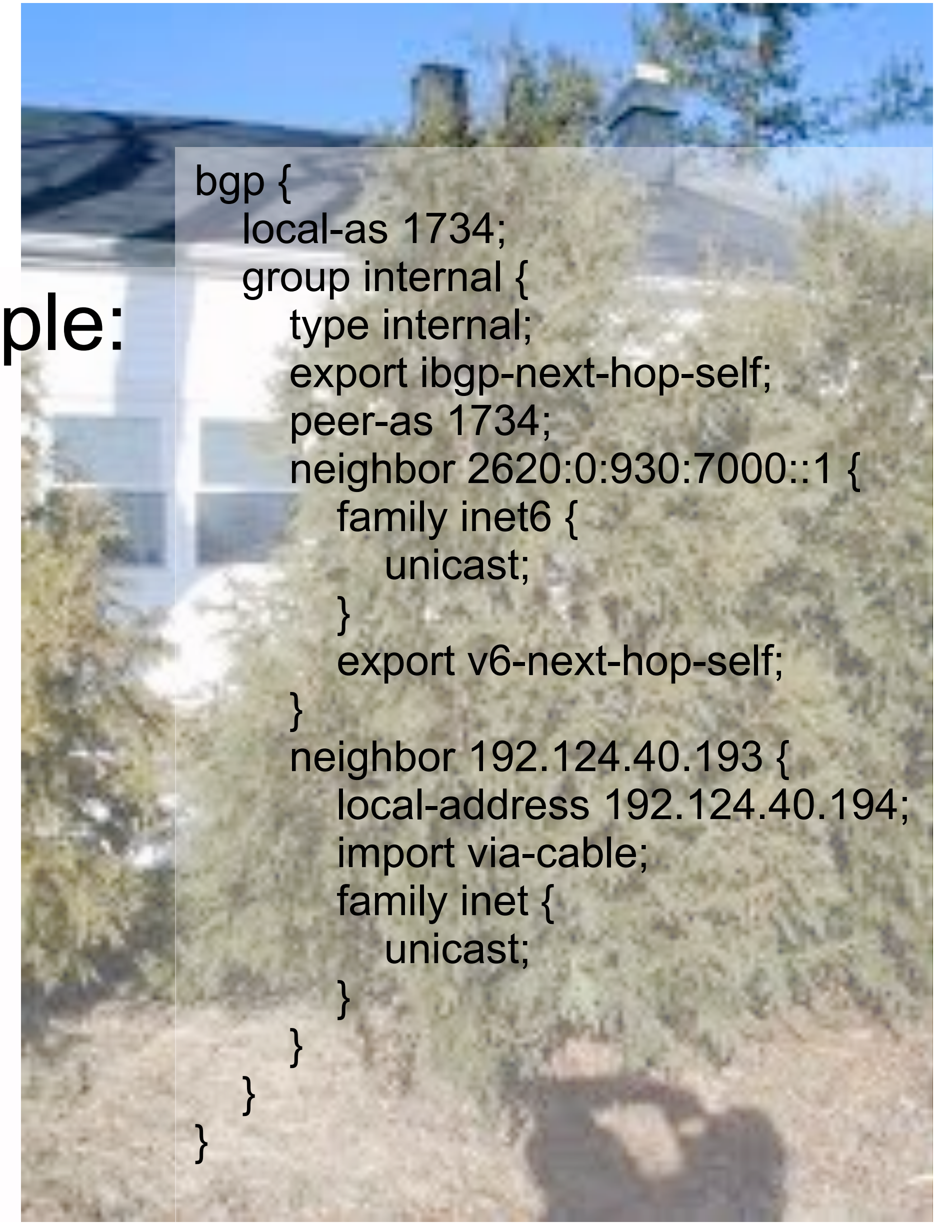
```
bgp {
    family inet {
        unicast;
    }
    family inet6 {
        unicast;
    }
    local-as 1734;
    group l42 {
        family inet {
            unicast;
        }
        family inet6 {
            unicast;
        }
        export to-l42;
        peer-as 8121;
        neighbor 192.124.40.129;
        neighbor 2620:0:930:7fff::1 {
            family inet6 {
                unicast;
            }
        }
    }
}
```

IPv6 Routing iBGP

- Same set of differences as IPv4 e vs. i BGP.
- All border routers must peer in a mesh
- IGP must provide reachable routes to all routers.
- Peering best between loopbacks.
- All the IPv4 lessons and best practices apply.

IPv6 Routing iBGP

- Configuration Example:
  - Juniper SRX-100



```
bgp {
    local-as 1734;
    group internal {
        type internal;
        export ibgp-next-hop-self;
        peer-as 1734;
        neighbor 2620:0:930:7000::1 {
            family inet6 {
                unicast;
            }
            export v6-next-hop-self;
        }
        neighbor 192.124.40.193 {
            local-address 192.124.40.194;
            import via-cable;
            family inet {
                unicast;
            }
        }
    }
}
```

IPv6 -- Much like IPv4 was supposed to be

- This time with enough Addresses
  - Basic weight comparison: If IP addresses were a unit of mass, then, if IPv4 weighed the same as 7 liters of water, IPv6 would weigh the same as the entire planet Earth.

IPv6 Operations
DHCPv6 Overview

- Can assign prefix lengths other than /64
- Bootstrapped by RS/RA process
- DHCP-PD -- Potentially very useful feature for ISPs
- Can provide DNS, NTP, and other server locations (RA/SLAAC cannot, but, ND has a process for doing DNS servers).
- Limited field deployment/support so far.
- IETF religious problem (SLAAC vs. DHCP war)

IPv6 Operations
SLAAC and the need for RA-Guard

- Stateless Autoconfiguration provides a very convenient way to number hosts.
- Unfortunately, all routers are created equal, including the ones created by someone else.
- In DHCP, a rogue server tends to break things.
- With RA, a rogue router can intercept traffic without breaking anything.
- RA Guard blocks RA transmissions on switchports not defined as routers.

IPv6 Transition Issues
Things that don't route packets

- Software Upgrades
  - In-house tools
    - Porting to IPv6 network
    - Adding IPv6 address capability to
      - databases
      - parsers
      - data structures
      - etc.
  - Vendor Provided Software
    - Start talking to them now if you aren't already
    - You're not the only one asking, no matter what they say.
    - Call their bluff.

IPv6 Transition Issues
Things that forward packets

- Hardware Upgrades
  - Firewalls
  - Really old routers
  - Intrusion Detection systems
  - Load Balancers
  - Printers -- Probably just leave on internal IPv4.
  - Clock sources
  - Other infrastructure
  - Other odd appliances

Parting Thoughts: What if IP Addresses were M&Ms -- IPv4

- IPv4: Standard network size: /24
  - Number of usable addresses: 254

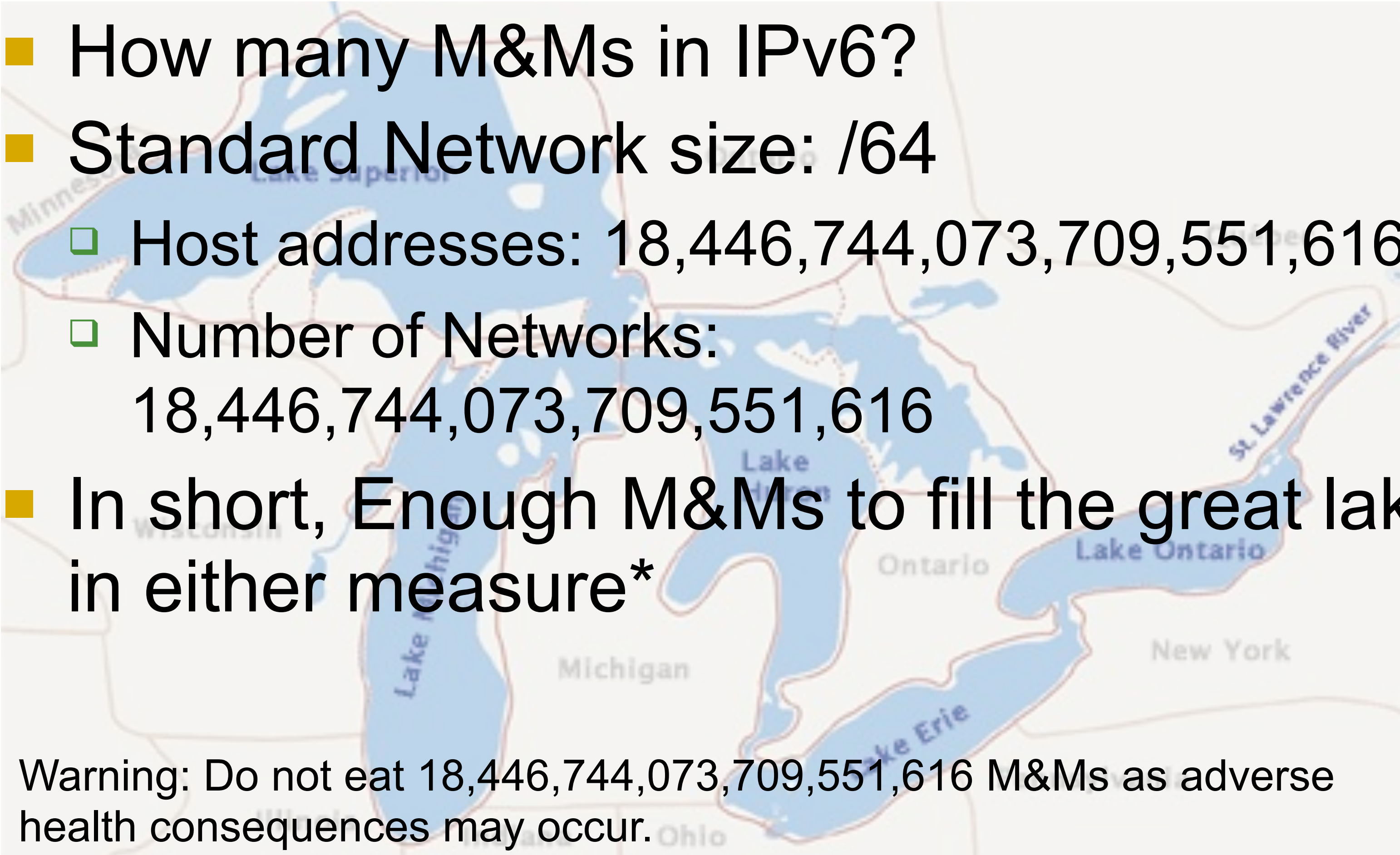
One IPv4 /24 -- 254 M&Ms

  - Number of /24s: Aprox. 14.5 Mililon

Full Address Space, One M&M per /24 covers 70% of a football* field

*An American Football field

Parting Thoughts: IP Addresses as M&Ms -- IPv6



- How many M&Ms in IPv6?
- Standard Network size: /64
  - Host addresses: 18,446,744,073,709,551,616
  - Number of Networks: 18,446,744,073,709,551,616
- In short, Enough M&Ms to fill the great lakes in either measure*

Warning: Do not eat 18,446,744,073,709,551,616 M&Ms as adverse health consequences may occur.

If IP addresses were M&Ms
Final thought:

If IP addresses were M&Ms Final thought:

They'd taste better.

OK.. The real conclusion

- You've learned about the structure of IPv6 and IPv6 addressing
- You've learned how to plan your addressing strategy.
- You've learned a little about deploying IPv6.
- Mostly, deployment is a lot like IPv4, but, without the address scarcity.

Similarities

- Same routing protocols (mostly)
- Same issues
- Same solutions (mostly)
- Similar security models and tactics
  - Note: NAT isn't security in IPv4. Stateful ispection provides security, NAT just depends on stateful inspection.
- Similar host configuration methods

Differences

- Much bigger address space
- Restoration of the End-to-End model of networking (Really, this is a good thing, even if you find it a little scary at first)
- Stateless Autoconfiguration
- (Slightly) Crippled DHCP
- DHCP Prefix Delegation
- Hex Notation
- No more writing out netmasks (YAY!)

IPv6 Deployment -- Next Steps

- This was just an introduction.
- Hopefully you have enough knowledge to build an IPv6 test lab and start gaining experience.
- Use a tunnel at first if you need to.
- Don’t forget to pressure your vendors so that they are ready when you need them to be (or only a little late)
  - Network
  - Equipment

IPv6 Deployment -- Next Steps part 2

- Build your test lab
- Try it out
- Try different scenarios
- Get to know what works and doesn't work in your particular environment
- Help is available
  - ARIN IPv6 wiki: http://www.getipv6.info
  - Hurricane Electric: http://www.tunnelbroker.net
  - ~~Rent~~ ;-) Email me: owend@he.net

Q&A

?

Contact:

Owen DeLong
IPv6 Evangelist
Hurricane Electric
760 Mission Court
Fremont, CA 94539, USA
http://he.net/

owend at he dot net
+1 (408) 890 7992

The end

Thank you

Contact:

Owen DeLong
IPv6 Evangelist
Hurricane Electric
760 Mission Court
Fremont, CA 94539, USA
http://he.net/

owend at he dot net
+1 (408) 890 7992