
# Developing an IPv6 Addressing Plan Guidelines, Rules, Best Practice

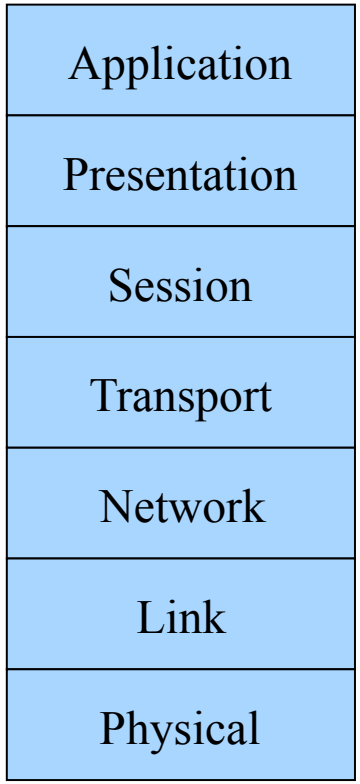
Ron Broersma
DREN Chief Engineer
SPAWAR Network Security Manager
ron@spawar.navy.mil

# Introduction

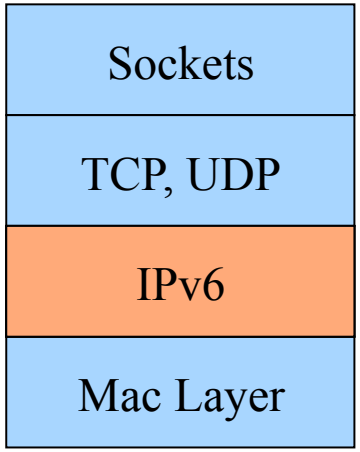
- IPv6 deployment includes:
  - obtaining a block of IPv6 addresses (a "prefix") for your organization and its networks.
  - establishing a plan for how those addresses will be assigned to your networks and subnets.
- Observation:  Many plans have serious flaws
  - usually takes about 3 times to get it right
  - many plans include the same basic mistakes
- Goal of this presentation:
  - not intended to be a comprehensive tutorial
  - review the common mistakes, and the reasons behind them
  - save everyone time and effort, by avoiding those mistakes
    - less re-numbering
  - take full advantage of the vast address space now available to us

# The piece that has changed

ISO 7 Layer Model

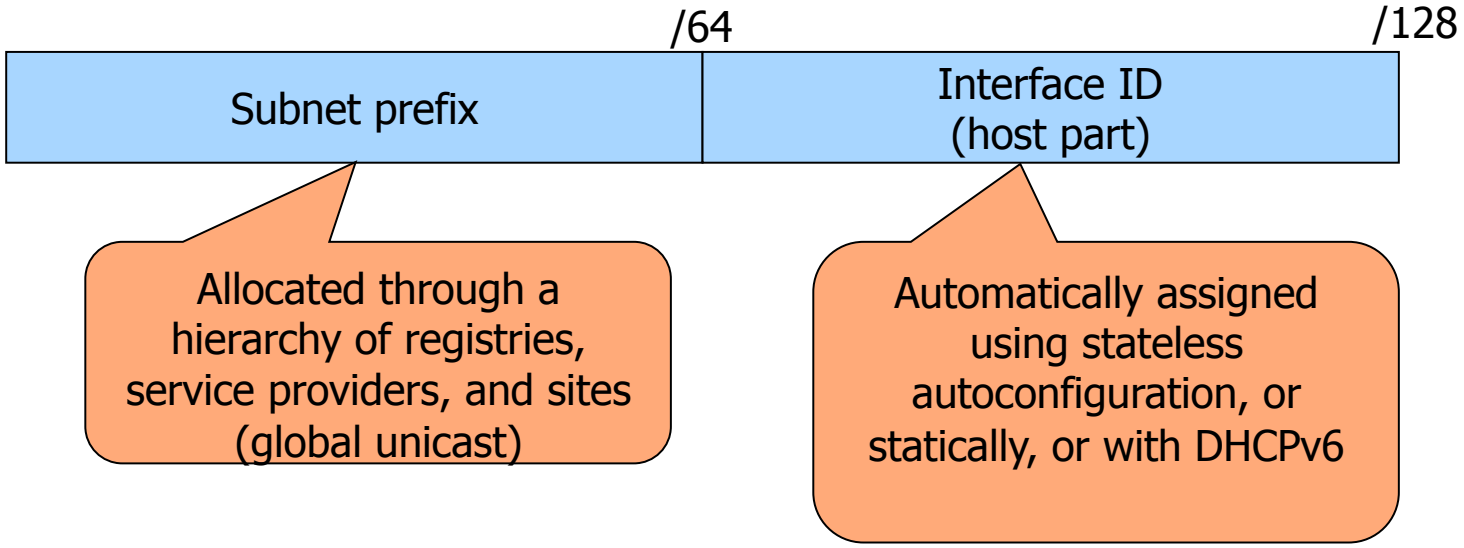



Internet Stack



# Address Structure

- Unicast addresses are structured as a subnetwork prefix and an interface identifier.




*Size of a given (sub)network is effectively not limited by the number of unique host values as was the case in IPv4 where a /24 (Class C) net can only have 254 hosts.*

# Address Types

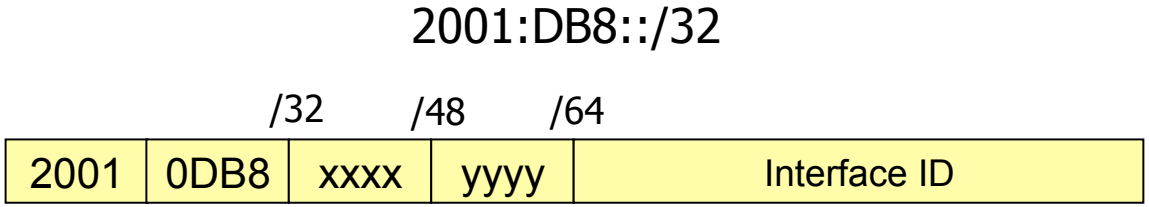
| Prefix | Designation and Explanation | IPv4 Equivalent |
|---|---|---|
| ::/128 | **Unspecified**<br>This address may only be used as a source address by an initialising host before it has learned its own address. | 0.0.0.0 |
| ::1/128 | **Loopback**<br>This address is used when a host talks to itself over IPv6. This often happens when one program sends data to another. | 127.0.0.1 |
| ::ffff/96<br><br>Example:<br>::ffff:192.0.2.47 | **IPv4-Mapped**<br>These addresses are used to embed IPv4 addresses in an IPv6 address. One use for this is in a dual stack transition scenario where IPv4 addresses can be mapped into an IPv6 address. See RFC 4038 for more details. | There is no equivalent. However, the mapped IPv4 address can be looked up in the relevant RIR's Whois database. |
| fc00::/7<br><br>Example:<br>fdf8:f53b:82e4::53 | **Unique Local Addresses (ULAs)**<br>These addresses are reserved for local use in home and enterprise environments and are not public address space.<br><br>These addresses might not be unique, and there is no formal address registration. Packets with these addresses in the source or destination fields are not intended to be routed on the public Internet but are intended to be routed within the enterprise or organisation.<br><br>See RFC 4193 for more details. | Private, or RFC 1918 address space:<br><br>10.0.0.0/8<br>172.16.0.0/12<br>192.168.0.0/16 |
| fe80::/10<br><br>Example:<br>fe80::200:5aee:feaa:20a2 | **Link-Local Addresses**<br>These addresses are used on a single link or a non-routed common access network, such as an Ethernet LAN. They do not need to be unique outside of that link.<br><br>Link-local addresses may appear as the source or destination of an IPv6 packet. Routers must not forward IPv6 packets if the source or destination contains a link-local address.<br><br>Link-local addresses may appear as the source or destination of an IPv6 packet. Routers must not forward IPv6 packets if the source or destination contains a link-local address. | 169.254.0.0/16 |

| Prefix | Designation and Explanation | IPv4 Equivalent |
|---|---|---|
| 2001:0000::/32<br><br>Example:<br>2001:0000:4136:e378:<br>8000:63bf:3fff:fdd2 | **Teredo**<br>This is a mapped address allowing IPv6 tunneling through IPv4 NATs. The address is formed using the Teredo prefix, the server's unique IPv4 address, flags describing the type of NAT, the obfuscated client port and the client IPv4 address, which is probably a private address. It is possible to reverse the process and identify the IPv4 address of the relay server, which can then be looked up in the relevant RIR's Whois database.<br><br>You can do this on the following webpage:<br>http://www.potaroo.net/cgi-bin/ipv6addr | No equivalent |
| 2001:0002::/48<br><br>Example:<br>2001:0002:6c::430 | **Benchmarking**<br>These addresses are reserved for use in documentation. They should not be used as source or destination addresses. | 198.18.0.0/15 |
| 2001:0010::/28<br><br>Example:<br>2001:10:240:ab::a | **Orchid**<br>These addresses are used for a fixed-term experiment. They should only be visible on an end-to-end basis and routers should not see packets using them as source or destination addresses. | No equivalent |
| 2002::/16<br><br>Example:<br>2002:cb0a:3cdd:1::1 | **6to4**<br>A 6to4 gateway adds its IPv4 address to this 2002::/16, creating a unique /48 prefix. As the IPv4 address of the gateway router is used to compose the IPv6 prefix, it is possible to reverse the process and identify the IPv4 address, which can then be looked up in the relevant RIR's Whois database.<br><br>You can do this on the following webpage:<br>http://www.potaroo.net/cgi-bin/ipv6addr | There is no equivalent but 192.88.99.0/24 has been reserved as the 6to4 relay anycast address prefix by the IETF. |
| 2001:db8::/32<br><br>Example:<br>2001:db8:8:4::2 | **Documentation**<br>These addresses are used in examples and documentation. They should never be source or destination addresses. | 192.0.2.0/24<br>198.51.100.0/24<br>203.0.113.0/24 |
| 2000::/3 | **Global Unicast**<br>Other than the exceptions documented in this table, the operators of networks using these addresses can be found using the Whois servers of the RIRs listed in the registry at:<br>http://www.iana.org/assignments/ipv6-unicast-address-assignments | No equivalent single block |
| ff00::/8<br><br>Example:<br>ff01:0:0:0:0:0:0:2 | **Multicast**<br>These addresses are used to identify multicast groups. They should only be used as destination addresses, never as source addresses. | 224.0.0.0/4 |

http://www.ripe.net/ipv6-address-types/ipv6-address-types.pdf

# Example Allocation

- Your enterprise is allocated a Global Unicast Prefix*

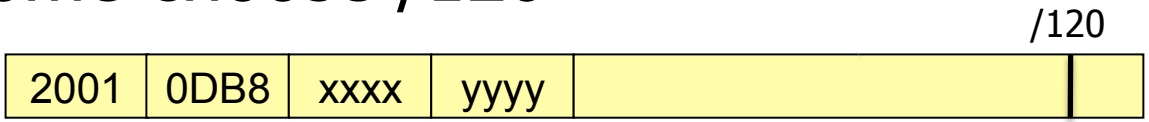



- How do you assign xxxx and yyyy throughout your enterprise?

* "The default provider allocation via the RIRs is currently a /32." (RFC 5375)

# Big mistake #1

- Using other than /64 for subnets
- Some choose /120




- Reasoning:
  - "host" part is same size as in IPv4 (8 bits)
  - /64 is wasteful
  - the security guy wants to be able to enumerate all hosts by scanning the subnet, just like in IPv4

# IPv4 practice gets in the way

- Being conservative with addresses
  - operating on the notion that addresses are very scarce
- Making the subnet mask long, to avoid waste. Examples:
  - /30 for point-to-point links
  - if you only have 10 hosts on a subnet, then allocate a /28
  - squeezing as many subnets as possible out of a /24

# Making the paradigm shift

- You may be un-qualified to develop a final IPv6 addressing plan if you think:
  - /64 for subnets is wasteful
  - /64 for point-to-point links is wasteful
  - /48 for small sites is wasteful

# Subnets are /64

- If you choose other than /64, the following things will not work:
  - Neighbor Discovery
  - Secure Neighbor Discovery
  - Stateless Address Autoconfiguration (SLAAC)
  - Microsoft DHCPv6
  - Multicast with Embedded-RP
  - Mobile-IPv6
  - and many other things in the future
- Using other than /64 for subnets goes against:
  - RFC 4291 “IPv6 Addressing Architecture”
  - RFC 5375 “IPv6 Addressing Considerations”

# Subnets are /64

*"For all unicast addresses, except those that start with the binary value 000, Interface IDs are required to be 64 bits long."* (RFC 4291)

*"Using /64 subnets is strongly recommended, also for links connecting only routers. A deployment compliant with the current IPv6 specifications cannot use other prefix lengths."* (RFC 5375)

# What about point-to-point links?

- Even if we finally agree that subnets are /64, some will argue that point-to-point links must be /126 (like an IPv4 /30) or /127.
  - Can't waste a whole /64 when you need only 2 addresses
- Best practice is to allocate /64 for point-to-point links
  - whether you need 2 out of 2**64 or 200 out of 2**64, there's not much difference in "waste"
- But what about that DoS problem from the ping-pong effect?
  - This will not happen on a RFC 4443 compliant IPv6 implementation
  - If you have a non-compliant device (Juniper), you can set the interface mask to /126 on the interface as a temporary workaround until your device is fixed, but you should still allocate a /64 for the link.
  - Never use /127 (See RFC 3627), but also look at RFC 6164.

# Mistake #2

- Thinking you have to get the addressing plan right the first time
  - Unless you have operational experience with IPv6 deployments and transition, you WILL get it wrong.
  - Usually takes about 3 times to get it right.
- Thinking you can't afford to re-address
  - Since the first plan is probably a throw-away, you will have to re-address when you come up with a revised plan.

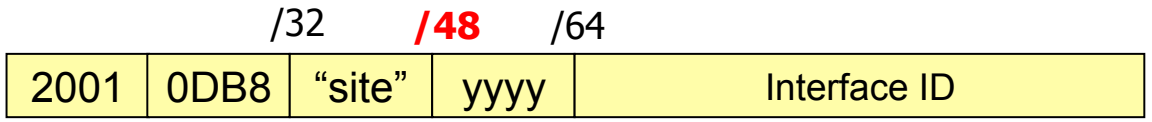
# Iterative planning approach

- Assume the first plan is a throwaway
  - Don't put too much energy into it, because it is only temporary
- Do some initial limited IPv6 deployments based on the initial addressing plan
  - testbeds
  - public facing services
- Gain operational experience
  - realize ways to improve the addressing plan
  - interact with the community to get ideas
- Develop your next addressing plan
  - put more energy into this one
  - readdress the existing IPv6 infrastructure
- Do a wider deployment with the new plan
  - internal servers, maybe clients.
- Iterate

# Mistake #3

- Trying to be too creative about how much address space to allocate to a "site"
  - Thinking you need to allocate large amounts of space to large sites, and much smaller amounts to small sites
- Assuming that large allocations to small sites is wasteful
  - Go back and review the slide on being stuck in the IPv4 conservation paradigm.

# "Sites" get a /48




- Here, the "site" field is 0x0000-0xFFFF
  - That gives you 65,536 sites!
  - That's not enough?
- And each site get 65,536 subnets
  - That's not enough? Its like a "Class A" block of huge subnets.
- Standardize!
  - It simplifies things administratively and operationally.

# Mistake #4

- Justify "upwards", rather than pre-allocate "downwards".
  - Requiring sites to develop documentation and justification for their address space requirements
  - Allocating to those groups or sites based on that justification

# Pre-allocation

- You can easily pre-allocate to the site level
  - see slide on "sites get a /48"
- Within sites, addressing can align with existing subnet structure
  - later, you may want to re-address your IPv4 networks (but don't worry about that just yet).

# Mistake #5

- Host-centric allocation rather than subnet-centric
  - Thinking that address allocation has anything to do with the number of hosts

# Focus on subnets

- A /64 subnet has enough room for this many hosts:

  18,446,744,073,709,551,616

- You don't have to think about whether a subnet is large enough for all your hosts.
- You don't have to worry about "growing" a subnet later if you get more hosts.
- Just focus on your network topology (links, subnets, VLANs, etc.) and align with that.

# Once again

When doing an address plan, a major driver in IPv4 was efficiency and conservation

In IPv6, efficiency and conservation is NOT a major driver, but instead it is all about better alignment with network topology, accommodation of security architecture, and operational simplicity through standardization

# Other Considerations

- In IPv6, every interface has multiple addresses
  - In IPv4, we thought of a "host" as having a single IP address
- Embedding IPv4 addresses in IPv6 addresses adds administrative burden and limits flexibility
  - limited long term benefit, so don't do it
  - It is reasonable to copy just the "host" part of the IPv4 address into the IID (host part) of the IPv6 address

# Other Considerations

- There is an opportunity to align the addressing plan with security topology, to simplify ACLs
  - This is the type of thing you may start to incorporate into your 3rd version of your plan.
- Internal aggregation is not nearly as critical as aggregating route announcements to your ISP
  - you can afford to carry a few thousand routes internally, but the Internet can't afford to carry all your /48's or longer.

# Other Considerations

- Most of the context here has been for large enterprises that aggregate into a very few connections to one or two ISPs, and use "provider-independent" (PI) space.
- If you have a lot of small outlier sites that are single-homed directly to an ISP, have them get their address space from that ISP, known as "provider-aggregatable" (PA) space.

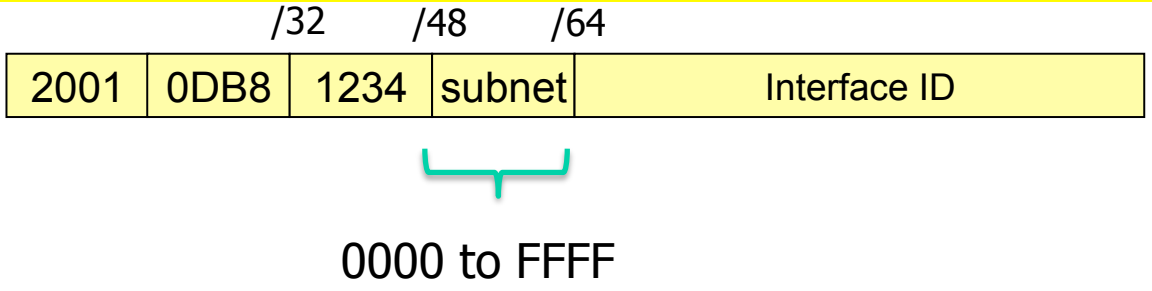
# Adding structure or hierarchy

- Examples:
  - grouping of sites by
    - region
    - service delivery point
  - grouping of subnets within a site to align with
    - IPv4 mapping
    - routing topology
    - security topology

# Adding structure or hierarchy

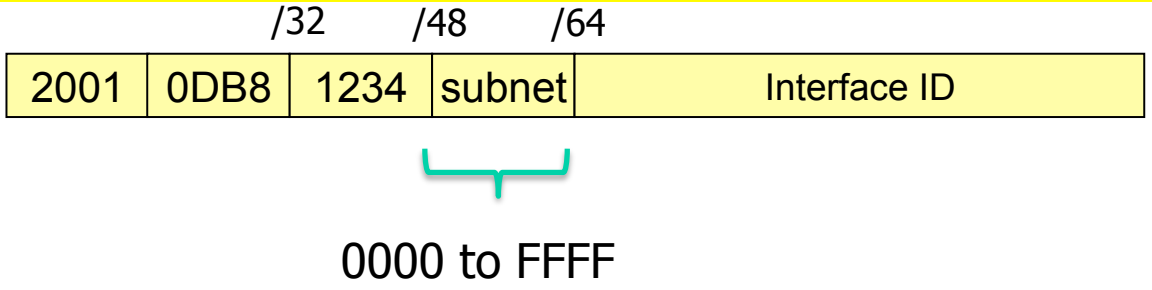
- Recommendation: add grouping or hierarchy on nibble (4 bit) boundaries
  - Aligns better with hex digits
  - Aligns better with grouping in DNS PTR records
- Examples:
  - /36 for regions
    - 16 regions with 4096 sites per region
  - /44 for service delivery points
    - 16 customers per SDP, up to 4096 SDPs
  - /52 to align with IPv4 allocations
    - can map up to 16 allocations

# Subnet numbering example




- You could just assign them incrementally
  - 0, 1, 2, 3, etc
- You could have them match some part of your existing IPv4 subnet numbers
  - Like the 3rd octet of your subnets addresses, if you have a "Class B" and all your subnets are /24's
- You may want to create some hierarchy, if you have separate enclaves or security zones or want to map to multiple existing IPv4 allocations.

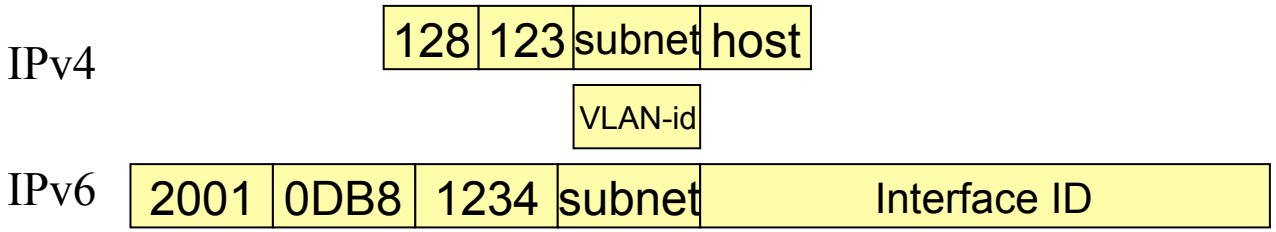
# Hierarchy Example




- Save the top 4 bits of the subnet number for mapping to IPv4 allocation (or other grouping)
  - That's a /52
- Subnet numbers are then 000 to FFF
  - 4096 subnets per /52 (you only need 256, but 3 hex digits allows you to keep decimal notation)

# Example Addressing Scheme

- Address the network for consistency between protocols
  - Align VLAN number with 3rd octet of IPv4 address
  - Align IPv6 "subnet number" with the above



# IPv6 Addressing Example

| Subnet | IPv4 | IPv6 |
|---|---|---|
| Offices | 128.123.1.0/24 | 2001:480:1234:1::/64 |
| Computer Room | 128.123.2.0/24 | 2001:480:1234:2::/64 |
| DMZ (BR) | 128.123.100.0/24 | 2001:480:1234:1100::/64 |
| DMZ (FW) | 128.123.101.0/24 | 2001:480:1234:1101::/64 |
| fw-to-br | 128.123.254.0/30 | 2001:480:1234:1000::/64 |
| fw-to-ir | 128.123.254.4/30 | 2001:480:1234:0000::/64 |
| | | |

Notes:

- Used subnet 000 for "infrastructure" links
- /52 used to designate security zone (0 – trust, 1 – untrust)
- IPv4 and IPv6 subnet numbers try to align, where possible (when IPv4 subnets are /24)
- didn't use /126's nor /127's for the point-to-point links

# Privacy Addresses (RFC 4941)

- Incompatible with many Enterprise environments
  - Need address stability for many reasons
    - Logging, Forensics, DNS stability, ACLs, etc.
- Enabled by default in Windows
  - Breaks plug-n-play because we have to visit every Windows machine to disable this feature.
- Just added in Mac OS X "Lion".
- Ubuntu thinking about making it default.

*[Privacy addresses] are horrible and I hope nobody really uses them, but they're better than NAT.*
… Owen DeLong, Hurricane Electric

# Living with Privacy addresses

- Where your clients support DHCPv6, use that to assign addresses
  - No DHCPv6 client support in Windows XP, Mac OSX before 10.7 (Lion), etc.
- If all your Windows systems are in Active Directory, use GPO to disable privacy addresses
- Options for other systems:
  - configure system to disable privacy addresses
    - registry setting in Windows (see below)
  - configure addresses statically on the hosts
  - keep a historical record of all MAC address to IPv6 address mappings for every host, for correlation in IDS and forensics tools

```
netsh interface ipv6 set privacy state=disabled store=persistent
netsh interface ipv6 set global randomizeidentifiers=disabled store=persistent
```

# Additional Guides

- Preparing an IPv6 Addressing Plan

http://www.ripe.net/lir-services/training/material/IPv6-for-LIRs-Training-Course/IPv6_addr_plan4.pdf

- IPv6 Address Design, a few practical principles

http://www.txv6tf.org/wp-content/uploads/2011/09/Doyle-TXv6TF_09142011.pdf


# End of Addressing discussion


# Other topics


# What's missing: IPv6 Operational Experience

- Lots of planning is underway
  - transition planning
  - address planning
- Much of this planning is done by individuals who have never touched an IPv6 packet
- Too much energy is being wasted on plans that are flawed, because they are not based on operational experience
- It is more important to turn on IPv6 now and start moving some IPv6 traffic, than it is to have a complete plan

# Getting IPv6 experience

- Run IPv6 at home
  - Get a tunnel from Hurricane Electric
- Get the IPv6 Certification from HE
- Managers:
  - make sure your network engineers are doing the above, or something similar
- Run IPv6 in a testbed environment
- IPv6-enable just your public-facing services to start with
- Then you can start comprehensive planning

# Go native

- "native IPv6" means "don't use tunnels".
  - some confuse this term to mean IPv6-only, but that is not the case.
- Access to Legacy IPv4 networks and systems will be necessary for years to come.
  - we need both IPv4 and IPv6 at the same time.
  - IPv4 and IPv6 are not directly interoperable
- Use "dual stack" as the IPv6 transition mechanism
  - can use translators in the interim, but NOT long term. goal is end-to-end native IPv6.

# About translators

- Common scenario:
  - Don't IPv6-enable your actual public web site, but instead front-end it with an IPv6-to-IPv4 translator
- This is OK as an interim step, because of the extreme importance of IPv6-enabling the public Internet
- But the target is end-to-end native IPv6, so consider any such translators to be very temporary
  - unless that translation device is already in the path for reasons unrelated to IPv6 transition

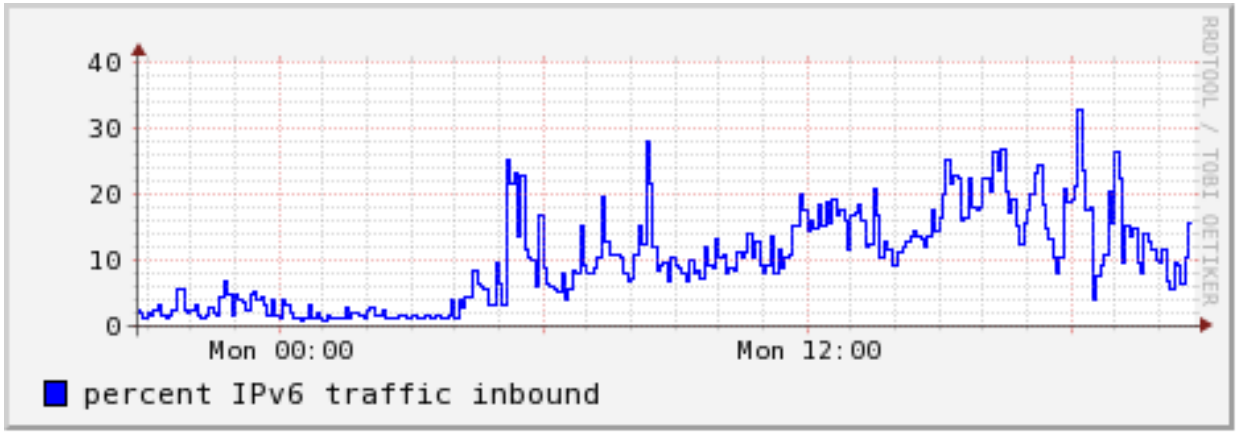
# From a Microsoft talk…

## Best Practices

- Leave Windows in the default configuration (IPv6 enabled)
- Block IPv6, IP Protocol 41, and Teredo at the perimeter
- Set up a test lab to test and learn IPv6
  - Use ISATAP for a low cost test lab deployment
- Monitor Production DNS Servers for AAAA records
  - The presence of AAAA records prior to rollout probably indicates Public IPv4 addresses are in use
- Document and test broadcast domains
- Link planned IPv6 subnets to existing Active Directory Sites
- Set High Priority on genuine Router Advertisements (RFC 4191)
- Use 802.1x when possible

# IPv6 traffic percentage

- From a server perspective, what percentage of the Internet will try to reach you over IPv6 today?
  - 0.4%
- From a client perspective, what percentage of Internet traffic is IPv6, where everything at your site is IPv6-enabled:



# Another event like World IPv6 Day?

- June 2012
- You should plan to IPv6-enable your public facing services before then

# Final Comments


# END
# Any Questions?

Contact me at:

ron@spawar.navy.mil