IPv6 Training

Joe St Sauver, Ph.D. (joe@oregon.uoregon.edu)

NCFTA, Pittsburgh, PA May 5th, 2010

http://www.uoregon.edu/~joe/ipv6-training/

Part 1. Do We Need IPv6?

Why Do We Need IPv6? There Are Many Legitimate Reasons

- Such as…

1.1 We're running out of IPv4 addresses (this is the one reason which everyone always thinks of)

1.2 IPv6 may allow us to regain end-to-end transparency (widely overlooked but very important)

1.3 Long term, we need to contain route table bloat

1.4 IPv6 has been mandated in some environments (e.g., theoretically for federal government networks)

Reasons for Deploying IPv6 Which Don't Tend to Survive Scrutiny

1.5 IPv6 will inherently improve Internet security (it won't -- in particular, IPSec is **NOT** being universally deployed on all IPv6 links!)

1.6 IPv6 will simplify renumbering, improve routing performance by simplifying packet formats, improves support for QoS (sigh!), facilitate mobility, etc. -- these may or may not be properties of IPv6 as actually deployed, but it doesn't matter because people aren't leveraging these even if they are.

So let me emphasize that these are all **non-reasons** for folks to be deploying IPv6. If you hear people tell you to deploy IPv6 for these reasons, be very skeptical.

Some Real Reasons for *NOT* Deploying IPv6

- Some sites may NOT be in any particular rush to deploy IPv6 because…

  1.7 Their site already has abundant IPv4 space

  1.8 Anything that's available via IPv6, is also available via IPv4 (at least for now)

  1.9 Their network uses middleboxes (such as firewalls or network load balancers) that are not fully IPv6 aware

  1.10 Their network provider is still dragging their heels when it comes to providing IPv6 connectivity.

Bogus "Reasons" for NOT Deploying IPv6

- There are also many, many bogus "reasons" why some sites may NOT want to deploy IPv6, including:

  1.11 "I'm too busy working on more important things."

  1.12 "This whole IPv4 exhaustion thing is a bunch of malarky -- folks will figure out some way to stretch out what space we've still got available."

  1.13 "IPv6 and those super long addresses are just too weird/hard."

  1.14 "Bottom line, customers (except you!) just aren't asking for IPv6 support."

The Real Reason Why The *Bad Guys*/ *Bad Gals* Are Interested in IPv6

- The real reasons why the Bad Guys/Bad Gals are interested in IPv6 is that at many sites:

  -- IPv6 network traffic isn't monitored on par with IPv4 traffic (if it is monitored at all), so IPv6 can be a great covert channel

  -- IPv4 security measures (such as perimeter firewalls or filter ACLs) are often not replicated for IPv6

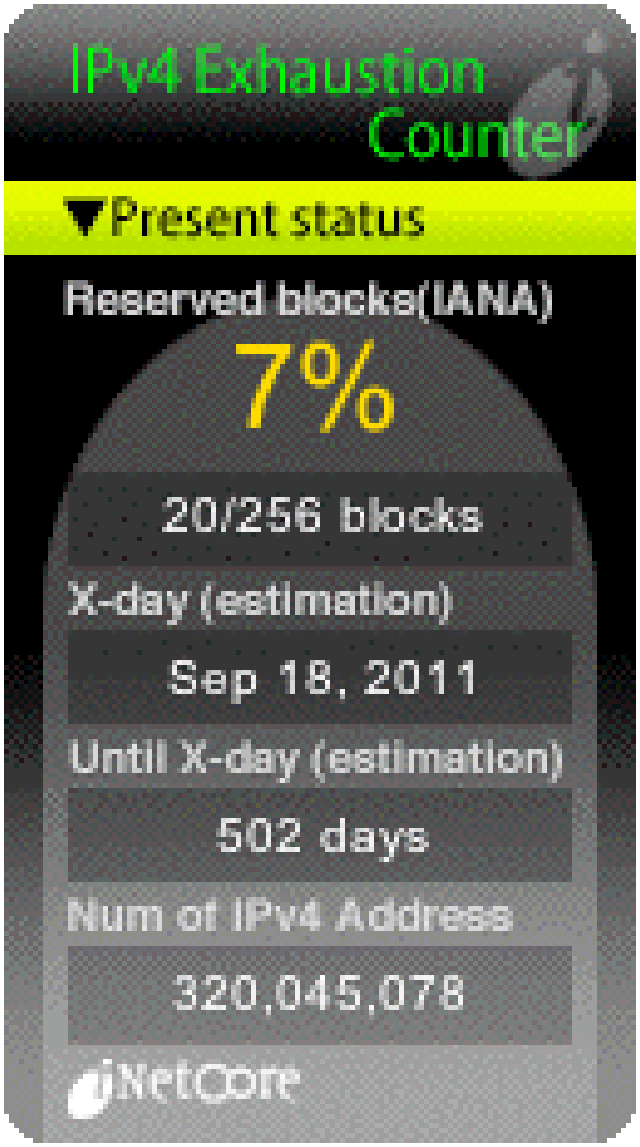
  -- Law enforcement isn't ramped up to deal with online badness that involves IPv6 (example: I suspect that few if any cybercrime cops have IPv6 connectivity

1.1 IPv4 Address Exhaustion

IPv4 Addresses: A Scarce Resource

- There is a finite pool of available IPv4 addresses, and we're really, really close to running out.
- Based on the best available forecasts, see http://www.potaroo.net/tools/ipv4/index.html , the last IPv4 blocks will be allocated by IANA on 18-Sep-2011
- The regional internet registries (RIRs), such as ARIN, RIPE, APNIC, LACNIC and AFRINIC will exhaust the address space they've received from IANA less than a year later, around 29-Apr-2012
- These best estimates are based on current trends, and actual exhaustion might accelerate (or might slow) depending on what the community does (but probably not by much). From now till June 12th, 2012 is roughly two years and one month away. That's not much time.

inetcore.com/project/ipv4ec/index_en.html



Just 25 Months…

- Twenty five months isn't much time if you're an ISP and you don't already have an IPv6-capable infrastructure (or plans and processes underway for getting there). You may need to do some "forklift upgrades" to at least some of your gear, you'll need to arrange to get IPv6 address space, and you'll need to update your provisioning systems and network monitoring systems, and you'll need to train your staff and end users, and…
- There's a lot to do, and not a whole lot of time left in which to do it.
- Moreover, there are a relatively limited number of people with IPv6 expertise available to help you through any rough spots you may encounter.
- Fortunately, this is something of a slow-speed "crash$_{11}$"

The Internet, Post-IPv4 Run Out

- Running out of IPv4 addresses isn't like running out of water in the desert, or air while SCUBA diving -- if you already have IPv4 address space, the IPv4 address space you already have will continue to work just fine.
- People who WILL run into problems, however, include:
  -- growing ISPs who need more IPv4 IP addresses
  -- new ISPs who need IPv4 addrs just to get rolling
  -- customers of existing IPv4-based ISPs who may need to access network resources available ONLY via IPv6
  -- customers behind weird/broken stopgap kludges
- Eventually, we risk the bifurcation of the Internet: part of the Internet may cling to IPv4 addressing, while the rest may end up having no choice but to use IPv6 addressing. Eventually, this will be a serious issue.

1.2 Internet Transparency

"But What About NAT?"

- While some sites (including uoregon.edu) assign each system on campus a globally routable IP address, other sites (including many home users and many corporate sites) routinely employ network address translation (or "NAT"). NAT (actually PAT) makes it possible for multiple workstations to all use a single shared globally routable IPv4 address. If all you do is browse the web or use a web email service such as Hotmail, or Yahoo! Mail, or Gmail, NAT may superficially work fine for your needs.
- On the other hand, if you want to do Internet video conferencing, or use peer-to-peer applications, or you're trying to track down and fix malware-infested hosts connecting from behind a NAT, you may find that NAT will make your head explode.

NAT: A (Semi) Protocol-Aware Protocol

- Some network protocols (such as H.323) embed IP addresses in the traffic generated by those protocols.
- Because NAT rewrites network addresses, it needs to know HOW each protocol embeds IP addresses in network traffic streams. That is, NAT boxes need to be "protocol aware," and thus networks using NAT are NOT "end-to-end transparent." (Packets get rewritten during transport while passing through a NAT)
- If a NAT box faces traffic of a type that it doesn't know how to handle, such as some new protocol, it can't rewrite that traffic, and as a result that application will fail when run behind a NAT. This is very commonly the case for H.323 video conferencing, for example.
- Because of this, NAT'd networks can stifle application layer network innovation, or at least make it far harder!

The "Two Port Internet"

- Because of the problems that application developers face getting past NAT boxes (and restrictive firewalls!) it is common for developers to implement new protocols over http instead of developing new native protocols. Some of my colleagues refer to this as the "two port Internet" -- in this model, virtually all user traffic is either http (port 80) or https (port 443).
- Obviously this is something of an exageration (they forgot about DNS for example :-)), but it isn't entirely an argument w/o merit. All you need to do is look at network traffic and try to identify what applications make up most of the packets or most of the octets to see the problem -- you can't do it just based on ports.
- C.F.: "A Look at the Unidentified Half of Netflow," www.uoregon.edu/~joe/missing-half/missing-half.pdf

End-To-End Transparency

- If you'd like to read about the importance of end-to-end transparency, some excellent starting points are:

  -- RFC2775, "Internet Transparency," B. Carpenter, February 2000, http://tools.ietf.org/rfc/rfc2775.txt

  -- RFC4924, "Reflections on Internet Transparency," B. Aboba and E. Davies, July 2007, http://tools.ietf.org/rfc/rfc4924.txt

- While Internet transparence is less often mentioned than imminent IPv4 address exhaustion as a reason why we need to deploy IPv6, transparency is nonetheless a very important underlying motivation for IPv6.

1.3 IPv6 and Controlling Route-Table Bloat

Controlling Route Table Bloat

- Another important (if little recognized) reason for promoting use of IPv6 has been the need to control the growth in the size of the global routing table. In fact, RFC4984 (http://www.ietf.org/rfc/rfc4984.txt) states,

  "[...] routing scalability is the most important problem facing the Internet today and must be solved [...]"

- If you're not a network engineer, you likely don't think much about growth in the size of the global routing table, so let's step back and do a little backfill on this topic.

What Is "Routing?"

- You may have wondered how packets know how to get from site A to site B. The answer is "routing."
- When a server at a remote location has network traffic for a site, a series of hop-by-hop decisions get made: at each router, a packet needs to decide where to go to get closer to its ultimate destination. A packet comes in on one interface, and may have a choice of two, three, or even a dozen or more outbound interfaces for the next step in its journey. Which path should it take next?
- Each router has a table of network IP address prefixes which point at outbound router interfaces, and that table guides packets on the next step of their journey.
- After the packet traverses that link, the process is then repeated again at the next router for the next link, etc

Most Little Sites: No Impact on Table Size

- If you're a small and simple site with just a single upstream provider, that upstream ISP may aggregate the network addresses you use with other customers it also services. Thus, the global routing table might have just a single table entry servicing many customers.
- Once inbound network traffic hits the ISP, the ISP can then figure out how to deliver traffic for customer A, traffic for customer B, etc. The ISP handles that, the Internet doesn't need to know the gory local details
- Similarly, outbound, if you're a small site with just a single upstream provider, your choice of where to send your outbound traffic is pretty simple: you've only **got** one place you can send it. This allows you to set a "default route," sending any non-local traffic out to your ISP for eventual delivery wherever it needs to go.

Sites With Their Own IP Address Space

- Sometimes, however, sites have their own address space.
- For example, UO has the prefix 128.223.0.0/16, the IPv4 addresses 128.223.0.0--128.223.255.255.
- That address block is not part of any ISP's existing address space.
- If UO wants to receive traffic intended for those addresses, it needs to announce (or "advertise") that network address block to the world.
- When UO's route gets announced, each router worldwide adds that route to its routers' routing tables, and thus know how to direct any traffic it may see that's destined for UO, to UO.
- Without that route, our address space would be unreachable.

Some Sites Have Multiple Prefixes

- Sometimes sites have more than one chunk of network address space. For example, Indiana University has 129.79.0.0/16, 134.68.0.0/16, 140.182.0.0/16, 149.159.0.0/16 149.160.0.0/14, 149.165.0.0/17, 149.166.0.0/16, 156.56.0.0/16, and 198.49.177.0/24, and IU has nine slots in the global routing table associated with those prefixes.
- Other sites may have a range of addresses which **could** be consolidated and announced as a single route, but they intentionally "deaggregate" that space, perhaps announcing a separate route for each /24 they use. For example, BellSouth announces roughly 4,000 routes globally, even though it could aggregate those routes down to less than 300 routes if they were so inclined.

"So What? Who Cares About Route Growth?"

- **Each route in the global routing table need to be carried by routers at every provider in the world.**
- Each route in the route table consumes part of a finite pool of memory in each of those routers. When routers run out of memory, "Bad Things" tend to happen. Some routers even have relatively small fixed limits to the maximum size routing table they can handle (see http://tinyurl.com/route-table-overflow).
- Each route in the route table will potentially change whenever routes are introduced or withdrawn, or links go up or down. The larger the route table gets, the longer it takes for the route table to reconverge following these changes, and the more CPU the router requires to handle that route processing in a timely way

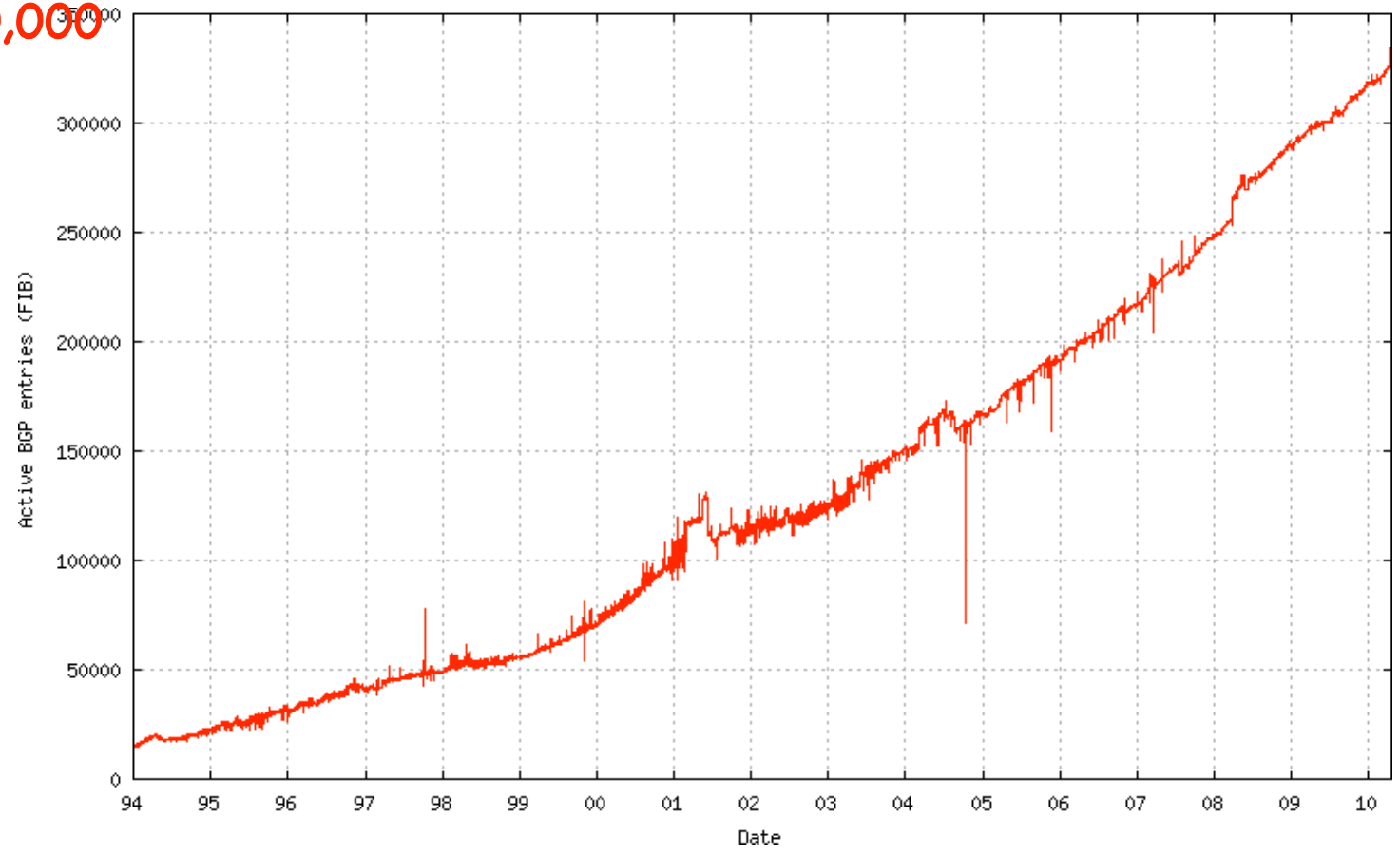
An Aside on Route Table Growth and Convergence

- There are some indications that we're getting luckier with route table performance than we might have expected; see Geoff Huston "BGP in 2009" talk from the recently completed ARIN Meeting in Toronto:

  https://www.arin.net/participate/meetings/reports/ARIN_XXV/PDF/Monday/Huston-bgp.pdf

The IPv4 Route Table DOES Continue to Grow…

350,000



Source: http://bgp.potaroo.net/as6447/

IPv6 Was Supposed to Help Fix That

- When IPv6 was designed, address assignment was supposed to be hierarchical. That is, ISPs would be given large blocks of IPv6 address space, and they'd then use chunks of that space for each downstream customer, and only a single entry in the IPv6 routing table would be needed to cover ALL the space used by any given ISP and ALL their downstream customers (see RFC1887, "An Architecture for IPv6 Unicast Address Allocation")
- But now, let's pretend that my Internet connectivity is important to me, so I don't want to rely on just a single ISP -- I want to connect via multiple ISPs so that if one provider has problems, the other ISPs can still carry traffic for my site. This connection to multiple sites is known as "multihoming."

If I'm Multihomed, Whose Address Space Do I Use?

- When I get connectivity from sites A, B and C, whose address space would I announce? Address space from A? Address space from B? Address space from C? No...
  -- A doesn't want me to announce part of its address space via B and C
  -- B doesn't want me to announce part of its address space via A and C
  -- C doesn't want me to announce part of its address space via A and B.
- I need to either assign each host multiple addresses (e.g., one address from A, one from B, and one from C), or I need to get my own independent address space which I can use for all three ISPs, but which will then take up a slot in the global routing table.

The Original Multiple IP Approach in IPv6

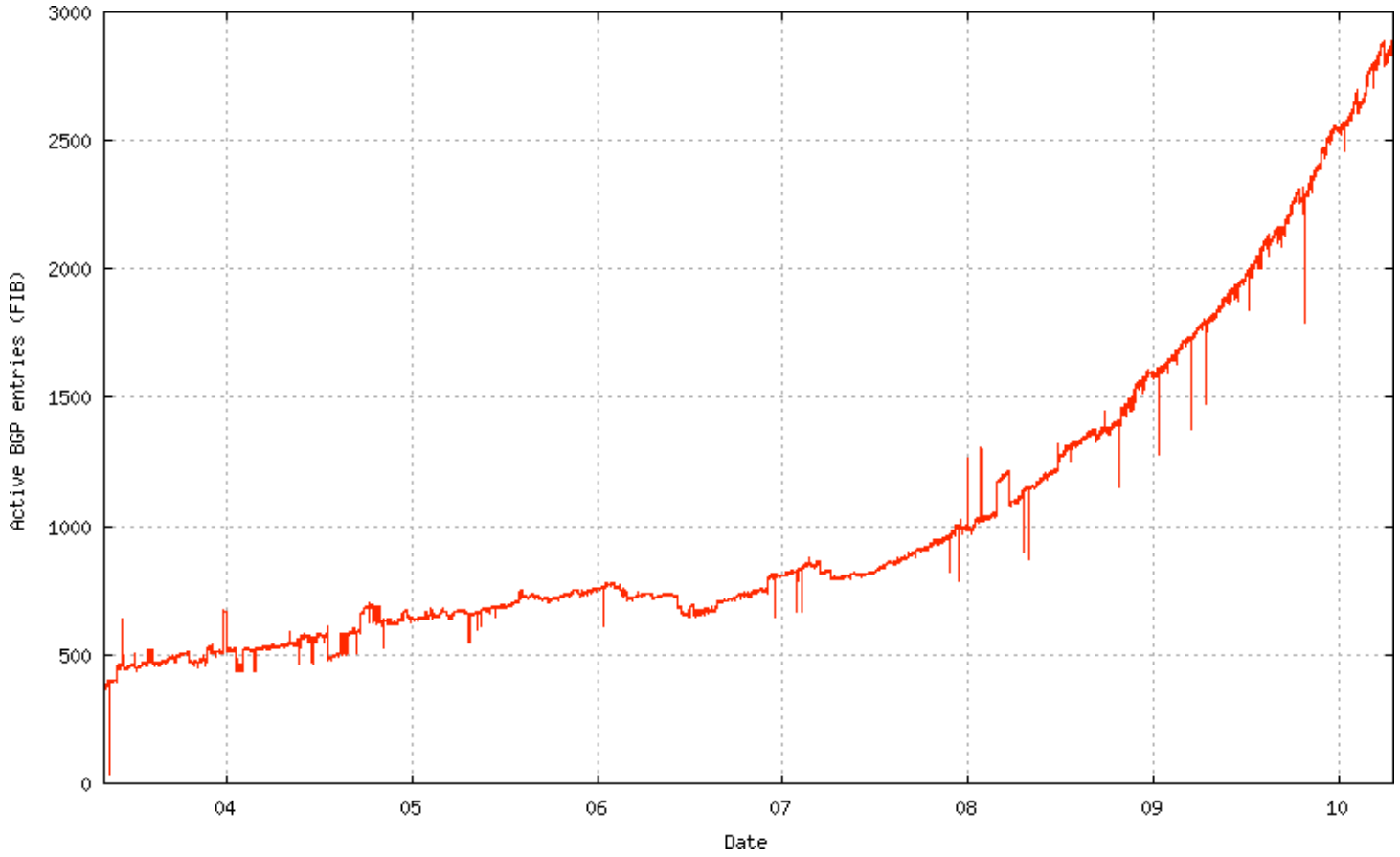
- The multiple IP approach was the original "answer" to this question in the IPv6 world.
- But if I assign multiple IPs to each host, one for each upstream ISP I connect to, how do I know which of those IP addresses I should use for outbound traffic generated by each host? Do I arbitrarily assign the address from A to some traffic? The address from B to other traffic? What about the address from C?
- Which of those addresses do I map to my web site or other servers via DNS? Do I use just A's address? Just B's? Just C's? All three of those addresses? What if one of my providers goes down? Will traffic failover to just the other two providers quickly enough?

The Multihoming Reality Today

- IPv6 multihoming without use of provider independent address space is one of the unsolved/open issues in the IPv6 world today. Operationally, in the real world, ISP customers who need to multihome request their own provider independent IPv6 address space (cue Sonny and Cher: "The beat goes on, and the beat goes on…")
- Route table growth may be a critical issue facing the Internet in the long term, but for now, the community has "dropped back into punt formation," and we're doing what needs to be done (at least for now) to get IPv6 deployed in a robust way (e.g., with multihoming). The good news is that the IPv6 table is still small (so we still have time to solve the IPv6 routing table growth issue); the bad news is that the IPv6 table is still small (which means many people still haven't deployed IPv6!)

IPv6 Route Table Growth

3000



Source: http://bgp.potaroo.net/v6/as6447/

1.4 IPv6 and Regulatory Compliance

Federal Networks, For Example, Are Supposed to Be IPv6 Ready

M-05-22

August 2, 2005

MEMORANDUM FOR THE CHIEF INFORMATION OFFICERS

FROM: Karen S. Evans
Administrator
Office of E-Government and Information Technology

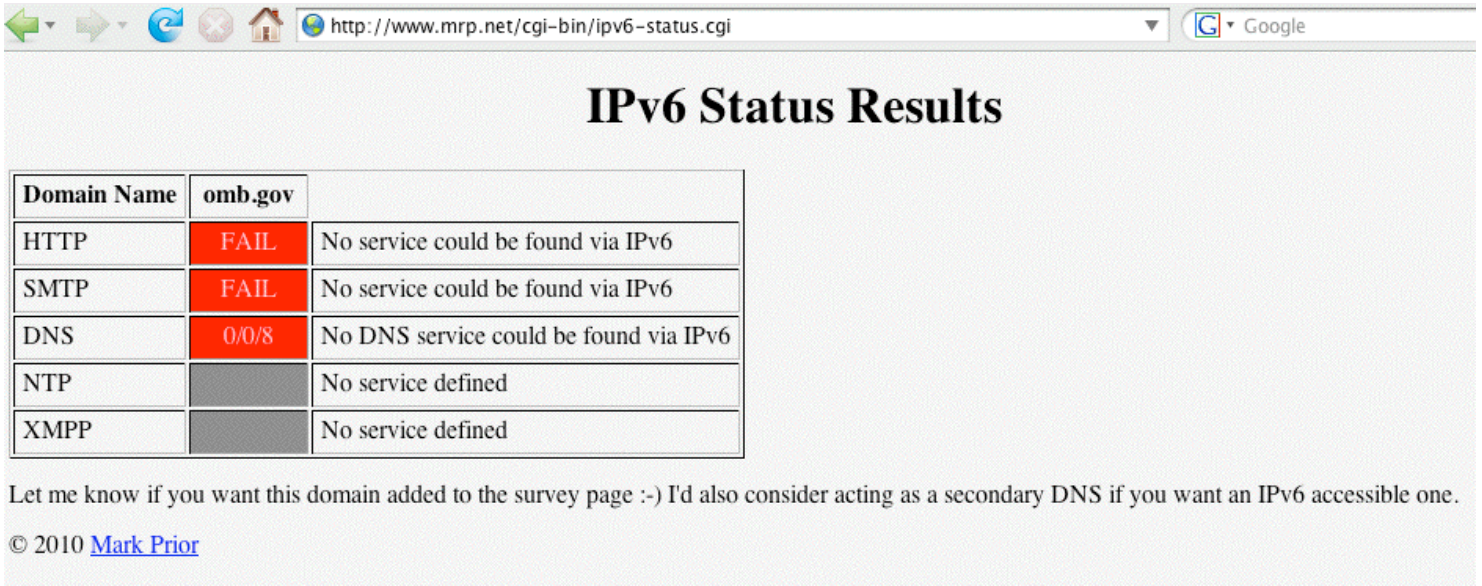
SUBJECT: Transition Planning for Internet Protocol Version 6 (IPv6)

As I stated in my testimony of June 29, 2005, before the House Committee on Government Reform, we have set June 2008 as the date by which all agencies' infrastructure (network backbones) must be using IPv6 and agency networks must interface with this infrastructure. This memorandum and its attachments provide guidance to the agencies to ensure an orderly and secure transition from Internet Protocol Version 4 (IPv4) to Version 6 (IPv6). Since the Internet Protocol is core to an agency's IT infrastructure, beginning in February 2006 OMB will use the Enterprise Architecture

Source: www.whitehouse.gov/omb/memoranda/fy2005/m05-22.pdf

Theoretically All Federal Networks (At Least Temporarily) Met That Mandate, But…

- Reportedly many federal networks, having passed one IPv6 packet (and thus, however briefly, demonstrated that their backbones were IPv6 capable), promptly "re-disabled" IPv6 (ugh!)
- Check your favorite fed sites -- are they v6 accessible? See: http://www.mrp.net/cgi-bin/ipv6-status.cgi



IPv6 and Fed Scorecard Network Homepages?

www.dhs.gov --> no
www.doc.gov --> no
www.dod.gov --> no
www.doe.gov --> no
www.dot.gov --> no
www.ed.gov --> no
www.epa.gov --> no
www.hhs.gov --> no
www.hud.gov --> no
www.doi.gov --> no
www.doj.gov --> no
www.dol.gov --> no
www.nasa.gov --> no
www.nsf.gov --> no
www.nrc.gov --> no
www.opm.gov --> no
www.sba.gov --> no
www.ssa.gov --> no
www.state.gov --> no
www.usaid.gov --> no
www.usda.gov --> no
www.ustreas.gov --> no
www.va.gov --> no

Or pick another federal agency of your choice: the pattern is pretty consistent I'm afraid...

"Planning Guide/Roadmap Toward IPv6 Adoption Within the US Government"

- This is a new document (ca. May 2009) from the Federal CIO Council Architecture and Infrastructure Committee, Technology Infrastructure Subcommittee, Federal IPv6 Working Group, see http://tinyurl.com/fed-cios-ipv6
- I quote: "The purpose of this document is to provide U.S. government agency leaders with practical and actionable guidance on how to successfully integrate Internet Protocol version 6 (IPv6) throughout their enterprise. [...] without a concentrated effort by Federal agencies to effectively and efficiently deploy secure IPv6 network services, the Government's technical advancement and ability to meet its mission needs will be critically impacted during the next 2 to 3 years."

A Major Potential Stumbling Block: Non-IPv6 Content Delivery Networks (CDNs)

- Many federal web sites (and key commercial web sites) use Akamai (or another CDN) in order to handle huge online audiences, deliver good performance worldwide, and to resist DDoS attacks.
- For example, www.irs.gov is actually just a cname for www.edgeredirector.irs.akadns.net; whois confirms that akadns.net actually belongs to Akamai.
- If Akamai doesn't do IPv6, will current major Akamai customers (such as Apple, Cisco, Microsoft, RedHat, the Whitehouse, etc.) be willing to deploy IPv6 for critical sites without them?
- BTW, at least one vendor, Limelight, DOES offer an IPv4 and IPv6 CDN service…

The Issue Isn't Just Web CDNs…

- A growing number of sites outsource their email operations.

- Unfortunately some email-as-a-service (and some cloud-based spam filtering services) don't support IPv6, thereby limiting the ability of their customers to integrate IPv6 into their existing IPv4-based services.

- CDNs and outsourced email and spam filtering services aren't the only reason why IPv6 adoption has been slow at some major Internet sites, but those are certainly important stumbling blocks that will need to get resolved.

1.5 Well, Won't IPv6 At Least Improve "Network Security" Due to IPv6 Having "Mandatory" IPSec?

(Sorry, No)

IPv6 and IPsec

- IPsec is not new with IPv6; in fact, IPsec dates to the early 1990's. What's different when it comes to IPv6 is that support for IPsec was made "mandatory" for IPv6 (see for example "Security Architecture for IP," RFC4301, December 2005 at section 10, and "IPv6 Node Requirements," RFC4294, April 2006 at section 8.)
- **If actually used**, IPsec has the potential to provide:
  -- authentication
  -- confidentiality
  -- integrity, and
  -- replay protection
- All great and wonderful security objectives -- IF IPsec gets used. **Unfortunately, as we'll show you, what was supposed to be the cornerstone of the Internet's security architecture has proven in fact to be widely non-used.**

How Might IPsec Be Used?

- IPsec can be used to authenticate (using AH (the Authentication Header), RFC4302), or it can encrypt and (optionally) authenticate (using ESP (the Encapsulating Security Protocol), RFC4303)
- IPsec can be deployed in three architectures:
  -- gateway to gateway (e.g., securing a network segment from one router to another)
  -- node to node (e.g., securing a connection end-to-end, from one host to another)
  -- node to gateway (e.g., using IPsec to secure a VPN connecting from a mobile device to a VPN concentrator)
- IPsec has two main encrypting modes:
  -- tunnel mode (encrypting both payload and headers)
  -- transport mode (encrypting just the payload)
- IPsec also supports a variety of encryption algorithms (including "null" and md5 (yech)), and a variety of key exchange mechanisms
- These alternatives obviously provides tremendous flexibility, but that flexibility also brings along a lot of complexity.

But IPsec ISN'T Getting Used "Everywhere"

- IPv6 can be brought up without IPSec getting enabled, and in fact this is routinely the case -- see an example on the next slide.
- More broadly, if people are doing cryptographically secured protocols of *any* sort, they inevitably run into problems -- crypto stuff just tends to be inherently complex and hard to learn to use. For example, how many of you routinely use PGP or GPG to cryptographically sign or encrypt your email, eh? How many of you are doing DNSSEC to cryptographically protect the integrity of your DNS traffic?
- Now think about how often you see people moaning about problems they're having getting IPSec to work with IPv6 -- do you EVER see that sort of thing on the mailing lists or discussion groups you're on? No, right? That's because hardly anyone is doing IPSec with IPv6.

Some IPv6 Traffic Statistics From A Mac OS X Host

```
# netstat -s -finet6
[snip]

ip6:
        124188 total packets received
        [snip]
        84577 packets sent from this host
        [snip]

    ipsec6:
        0 inbound packets processed successfully
        0 inbound packets violated process security policy
        [snip]
        0 outbound packets processed successfully
        0 outbound packets violated process security policy
        [snip]
```

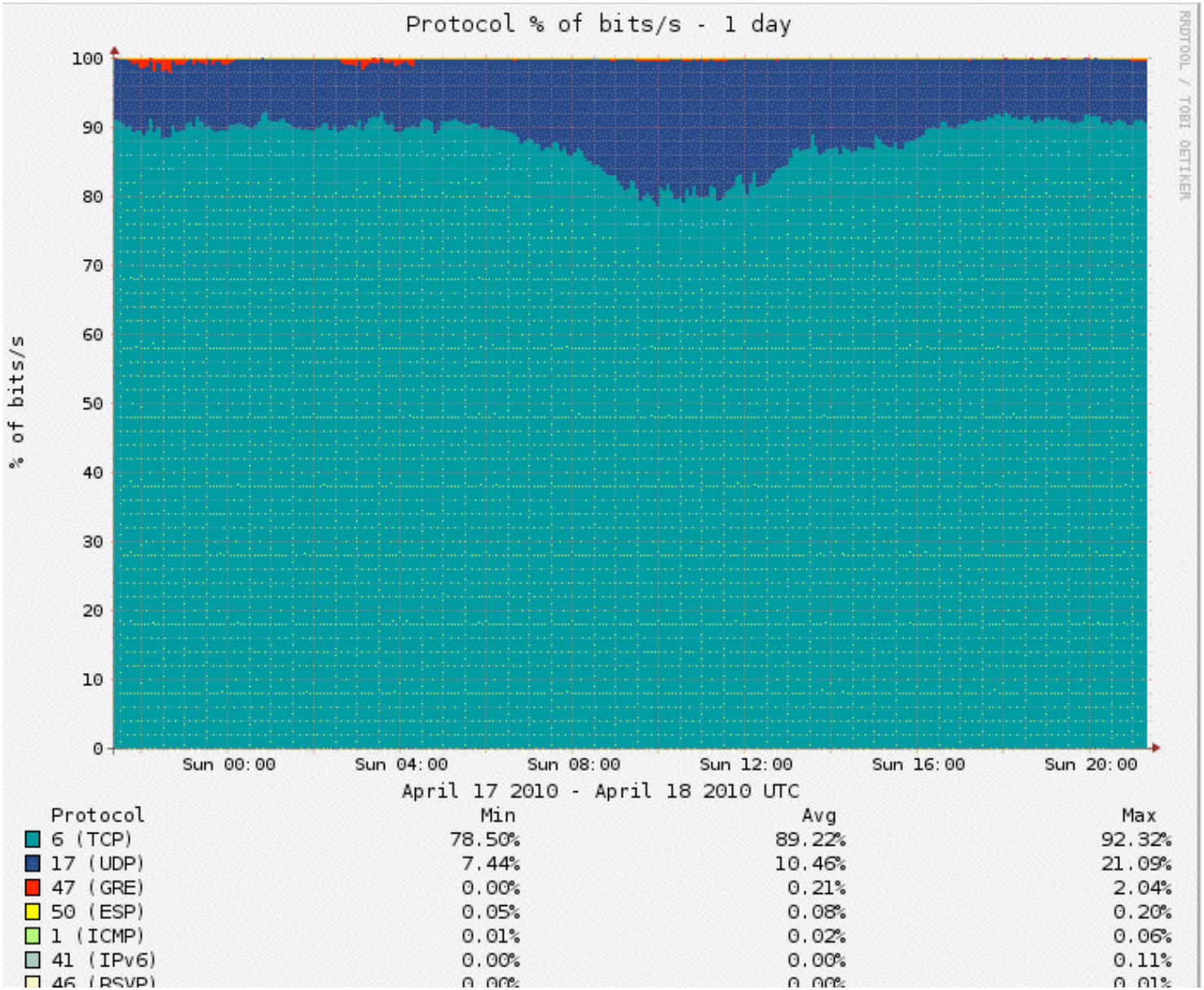
IPsec (Even on IPv4!) Isn't Getting Much Use

- Raw IPsec traffic (AH+ESP, protocols 50 & 51) isn't seen much on the commercial IPv4 Internet.

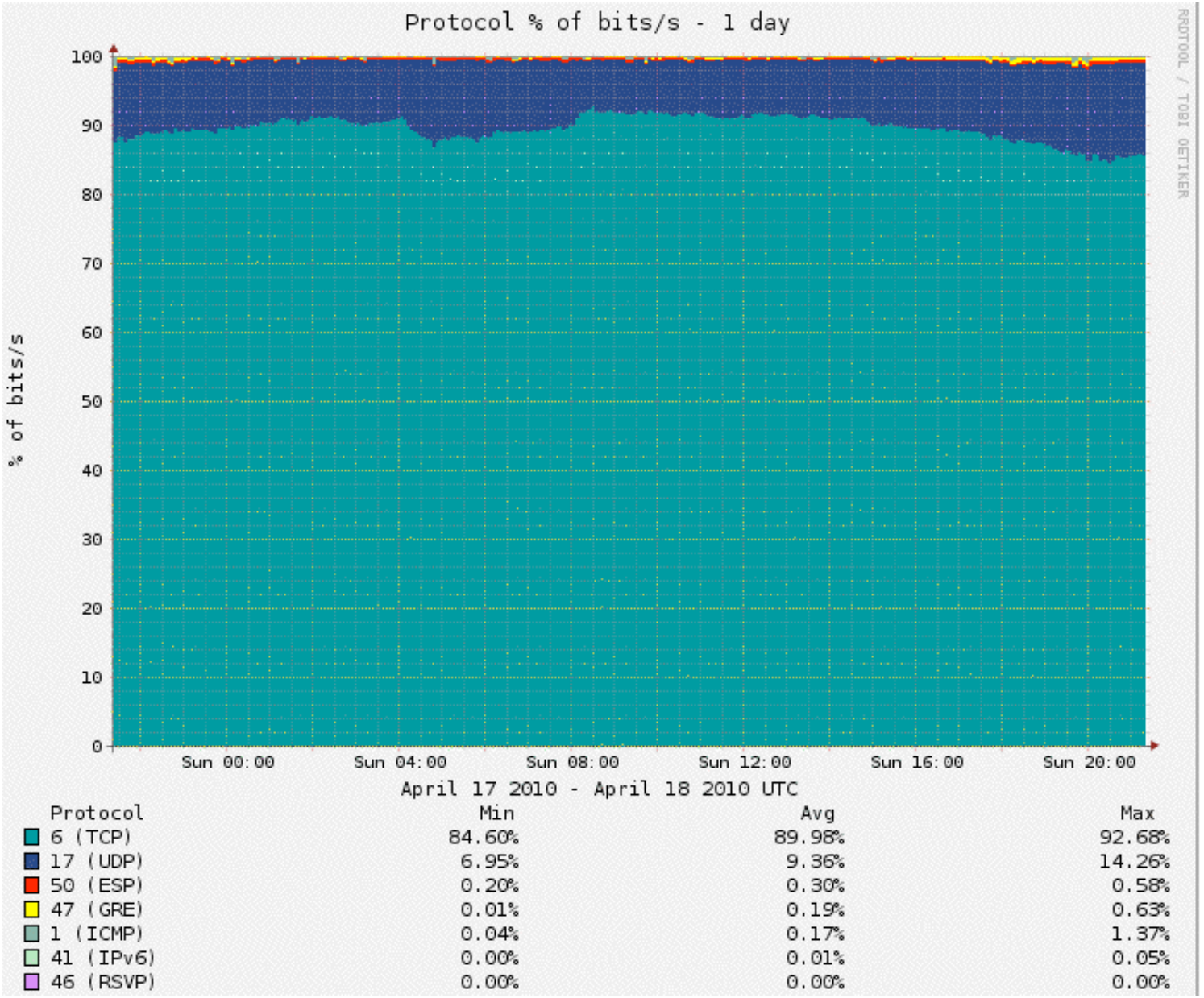
- For example, about one year ago, Jose Nazario of Arbor Networks estimated IPsec traffic at 0.9% of octets (statistic courtesy the ATLAS project).

- CAIDA (thanks kc!) also has passive network monitoring data available; see
  http://www.caida.org/data/passive/monitors/equinix-chicago.xml
  You can see the protocol distribution from a couple of CAIDA's monitors for one recent day on the next couple of slides. IPsec traffic is basically too small to even be seen for the most part.

Protocol Distribution From One of CAIDA's Passive Monitors



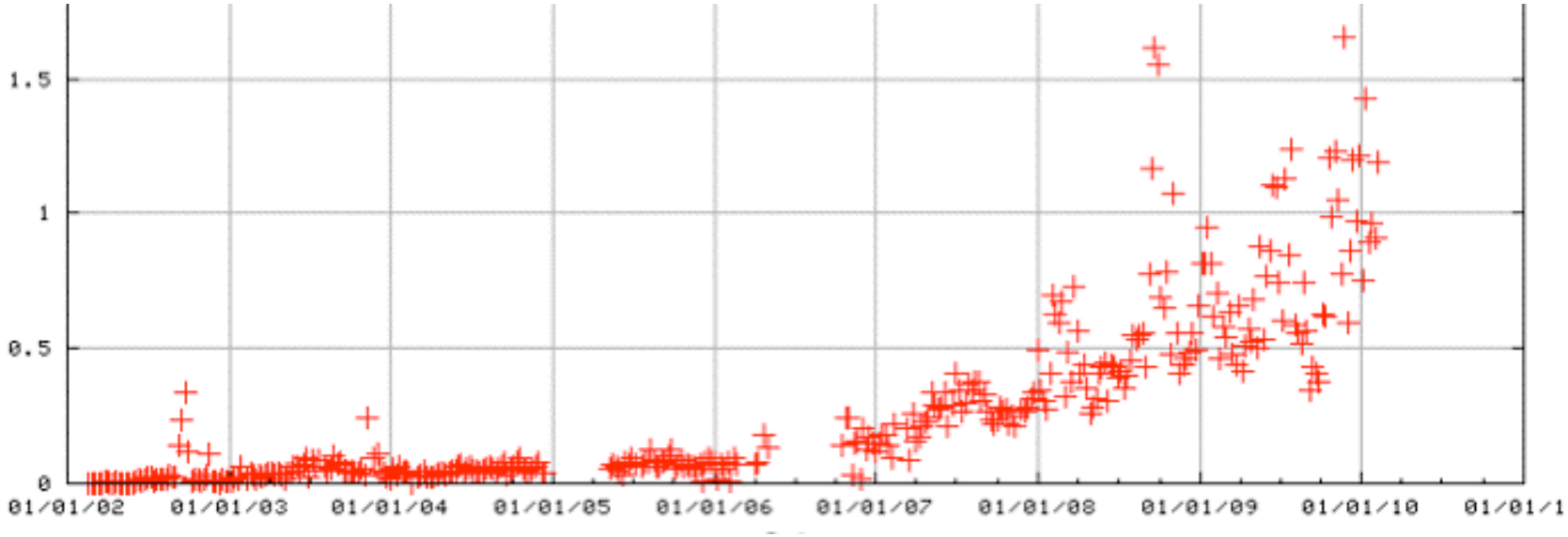
And The CAIDA Distribution Seen From Another Monitored Link



Protocol	Min	Avg	Max
6 (TCP)	84.60%	89.98%	92.68%
17 (UDP)	6.95%	9.36%	14.26%
50 (ESP)	0.20%	0.30%	0.58%
47 (GRE)	0.01%	0.19%	0.63%
1 (ICMP)	0.04%	0.17%	1.37%
41 (IPv6)	0.00%	0.01%	0.05%
46 (RSVP)	0.00%	0.00%	0.00%

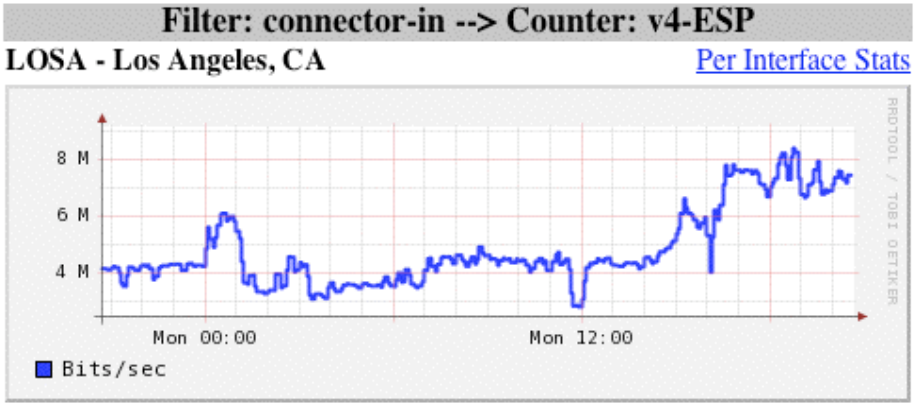
IPv4 IPSec Traffic on Internet2?

- Raw IPv4 IPsec traffic is quite rare on Internet2 as well, usually running well under 1.5% of octets (see table 7, http://netflow.internet2.edu/weekly/20100208/).
- Raw IPv4 IPsec traffic has been (gradually) growing, however. See Internet2 IPv4 IPsec ESP traffic levels (as a percent of all octets over time) by way of example:



IPv4 IPsec Traffic May Be From A Limited Number of Users/Systems (IPsec VPNs?)

- Protocols that are used by a **small** number of users or systems tend to exhibit "spikey" or rapidly varying aggregate traffic patterns while protocols which are in ubiquitous use tend to "average out" or be "smooth."
- The appearance of the firewall-based graph below (see http://vixen.grnoc.iu.edu/jfirewall-viz/index-bits.html) is consistent with IPsec traffic from just a few users (FWIW, total IPv4 traffic on this link runs ~25-40Gbps)



An Aside: IPv6 Traffic Visibility

- Ideally, for production IPv6 traffic, one would want **full IPv6 SNMP support** and **full IPv6 Netflow (V9) support**. Regretably, native IPv6 SNMP support and IPv6 V9 Netflow support remains elusive. That's increasingly unfortunate for IPv6 as a production protocol that is, or should be, on par with IPv4.
- One way to improve IPv6 visibility on ISP backbones would be to deploy at least a limited number of dedicated, IPv6-aware, passive measurement appliances. For instance, some network measurement researchers have been pleased with the IPv6 support available from InMon Corporation's Traffic Sentinel product (e.g., see http://www.inmon.com/products/trafficsentinel.php) or Lancope's StealthWatch (see http://www.lancope.com/)

What About IPv6 and Lawful Intercept?

- While network traffic visibility, including IPv6 traffic visibility, is important for network management and operations, network operators also need to meet their obligations to provider access for lawful intercept by law enforcement or the national security/homeland security community. It's not clear that all (any?) commercial or open source lawful intercept solutions fully support IPv6.

- I discussed this shortcoming years ago in "Upcoming Requirements from the US Law Enforcement Community to Technically Facilitate Network Wiretaps," May 2007, www.uoregon.edu/~joe/calea-requirements/terena.pdf

- But coming back to IPSec…

Why *Aren't* We Seeing More IPSec Traffic?

- Sites may not be deploying IPsec because IPsec (like many crypto-based security solutions) has developed a reputation as:

  -- not completely baked/still too-much under development
  -- too complex
  -- hard to deploy at significant scale
  -- less than perfectly interoperable
  -- likely to cause firewall issues
  -- potentially something of a performance hit (crypto overhead issues)
  -- congestion insensitive (UDP encapsulated IPsec traffic)
  -- something which should be handled as an end-to-end matter by interested system admins (from a network engineer perspective)
  -- something to be handled at the transport layer router-to-router (from an overworked system administrator's perspective)
  -- duplicative of protection provided at the application layer (e.g., encryption is already being done using ssh or ssl)
  -- complicating maintaining/debugging the network, etc., etc., etc.

- Regardless of whether those perceptions are correct (some may be, some may **not** be), IPsec adoption hasn't happened much to date.

But That's All Moot Relative to The Key Point…

- It would be foolhardy to expect IPsec to provide any material improvement to your site's security since the vast majority of your aggregate traffic (including virtually all your IPv4 traffic) will NOT be IPsec secured.

- On the other hand, the "good news" is that a lack of IPsec usage in the IPv6 world is substantively no worse than a lack of IPsec usage in the IPv4 world.

1.6 "IPv6 Will Simplify Renumbering, Improve Routing Performance By Simplifying Packet Formats, Will Improve Support For QoS, Will Facilitate Mobility, etc."

(maybe, but frankly, no one cares)

I Really Don't Mean to Sound Harsh, But…

- IPv6 may very well bring many cool new features to networking, but quite frankly, all of these incidental new features really don't matter -- they're not "make or break" drivers when it comes to adoption of IPv6.

- I'd love to hear hard evidence to the contrary, but truly, I've seen no indication that any of these other factors carry much weight in helping to shape the IPv6 go/no-go decision.

- On the other hand, there are some genuine reasons why people adamantly do NOT want to do IPv6.

1.7 One Real Reason For NOT Caring About IPv6: Some Sites Have Abundant IPv4 Space

If You Have Abundant IPv4 Address Space, It Can Be Hard to Get Excited About IPv6

- The definition of "abundant" will vary from site to site, but many colleges and universities have legacy /16's, and 2^16 (or roughly 65,500 addresses per /16) can seem like "a lot" of addresses (even though they can go awfully fast when you have a campus of 20,000-25,000 people, most of whom have multiple networked devices, plus lots of printers and other networked infrastructure).
- Others, however, have /8's (2^24, or roughly 16,777,200 addresses per /8), and that may be large enough to eliminate worries at those sites about address scarcity.
- If you haven't recently looked at the list of who has /8's, you can check http://www.iana.org/assignments/ipv4-address-space/ipv4-address-space

Considering Just Large Federal Netblocks…

- The US government (typically as the US Department of Defense) controls a relatively large fraction of the entire Intenet IPv4 name space, including a dozen slash 8's: 6/8, 11/8, 21/8, 22/8, 26/8, 28/8, 29/8, 30/8, 33/8, 55/8, 214/8, and 215/8 (to say nothing of additional /8 blocks controlled by defense industrial contractors with close/ extensive military contracts, plus miscellaneous smaller netblocks scattered hither and yon). Given that level of public IPv4 address space availability, one can understand that some federal agencies may not feel a particularly pressing need to move to IPv6 real ricky-ticky soon now.
- An aside about those large federal netblocks: even though addresses from some of those block may not show up in public routing tables, they <u>are</u> still being used, you're just not seeing them in some <u>public</u> routing tables.

Another Manifestation of the "Oh, No Worries, I've Got Plenty of Address Space" Phenomena

- While there are obviously only a comparative handful of sites which have their own public /8, many sites use RFC1918 "private" non-publicly routable address space, such as addresses from 10/8.
- If NAT currently meets your needs, as long as you have 10.0.0.0-10.255.255.255 (and 172.16.0.0-172.31.255.255, and 192.168.0.0-192.16.255.255) available, again, you may feel like your private IPv4 addressing needs are generally being well met (provided, of course, that you can also get the comparative handful of public IPv4 addresses you may also need).
- But, as we'll see later, use of NAT forecloses some of the easiest IPv6 transition mechanisms, such as use of 6to4.

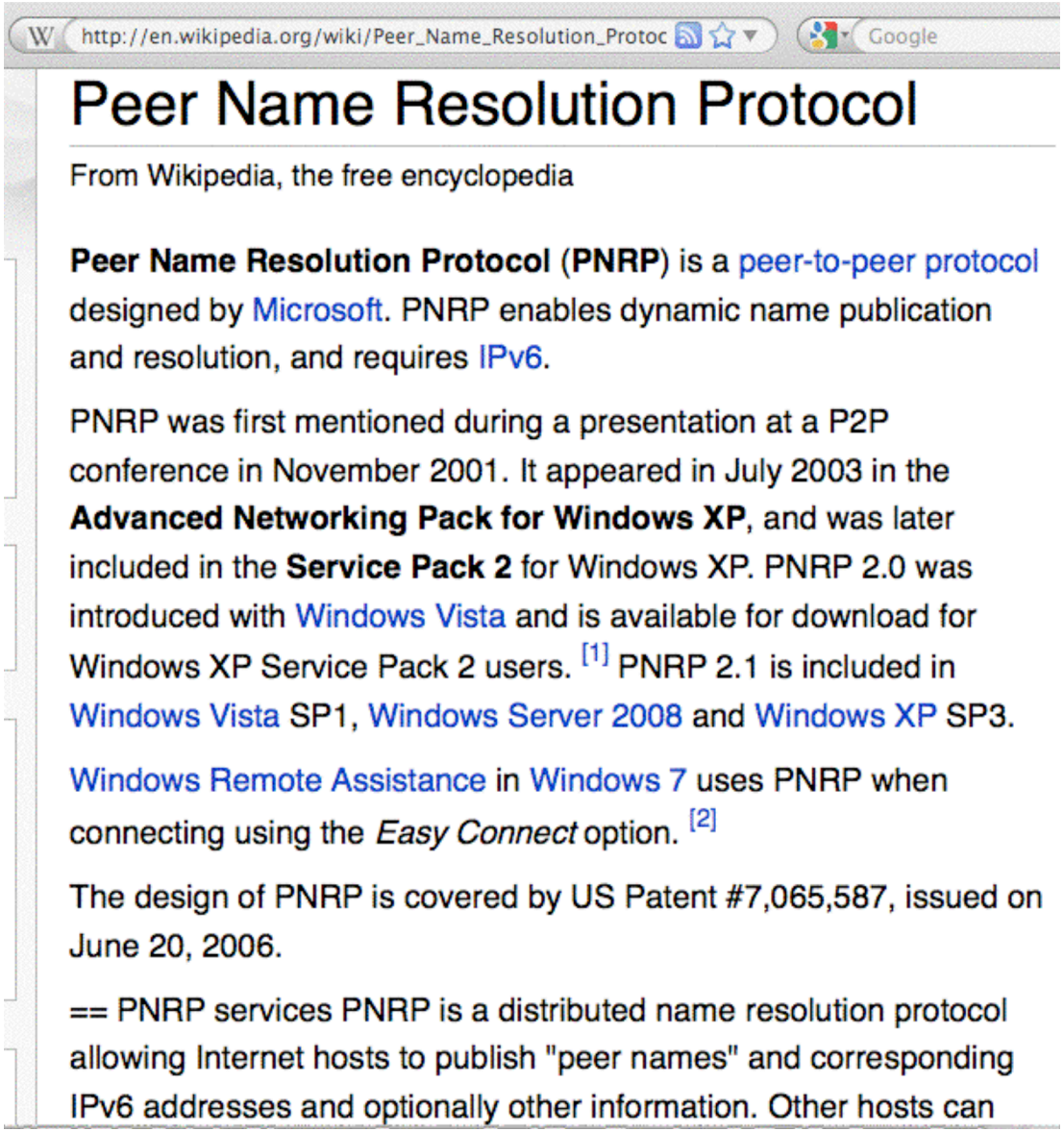
1.8 Anything That's Available Via IPv6, Will Also Be Available via IPv4

(So You Won't Suddenly Get Access To Lots of "Cool New Stuff" if You Start Doing IPv6)

An Important/Subtle Point to Understand

- Sometimes folks ask, "So if I begin to do IPv6, ***what new stuff can I get at that I can't get at already?***"
- Can you imagine Google, or Amazon, or Microsoft, or CNN, or <fill in the name of an important Internet site here> making their web site or other online resources ONLY available via IPv6? No, probably not.
- Any/all important Internet resources will ALWAYS be available via IPv4, even if those resources are ALSO available via IPv6.
- Thus moving to IPv6 does NOT magically give you access to new stuff that you couldn't get to via IPv4 (well, technically there are a few IPv6-only things mentioned at http://www.sixxs.net/misc/coolstuff/ , but nothing in and of itself that's enough to justify deploying IPv6)

Aside: Is PNRP An Exception to the Rule?



At Some Point In The Future, Though, The Default *Will* Change

- At some point in the future, people will eventually ask, "So if I still do IPv4, ***what sort of 'old stuff' can I get at that I can't get at already via IPv6?***"

- We're still a LONG way off from that point, but it WILL eventually happen.

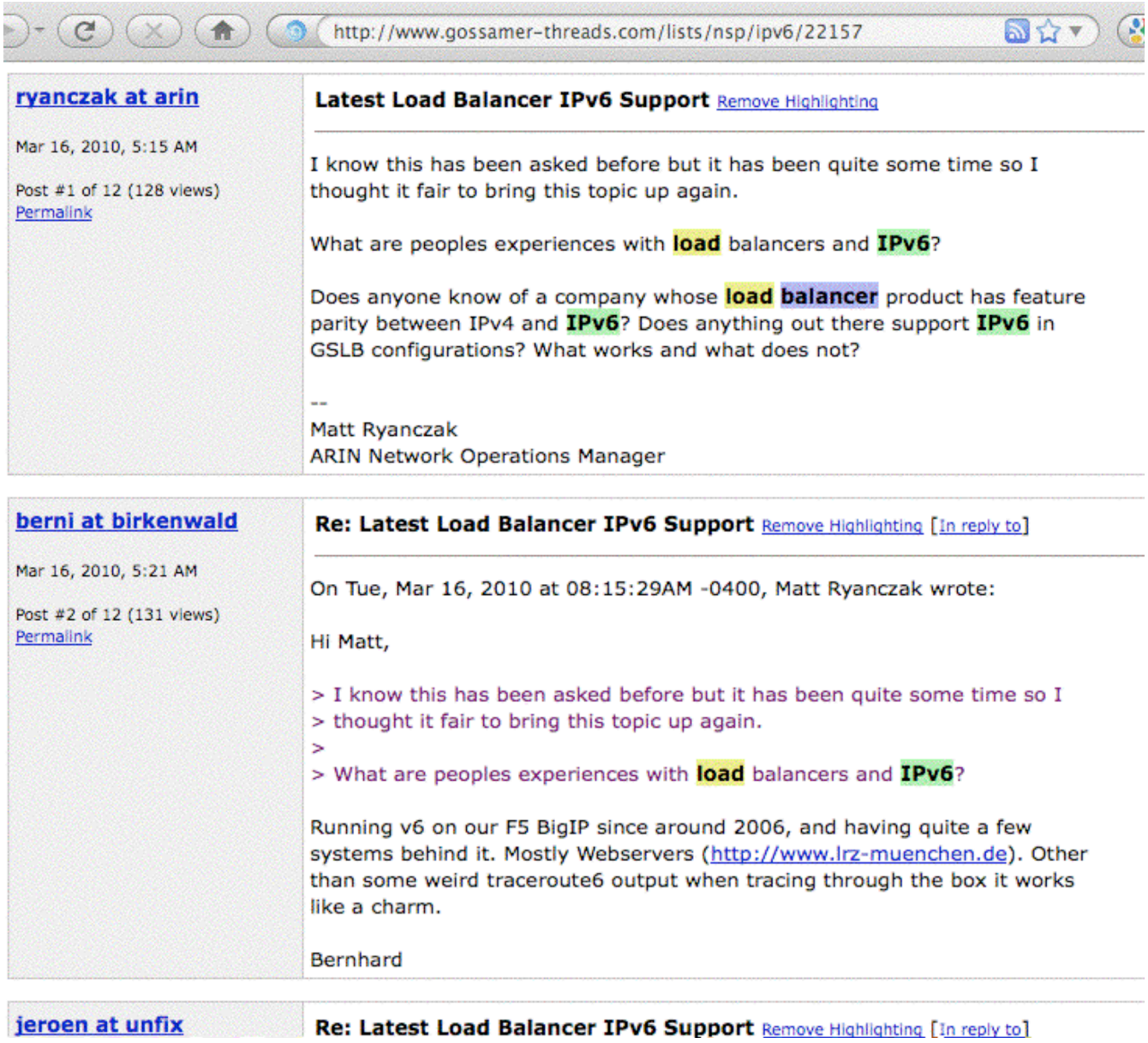
- Remember when people used to carry AppleTalk or IPX or DECNet on their local area networks? They sure don't anymore (or at least no one I know still does!)

1.9 "I'd Love to Move to IPv6, But My Network Uses Critical Middleboxes (Such as Firewalls or Network Traffic Load Balancers) That Aren't Fully IPv6 Aware"

Middleboxes Are Potentially A Major PITA

- The more I talk with sites about IPv6, the more I hate network middleboxes such as firewalls or network traffic load balancers. Sometimes those devices simply do not understand IPv6 at all. Other times they may have a primitive or incomplete implementation of IPv6, or require users to license an expensive "enhanced" software image to support IPv4 and IPv6.
- In general, I'd recommend moving firewalls as close to the resources they're protecting as possible (e.g., down to a subnet border, or even down to the individual ethernet port level), assuming you can't get rid of them altogether
- If you need to pay extra for IPv6 support in middleboxes, complain to your vendor or vote with your checkbook. IPv6 support is a core feature, not an enhanced or optional one

Loadbalancers: Not ALWAYS Broken



http://www.gossamer-threads.com/lists/nsp/ipv6/22157

ryanczak at arin

Mar 16, 2010, 5:15 AM

Post #1 of 12 (128 views)
Permalink

Latest Load Balancer IPv6 Support Remove Highlighting

I know this has been asked before but it has been quite some time so I thought it fair to bring this topic up again.

What are peoples experiences with **load** balancers and **IPv6**?

Does anyone know of a company whose **load balancer** product has feature parity between IPv4 and **IPv6**? Does anything out there support **IPv6** in GSLB configurations? What works and what does not?

--
Matt Ryanczak
ARIN Network Operations Manager

berni at birkenwald

Mar 16, 2010, 5:21 AM

Post #2 of 12 (131 views)
Permalink

Re: Latest Load Balancer IPv6 Support Remove Highlighting [In reply to]

On Tue, Mar 16, 2010 at 08:15:29AM -0400, Matt Ryanczak wrote:

Hi Matt,

> I know this has been asked before but it has been quite some time so I
> thought it fair to bring this topic up again.
>
> What are peoples experiences with **load** balancers and **IPv6**?

Running v6 on our F5 BigIP since around 2006, and having quite a few systems behind it. Mostly Webservers (http://www.lrz-muenchen.de). Other than some weird traceroute6 output when tracing through the box it works like a charm.

Bernhard

jeroen at unfix

Re: Latest Load Balancer IPv6 Support Remove Highlighting [In reply to]

Broadband Customer Premises Equipment (CPE)

- Lots of Broadband CPE does NOT support IPv6
- One list of products that does have at least some IPv6 support can be found at http://www.getipv6.info/index.php/Broadband_CPE
- Customer security software can also be a source of frustration…

Zone Alarm and IPv6

1.10 "My Network Provider Is Still Dragging Their Heels When It Comes to Providing IPv6 Connectivity"

IPv6 Transit Providers (e.g., NSPs)

- We'll talk about some other options later, but at least for now you should know that MANY major network service providers DO offer IPv6 transit; see the list at http://www.sixxs.net/faq/connectivity/?faq=ipv6transit
- That list includes most of the usual suspects, including:
  AS701 Verizon
  AS1239 Sprint
  AS2686 AT&T
  AS2914 NTT/Verio
  AS3356 Level3
  AS6939 Hurricane Electric
  plus many others…
- Higher education users should also know that Internet2 has long supported native IPv6 on its national backbone.

And Since We're Here In Pittsburgh…

https://www.3rox.net/index.php?option=com_content&view=article&id=23&Itemid=61

Google

Native IPv6 Connectivity

3ROX is ready for the future. Since the summer of 2005, 3ROX has been operating native IPv6 connectivity to the commodity Internet over its Global Crossing connection. For 3ROX customers, IPv6 is here now whether you're using the commodity Internet or high-performance research networks.

Wide Area Network Connections

Commodity Internet Connections	Available Bandwidth
Global Crossing	1 Gb/s
Sprint	1 Gb/s
Research Network Connections (Layer 3)	**Available Bandwidth**
National LambdaRail PacketNet	10 Gb/s
Internet2 IP Network	1 Gb/s (shared with the Internet2 Content Peering Service)
Teragrid Extensible Backplane Network	10 GE dedicated / 10 GE shared
Peering Connections	**Available Bandwidth**
ESnet	1 Gb/s
TransitRail	1 Gb/s
SOX	1 Gb/s
OARNet	1 Gb/s
Internet2 Content Peering Service	1 Gb/s (shared with the Internet2 IP network)
Dedicated Research Connections (Layer 2	**Available Bandwidth**

IPv6 Access Providers (e.g., ISPs)

- But what if you're an end user, and you just want connectivity via your ISP?
- There are IPv6 access providers offer native IPv6 service to end user customers. Examples of IPv6-enabled access providers can be found at: http://www.sixxs.net/faq/connectivity/?faq=native
- End users whose ISP doesn't offer native IPv6 may want to explore tunneled IPv6 connectivity from a tunnel broker; see: http://www.sixxs.net/tools/aiccu/brokers/
- Other users may only need IPv6 web hosting (and yes, it is available from a variety of providers)
- We're also starting to hear exciting news from some major broadband ISPs in the US…

Comcast IPv6 Trials

Comcast IPv6 Trials to Begin Soon

This site is intended to provide the latest information about Comcast's IPv6-related work. In a few weeks time we will begin conducting several IPv6 technical trials in our production network, with customers, in order to prepare for the IPv6 transition. This site will be updated as new information about these trials comes out, and as other IPv6-related work occurs.

IPv6 Trial News and Information:

Details for Trial #2: Native Dual-Stack for Cable Modem (DOCSIS) Customers
Thursday, April 22, 2010

This trial depends upon a Cable Modem Termination System (CMTS) being upgraded to support IPv6. Two of our CMTS vendors, Cisco and Arris, have software ready for IPv6 testing on their CMTS platforms.* As such, our initial trial areas will be restricted to selected markets with Arris or Cisco CMTSes. We have selected the San Francisco / Bay Area market, Chicago market, and Philadelphia market for our initial trial areas. That does not mean that all customers in these areas will be able to trial IPv6, however. In each market, we are working to select a small number of CMTSes that will be upgraded, based on where our trial volunteers are located. We are working with trial volunteers in these areas to gather additional information to assist in finalizing which specific customers can participate.

Once trial users are selected, they may be sent a new Arris DOCSIS 3.0 cable modem if their existing device

Other Comcast IPv6 Trials

- In addition to native dual stack, Comcast will also be doing other IPv6 trials, including tests of:

  -- 6RD (see RFC5569 and http://en.wikipedia.org/wiki/IPv6_rapid_deployment)

  -- Dual Stack Lite (see http://smakd.potaroo.net/ietf/idref/draft-ietf-softwire-dual-stack-lite/index.html)

Verizon IPv6 Trials

Verizon Begins Testing IPv6 on FiOS Services

Use of Emerging Protocol Will Enable the FiOS Network to Accommodate Long-Term Growth and Support Future Innovative Services

PR Newswire

April 06, 2010: 07:00 AM ET

BASKING RIDGE, N.J., April 6 /PRNewswire/ -- Verizon has begun testing on its all-fiber FiOS network a new Internet communications protocol that will enable the Internet to continue to expand and facilitate the future development of innovative services.

The new protocol – known as Internet Protocol version 6, or IPv6 – is designed to eventually replace the current Internet Protocol version 4 (IPv4), which over the next few years is expected to reach the maximum number of IP addresses it can accommodate, due to the rapid growth of the World Wide Web and IP-connected devices.

IPv6 expands the number of possible addresses from approximately 4 billion with IPv4 to roughly 340 trillion trillion trillion IPv6 addresses.

Because both IPv4 and IPv6 will be in use during the expected lengthy transition period, network-connected equipment and network operating systems must be able to handle both protocols.

"FiOS is a key service that can take advantage of IPv6," said Jean McManus, executive director – packet network technology for Verizon. "We've been working on an IPv6 transition plan for FiOS along with our other residential and enterprise services, and this work involves testing network equipment and making necessary customer premises equipment changes to ensure interoperability and proper operation of equipment. The FiOS trial is a key step toward enabling IPv6 in our core network, on edge routers and on CPE."

Verizon's month-long trial of IPv6 involves FiOS-enabled customer homes with customized CPE – provided by Verizon – that can support both IPv6 and IPv4. The dual protocol setup will also be implemented on Verizon's edge gateway routers. Verizon employs 6PE technology, which uses IPv6-provider edge routers to connect across the company's IPv4 MPLS core. The IPv6 traffic is then sent over IPv6-capable peering connections.

1.11 "I'm Too Busy Working on More Important Things"

Track Your Time For a Week; Make a Prioritized To-Do List

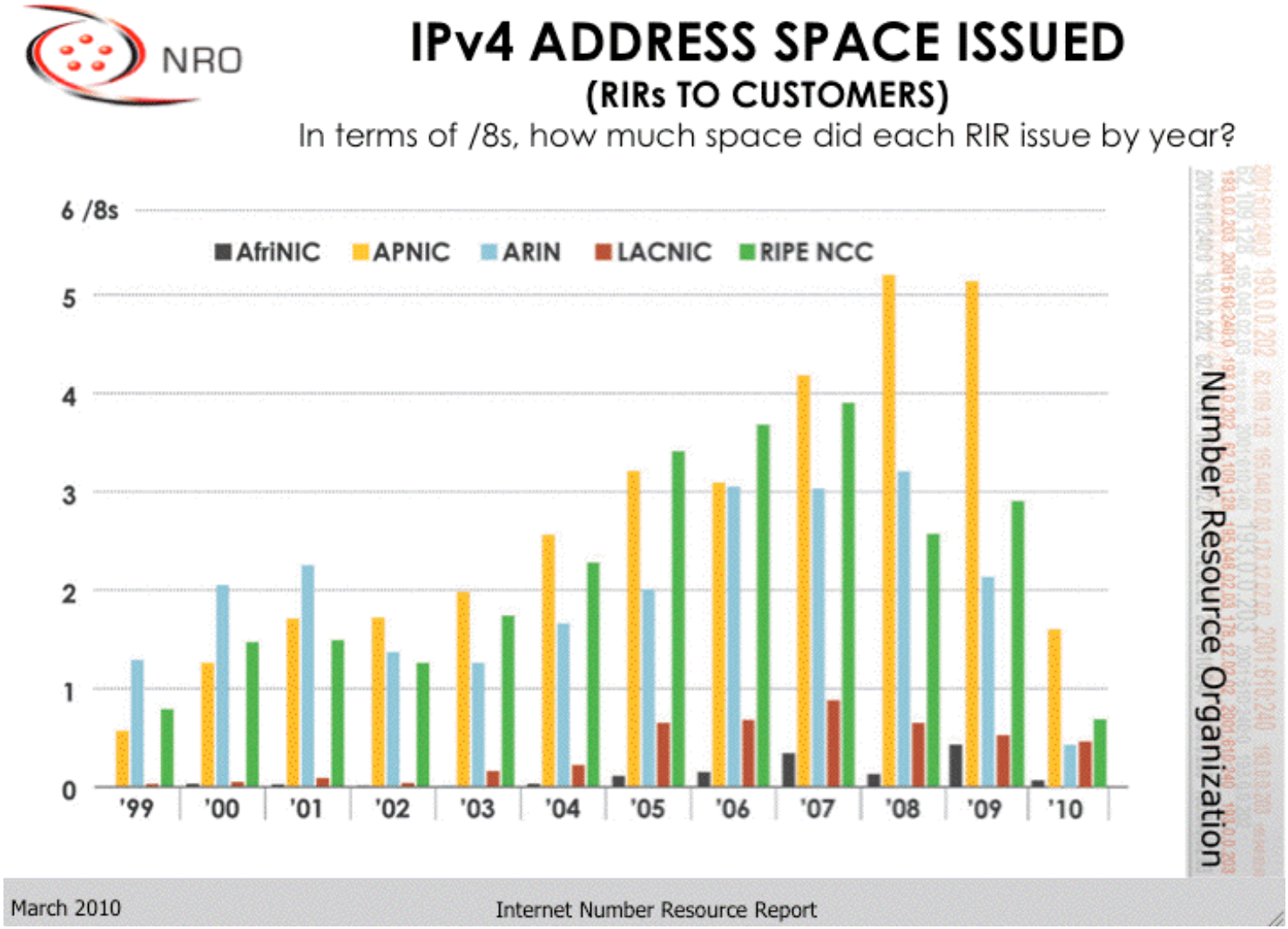
- Many people have no idea how they spend their time, nor do they have any sort of formal prioritized to-do list with deadlines -- stuff just gets handled, and the wheel that currently squeaking the loudest gets greased.
- Try tracking your time 24/7 for a week. Where does it all go? How much of it is spent attending meetings? Conference calls? Paperwork? ***Anything you could skip?***
- What would happen if you got sick or went on vacation? ***Could you pretend you were sick and then work on IPv6?***
- What's on your prioritized to-do list ahead of doing IPv6? ***When do the higher priority items need to be done?*** If you've actually got some slack time, maybe you can use some of that time to do IPv6 stuff.

You Need To Take Control Of Your Time

- I would assert that if you can't take control of your schedule at least to the point where you can make some time to do something really important (like getting radiating chest pains diagnosed, :-), or getting IPv6 installed), you may have bigger problems than just IPv4 exhaustion.
- Try reducing your contact channels: you don't need to have a desk phone and email and an instant messenger client and your cell phone hitting you with calls and text messages plus <fill in the blank> all strobing interrupts at you around the clock.
- Cut back. Stop using some of those channels, or at least turn some stuff off for part of the day. (And for that matter, spend some time with your spouse/kids/friends!)

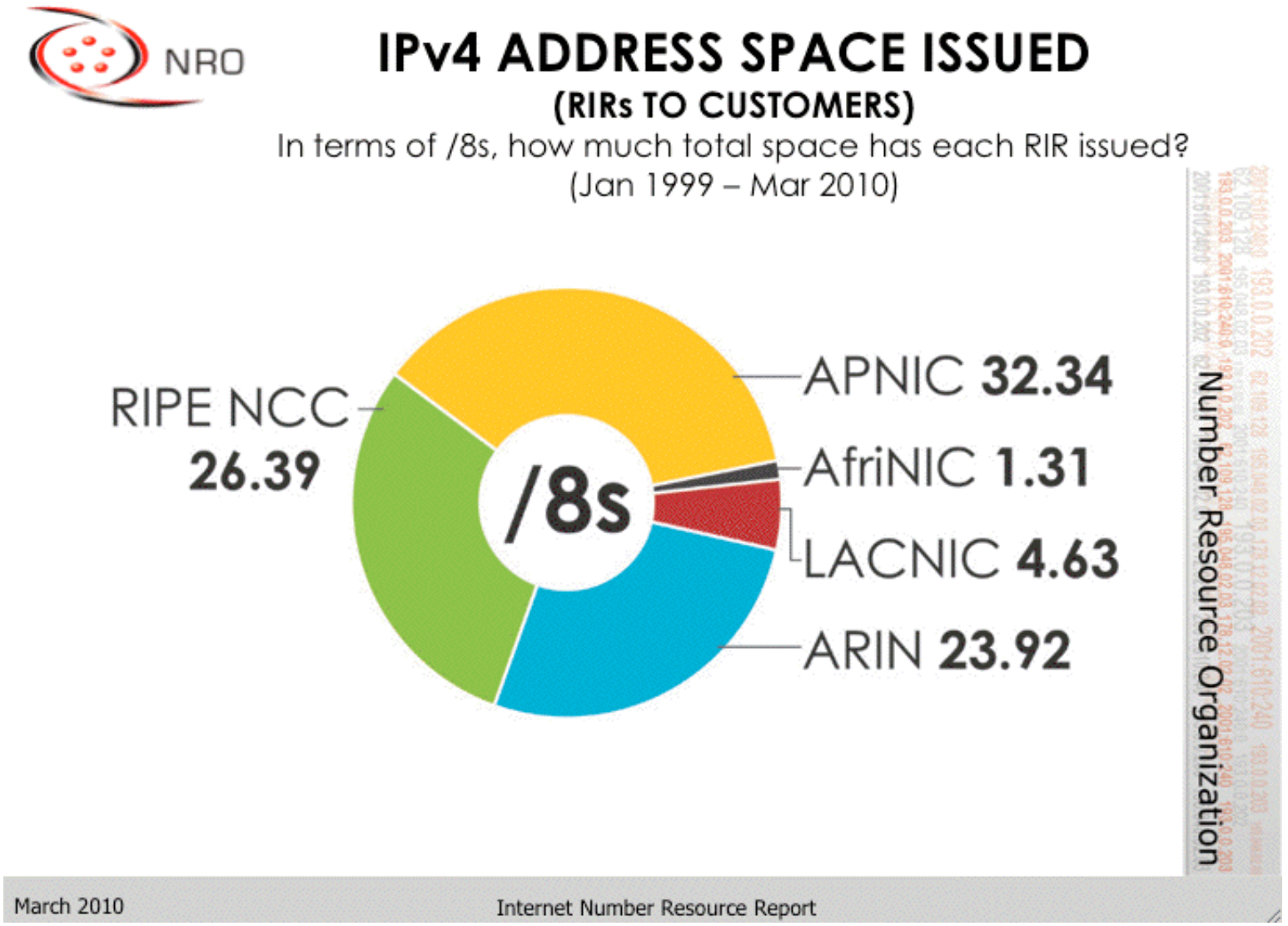
1.12 "This Whole IPv4 Exhaustion Thing Is A Bunch of Malarky -- Folks Will Figure Out Some Way To Stretch Out What IPv4 Space We've Still Got Available."

We're Not The (Only) Ones Driving The Address Consumption Bus!



http://www.arin.net/participate/meetings/reports/ARIN_XXV/PDF/Monday/Nobile_NRO_joint_stats.pdf

A Cumulative View



http://www.arin.net/participate/meetings/reports/ARIN_XXV/PDF/Monday/Nobile_NRO_joint_stats.pdf

What If IPv4 Address Usage Was Proportionate to Regional Population?

	Population	%	/8's	%	Ratio
• Asia:	4,121,097	60.3%	32.34	36.5%	0.605
• Africa:	1,009,893	14.7%	1.31	1.4%	0.095
• Europe:	732,206	10.7%	26.39	29.7%	**2.775**
• L. Amer.:	582,418	8.5%	4.63	5.2%	0.611
• N. Amer.:	348,360	5.1%	23.92	27%	**5.29**
• Oceania:	35,387	0.5%			
• Total:	6,829,360		88.56		

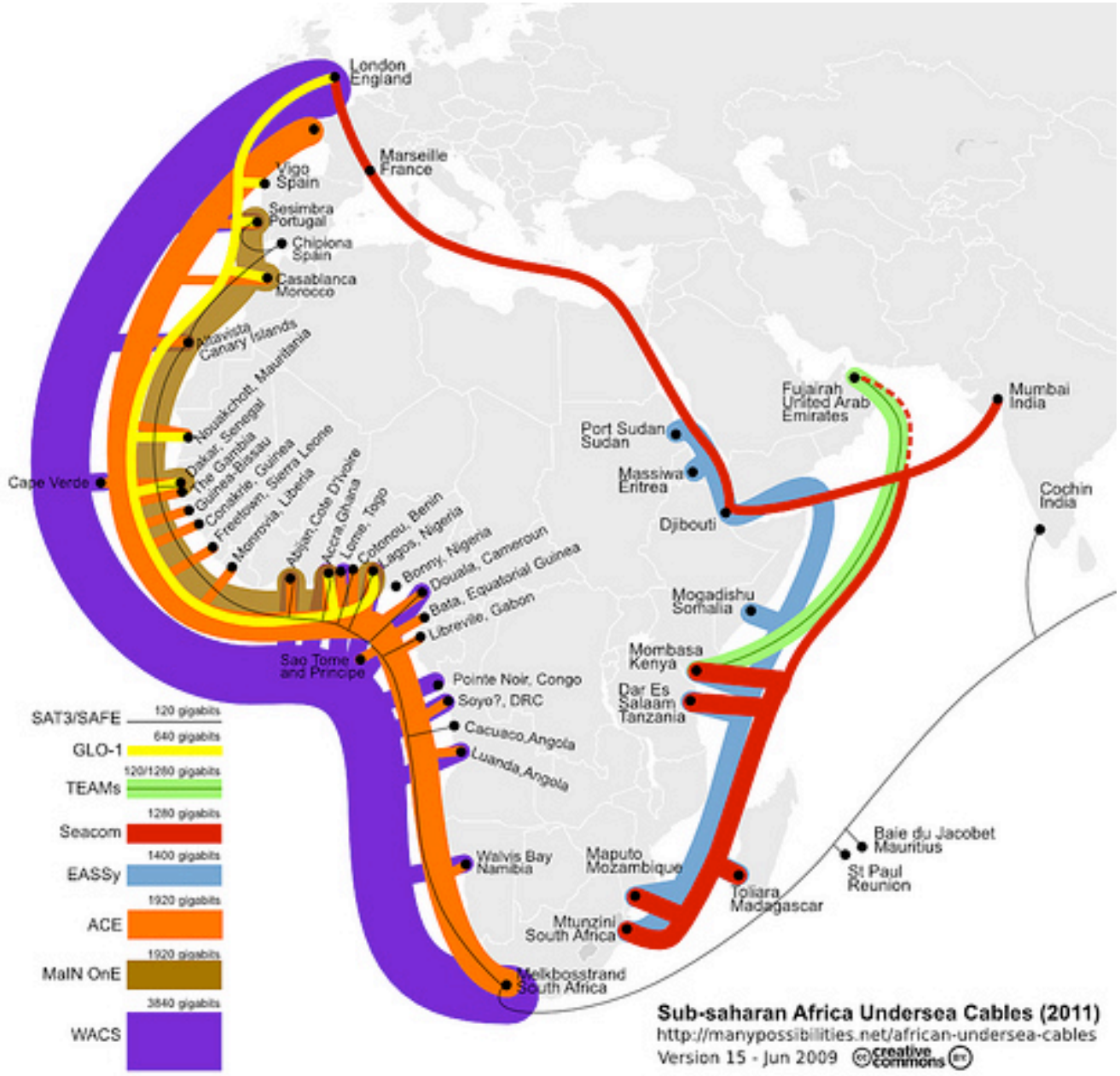
Population in thousands, mid year 2009 estimates

Note: Oceania's addresses are handled by APNIC (e.g., Asia)

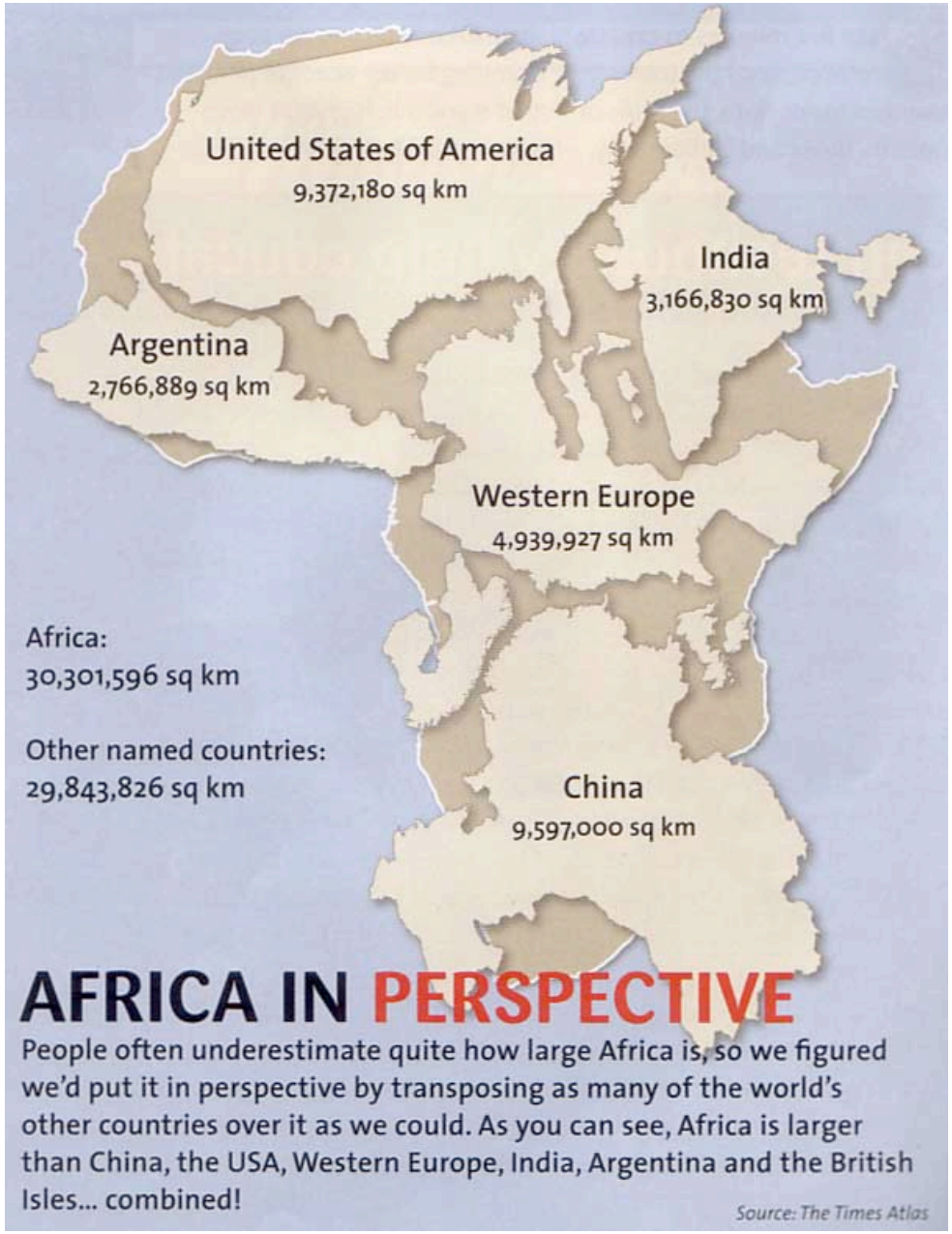
http://esa.un.org/unpd/wpp2008/jpg/WPP2008_Wall-Chart_Page_1.jpg

The Future: Notice Relatively "Tiny" Africa

- Historically, Africa's IP address usage to date has been minimal, less than one and a half /8s.
- This was likely due to a variety of factors, but at least one important factor was the high cost of connectivity (thousands of dollars per Mbps per month vs. just dollars per Mbps per month in the US (for large customers)).
- Another driver was widespread use of satellite Internet connectivity, with high latency, NAT'd connections and IPs provided by the satellite operator.
- Improved fiber connectivity is changing all that. Some of the world's largest and most densely populated regions in both Africa and Asia are now coming online, and I believe the improved connectivity to those areas will result in a substantial demand for new IPv4 addresses.



http://blog.foreignpolicy.com/files/images/090618_africa_underseas_cables.jpg



http://strangemaps.files.wordpress.com/2006/11/africa_in_perspective_map.jpg

If You Believe We Have Enough IPv4 Addresses

- Given the preceding slides, you must also believe in miracles! :-)
- In my case, I'd rather be prepared for IPv6 :-)

1.13 "IPv6 and Those Super Long Addresses Are Just Too Weird/Hard."

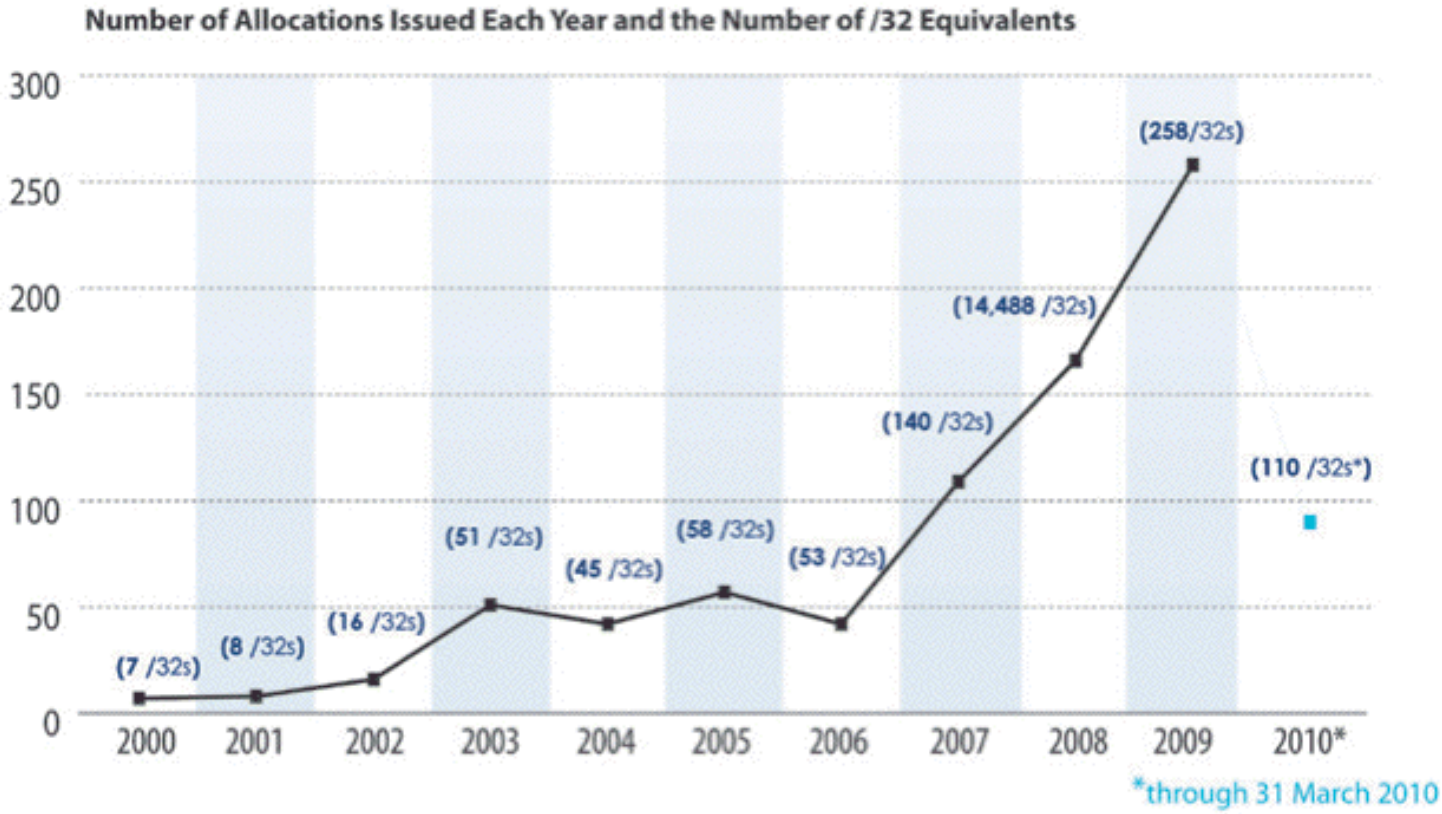
We'll See What You Think After This Workshop

- I'll let you make up your own mind about this particular point.

1.14 "Customers Just Aren't Asking For IPv6 (Except You!)"

That's Not The Impression I'm Getting…

IPv6 Allocations Issued (/32s)



Source: https://www.arin.net/participate/meetings/reports/ARIN_XXV/PDF/Wednesday/Nobile_RSD.pdf

An Anecdote from NANOG

- > What I heard at a recent (within the past six months)
  > conference was that "there is no customer demand for v6" so it
  > isn't on the immediate needs list. He said they had a lot of
  > inquiries about v6, but to date not having native v6 wasn't a
  > deal breaker with anyone

  I watched a vendor at one conference tell 20 people in a row that each one of them was the only one asking for IPv6. I mentioned to him that he should have his short-term memory loss checked out by a physician. At first he was confused. When I pointed out what I had just seen him do, he went from confused to embarrassed and admitted that it was the party line from his marketing department and they knew IPv6 was important, but, didn't have a story to tell yet, so, they were trying to spin for delay.

  http://seclists.org/nanog/2010/Apr/71

But The Perception Persists…

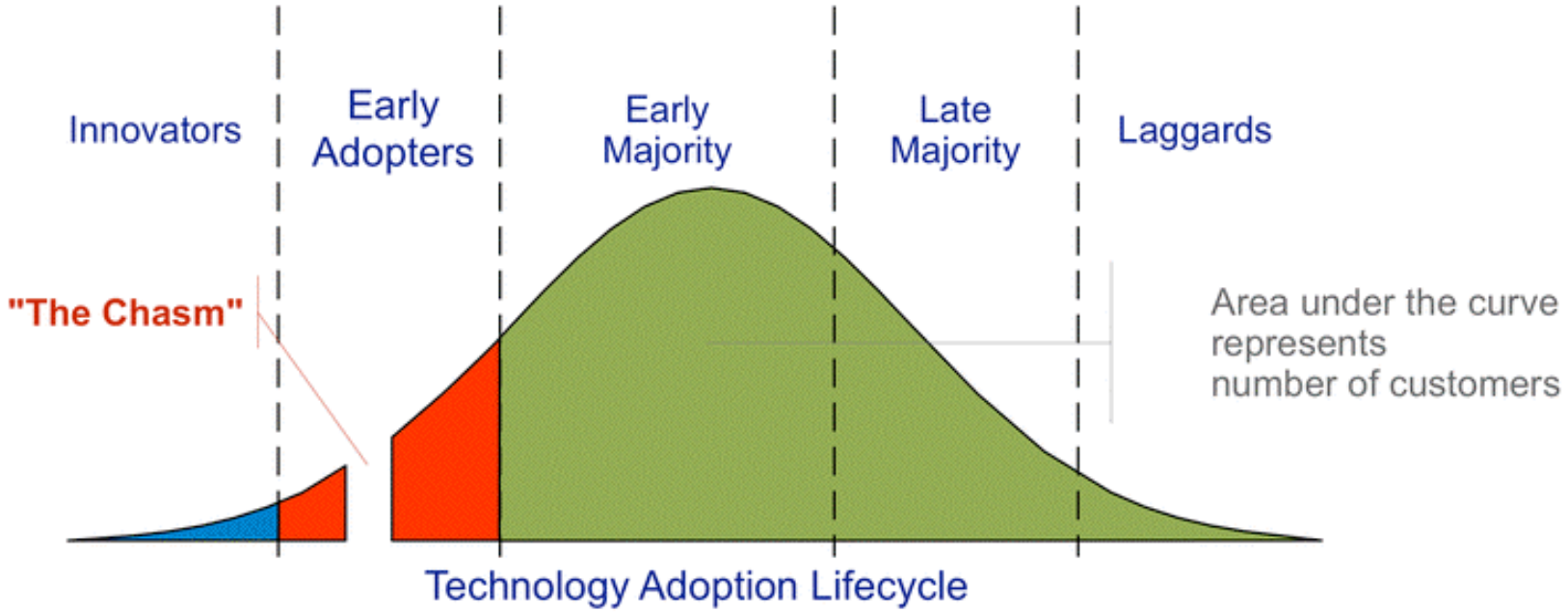
- Nobody actually cares*
  - Very little market demand
  - A "checkbox" feature
  - Few real endpoints

* Except Asia, US Government, Internet2

www.metasploit.com/data/confs/sector2008/exploiting_ipv6.pdf

The Question That's Really Being Asked Is…

- "What's the business case?"
- Vendors who ask that question obviously fail to understand the technology adoption lifecycle:



http://commons.wikimedia.org/wiki/File:Technology-Adoption-Lifecycle.png

We're Crossing "The Chasm" Right Now

- The companies that **are** ready to support IPv6 will be well positioned to support emerging customer needs and to thrive in an IPv6 environment

- The companies that **are not** ready will find their customers doing what they need to do (nothing personal, it's just business, and since "no one but me" cares about IPv6, I'm sure you won't mind that I'm taking my business elsewhere, see ya…)

A Brief Exercise

- Form groups of four to six people. Half of each group will initially advocate deployment of IPv6, the other half will initially argue against deployment of IPv6.
- After arguments have been aired, group participants may select either of the two positions, according to how they feel after listening to arguments both ways.
- Hypothetical factors that may potentially impact your decision with regard to IPv6:
  1) Costs associated with implementing IPv6, either now or on a crash basis later, will come out of **your** budget
  2) If an IPv6 project proceeds, **you** will be in charge of it
  3) Your agency's network continues to expand rapidly
  4) Your agency has zero tolerance for downtime or negative publicity

Part II. Basics of The Technology

2.1 IPv6 Addresses

Starting With What We Know: IPv4

- IPv4 addresses are 32 bits long
- 2^32=4,294,967,296
- Normally represented in "dotted decimal" format:
  -- four 8 bit octets (0 to 255 decimal)
  -- each octet is seperated from the next by a dot
  -- leading zeroes in each octet may be omitted
- Examples:
  -- 127.0.0.1
  -- 128.223.142.89
  -- 64.170.98.32

Something A Little Different: IPv6

- IPv4 addresses are 128 bits / 32 hexadecimal digits long
- 2^128=340,282,366,920,938,463,463,374,607,431,768,211,456 (e.g., 3.4x10^38 addresses)
- Normally represented in "colon separated" format:
  -- eight sets of four hex digits (0000 to FFFF hex)
  -- chunks are separated with colons (:)
  -- leading zeroes in each chunk may be omitted
  -- for convenience, :: (two successive colons) may replace one or more all-zero chunks, but only once in any addr
- Examples:
  2001:48a8:6880:0095:0000:0000:0000:0021
  ::1
  2001:468:d01:d6::80df:d617
  fe80::203:93ff:fecf:b638

Quick Quiz

- Structurally/superficially valid or invalid?
  If invalid, why?

  A) 2001:468:0d01:003c:0000:0000:80df:3c15
  B) 2001:468:0d01:003c::80df:3c15
  C) 2001:468:d01:3c::80df:3c15
  D) 2001:760:2e01:1::dead:beef
  E) 2001:480:10:1048:a00:20ff:fe9a:58c1:80
  F) 2001:500::4:13::80
  G) 2001:13G7:7002:4000::10
  H) 2607:f278:4101:11:209:5bff:fe8f:6609
  I) fe80::209:3dff:fe13:fcf7
  J) ::

Quick Quiz Answers

- Structurally/superficially valid or invalid?
  If invalid, why?

  A) Valid
  B) Valid
  C) Valid
  D) Valid
  E) Invalid (nine chunks instead of eight)
  F) Invalid (double colons appear more than once)
  G) Invalid (G is not a valid hexadecimal digit)
  H) Valid
  I) Valid
  J) Valid (albeit as the IPv6 "unspecified address")

2.2 IPv6 Prefixes

Starting With Something We Know: IPv4 Prefixes

- We originally had class A, class B and class C addresses:

Name	CIDR equiv	Number	Addresses Per Block
Class A	/8	128	16,777,216
Class B	/16	16,384	65,536
Class C	/24	2,097,152	256

- Like goldilocks, some of those were too large, and some of those were too small. We needed something a little more flexible, and that's what we got from CIDR, or Classless Inter-Domain Routing.

Common IPv4 CIDR Prefix Lengths

- /8 ==> 16,777,216 addresses
  /9 ==> 8,388,608
  /10 ==> 4,194,304
  /11 ==> 2,097,152
  /12 ==> 1,048,576
  /13 ==> 524,288
  /14 ==> 262,144
  /15 ==> 131,072
  /16 ==> 65,536
  /17 ==> 32,768
  /18 ==> 16,384
  /19 ==> 8,192
  /20 ==> 4,096
  /21 ==> 2,048
  /22 ==> 1,024
  /23 ==> 512
  /24 ==> 256
  /25 ==> 128
  /26 ==> 64
  /27 ==> 32
  /28 ==> 16
  /29 ==> 8
  /30 ==> 4
  /31 ==> 2
  /32 ==> 1

- It's common for IPv4 subnets to be /24's (or maybe /23's or /25's)

IPv6 Uses Prefixes, Too

- The prefixes are just usually a "little" bigger…

Prefix	IPv6 Addresses
/32	2^(128-32) = 79,228,162,514,264,337,593,543,950,336 (4,294,967,296 /64 subnets)
/48	2^(128-48) = 1,208,925,819,614,629,174,706,176 (65,536 /64 subnets)
/56	4,722,366,482,869,645,213,696 (256 /64 subnets)
/64	18,446,744,073,709,551,616 IPv6 addresses

General Rules for IPv6 Allocations/Assignments/Subneting

- Local Internet Registries (LIRs), e.g., ISPs, statewide networks, etc., will get one (or more) IPv6 /32 from ARIN, RIPE, APNIC, etc.
- Large sites will get an IPv6 /48 from their LIR's /32
- Small sites needing only a few subnets over 5 years will get an IPv6 /56 from their LIR's /32
- If one and only one subnet is needed, that entity gets an IPv6 /64
- Hosts get one or more IPv6 /128s out of a /64

- ALL SUBNETS at most sites will normally be /64s (even if they have only a small handful of hosts. Do NOT try to get clever and do something exotic when it comes to subnetting.

- [BTW, don't the /32, /48, /56 cut points feel a lot like the old IPv4 classful address days? They sure do to me...]

2.3 Types of IPv6 Addresses

Types of IPv6 Addresses

- **Global Unicast:** **2000::/3**
- **Link Local Unicast:** **FE80::/1**
- **Loopback:** **::1/128**
- 6to4: 2002::/16
- Teredo: 2001:0000::/32
- Unique Local Unicast: FC00::/7
- Multicast: FF00::/8
- IPv4-Mapped: ::ffff:128.223.214.23
- Deprecated: Site Local addrs and IPv4-Compatible addrs.
- For more on IPv6 addresses, see RFC4291 (+ other RFCs).
- For the most part, we primarily care about global unicast, link local unicast, and loopback addresses

Address Type Discussion

- **Global Unicast** addresses are globally unique, "real" IP addresses. These are the way you'll normally refer to most IPv6 hosts
- **Link Local Unicast** addresses are used for some purposes local to a particular link; outside the extent of that link, they don't get used.
- **Loopback** addresses -- just like 127.0.0.1 in IPv4 space, the IPv6 loopback address (::1/128) is an internal virtual address that the server can use to refer to itself.
- **6to4 and Teredo** addresses are special IPv4-to-IPv6 transition mode technologies. We'll talk about them later.
- **Unique Local Unicast** addresses (RFC4193) are the IPv6 equivalent of RFC1918 IPv4 addresses. Don't use them.
- **Multicast** addresses are just like multicast in IPv4, except that in IPv6 they're used extensively on the LAN, but IPv6 multicast traffic rarely appear on the wide area Internet, unlike IPv4 where the exact opposite is largely true.
- **IPv4-Mapped** addresses allow hosts that only bind IPv6 sockets to also accept IPv4 addresses.

IPv4-mapped Addresses in *BSD

- "One complicating factor for Apache administrators is whether or not an IPv6 socket can handle both IPv4 connections and IPv6 connections. Handling IPv4 connections with an IPv6 socket uses IPv4-mapped IPv6 addresses, which are allowed by default on most platforms but are disallowed by default on FreeBSD, NetBSD, and OpenBSD in order to match the system-wide policy on those platforms."
  http://httpd.apache.org/docs/2.0/bind.html
- "In FreeBSD, enabling IPv4-mapped addresses is done by adding ipv6_ipv4mapping="YES" to /etc/rc.conf (in addition to ipv6_enable="YES")."
  www.washington.edu/imap/documentation/IPv6.txt.html
- NEVER SEEN ON THE WIRE!

Anycast

- IPv6 also uses "anycast" addresses. Anycast addresses look just like regular global unicast addresses, however the same prefix gets advertised from multiple locations.

- Through the magic of routing, users automatically connect to the closest anycast instance when accessing a UDP service (such as DNS) hosted on an anycast prefix.

- If one anycast instance goes down, routes adjust, and the other anycast instances seamlessly assume the additional load.

2.4 IPv6 Scoping

Scoping Is A Corner Case

- Quoting "IPv6 Addresses," http://msdn.microsoft.com/en-us/library/aa921042.aspx

  "The scope ID identifies a specific area within the reachability scope for non-global addresses. A node identifies each area of the same scope with a unique scope ID."

- Scoped IPv6 addresses are rarely used by anyone except Microsoft, and generally you should ignore this concept. For justification, see the discussion of scoping at section 2.1 of "Deprecating Site Local Addresses," http://www.rfc-editor.org/rfc/rfc3879.txt

2.4 Addresses and Systems

Expect Multiple Addresses

- When you look at the interfaces on an IPv6-enabled system, it will be routine for it to have multiple addresses/interface.
- Sometimes this will just be a single globally unique unicast address, plus a link local address, plus some other bits and pieces, other times you may see multiple globally unique unicast addresses.
- Do not let this shake you up.

- Speaking of multiple addresses, most hosts will have both IPv6 AND IPv4 addresses on some interfaces. This is known as being "dual stacked" and is perfectly normal and acceptable.

Looking at Addresses on Interfaces on Linux

```
% ifconfig -a
eth0      Link encap:Ethernet  HWaddr 00:09:3D:13:FC:F7
          inet addr:128.223.142.32  Bcast:128.223.143.255  Mask:255.255.254.0
          inet6 addr: 2001:468:d01:8e:209:3dff:fe13:fcf7/64 Scope:Global
          inet6 addr: fe80::209:3dff:fe13:fcf7/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1187468996 errors:0 dropped:1805 overruns:0 frame:0
          TX packets:1338373204 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:232065679216 (216.1 GiB)  TX bytes:915094219311 (852.2 GiB)
          Interrupt:185
[snip]
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:8143461 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8143461 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:4295055907 (4.0 GiB)  TX bytes:4295055907 (4.0 GiB)
```

Looking at Addresses on Interfaces on Mac OS X

```
% ifconfig -a
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
        inet6 ::1 prefixlen 128
        inet6 fe80::1 prefixlen 64 scopeid 0x1
        inet 127.0.0.1 netmask 0xff000000
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        inet 128.223.214.23 netmask 0xfffffe00 broadcast 128.223.215.255
        inet6 fe80::203:93ff:fecf:b638 prefixlen 64 scopeid 0x4
        inet6 2001:468:d01:d6::80df:d617 prefixlen 64
        ether 00:03:93:cf:b6:38
        media: autoselect (1000baseTX <full-duplex>) status: active
        supported media: none autoselect 10baseT/UTP <half-duplex> 10baseT/UTP
      <full-duplex> 10baseT/UTP <full-duplex,hw-loopback> 100baseTX <half-duplex>
      100baseTX <full-duplex> 100baseTX <full-duplex,hw-loopback> 1000baseTX
      <full-duplex> 1000baseTX <full-duplex,hw-loopback> 1000baseTX <full-
      duplex,flow-control> 1000baseTX <full-duplex,flow-control,hw-loopback>
[etc]
```

How Addresses Get On Interfaces: IPv4

- Starting with something you know, IPv4, most **end-user workstations** get their IP addresses automatically assigned via DHCP.
- Besides IPv4 addrs, RFC2132 describes additional useful bits that a host can get from a DHCP server, including:
  -- subnet mask
  -- router address(es)
  -- time server(s)
  -- domain name server(s)
  -- the host name
  -- the domain name
  -- MTU information
  -- broadcast address
  -- plus an amazing amount of other bits and pieces (check it out if you don't believe me!)

How Addresses Get On Interfaces: IPv4 (2)

- Some sites may use host-specific DHCP entries to configure servers, too, but most IPv4 **servers** tend to be manually configured by a system administrator with a statically assigned IP addresses, host name and domain information, name server information, broadcast address, MTU information, etc.

- Some large sites which manage hundreds or thousands of IPv4 servers may use automatic host configuration tools such as Puppet (see http://www.puppetlabs.com/) or cfengine (see http://www.cfengine.org/) to help automate and improve the scalability and accuracy of bulk server configuration (and to keep their server admins from going nutz from tedious repetition).

How Addresses Get On Interfaces: IPv6

- In IPv6 world, most user workstations will get IPv6 addresses from state-less address auto configuration, or SLAAC. SLAAC runs on a router (not on a separate DHCP server), and as you might expect from the name, the IPv6 addresses that one gets via SLAAC are not maintained in a table anywhere (no "state" gets created when an IPv6 address is assigned via SLAAC).

- So how does the router know that it won't accidentally give you the same address as someone else (e.g., assign a duplicate address) if it doesn't keep track of who it has given an address to? Answer: it derives the address it gives you from something only you have, namely the MAC (hardware ethernet) address of your NIC

IPv6 Modified EUI-64 Format Identifier

- So, we need to take the 48 bit MAC address from your NIC, and convert that into a 64 bit dynamically assigned address. What do we do?

  -- The left most 24 bits of the MAC form the left most 24 bits of the EUI-64 format identifier
  -- The right most 24 bits of the MAC form the right most 24 bits of the EUI-64 format identifier
  -- We cram the constant FFFE in the middle 16 bits
  -- We tweak bit 7 (counting from the left) from zero to one (this is the "universal/local" bit)

  The "front half" of the 128 bit address comes from the network (remember our rule that all subnets are /64s!)

"But Joe!"

- "The whole world will know my unique and unvarying hardware MAC address! Evil marketers will track and correlate my every move wherever I may connect my IPv6 device based on my MAC address! This is worse than cookies!"

- True (especially the evil marketers bit).

- This concern spawned another type of IPv6 address, so-called RFC3041 Privacy Addresses. These addresses effectively use a random address for the low order 64 bits of the IPv6 address, instead of a value derived from the host's MAC address. We may also change those addresses from time to time.

"BUT JOE!!!!!"

- "That's NOT what I wanted! If we give users random network addresses, how will we be able to track down abusers??? The same user may have one IPv6 address now, and another completely different IPv6 address later, and I don't see how we'd keep track of who's got what address when!!! What a pain!"

- One bit of potential happiness: NDPmon (think of this as "arpwatch for IPv6"). See ndpmon.sourceforge.net

- Another potential option: control network access just as you currently control network access for wireless networks (e.g., use 802.1x or a homegrown authentication solution)

Selecting/Deselecting Privacy Addresses

- **Windows:** privacy addresses are **enabled** by default when IPv6 is enabled on Windows XP. To disable them, see the next slide.
- **Macs:** privacy addresses are **disabled** by default.
  To enable them:
  # sysctl net.inet6.ip6.use_tempaddr=1
- **Linux:** like Macs, privacy addresses are disabled by default. To enable them:
  # sysctl net.ip6.conf.all.use_tempaddr=2
  # sysctl net.ip6.conf.default.use_tempaddr=2
- Periodically recheck your assigned addresses if this is a big deal for you, and remember, this is NOT life-and-death privacy, it is just "something-to-make-life-hard(er)-for-intrusive-marketers"-grade privacy.

Disabling IPv6 Privacy Addresses

Disabling privacy addresses

- Windows XP

```
ipv6 -p gpu UseTemporaryAddresses no
```

- Windows 2003

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

- Windows Vista

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

- Windows 2008

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


http://www.internet2.edu/presentations/jt2010feb/20100202-broersma.pdf

An Aside: Trying to Anonymize IPv6 Flow Data

- Speaking of IPv6 and privacy…
- Internet2 routinely makes anonymized Netflow data available for researcher use (see, for example
  ·"A Look at the Unidentified Half of Netflow (With an Additional Tutorial On How to Use the Internet2 Netflow Data Archives),"
  www.uoregon.edu/~joe/missing-half/missing-half.pdf
- In the IPv4 case, the anonymization process is easy: the low order 11 bits of each IPv4 address are zeroed.
- The IPv6 case is significantly more complex. For a copy of the draft policy Internet2 finally approved, see http://www.uoregon.edu/~joe/ipv6-mask.pdf

A Stateful Alternative to SLAAC: DHCPv6

- An alternative to SLAAC for workstations is DHCPv6, much like DHCP for IPv4.
- One critical difference: while DHCPv6 is well supported by IPv6ified versions of Microsoft Windows, **at least some important vendors (read that as "Apple") do not support DHCPv6 at this time (and may never do so).**
- If you're using a Mac OS X box and you want or need to do DHCPv6, you will need to run a third party DHCPv6 client (Dibbler is a commonly suggested option, see http://klub.com.pl/dhcpv6/ , but note the comment there that "Due to work on my Ph.D, the Dibbler project is in the maintenance mode. Active development and non-critical bug fixing is on hold, until I finish my dissertation. Sorry.")

Another DHCPv6 "Oddity:" "M vs. O" Flags

- DHCPv6 can be used in two different modes, often referred to as "M" and "O" modes:

  M :

  1-bit "Managed address configuration" flag. When set, it indicates that Dynamic Host Configuration Protocol [DHCPv6] is available for address configuration in addition to any addresses autoconfigured using stateless address autoconfiguration. The use of this flag is further described in [omited].

  O :

  1-bit "Other configuration" flag. When set, it indicates that [DHCPv6lite] is available for autoconfiguration of other (non-address) information. Examples of such information are DNS-related information or information on other servers within the network.

  See http://tools.ietf.org/id/draft-ietf-ipv6-ra-mo-flags-01.txt [127]

An Aside: "M" vs "O" in Cisco IOS

While most of you will not be configuring routers to do DHCPv6, just in case:

```
Router(config-if)#ipv6 nd ?
dad                  Duplicat Address Detection
managed-config-flag  Hosts should use DHCP for address config
ns-interval          Set advertised NS retransmission interval
other-config-flag    Hosts should use DHCP for non-address config
prefix               Configure IPv6 Routing Prefix Advertisement
ra-interval          Set IPv6 Router Advertisement Interval
ra-lifetime          Set IPv6 Router Advertisement Lifetime
reachable-time       Set advertised reachability time
suppress-ra          Suppress IPv6 Router Advertisements
```

What If We're Not Doing DHCPv6 "O" Mode?

- If you're running a Mac, or the local IPv6 network isn't doing "O" mode DHCPv6, you may wonder how you learn things like the DNS servers you should be using.
- The default answer (a bit of a disappointment, but *c'est la vie*) is that often users will rely on IPv6-aware name servers which have IPv4-transport, typically name servers learned from IPv4 DHCP.
- The alternative answer is that users may need to manually configure one or more IPv6-aware name servers. In the absence of more specific guidance, you may want to use Google's intentionally open name servers at 8.8.8.8 and 8.8.4.4 -- they're IPv6-aware.

Manually Assigned Native IPv6 Addresses

- Just like in IPv4, you also have the option of manually assigning native IPv6 addresses for things like servers. Recipes for some common OS's:

  -- FreeBSD and Friends:
  http://www.cyberciti.biz/faq/freebsd-configure-ipv6-networking-static-ip-address/
  -- Redhat/CentOS:
  http://www.cyberciti.biz/faq/rhel-redhat-fedora-centos-ipv6-network-configuration/
  -- SuSE Linux:
  http://www.cyberciti.biz/faq/configuring-ipv6-in-sles10-opensuse-linux/
  -- Ubuntu Linux:
  http://www.cyberciti.biz/faq/ubuntu-ipv6-networking-configuration/
  -- Windows Server 2008/R2
  http://technet.microsoft.com/en-us/library/cc732106.aspx

  Mac user? Just set a static IP in System Preferences -> Network -> Configure -> Configure IPv6 -> Manually

"What's My Subnet Length and Router Addr?"

- When statically configuring…
- Unless you're told otherwise, as a general rule of thumb, the subnet length will always be /64
- Unless you're told otherwise, again as a rule of thumb, the router address will always have the same first 64 bits as your host's static address, followed by ::1
- Don't forget to also define IPv6-aware name servers. If you don't have a suitable local alternative, Google's intentionally open name servers, 8.8.8.8 and 8.8.4.4 will usually work fine as long as you have IPv6 AND IPv4 connectivity to your host.
- Q. "In IPv4 I usually configure a broadcast address. What's my broadcast address for IPv6?"
  A. Broadcast isn't needed and doesn't exist in IPv6.

2.5 IPv6 Internet Connectivity

Native IPv6 Connectivity

- Native IPv6 connectivity is preferred.

- Native IPv6 connectivity is the IPv6 analog of normal IPv4 connectivity, and would ideally come from your current network service provider.

- UO, for example, has native IPv6 connectivity from Internet2 via the Oregon Gigapop.

- If you have the bad luck to have a non-IPv6 aware service provider, see the pointers to IPv6 options at section 1.10 earlier in this talk.

IPv6 Peering

- Large providers doing IPv6 should also be aware that there may be opportunities for IPv6 peering (e.g., the exchange of customer traffic, and only customer traffic, over IPv6).

- A good presentation on this topic is available at:

  www.nanog.org/meetings/nanog43/presentations/Levy_IPv6_%20Peering_N43.pdf

Manually Configured IPv6 Tunnels

- Another alternative is to arrange for a manually configured IPv6 tunnel from an IPv6 tunnel broker.
- Free tunneled IPv6 connectivity is available from a variety of providers, including most notably:
  -- Hurricane Electric, http://tunnelbroker.net/
  -- SixXS, https://www.sixxs.net/main/
- When establishing a manually configured IPv6 tunnel, beware of tunneling to a very distant tunnel endpoint -- all your traffic will have to make that long trip, and that will add (potentially substantial) latency. Keep tunnels as short as possible!
- Also beware of firewall issues. If you have a firewall between you and your tunnel provider, it will need to permit proto 41 ("6in4") traffic.

Manually Configured Tunnel Security

- Manually configured tunnels are subject to a number of security vulnerabilities, including endpoint spoofing/traffic injection (unless specific precautions, such as using IPSec are taken, which we've already established is normally not the case).

- For an interesting non-malicious discussion of tunnel endpoint spoofing, see http://www.dia.uniroma3.it/~compunet/tunneldiscovery/

- Remember, use native IPv6 whenever you possibly can, and always encrypt all your traffic end to end.

"Automatic" IPv6 Connectivity: 6to4

- Another alternative to native IPv6 connectivity or manually configured IPv6 tunnels is 6to4.
- 6to4 was meant as a temporary transition mechanism, to help people use IPv6 until they could get native IPv6 (or at least tunneled IPv6) deployed.
- Two issues with 6to4:
  -- it sends traffic to 192.88.99.1, part of the anycast block 192.88.99.0/24; you *will* use the "closest" 192.88.99.0/24 that's out there, whether friend or foe :-;
  -- 6to4 cannot traverse a firewall (including those ubiquitous little blue Linksys boxes); Mac Airport Express and Airport Extreme boxes reportedly DO know how to handle 6to4 correctly, however.

Enabling 6to4 on a Mac

- ***N.B.:*** 6to4 (RFC3056) usually **won't work** behind a firewall
- -- Apple Menu ==> System Preferences ==> Network ==>
  Show: Network Port Configuration
  -- If no 6 to 4 port already exists, click "New"
  -- Select 6 to 4 for the port from the pull down list of ports
  -- Enter "6 to 4" for the port's name
  -- Click OK
  -- Make sure "6 to 4" is checked as "On"
  -- Click "Apply Now"

  [the above details may vary on some versions of OS X]
- If you're using Firefox 2.x on a Mac, you may also need to tell Firefox to allow IPv6 DNS resolution to occur
  -- In Firefox go to the URL about:config
  -- Filter on the string IPv6
  -- Set network.dns.disableIPv6 to be false
  -- Try going to http://ipv6.google.com/ (the logo should dance)
- To disable 6to4, use System Preferences to set 6to4 to be "Off"

Teredo/Miredo

- Teredo is another automatic IPv6 tunneling protocol; this one differs from 6to4 in that it can successfully traverse network firewall boxes (unless the firewall blocks outgoing IPv4 traffic on 3544/UDP)

- Teredo ships with Microsoft Windows; if you're running Linux, you'll need to install Miredo for Teredo functionality. Miredo is available from http://www.remlab.net/miredo/

Enabling Teredo on a Windows XP SP2 PC

To set up IPv6 and Teredo on a Windows XP SP2 system, do:

Start ==> Accessories ==> Command Prompt

netsh interface ipv6 install

netsh interface ipv6 set teredo client

To disable it:
netsh interface ipv6 set teredo disabled
netsh interface ipv6 uninstall

The Special Case of IPv6 Unexpectedly Not Working Under Windows Vista

- Vista users should have IPv6 work "out of the box," although Firefox users will need to tweak their browser to enable IPv6 DNS
- **But what if you're a Vista users and IPv6 doesn't work even when you use the latest version of IE (or some web browser other than Firefox?)** You **may** be on a managed laptop where the system administrator has "intentionally broken" IPv6 (see http://support.microsoft.com/kb/929852).
- We can fix that, but in some cases it can involve editing the registry (yuck), while in other cases you may "just" need to re-enable use of IPv6 (Start ==> right click on Network, select Properties, click on Manage Network Connections. Right click on the wireless connection. Select Properties. Tick the IPv6 box.)
- Note: if you "unbreak" IPv6 on your official laptop, visit with your system admin after you return home to confess your sins. :-)

Key Point: Even If Your Site "Officially" Foregoes IPv6, Users May Still Decide to Try It…

- Some sites which rely heavily on firewalls and perimeter security may decide to forego or postpone deployment of native IPv6. Having made the decision to do so, folks may emit a big relieved sigh, believing that by "sitting this dance out," they will have foreclosed any possibility of user access to IPv6-only resources.
- Unless that policy is **very** carefully enforced on a technical basis, you may be in for a surprise or two because users may be able to easily work their way around your non-implementation or filters.
- This is particularly important if you're relying primarily on perimeter filtering to control either the **infiltration** of malware (or other unwanted content, e.g., "adult entertainment"), or the **exfiltration** of site-sensitive information (as at some federal sites).
- BTW, a very cool IPv6 web hack is sixxs.org's IPv6 web gateway: try www.cnn.com.sixxs.org (for example), from an IPv6-ified box

Your Users *Will* Be Fulfilled

- It is natural and entirely appropriate that your users will want to try new things, such as things they may hear about from their friends and colleagues. One of those things may be IPv6.
- If a technology they're interested in (such as IPv6) isn't one that you're currently supporting, they may search for and find "ad hoc" approaches which they can try without "having to bother you."
- Sometimes there's a hope that obscurity or technical difficulty will keep users from trying some work-arounds, but I wouldn't count on "security through obscurity" in the case of IPv6.
- For example, if a user is on a Mac at a "non-IPv6 site" and that site also doesn't have a perimeter or interior firewall, one option would be for him to enable "6to4." You've seen how easy that was…
- As another example, assume a user is behind a firewall and is using a PC running Windows XP at a "non-IPv6 site." You've also seen how easy that was…

Immense Technical Skill Is *Not* Required

- In my opinion, **pretty much any "reasonably motivated" semi technical user will be able to successfully enable 6to4 or Teredo on their desktop or laptop**, even if they don't fully understand the technology or the implications of having done so.
- And even if you block 6to4 and Teredo, users can still use RFC3053 IPv6 tunnel brokers (there's a nice list of them at http://en.wikipedia.org/wiki/List_of_IPv6_tunnel_brokers for example), and so on and so forth.
- On the other hand (and for interesting reasons), there is still no IPv6 version of Tor (the onion routing protocol) yet, see http://wiki.noreply.org/noreply/TheOnionRouter/TorFAQ at section 8.13
- Anyhow, rather than playing "IPv6 cat and mouse" with your users, why not just buckle down and run native IPv6 instead? Trying to fight transition mode IPv6 traffic will ultimately become really trickier and trickier over time, particularly if users encrypt.

6to4 & Teredo May Rely on "Remote Resources"

- In addition to things like 6to4 and Teredo traffic posing surprises for things like border filtering and traffic monitoring, tunneled traffic may also rely on comparatively **remote resources**.
- Depending on how far away some of those resources may be, the additional **latency** associated with reaching those gateways may impact the performance of untuned network connections.
- Remote resources may also be a sign that there's only a **limited pool** of available gateways (if the pool was large and well distributed, presumably you'd be using a nearby gateway rather than a remote one). When the pool of available resources is constrained, it may eventually get "**loved to death**" (overloaded).
- One could also imagine a site run by a cyber criminal, kindly offering free gateway services in an effort to attract your customer's traffic for surreptitious **MITM**-ish monitoring.
- Services such as 6to4 and Teredo which do not require any sort of registration or authentication may also end up being **abused** by bad guys just as **open SMTP relays** once were.

Magic Addresses

- 6to4 uses 192.88.99.1 as a magic address, anycast via the magic prefix 192.88.99.0/24 (see RFC3068 at 2.3 and 2.4)
- Do you know where **your** 192.88.99.1 traffic is going? (simple test: traceroute to 192.88.99.1 from a machine at your home site) [Maybe you want to *routinely* monitor the path to 192.88.99.1?]
- When I looked at some examples from public traceroute servers, (examples which I'll omit here), I've seen:
  -- large academic sites whose customers may end up using anycast 6to4 relays located clear across the country,
  -- government mission networks whose customers may rely on 6to4 anycast relays hosted on the campus of academic sites
  -- commercial providers whose customers may rely on anycast 6to4 relays hosted by some of their competitors.
- Or consider Teredo -- Teredo relies on Teredo servers and Teredo relays. Do you know which ones ***your*** folks may be using? http://technet.microsoft.com/en-us/library/cc722030.aspx mentions the Teredo ***server*** teredo.ipv6.microsoft.com

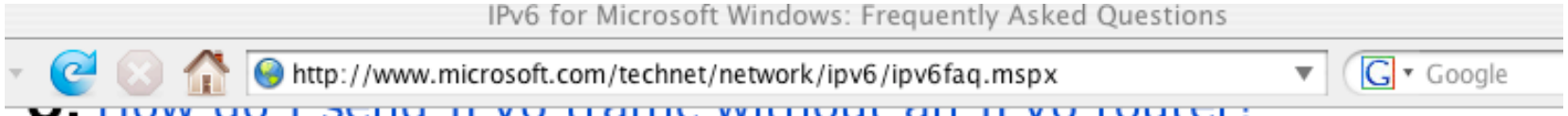
Where's UO's Closest 192.88.99.1?

```
% traceroute 192.88.99.1
traceroute to 192.88.99.1 (192.88.99.1), 30 hops max, 46 byte packets
 1  vl-142.uonet2-gw.uoregon.edu (128.223.142.3)  0.341 ms  0.231 ms  0.667 ms
 2  3.xe-1-3-0.uonet10-gw.uoregon.edu (128.223.3.10)  0.180 ms  0.159 ms  0.152 ms
 3  vl-105.ge-2-0-0.core0-gw.pdx.oregon-gigapop.net (198.32.165.89)  2.805 ms  2.682 ms  2.679 ms
 4  vl-101.abilene-losa-gw.oregon-gigapop.net (198.32.165.66)  24.511 ms  24.567 ms  24.548 ms
 5  xe-0-1-0.0.rtr.hous.net.internet2.edu (64.57.28.97)  56.585 ms  56.555 ms  56.525 ms
 6  xe-2-3-0.0.rtr.atla.net.internet2.edu (64.57.28.113)  130.461 ms  79.937 ms  79.949 ms
 7  xe-0-2-0.110.rtr.ll.indiana.gigapop.net (149.165.254.20)  95.113 ms  95.063 ms  95.068 ms
 8  xe-0-0-0.1.rtr.ictc.indiana.gigapop.net (149.165.254.25)  115.358 ms  95.126 ms  95.102 ms
 9  rtr3.ul.indiana.gigapop.net (149.165.255.129)  95.359 ms *  95.302 ms
```

Example From The Local Hotel's Network

```
%    traceroute 192.88.99.1
traceroute to 192.88.99.1 (192.88.99.1), 64 hops max, 40 byte packets
 1  83.7.210.65.in-addr.arpa.noptr.antlabs.com (65.210.7.83)  1.941 ms  1.034 ms  0.992 ms
 2  81.7.210.65.in-addr.arpa.noptr.antlabs.com (65.210.7.81)  1.935 ms  1.472 ms  3.285 ms
 3  * mfr103-500.gw3.pit1.alter.net (152.179.63.201)  5.896 ms *
 4  518.at-4-0-0.cl1.pit1.alter.net (152.63.36.250)  179.161 ms *  17.748 ms
 5  0.so-6-0-0.xl3.iad8.alter.net (152.63.0.138)  89.683 ms  37.278 ms  14.802 ms
 6  tengige0-6-1-0.gw1.iad8.alter.net (152.63.35.137)  167.029 ms  164.967 ms tengige0-6-0-
    0.gw1.iad8.alter.net (152.63.36.45)  42.621 ms
 7  teliasonera-gw.customer.alter.net (63.125.125.42)  91.004 ms  14.806 ms  14.778 ms
 8  ldn-bb1-link.telia.net (80.91.251.206)  187.366 ms 97.65.248.213.in-addr.arpa.noptr.antlabs.com
    (213.248.65.97)  94.529 ms ldn-bb2-link.telia.net (80.91.251.208)  98.843 ms
 9  ldn-b5-link.telia.net (80.91.250.168)  211.678 ms  91.400 ms  90.145 ms
10  * * *
11  * * *
12  * * *
```

But What About Teredo *Relays*, Where the Bandwidth Intensive "Heavy Lifting" Happens?



Q. Why can't I reach locations on the IPv6 Internet using the Teredo client in Windows?

A. To reach the IPv6 Internet from behind a network address translator (NAT) using the Teredo client included with Windows, there must be an operational Teredo relay attached to the IPv4 and IPv6 Internets. At this time, Microsoft is not providing any operational Teredo relays for reachability to locations on the IPv6 Internet for Windows-based Teredo clients.

Sites Which *Are* Advertising 2001:0::/32

- RFC4380 at 2.6 specifies 2001:0::/32 for the Teredo relay service. Martin Levy recently presented "IPv6 Traffic Levels on Hurricane Electric's Backbone," (see www.nanog.org/meetings/nanog45/presentations/Tuesday/Levy_traffic_level_hurricane_N45.pdf):

  > "**[Teredo] traffic is all eastward across the Atlantic**
  > Flows toward teredo.bit.nl AS12859 via AMS-IX
  > 2001::/32 announce by other networks including
  > AS12637 Seeweb, AS1257 Tele2, etc." [emphasis added]

- If you telnet to one of the IPv6 aware routeviews.org nodes (such as route-views.linx.routeviews.org), you can see sites advertising 2001:0::/32 by using the command "show ipv6 bgp 2001:0::/32"
- When I last checked, I saw 2001:0::/32 from AS1257 (Tele2), AS6939 (Hurricane), AS12637 (Seeweb), AS12859 (Bit.NL) and AS21155 (ProServe).
- If you are globally advertising 2001:0::/32, but for some reason your ASN isn't listed here, I'd love to hear from you.

'So Are You Telling Me That I Should Try To "Break" or "Disable" 6to4 and/or Teredo?'

- Encountering 6to4 or Teredo is like encountering extra-terrestrial intelligence. Squelch any immediate reptilian instinct to smash/kill/eat anything which is new/different/potentially threatening. :-)
- At the same time, let's avoid philosophically overanalyzing this. We should not let "the perfect" get in the way of the "adequate." While I **really** want to see native IPv6 deployed end-to-end, 6to4 or Teredo (at least as long as it works and isn't being abused), is better for many users than no IPv6 service at all.
- **Thus, notwithstanding some of the issues mentioned on previous slides, please refrain from breaking 6to4 or Teredo.**
- **You should consider fielding a carefully monitored version of those services, accessible only by your local users, thereby soaking up the local demand for those services (and if you do see folks using 'em, nudge them toward native IPv6 instead)**

Historical Trivia: What Was the "6Bone?"

- The 6bone was an IPv6 testbed using configured tunnels.

- 6bone IPv6 addresses began with 3FFE (sadly, you may still see some people trying to use 3FFE addresses today, even though the 6bone officially was decommissioned in June 2006)

- For historical information about this experiment, see http://go6.net/ipv6-6bone/

What About ISATAP?

- ISATAP is yet another IPv6 transition mechanism, this one defined in RFC5214 (seems like they're millions of transition mechanisms, doesn't it?)
- It violates fundamental network layering principles.
- ISATAP also relies on the presence of a magic isatap.<domain> domain name. You *are* making sure that no one else is registered that magic name, right?
  Just as you're making sure no one except your authorized proxy admin registers wpad.<domain>?
- I explicitly urge you NOT to deploy ISATAP at your site!
- Nonetheless, if you want to see an example of how one could set up ISATAP, see
  http://technet.microsoft.com/en-us/magazine/2008.03.cableguy.aspx

2.6 IPv6 Switching and Routing

This Is Not A Switching/Routing Tutorial

- I'm not a network engineer, and neither are most of you
- This is just enough background to make you dangerous/get you started...
- On most hosts, IPv6 routing Just Works as shipped.
- At layer 2 (e.g., over ethernet switches), ethernet switches only care about MAC addresses, so when it comes to layer 2, IPv6 generally Just Works there too.
- Locally at layer 3, IPv6 may be routed by OSPF3 or ISIS. See your favorite vendor's documentation for details.
- IPv6 uses BGP4 in the wide area, just like IPv4. A nice quick IPv6 BGP recipe for native IPv6 peering with Internet2 can be found in the Internet2 IPv6 Cookbook, see http://noc.net.internet2.edu/i2network/ipv6-cookbook.html
- See also "Configuring IPv6 for Cisco IOS" by Sam Brown, et. al.

What About Open Source Routing Product?

- If you don't have large piles of IPv6-capable name brand routers floating around on which to experiment, but you do have some surplus PCs, you may want to check out Quagga, an open source/GPL licensed software routing suite which supports IPv6.
- As a sign of how we feel about it, I'd note that Oregon Routeviews uses Quagga for some of its RouteViews boxes.
- See http://www.quagga.net/ for more information

- Also worthy of note for those using Linux:
  www.linuxfoundation.org/collaborate/workgroups/networking/iproute2

Routing Gone Wrong #1: Rogue RAs

- http://tools.ietf.org/html/draft-chown-v6ops-rogue-ra-03

When deploying IPv6, whether IPv6-only or dual-stack, routers are configured to send IPv6 Router Advertisements to convey information to nodes that enable them to autoconfigure on the network. This information includes the implied default router address taken from the observed source address of the Router Advertisement (RA) message, as well as on-link prefix information. However, unintended misconfigurations by users or administrators, or possibly malicious attacks on the network, may lead to bogus RAs being present, which in turn can cause operational problems for hosts on the network. In this draft we summarise the scenarios in which rogue RAs may be observed and present a list of possible solutions to the problem. We focus on the unintended causes of rogue RAs in the text. The goal of this text is to be Informational, and as such to present a framework around which solutions can be proposed and discussed.

Routing Gone Wrong #2: Gratuitous Provision of IPv6 Transit

- Another example of how IPv6 connectivity can be at times less robust than IPv4 can be seen in problems associated with things like the "gratuitous provision of global transit."
- While offering to route anyone's IPv6 transit traffic at no charge and without prearrangement may seem like an incredibly generous thing to do, it can cause problems when production IPv6 traffic suddenly follows a "shorter" (BGP) path that flows indirectly via geographically remote parts of the world (or attempts to flow via circuits not provisioned to carry a material fraction of the whole world's IPv6 transit bandwidth). Fortunately, better BGP filtering has largely reduced or eliminated this issue today.
- A set of IPv6 BGP filters meant to provide a nice start at reducing the number of "problematic" global IPv6 routes is available at http://www.space.net/~gert/RIPE/ipv6-filters.html
  As always, the more strictly you filter, the more carefully/closely you'll need to work at keeping your filters updated.

DDoS Attacks Against IPv6 Sites

- Another example of a security-related routing issue that may arise in conjunction with IPv6 sites is mitigating distributed denial of service (DDoS) attacks. In the IPv4 world, a common option to avoid having DDoS traffic saturate downstream links is the use of blackhole routes.
- For example, Internet2's IPv4 BGP policy allows connectors to advertise BGP discard routes tagged with the BGP Community 11537:911 and a mask length from /24 to /32, in which case all packets arriving for that route will be discarded by all Internet2 Network routers, before those packets can saturate downstream customer links.
- The Internet2 community has discussed offering a comparable policy, obviously adjusted for IPv6 address lengths and prefix usage patterns (e.g., perhaps accepting masks from /64 to /128) should be implemented for IPv6 on Internet2.

Looking at Local IPv6 Routes on Mac OS X

```
% netstat -nr -finet6

Routing tables

Internet6:

Destination                       Gateway                       Flags       Netif
default                           2001:468:d01:d6::1            UGSc         en0
::1                               ::1                           UH           lo0
2001:468:d01:d6::/64              link#4                        UC           en0
2001:468:d01:d6::1                0:d0:1:95:e0:0                UHLW         en0
2001:468:d01:d6::80df:d617        0:3:93:cf:b6:38               UHL          lo0
fe80::/64                         fe80::1                       Uc           lo0
fe80::1                           link#1                        UHL          lo0
fe80::/64                         link#4                        UC           en0
fe80::203:93ff:fecf:b638          0:3:93:cf:b6:38               UHL          lo0
fe80::2d0:1ff:fe95:e000           0:d0:1:95:e0:0                UHLW         en0
fe80::/64                         link#5                        UC           fw0
fe80::203:93ff:fecf:b638          0.3.93.ff.fe.cf.b6.38         UHL          lo0
ff01::/32                         ::1                           U            lo0
ff02::/32                         ::1                           UC           lo0
ff02::/32                         link#5                        UC           fw0
```

Mac OS X NDP (e.g., Layer 2 Neighbors)

```
% ndp -a
Neighbor                        Linklayer Address  Netif Expire    St Flgs Prbs
vl-214-gw.uoregon.edu           0:d0:1:95:e0:0       en0 23h46m55s S  R
canard.ipv6.uoregon.edu         0:3:93:cf:b6:38      en0 permanent R
fe80::1%lo0                     (incomplete)         lo0 permanent R
fe80::203:93ff:fecf:b638%en0    0:3:93:cf:b6:38      en0 permanent R
fe80::2d0:1ff:fe95:e000%en0     0:d0:1:95:e0:0       en0 23h47m4s  S  R
fe80::203:93ff:fecf:b638%fw0    0:3:93:ff:fe:cf      fw0 permanent R
```

Looking at Local IPv6 Routes and Neighbors on Linux

```
% netstat -r --inet6
Kernel IPv6 routing table
Destination                                 Next Hop      Flags Metric  Ref     Use Iface
localhost/128                                    *        U     0       7264      2 lo
2001:468:d01:8e:209:3dff:fe13:fcf7/128           *        U     0       17551     2 lo
2001:468:d01:8e::/64                             *        UA    256     7935      0 eth0
fe80::209:3dff:fe13:fcf7/128                     *        U     0       242       2 lo
fe80::209:3dff:fe13:fcf8/128                     *        U     0       0         2 lo
fe80::/64                                        *        U     256     0         0 eth0
fe80::/64                                        *        U     256     0         0 eth0
fe80::/64                                        *        U     256     0         0 eth1
ff00::/8                                         *        U     256     0         0 eth0
ff00::/8                                         *        U     256     0         0 eth1
*/0                      fe80::2d0:1ff:fe95:e000     UGDA  1024    650       0 eth0

% ip -6 neighbor list
fe80::2d0:1ff:fe95:e000 dev eth0 lladdr 00:d0:01:95:e0:00 router nud stale
```

Magic Multicast Addresses

```
% ping6 -I eth0 ff02::1
% ping6 -I eth0 ff02::2
% ping6 -I eth0 ff02::5
```

[See http://www.iana.org/assignments/ipv6-multicast-addresses/]

Magic Multicast Address Output

```
% ping6 -I en0 ff02::1
PING6(56=40+8+8 bytes) fe80::203:93ff:fecf:b638 --> ff02::1
16 bytes from fe80::203:93ff:fecf:b638, icmp_seq=0 hlim=64 time=0.248 ms
16 bytes from fe80::20f:1fff:fe98:e548, icmp_seq=0 hlim=64 time=0.761 ms(DUP!)
16 bytes from fe80::213:faff:fe01:a6a4, icmp_seq=0 hlim=64 time=0.898 ms(DUP!)
16 bytes from fe80::2e0:29ff:fe3c:9a3a, icmp_seq=0 hlim=64 time=0.951 ms(DUP!)
16 bytes from fe80::2e0:dbff:fe10:75c, icmp_seq=0 hlim=64 time=1.254 ms(DUP!)
16 bytes from fe80::2d0:1ff:fe95:e000, icmp_seq=0 hlim=64 time=1.376 ms(DUP!)
16 bytes from fe80::2e0:dbff:fe10:7c6, icmp_seq=0 hlim=64 time=1.832 ms(DUP!)
16 bytes from fe80::216:d3ff:fe15:3045, icmp_seq=0 hlim=64 time=6.647 ms(DUP!)
16 bytes from fe80::21e:bff:fe17:b2a2, icmp_seq=0 hlim=64 time=202.283 ms(DUP!)
16 bytes from fe80::217:f2ff:fe08:93e4, icmp_seq=0 hlim=64 time=634.863 ms(DUP!)
[etc]

% ping6 -I en0 ff02::2
PING6(56=40+8+8 bytes) fe80::203:93ff:fecf:b638 --> ff02::2
16 bytes from fe80::2d0:1ff:fe95:e000, icmp_seq=0 hlim=64 time=5.206 ms
[etc]
```

Pre-Attack Network Reconnaissance

- It is common for miscreants to remotely scan IPv4 network addresses in an effort to identify active addresses, operating systems in use, open ports, etc., intelligence which may help them plan an attack against you. An increasingly common (if unfortunate) response to that threat has been to insert a firewall between the Internet and local users, thereby deflecting some scans and probes, albeit at the cost of a loss of transparency.
- Because IPv6-connected sites typically have a far larger number of addresses than IPv4-only sites, and end-to-end connectivity was another key objective of IPv6's architecture, some have suggested that it might be harder for attackers to do exhaustive scans of IPv6 sites simply because of the vastly larger number of addresses involved. That's true, as far as it goes, but that's not the whole story. If you haven't seen RFC 5157 ("IPv6 Implications for Network Scanning," March 2008), I'd urge you to look it over.
- So why did I mention ff02::1? If a miscreant can get a toehold on a local IPv6 connected host, they can obviously easily discover all the other hosts on that same subnet… :-;

2.7 IPv6 Address Space

Ironically, Until Now, We Haven't Talked Much About Getting IPv6 Address Space

- That's in part because getting IPv6 address space is rather easy.
- If you're an end user, your friendly local router or your local DHCPv6 server or your DNS admin will give you an address
- If you're running your own site, need less than a /32, and don't need to multihome, your LIR (e.g., your ISP) can fix you up with the addresses you need.
- If you need to multihome, or you need a /32 or more, you'll want to get provider independent (PI) IPv6 address space from ARIN…

Internet2 Participants With PI IPv6 Allocations

To see these, at http://routerproxy.grnoc.iu.edu/internet2/ select a node and then do:
show route table inet6.0 community 11537:950 terse (you're looking for non 2001:468 prefixes)
[BGP community info is at www.abilene.iu.edu/i2network/maps--documentation/cookbooks.html]

2001:4d0:9c00::/40	NASA
2001:4e0::/32	WiscNet
2001:5e8::/32	Pittsburgh Supercomputing Center
2001:1458::/32	CERN
2001:1860::/34	Pacific Northwest Gigapop
2001:1860:c000::/34	Pacific Northwest Gigapop
2001:18e8::/32	Indiana U
2001:1948::/32	Utah Education Network
2001:4898::/32	Microsoft
2001:48a8::/32	Merit
2001:48d0::/32	San Diego Supercomputer Center
2001:4930::/32	State of North Dakota ITD
2001:49d0::/32	Kansas Research and Education Network
2001:49d8:40::/42	Commonwealth of PA - OA / Integrated Network Management Services
2002::/16	6to4
2607:f010::/32	UCLA
2607:f140::/32	Berkeley
2607:f290::/32	UC Riverside
2607:f320::/32	U Nebraska-Lincoln
2607:f378::/32	UC Santa Barbara

I2 Participants With PI IPv6 Alloc. (cont.)

2607:f388::/32	U Wisconsin Madison
2607:f390::/32	Louisiana Board of Regents/Louisiana Optical Network Init
2607:f3b0::/32	NJEDge.Net, Inc.
2607:f470::/32	U Pennsylvania
2607:f600::/32	NYU
2610:8::/32	Penn State
2610:20:8000::/35	US Dept of Commerce
2610:28::/32	NCREN
2610:48::/32	U Maine System
2610:58::/32	Boston U
2610:a8::/32	OARnet
2610:d0::/32	Cisco
2610:e0::/32	U Missouri - dba the Missouri Research and Education Net
2610:130::/32	Iowa Communications Network
2610:148::/32	Georgia Tech
2610:1e0::/32	Kentucky Educational Computing Network
2620:0:270::/48	U Texas Health Science Center at Houston
2620:0:bc0::/48	George Mason U
2620:0:c30::/48	U South Florida
2620:0:c80::/48	NCSA
2620:0:df0::/48	Bryant U, RI
2620:0:e50::/48	U Iowa

I2 Members with Non-PI IPv6 Address Assignments From I2's IPv6 Allocation

- Internet2's IPv6 address block is 2001:0468::/32

  You can see documented assignments from within that block via whois. If you have a Linux box or Mac, pop up a terminal window and then enter…

  ```
  % whois -h whois.arin.net \>\ 2001:468::
  ```

- There's also an HTML page that lists all assignments from within that block:
  http://ipv6.internet2.edu/Abilene_Allocations.shtml

If Your Site Wanted To Get PI IPv6 Space

- In the ARIN region, the Number Resource Policy Manual describes the minimum requirements which a LIR (e.g., a service provider such as a RON/Gigapop) must meet in order to receive an initial minimum allocation of an IPv6 /32 (see http://www.arin.net/policy/nrpm.html#six):

  6.5.1.1. Initial allocation criteria

  To qualify for an initial allocation of IPv6 address space, an organization must:

  1. be an LIR;
  2. not be an end site;
  3. plan to provide IPv6 connectivity to organizations to which it will assign IPv6 address space, by advertising that connectivity through its single aggregated address allocation; and
  4. be an existing, known ISP in the ARIN region or have a plan for making at least 200 end-site assignments to other organizations within 5 years.

Once You Have Address Space, Decide On An IPv6 Address Plan

- You can use IPv6's vastly expanded address space to try new and innovative architectures, but you can also just map your static IPv4 address space to your IPv6 address space as UO does. For example:

```
% dig phloem.uoregon.edu +short
128.223.32.35
% dig phloem.uoregon.edu aaaa +short
2001:468:d01:20::80df:2023
```

```
128 ==> 80
223 ==> df
 32 ==> 20
 35 ==> 23
```

See also http://www.getipv6.info/index.php/IPv6_Addressing_Plans and http://www.ipv6book.ca/allocation.html

2.8 IPv6 DNS

If You Run Your Own Name Servers: Make Your Name Servers IPv6 Aware

- You'll need to think about making your name servers IPv6 aware. By this I mean your recursive resolvers and your authoritative name servers should both "know about" AAAA records.
- BIND from ISC, the most common name server, is fully able to support IPv6
- Your ***access*** to your name servers need not change -- you can continue to access them over IPv4 -- you just need to make sure that they understand AAAA records.
- If you have off site secondary name servers, make sure that those secondary name servers are also IPv6 aware
- A new option for folks is to outsource recursive DNS to Google -- 8.8.8.8 and 8.8.4.4 are already IPv6 aware[74]

IPv6 DNS Is Fairly Similar to IPv4 DNS

- In IPv4 world, servers and other hosts use "A" records to map fully qualified domain names to dotted quads:

  % dig www.uoregon.edu a +short
  128.223.142.89

- In IPv6 world, we use "AAAA" ("quad A") records instead of A records to map fully qualified domain names to IPv6 addresses:

  % dig network-services.uoregon.edu aaaa +short
  2001:468:d01:3c::80df:3c15

Inverse Address Records Are Also Similar

- IPv4 world:

  % dig -x 128.223.142.89 +short
  www.uoregon.edu.

- IPv6 world:

  % dig -x 2001:468:d01:3c::80df:3c15 +short
  network-services.uoregon.edu.

- If you need a web-accessible IPv6 dig interface, try http://www.digwebinterface.com/

Complication: IPv6 Inverse Address Records

- There aren't a lot of good solutions for IPv6 inverse address record creation (other than manually creating them for servers, publicly visible network links, etc.).

- Dynamic DNS isn't such a hot idea, nor is building an inverse address record for millions of IPv6 addresses which may never get used!

- Ron Broersam from DREN described one potential approach in a talk he did at Joint Techs, see:
  http://www.internet2.edu/presentations/jt2010feb/20100202-broersma.pdf
  but it is unclear if the code he mentions there will be available for others to also use. (I hope it will be!)

Complications: IPv6 AND IPv4 Domain Names

- If a fully qualified domain name is bound to both IPv4 and IPv6 addresses, which one gets used? Which one should be "preferred?" The IPv6 one or the IPv4 one?
- This may be determined by the application (e.g., it may ask for both, and then use its own internal precedence information to determine which it will use), or by the DNS server (if it can give you an IPv6 address for a host, it will, and it will then stop; this is a problem if you advertise an IPv6 address for a host but then don't actually offer IPv6 connectivity for that domain name!)
- One sort of kludgy solution: put the IPv6 address in a subdomain. E.g., www.example.com (IPv4 only) might also have a related domain www.ipv6.example.com (IPv6 only)

What Address Do I Appear to Be Using?

- Sometimes you just want to know what address your system appears to be using.

- When that's the case, try:

  http://whatismyv6.com/

- If you find yourself routinely curious about whether you're connecting to a site via IPv4 or IPv6, you can install the ShowIP Firefox addon available from

  https://addons.mozilla.org/en-US/firefox/addon/590

  It will then report the IP address that it is using for each web site you visit, green for IPv6 and red for IPv4

Protocol Precedence More Generally

- Assume you now have at least IPv4 connectivity as well as IPv6 connectivity via Teredo (you may also have native IPv6 connectivity, IPv6 connectivity via 6to4, IPv6 connectivity via ISATAP, IPv6 connectivity via a configured tunnel, etc.).

- If a host is available via IPv4 and at least one IPv6 connection modality, what's the right order (or precedence) with which traffic to that dual stack host should be tried? Should traffic go over IPv4 whenever possible, or over IPv6 whenever possible? Obviously IPv4 is the old-and-proven familiar/safe/production quality choice, but the object of this whole exercise is to try IPv6, right?

- On the other hand, some IPv6 connectivity (such as Teredo) may be less direct/more circuitous than production IPv4 connections, with greater latency and maybe even some packet loss. Do we really want to force as much traffic as possible over that sort of disadvantaged route?

Protocol Precedence More Generally (2)

- In IPv6, where a single computer may have multiple IPv6 addresses assigned, protocol preferences also influence which of multiple IPv6 address will be used when composing outbound IPv6 traffic. To see the current policy for a Windows IPv6 host, in a terminal window use the command:

  C:\> netsh interface ipv6 show prefixpolicy

  For example, a typical default prefix policy (matching well with RFC3484 recommendations) might look like:

```
Precedence   Label  Prefix ----------------------------------------------------------------
5             5     2001::/32      <== Teredo-tunneled IPv6 connectivity (least preferred)
10            4     ::ffff:0:0/96  <== IPv4 connectivity (represented here in IPv6 mapped format)
20            3     ::/96           <== deprecated; see RFC4291 at section 4
30            2     2002::/16       <== 6to4 IPv6 connectivity
40            1     ::/0            <== native IPv6 connectivity
50            0     ::1/128         <== IPv6 localhost (like 127.0.0.1 in IPv4) (most preferred)
```

  While theoretically you can change those preferences with netsh commands, in practice it appears difficult to force IPv6 to be prefd over IPv4 in practice under Windows XP Service Pack 2 when a hostname resolves to both an IPv6 and an IPv4 address. This is actually probably a good thing in practice, although it means that if you want to insure that traffic to a given host goes via IPv6, you may need an IPv6-only address for that host (e.g., www.ipv6.example.com is a common format for IPv6-only web servers)

3. Enabling IPv6 in Operating Systems

Most OS's Will Autoconfigure (Except XP)

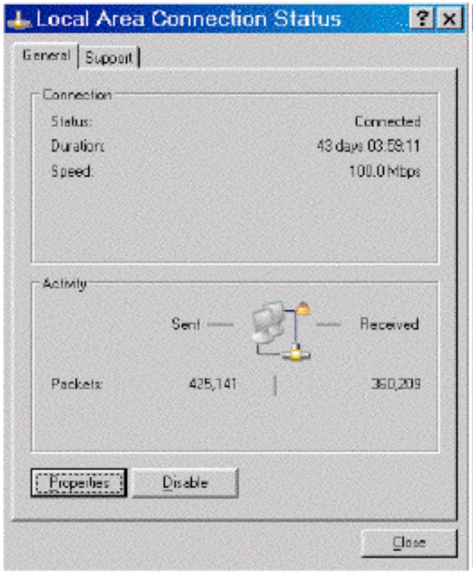
To enable IPv6, begin by putting up a Windows command prompt terminal window

Start ⟹ Programs ⟹ Accessories ⟹ Command Prompt
C:\> netsh interface ipv6 install (if you ever want to remove, change install to uninstall)
C:\> netsh interface ipv6 set teredo client (member of a domain? use enterpriseclient not client) (to disable, use disabled, not client)
C:\> ping ipv6.google.com (no reply? you probably need to configure an alternative IPv6 aware name server)

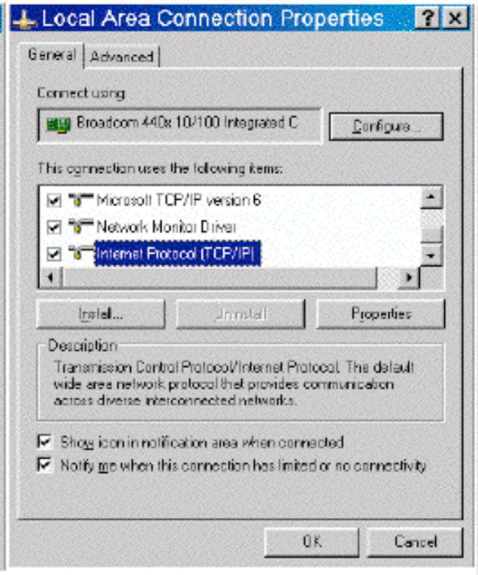
Configuring An IPv6 Aware Name Server, If Needed

Windows XP Service Pack 2 needs to one or more IPv6-**aware** name servers defined (an IPv6-aware name server is one which can return AAAA records for IPv6-connected fully qualified domain names). Note that Windows XP does **not** support name servers **accessible** only via IPv6. You may already have received a suitable name server as part of your existing IPv4 configuration, but if not, you will want to add an IPv6-aware name server. To manually configure an IPv6-aware name server, do:

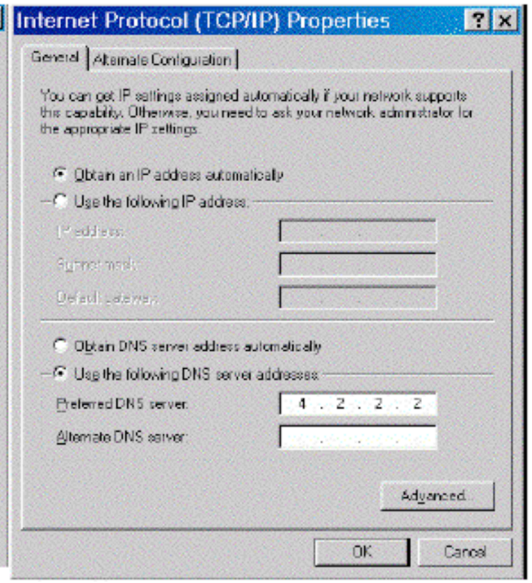
Start ⟹ Settings ⟹ Control Panel ⟹ Network Connections ⟹ Local Area Connection



Click Properties.



Come down to Internet Protocol (TCP/IP) and select that item. Now click Properties.



Click Use the following DNS server address and enter the name of the IPv6 aware name server you want to use (such as 4.2.2.2).

Enabling the Windows Software Firewall for Teredo

By default, Teredo is not subject to the Windows software firewall that ships with Windows. To enable/confirm IPv6 firewall coverage:

Start ==> Settings ==> Control Panel ==> Windows Firewall ==> Advanced tab
Local Area Connection should be checked (and should stay checked)
Select Local Area Connection, then click on Settings
If Teredo is listed as an accessible service (e.g., it is checked), uncheck it
Click OK, OK

If you are using a 3rd party Windows software firewall instead of the integrated Microsoft Firewall, consult your vendor for advice with respect to using that firewall in a Teredo-enabled IPv6 environment.

Confirming You Have IPv6 Connectivity

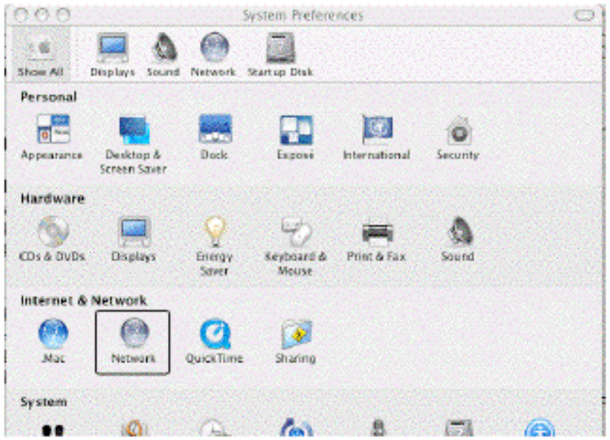
To confirm that you have IPv6 connectivity, try tracing to an IPv6 only destination such as ipv6.google.com

Start ==> Programs ==> Accessories ==> Command Prompt
C:\> tracert6 ipv6.google.com

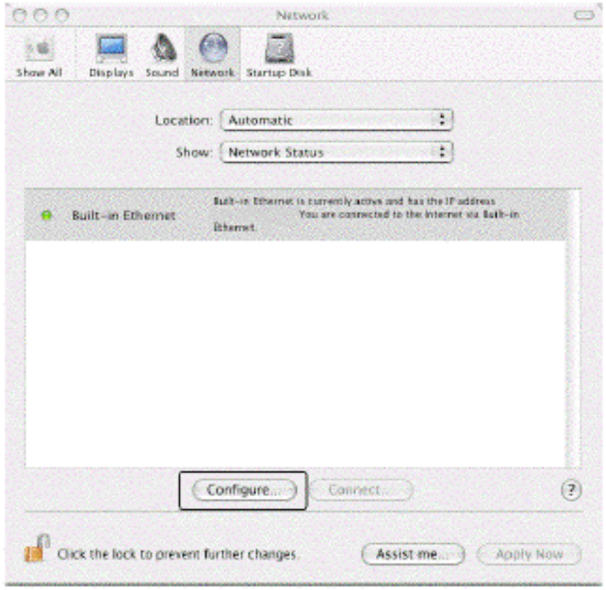
If you see a series of hops with IPv6 addresses, you have IPv6 connectivity. You may also find it interesting to see the path your traffic takes when using Teredo for IPv6 connectivity.

http://www.uoregon.edu/~joe/ipv6/windows-xp-ipv6-with-teredo.pdf

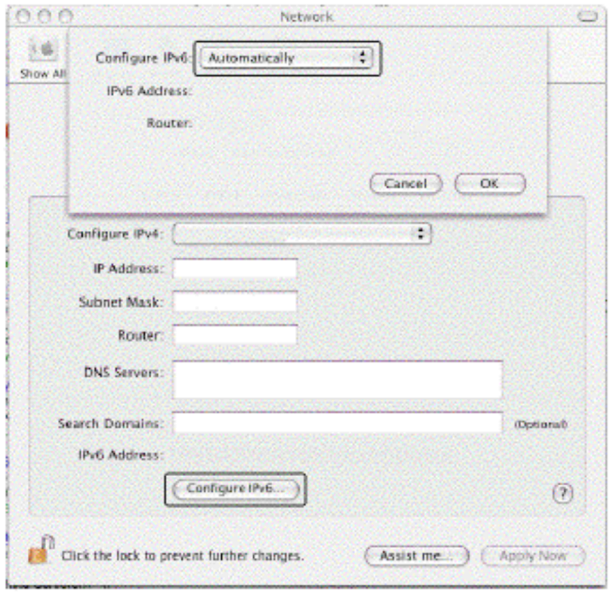
Mac OS X



Apple Menu ⟹ System Preferences ⟹ Network



On the network preferences screen, choose the drop down item: Network Status
Select the interface to be configured (e.g., wireless, built-in ethernet, etc.)
Click Configure



On the TCP/IP tab
Click on Configure IPv6 near the bottom of the pane
When the Configure IPv6 panel comes up, select Automatically
Click OK to close the Configure IPv6 panel

Click Apply Now
Close the Network Preferences window by clicking the red X button in the upper left corner
You should now have IPv6 enabled on your Mac

4. IPv6 Client Applications

4.1 IPv6 Web Browsers

Enabling IPv6 in Firefox for the Web

- If you don't already have Firefox, download it from: http://www.mozilla.com/firefox/
- Firefox are IPv6 aware, but may need to be tweaked the first time you use it (it will remember this setting after that):
  -- Launch Firefox
  -- In the address bar, enter about:config and hit return
  -- Filter on ipv6
  -- Make sure network.dns.disableIPv6 is set to false
  -- If it isn't, double click on it to toggle it to false

- Now go to http://ipv6.google.com/ -- you should see the Google icon "dance"

A Somewhat Random Selection of IPv6-Accessible Web Sites

Google	**http://ipv6.google.com/**	**2001:4860:b005::68**
AFRINIC	http://www.afrinic.net/	2001:42d0::200:80:1
APNIC	http://www.apnic.net/	2001:dc0:2001:b:4608::115
Arctic Regional Supercomputer Center	http://www.arsc.edu/	2001:480:150:75::109
Argonne National Lab Public Software Mirror	http://mirror.anl.gov/	2620::dc0:1800:214:4fff:fe7d:1b9
ARIN	http://www.arin.net/	2001:500:4:13::80 and 2001:500:4:13::81
Berkeley	http://www.ipv6.berkeley.edu/	2607:f140:ffff:ffff::80
DANTE	http://www.dante.net/	2001:798:2:284d::60
deepspace6.net (Linux IPv6 portal)	http://www.deepspace6.net/	2001:760:2e01:1::dead:beef
DREN	http://www.ipv6.dren.net/	2001:480:10:1050::5
ESNet	http://www.es.net/	2001:400:14:3::d
Fort Scott Community College	http://www.fortscott.edu/	2001:49d0:2c02::214:22ff:fe0f:38f4
Hexago	http://www.hexago.com/	2001:5c0:1000:10::2
HPCMO	http://www.hpcmo.hpc.mil/	2001:480:430:dddd:68b0:bff:fed5:f128
Hurricane Electric	http://ipv6.he.net/	2001:470:0:64::2
IANA	http://www.iana.org/	2620:0:2d0:1::193
ICANN	http://www.icann.org/	2620:0:2d0:1::103
IETF	http://www.ietf.org/	2001:1890:1112:1::20
Internet2	http://www.ipv6.internet2.edu/	2001:468:1420::151
Iowa State University	http://www.iastate.edu/	2610:130:101:100::7
IPv6.org	http://www.ipv6.org/	2001:6b0:1:ea:202:a5ff:fecd:13a6
IPv6 Portal	http://www.ipv6tf.org/	2a01:48:1::2e0:81ff:fe05:4658
IPv6 Task Force	http://www.ipv6.eu/	2001:690:1fff:200::226
ISC	http://www.isc.org/	2001:4f8:0:2::d
KAME Project	http://www.kame.net/	2001:200::8002:203:47ff:fea5:3085
KANREN	http://www.kanren.net/	2001:49d0:3c00:1:209:6bff:fe7f:6c06
Kenya NIC	http://www.kenic.or.ke/	2001:43f8:10:50::2655
LACNIC	http://www.lacnic.net/	2001:13c7:7002:4000::10

Linkopings University	http://www.liu.se/	2001:6b0:17:f005::148
MAGPI	http://www.magpi.net/	2001:468:1802:101::805b:22c
Michigan Tech	http://www.ipv6.mtu.edu/	2001:48a8:0:2::1:22
NANOG	http://www.nanog.org/	2001:48a8:6880:95::21
Nederlandse IPv6 Task Force	http://www.ipv6-taskforce.nl/	2001:610:512::1000
NREN	http://www.nren.nasa.gov/	2001:4d0:8102:8:198:10:138:131
NTT	http://www.nttv6.net/	2001:fa8::80
Peter Bieringer	http://www.ipv6.bieringer.de/	2001:a60:9002:1::186:6
RIPE	http://www.ripe.net/	2001:610:240:11::c100:1319
Sauk Valley Community College	http://www.svcc.edu/	2001:470:c10b::214:5eff:fe28:7878
Space and Naval Warfare Systems Command	http://www.spawar.navy.mil/	2001:480:10:1048:a00:20ff:fe9a:58c1
SURFNet	http://www.ipv6.surfnet.nl/	2001:610:510::192:42:113:60
SWITCH	http://www.switch.ch/	2001:620:0:1b::b
TERENA	http://www.terena.org/	2001:610:148:dead::6
Tsinghua University	http://www.tsinghua.edu.cn/	2001:da8:200:200::4:100
UCLA	http://www.ucla.edu/	2607:f010:3fe:301:101d:9ff:fe32:a7db (etc)
University of Hawaii	http://www.ipv6.hawaii.edu/	2607:f278:4101:11:209:5bff:fe8f:6609
University of Oregon video archive	http://limestone.uoregon.edu/	2001:468:d01:103::80df:9d0c
Virginia Tech	http://www.ipv6.vt.edu/	2001:468:c80:210f::173:2bc6:4145
WPI	http://www.wpi.edu/	2001:468:616:824::31
3ROX	http://www.3rox.net/	2001:5e8:0:1000::fa
6TAP	http://www.6tap.net/	2001:400:14:3::9

Track the IPv6-ification of some of your favorite web sites at http://www.mrp.net/IPv6_Survey.html

Access most any IPv4-only web site via IPv6 using sixxs.org s IPv4-to-IPv6 gateway by just appending sixxs.org to the normal IPv4 only domain name. For example, to access www.cnn.com via IPv6, you d go to http://www.cnn.com.sixxs.org/

What About Other GUI Web Browsers?

- Opera (http://www.opera.com/) is ready to use IPv6 as shipped with no tweaking required, likewise Internet Explorer 8.
- Safari is IPv6 enabled, but some versions of Safari may prefer IPv4 to IPv6 when both are available. To reverse that preference for dual stack sites:
  1) In a Mac terminal window (Applications --> Utilities --> Terminal) enable the Safari Debug menu by entering:
  defaults write com.apple.Safari IncludeDebugMenu 1
  You can then close the terminal window.
  2) Launch Safari, then go to Safari's Debug Menu (on the far right side of the main Safari menu), and come all the way down to Supported Protocols and **uncheck** http: (Simple Loader). You only need to do this once.

lynx and IPv6

- Get and build lynx from http://lynx.isc.org/

  Remember when ./configure'ing to include the option --enable ipv6

- If a site is available via both IPv6 and IPv4, lynx will prefer IPv6 by default

- Visiting an IPv6 site via the site's hex IPv6 address? Remember to use brackets around the literal address:

  % lynx "http://[2001:48a8:6880:95::21]"

wget and IPv6

- Get and build wget from ftp://ftp.gnu.org/gnu/wget/

- To force retrieval of a web page via IPv6, add -6:
  % wget -6 "http://www.example.com"

- To force retrieval of a web page via IPv4, add -4:
  % wget -4 "http://www.example.com"

- Need to specify an IPv6 hex literal? Remember brackets!
  % wget "http://[2001:48a8:6880:95::21]"

curl and IPv6

- Get curl from http://curl.haax.se/download.html
  (see also the additional libraries at
  http://curl.haxx.se/docs/libs.html)

- To force retrieval of a web page via IPv6, add -6:
  % curl -6 "http://www.example.com"

- To force retrieval of a web page via IPv4, add -4:
  % curl -4 "http://www.example.com"

- Need to specify an IPv6 hex literal? Remember brackets and -g (http://curl.haxx.se/docs/knownbugs.html at 30)
  % curl -6 -g "http://[2001:48a8:6880:95::21]"

4.2 IPv6 Email Clients

Modern MUAs Support IPv6

- Apple Mail.App
- Opera Mail client
- Outlook 2007
  (see http://support.microsoft.com/kb/924469)
- Thunderbird: (see the next slide)
- Other MUAs?

Enabling IPv6 in Thunderbird for Email

- Thunderbird (Mozilla's graphical email client, sibling application to the Firefox web browser) may ship with IPv6 DNS queries disabled by default.

- Enabling IPv6 DNS queries requires you to:
  -- Start Thunderbird
  -- Go to Thunderbird --> Preferences
  -- Go to Advanced (the "sprocket" or "gear" at the top)
  -- Click the "Config Editor" button near the bottom
  -- In the about:config window's Filter: box, enter **ipv6**
  -- You should see:
     network.dns.disableIPv6 [etc] false
  -- If, however, that setting is true, double click true to toggle it to false

alpine and IPv6

- Alpine (a replacement for good old pine) offers the ability to both do email and usenet news over IPv6.

- Get alpine from http://www.washington.edu/alpine/acquire/ (you may also need to install OpenSSL from http://www.openssl.org/source/ if you don't already have it)

Public IPv6-Accessible Web Email

- As another example of IPv6 email lagging relative to where it should be, consider web-based email, one of the most popular and widely used applications on the Internet today.
- To the best of my knowledge, there is currently only ONE (1) public web email provider which has even an experimental web email service accessible via IPv6, and that's http://ipv6.rollernet.us/
- Technically, if you're able to manipulate your local hosts file, you can also access Google's Gmail via IPv6, see http://jeremy.visser.name/2008/11/25/how-to-access-gmail-and-google-reader-over-ipv6/ (URL wrapped due to length), but that's hardly the sort of thing that a typical user should be expected to be able to do.
- **If you're aware of any additional IPv6-accessible public web email service providers, I'd love to know about them.**

4.3 ssh for IPv6

ssh and IPv6

- ssh supports IPv6. To get OpenSSH (5.5p1 was released 4/15/2010!) go to www.openssh.com/portable.html (be sure you're also running the latest OpenSSL -- 1.0.0 was released 3/29/2010 -- see www.openssl.org/ and don't forget to set a sane installation prefix!)

- To force ssh to use IPv6, use:
  % ssh -6 joe@example.com

- To force ssh to use IPv4, use:
  % ssh -4 joe@example.com

4.4 Usenet and IPv6

Reading Usenet News via alpine and IPv6

- Edit .pinerc, setting nntp-server=newszilla6.xs4all.nl
  (other free IPv6 news servers are also available, see
  http://www.xsnews.com/ipv6/ipv6_aanvraag.php and
  http://www.xsnews.com/ipv6/ for IPv6 traffic graphs!)
- % alpine
  L (go to folder list or news group to view)
  n (go down to pick the news server you've configured)
  return (to go to that new server)
  A (to add newsgroups to your subscribed list)
  ^T (get list of all available newsgroups; caution, large!)
  move up and down with your arrow keys/spacebar
  s (subscribe)
  e (exit subscribe mode)
- You can also use Thunderbird if you like GUI clients

5. IPv6 Server-Side Applications

5.1 Web Servers

What About YOUR Web Site?

- **Is your agency's web site IPv6 accessible?** If not, why not? The most popular web servers, e.g., Apache and Microsoft IIS, both support IPv6 these days! So what's the "holdup," eh?
- Often, your site may not have IPv6 connectivity (but it should!)
- Another potential roadblock to explore is whether hardware load balancers are IPv4 only. Hardware load balancers commonly sit in front of multiple physical computers, making a pile of systems act as if it were one computer. Some popular hardware load balancers license IPv6 functionality separately from IPv4.
- Other times the issue may be the use of outsourced content delivery networks (CDNs) which may not be IPv6-enabled (previously discussed)
- Lastly, at least sometimes, no one may every have asked, "Hey, why isn't our web site IPv6 accessible?" Perhaps that's a question that YOU should ask once you're done with this training?

Apache Web Server and IPv6

- Get Apache 2.2.15 (or whatever's the latest stable version) from http://httpd.apache.org/
- Review httpd.apache.org/docs/2.2/bind.html#ipv6 but otherwise build, install and configure as normal
- When configuring for IPv6, in /etc/httpd/httpd.conf, bind to an appropriate static IPv6 address:
  BindAddress [2001:468:d01:d6::80df:d617]
- Check your config and start httpd; typically:
  /usr/local/apache2/bin/apachectl configtest
  /usr/local/apache2/bin/apachectl start
- Confirm that you can connect OK to your IPv6 httpd:
  % telnet 2001:468:d01:d6::80df:d617 80
  GET / (note: case matters, GET, not get)
- Problems? Likely a firewall thing, as "always!" :-;

ip6fw on a Mac

```
# ip6fw list                <-- see what rules are currently defined
# ip6fw list > orig.txt     <-- grab a copy of those rules before we play...
# ip6fw flush               <-- dump all rules (e.g., permit all traffic)
# ip6fw add 100 deny ipv6 from any to any    <-- block all traffic
# ip6fw delete 100          <-- delete rule 100
```

Note: rule order counts! First match "wins" (that's why you get (and need!) the ability to specify rule numbers (and thus precedence))

If you need to reload orig.txt, edit that file and add ***ip6fw add*** in front of each saved line; if there are any wildcards (e.g., *'s) in the rules, escape those with a \ to prevent unintended shell expansion.

Then execute that file, watching for any issues.

```
# sh -x orig.txt
```

How Can I See What's Running Using IPv6?

```
# lsof -i6
```

```
COMMAND     PID   USER   FD   TYPE     DEVICE SIZE/OFF NODE NAME
syslogd      81   root    4u  IPv6 0x01c01f20      0t0  UDP *:syslog
mDNSRespo   167 nobody    8u  IPv6 0x01c01220      0t0  UDP *:mdns
xinetd      307   root    5u  IPv6 0x021bee80      0t0  TCP *:ssh
couriertc  3680   root    3u  IPv6 0x0362a970      0t0  TCP canard.ipv6.uoregon.edu:imap
master    21204   root   13u  IPv6 0x021be610      0t0  TCP *:smtp (ESTABLISHED)
httpd     29156   root    5u  IPv6 0x021be2b0      0t0  TCP canard.ipv6.uoregon.edu:http (CLOSED)
httpd     29253 daemon    5u  IPv6 0x021be2b0      0t0  TCP canard.ipv6.uoregon.edu:http (CLOSED)
httpd     29254 daemon    5u  IPv6 0x021be2b0      0t0  TCP canard.ipv6.uoregon.edu:http (CLOSED)
httpd     29255 daemon    5u  IPv6 0x021be2b0      0t0  TCP canard.ipv6.uoregon.edu:http (CLOSED)
httpd     29256 daemon    5u  IPv6 0x021be2b0      0t0  TCP canard.ipv6.uoregon.edu:http (CLOSED)
httpd     29257 daemon    5u  IPv6 0x021be2b0      0t0  TCP canard.ipv6.uoregon.edu:http (CLOSED)
httpd     29259 daemon    5u  IPv6 0x021be2b0      0t0  TCP canard.ipv6.uoregon.edu:http (CLOSED)
```

Don't Forget About IPv6 Addrs in Log Files

```
# cd /usr/local/apache2/logs
# cat access_log
[...]
2001:468:d01:d6::80df:d617 - - [23/Apr/2010:10:20:29 -0700]
"GET / HTTP/1.1" 200 54
[etc]
```

Does your log file analyzer product support IPv6 addresses? Some, like AWStats from http://awstats.sourceforge.net/ require a separate plugin to enable some IPv6 functionality; other functionality, like mapping addresses to geographic locations, may simply not be available for IPv6)

5.2 IPv6 SMTP and IMAP Servers

Email Is The "Forgotten" Application of IPv6

- While many people are very excited about the thought of using IPv6 for web servers, for some reason there seems to be less excitement about using IPv6 for email.
- Let's consider a few examples of this:
  -- Many mainstream mail software products support IPv6, but relatively few mail administrators apparently enable IPv6 support
  -- IPv6 DNS Blocklist suport is missing
  -- IPv6-accessible public web email services are nearly nil
- But some sites ARE deploying IPv6-accessible mail servers right now. For example...

Sample Institutional IPv6 Enabled MX

```
% dig ucla.edu mx +short
5 smtp.ucla.edu.

% dig smtp.ucla.edu a +short
169.232.46.240
169.232.46.241
169.232.46.242
169.232.46.244
etc.

% dig smtp.ucla.edu aaaa +short
2607:f010:3fe:302:1013:72ff:fe5b:60c3
2607:f010:3fe:102:101c:23ff:febe:116e
2607:f010:3fe:102:101c:23ff:febf:cfa7
2607:f010:3fe:102:101c:23ff:fed0:918c
etc.
```

IPv6 Support In Mainstream Email Software Products

- Virtually all modern mail transfer agents support IPv6:

  -- Exchange 2007 SP1 (only under Windows Server 2008, and only with both IPv4 and IPv6 enabled); see http://technet.microsoft.com/en-us/library/bb629624.aspx
  -- Exim (http://www.exim.org/exim-html-current/doc/html/spec_html/ch04.html at section 4.8)
  -- Postfix (http://www.postfix.org/IPV6_README.html)
  -- Qmail (via Qsmtp, see http://opensource.sf-tec.de/Qsmtp/)
  -- Sendmail (see the Sendmail Installation and Operation Guide)

- What about Procmail as a local mail delivery agent? Umm... see http://www.procmail.org/todo.html

IPv6 Support for imapd

- Yep, imapd support under IPv6 is available as well…
  For example:

  -- Courier: http://www.courier-mta.org/imap/features.html

  -- Dovecot: http://www.dovecot.org/

  -- UW: www.washington.edu/imap/documentation/IPv6.txt.html

  -- etc.

An Example of How the Email World Is Lagging in IPv6 Space: DNS Blocklists

- DNS blocklists, such as those offered by Spamhaus, are a key anti-abuse tool in today's IPv4-dominated Internet, directly blocking spam while also encouraging ISPs to employ sound anti-abuse practices.
- Virtually all sites that use DNS-based blocklists rely on rbldnsd (see http://www.corpit.ru/mjt/rbldnsd/rbldnsd.8.html).
- rbldnsd does NOT support IPv6 records at this time
- Spamhaus (and all other block list operators I'm aware) also do not maintain any IPv6 blocklists.
- How, then, is the mail community to block abusive traffic sources coming from IPv6 space?
- If we cannot support IPv6 entries in blocklists, I believe we have a fundamental deficiency we still need to address.

Redeeming Features of IPv6 Email

- Just as in the "good old days" of IPv4, most of the people who are doing IPv6 email today are pretty responsible folks so thankfully there hasn't been much IPv6-delivered abuse.

- At some point, however, we will begin to see unwanted traffic coming in over IPv6, and at that point it would sure be great if we were ready to block it, eh?

- Some anti-abuse email technologies are ready today, e.g., SPF does support IPv6 (see http://www.openspf.org/SPF_Record_Syntax)

Enabling IPv6 in postfix

- Get postfix 2.7 (or whatever's the latest stable version) from http://www.postfix.org/download.html
- Review http://www.postfix.org/IPV6_README.html
- When configuring for IPv6, in /etc/postfix/main.cf, set inet_protocols = ipv6, ipv4 (if you're dual stacking)
- Also include in /etc/postfix/main.cf the address you want to use for outgoing IPv6 SMTP connections:
  smtp_bind_address6 = 2001:468:d01:d6::80df:d617
- Check your config and start postfix; typically:
  /usr/sbin/postfix check
  /usr/sbin/postfix start
- Confirm that you can connect OK to your IPv6 smtpd:
  % telnet 2001:468:d01:d6::80df:d617 25
  quit

imapds and IPv6

- There are many different SSL-secured imap servers (for example, at UO we run dovecot), but just for something different, let's do http://www.courier-mta.org/imap/
- Build Courier imapd the way you normally would (e.g., begin by viewing 00README.NOW.OR.SUFFER :-), and be sure to create an imapd cert with mkimapdcert)
- Then, as part of the configuration process, cd to /usr/lib/courier-imap/etc and edit imapd-ssl so that the PORT line lists the IPv6 address of your server with a dot 993 at the end. For example: PORT=2001:468:d01:d6::80df:d617.993
- Start Courier as you normally would, e.g.: /usr/lib/courier-imap/libexec/imapd-ssl.rc start

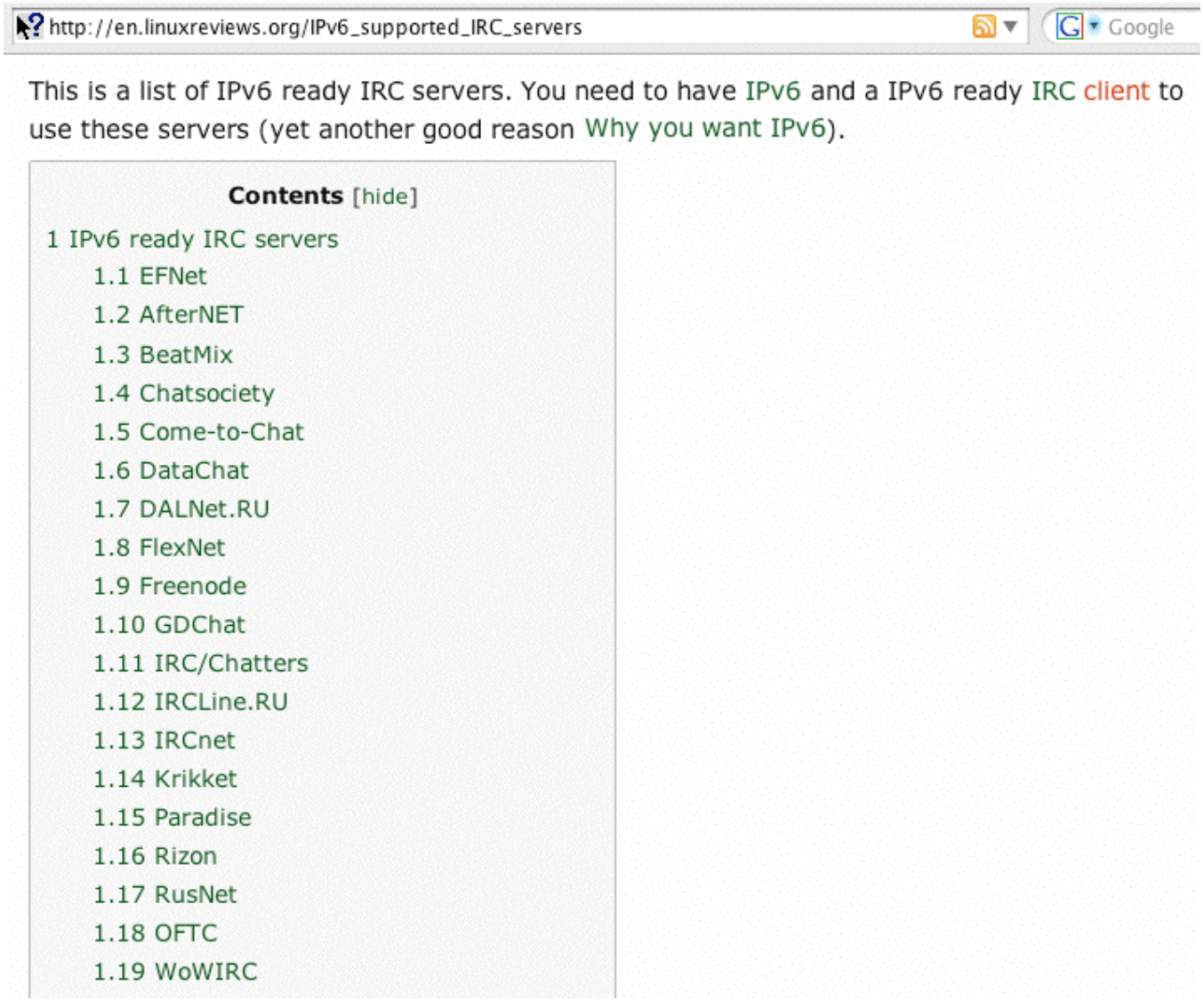
5.3 IPv6 Usenet Servers

Usenet Servers and IPv6

- INN supports IPv6, as does Leafnode
- Get INN inn-2.5.2.tar.gz from ftp://ftp.isc.org/isc/inn
- Get Leafnode from http://leafnode.sourceforge.net/

5.4 IPv6 IRC Servers

IRC Servers Running on IPv6



6. Miscellaneous Tools and IPv6

6.1 IPv6 Ping

- Ping in IPv6 works basically the same as ping in IPv4 (although you may need to use the command name "ping6" instead of just "ping" or you may need to add a flag to tell ping to use IPv6 instead of IPv4).
- For example:

```
% ping6 www.nanog.org
PING www.nanog.org(s1.nanog.org) 56 data bytes
64 bytes from s1.nanog.org: icmp_seq=0 ttl=55 time=107 ms
64 bytes from s1.nanog.org: icmp_seq=1 ttl=55 time=106 ms
64 bytes from s1.nanog.org: icmp_seq=2 ttl=55 time=107 ms
64 bytes from s1.nanog.org: icmp_seq=3 ttl=55 time=107 ms
[etc]
```

Some Notes About IPv6 Ping

- If you need to do an IPv6 ping from a Windows system, open a CMD window and use the "ping6" command
- Need a web-accessible IPv6 ping'er? Try: www.subnetonline.com/pages/ipv6-network-tools.php
- You can also ping from Route-Views. To do so, telnet to route-views.oregon-ix.net , login as rviews (no password required) then use "ping ipv6". For example:

```
% telnet route-views.oregon-ix.net
Username: rviews
route-views>ping ipv6 www.ietf.org
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:1890:1112:1::20, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 196/197/200 ms
```

6.2 IPv6 Traceroute

- Traceroute in IPv6 works basically the same as traceroute in IPv4 (although you may need to use the command name "traceroute6" instead of just "traceroute" or you may need to add a flag to tell traceroute to use IPv6 instead of IPv4)

```
% traceroute6 www.ietf.org
[snip]
1    vl-142-gw.uoregon.edu (2001:468:d01:8e::1)  2.337 ms  1.039 ms  0.941 ms
2    vl-3.uonet9-gw.uoregon.edu (2001:468:d01:3::9)  0.3 ms  0.261 ms  0.259 ms
3    2001:468:d00:3::1 (2001:468:d00:3::1)  2.717 ms  2.725 ms  2.71 ms
4    2001:468:ffff:54d::1 (2001:468:ffff:54d::1)  24.577 ms  24.595 ms  24.569
     ms
5    pao-ipv6.gblx.net (2001:504:d::37)  33.242 ms  169.607 ms  33.134 ms
6    2001:1890:1fff:308:192:205:34:77 (2001:1890:1fff:308:192:205:34:77)
     82.432 ms  82.434 ms  82.478 ms [etc]
```

Some Notes About IPv6 Traceroutes

- Need to do an IPv6 traceroute from a Windows system? Open a CMD window and use the "tracert6" command
- Need a web-accessible IPv6 traceroute server? One's available at http://4or6.com/
- Want a traceroute augmented with routing information? telnet to route-views.oregon-ix.net , login as rviews (no password required) then use "traceroute ipv6"
- Many hops in an IPv6 traceroute may lack rDNS (in-addrs are often something of an afterthought)
- IPv6 traceroute paths will often follow bizarre paths due to poor route filtering policies (or the use of IPv6 transition technologies or IPv6 tunnel provider services)
- IPv4 and IPv6 traceroute paths may routinely follow radically different paths; make no assumptions!

6.3 tcptraceroute (via ndisc6)

- "NDisc6 is a small collection of useful tools for IPv6 networking. It includes the following programs :

  ndisc6: ICMPv6 Neighbor Discovery tool [...]
  rdisc6: ICMPv6 Router Discovery tool [...]
  tcptraceroute6: lightweight IPv6 tcptraceroute [...]
  traceroute6: IPv6 traceroute [...]
  rdnssd: Recursive DNS Servers discovery Daemon [...]
  isatapd: ISATAP daemon [...]"

  See http://www.remlab.net/ndisc6/

  But note: I see errors when building; YMMV

6.4 Routeviews and IPv6

- IPv6 aware route servers are available; one list of IPv6-enabled route servers can be found at www.bgp4.net/wiki/doku.php?id=tools:ipv6_route_servers

```
% telnet route-views.eqix.routeviews.org
route-views.eqix-ash> show ipv6 bgp 2001:468:d01:d6::80df:d617
BGP routing table entry for 2001:468::/32
Paths: (9 available, best #9, table Default-IP-Routing-Table)
Not advertised to any peer
33437 6939 11537
  2001:504:0:2:0:3:3437:1 from 2001:504:0:2:0:3:3437:1
  (66.117.47.226)
  (fe80::2b0:c2ff:fe13:b01c)
  Origin IGP, localpref 100, valid, external
  Last update: Thu Apr 29 16:46:47 2010

4436 11537
  2001:504:0:2::4436:2 from 2001:504:0:2::4436:2 (69.31.31.244)
  [etc]
```

6.5 nc6 (netcat for IPv6)

- Quoting from http://www.deepspace6.net/projects/netcat6.html

  "In the simplest usage, "nc6 host port" creates a TCP connection to the given port on the given target host (using either IPv4 or IPv6 as appropriate). Your standard input is then sent to the host, and anything that comes back across the connection is sent to your standard output. This continues indefinitely, until the network side of the connection shuts down. [...] Netcat6 can also function as a server, by listening for inbound connections on arbitrary ports and then doing the same reading and writing when a client connects. With minor limitations, netcat doesn't really care if it runs in "client" or "server" mode -- it still shovels data back and forth until there isn't any more left. Netcat6 can also be used over UDP [...]"

nc6 example (from man nc6)

- Server:
  % nc6 --continuous --exec cat -l -p 2345
  -- continuous (act like inetd)
  -- exec (exec this command)
  cat (the regular old unix cat command)
  -l listen for inbound connects
  -p on port number...

- Client:
  % telnet <ipv6 addr of server> 2345
  hello world <-- your input
  hello world <-- echo'd back

- Use ^C to kill the server, ^] q to kill the client

6.6 tcpdump and IPv6

- Build and install tcpdump (4.1.1) and libpcap (1.1.1) from http://www.tcpdump.org/

- For IPv6, add the tcpdump ip6 option. For example:
  # tcpdump -xvs0 ip6 and tcp port 80

  x --> print each packet (except link layer hdr) in hex
  v --> slightly verbose
  s0 --> capture the whole packet, not just 68 bytes
  ip6 --> ipv6 traffic only
  tcp port 80 --> only tcp traffic on port 80

- If IPSec is being used on IPv6 links, obviously traffic may be obfuscated (unless you know the IPSec key)

Sample tcpdump output (ipv6.google.com)

- tcpdump: listening on en0, link-type EN10MB (Ethernet), capture size 65535 bytes 09:15:39.691986 canard.ipv6.uoregon.edu.59761 > **pv-in-x68.1e100.net**.http: P [tcp sum ok] 3681843577:3681844207(630) ack 2324549898 win 65535 <nop,nop,timestamp 365589597 2712339760> (len 662, hlim 64)

```
        0x0000:  6000 0000 0296 0640 2001 0468 0d01 00d6  `......@...h....
        0x0010:  0000 0000 80df d617 2001 4860 b006 0000  ..........H`....
        0x0020:  0000 0000 0000 0068 e971 0050 db74 7979  .......h.q.P.tyy
        0x0030:  8a8d d10a 8018 ffff 7235 0000 0101 080a  ........r5......
        0x0040:  15ca 745d a1ab 0530 4745 5420 2f20 4854  ..t]...0GET./.HT
        0x0050:  5450 2f31 2e31 0d0a 486f 7374 3a20 6970  TP/1.1..Host:.ip
        0x0060:  7636 2e67 6f6f 676c 652e 636f 6d0d 0a55  v6.google.com..U
        0x0070:  7365 722d 4167 656e 743a 204d 6f7a 696c  ser-Agent:.Mozil
        [snip]
```

09:15:39.755859 **pv-in-x68.1e100.net**.http > canard.ipv6.uoregon.edu.59761: . [tcp sum ok] 1:1209(1208) ack 630 win 285 <nop,nop,timestamp 2712558929 365589597> (len 1240, hlim 54)

```
        0x0000:  6000 0000 04d8 0636 2001 4860 b006 0000  `......6..H`....
        0x0010:  0000 0000 0000 0068 2001 0468 0d01 00d6  .......h...h....
        0x0020:  0000 0000 80df d617 0050 e971 8a8d d10a  .........P.q....
        0x0030:  db74 7bef 8010 011d 05d1 0000 0101 080a  .t{.............
        0x0040:  a1ae 5d51 15ca 745d 4854 5450 2f31 2e31  ..]Q..t]HTTP/1.1
        0x0050:  2032 3030 204f 4b0d 0a44 6174 653a 2053  .200.OK..Date:.S
        0x0060:  6174 2c20 3031 204d 6179 2032 3031 3020  at,.01.May.2010.
        0x0070:  3136 3a31 353a 3339 2047 4d54 0d0a 4578  16:15:39.GMT..Ex
        [etc]
```

Watch An Amsterdam Street Cam via IPv6

- http://www.terena.org/webcam/

  (you can use tcpdump to confirm that yes, it really IS transmitting via IPv6, e.g.:

  # tcpdump -i en0 -xvs0 ip6

  will show traffic from wowza.terena.org)

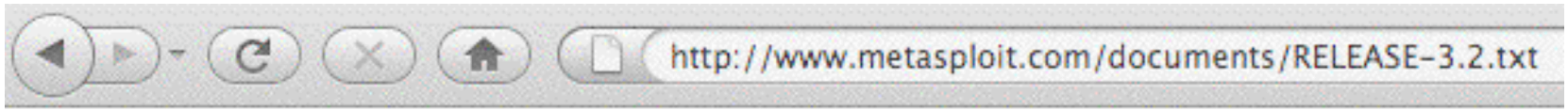
Additional ipv6 tcpdump commands

- Want more? All IPv6 traffic verbosely:
  # tcpdump -xxvvves0 ip6
- Want less? Just IPv6 traffic to/from a specific host:
  # tcpdump -xvs0 ip6 host foo.example.com *or*
  # tcpdump -xvs0 ip6 host fe80::203:93ff:fecf:b6a2
- Multiple interfaces? You can specify a specific interface (use ifconfig -a *or* tcpdump -D to see what's available to pick).
  For example, want to see if any ping6's are getting through on en0?
  # tcpdump -i en0 icmp6
- IPv6 multicast traffic can be interesting:
  # tcpdump -i en0 ip6 multicast
- Capture/replay tcpdump files:
  # tcpdump -i en0 -w sample.dump ip6
  [after a while, ^C]
  # tcpdump -r sample.dump
- Need the contents of tunneled IPv6 traffic? Check out teredont:
  http://speakeasy.wpi.edu/teredont/

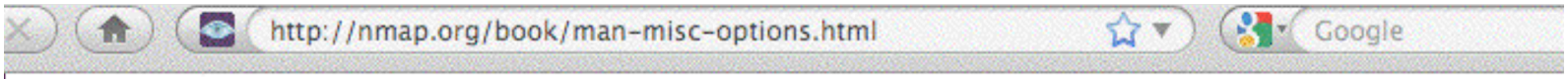
IPv6 Packet Formats

- Cool little pocket guide from SANS:
  http://www.sans.org/security-resources/ipv6_tcpip_pocketguide.pdf

6.7 A Couple Other Staple Security Tools




```
  Nearly all Metasploit modules now support IPv6 transports. IPv6 stagers
exist for the Windows and Linux platforms, opening the door for penetration
testing of pure IPv6 networks. The VNCInject and Meterpreter payloads have
been extensively tested over IPv6 sockets.
```



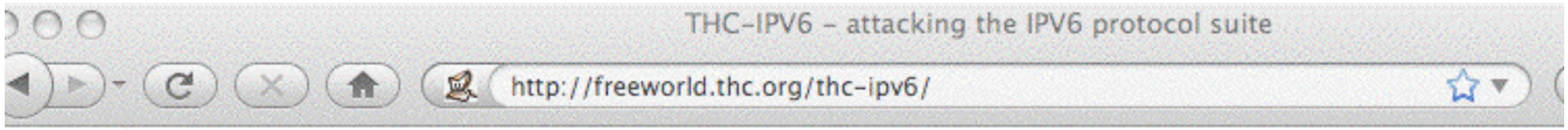
Miscellaneous Options

This section describes some important (and not-so-important) options that don't really fit anywhere else.

`-6` (Enable IPv6 scanning)

> Since 2002, Nmap has offered IPv6 support for its most popular features. In particular, ping scanning (TCP-only), connect scanning, and version detection all support IPv6. The command syntax is the same as usual except that you also add the `-6` option. Of course, you must use IPv6 syntax if you specify an address rather than a hostname. An address might look like `3ffe:7501:4819:2000:210:f3ff:fe03:14d0`, so hostnames are recommended. The output looks the same as usual, with the IPv6 address on the "interesting ports" line being the only IPv6 give away.
>
> While IPv6 hasn't exactly taken the world by storm, it gets significant use in some (usually Asian) countries and most modern operating systems support it. To use Nmap with IPv6, both the source and target of your scan must be configured for IPv6. If your ISP (like most of them) does not allocate IPv6 addresses to you, free tunnel brokers are widely available and work fine with Nmap. I use the free IPv6 tunnel broker service at http://www.tunnelbroker.net. Other tunnel brokers are listed at Wikipedia. 6to4 tunnels are another popular, free approach.

6.8 IPv6 Attack/Pen Testing Tools?




```
[0x03] The Included Tools
       - parasite6: icmp neighbor solitication/advertisement spoofer, puts you
         as man-in-the-middle, same as ARP mitm (and parasite)
       - alive6: an effective alive scanng, which will detect all systems
         listening to this address
       - fake_router6: announce yourself as a router on the network, with the
         highest priority
       - redir6: redirect traffic to you intelligently (man-in-the-middle) with
         a clever icmp6 redirect spoofer
       - toobig6: mtu decreaser with the same intelligence as redir6
       - detect-new-ip6: detect new ip6 devices which join the network, you can
         run a script to automatically scan these systems etc.
       - dos-new-ip6: detect new ip6 devices and tell them that their chosen IP
         collides on the network (DOS).
       - fake_mld6: announce yourself in a multicast group of your choice on the net
       - fake_mipv6: steal a mobile IP to yours if IPSEC is not needed for authentication
       - fake_advertiser6: announce yourself on the network
       - smurf6: local smurfer
       - rsmurf6: remote smurfer, known to work only against linux at the moment
       - sendpees6: a tool by willdamn(ad)gmail.com, which generates a neighbor
       solicitation requests with a lot of CGAs (crypto stuff ;-) to keep the CPU busy. nice.
```

7. IPv6 Network Management

7.1 Network Device Monitoring

- Monitoring devices

  -- Are all your network devices accessible via IPv6 ssh for CLI-based management chores?

  -- Can you do TFTP (if you need to) over IPv6 transport?

  -- Do all those devices support SNMP over IPv6 transport?

  -- Do all the device MIBs know about IPv6 SNMP data structures? For example, can you query and get just IPv6 octets in/out each interface?

7.2 Netflow

- Netflow is a fundamental tool for monitoring and managing your network, however the version of netflow that most sites run, Netflow v5, doesn't know about IPv6.
- In a mixed IPv4/IPv6 environment, you need Netflow v9.

7.3 NTP (Network Time)

- Keeping closely synchronized time is another important network management (and system management function).

- For more information about NTP, see http://www.ntp.org/

- A list of NTP Stratum 2 servers is available at: http://support.ntp.org/bin/view/Servers/StratumTwoTimeServers

  Some of those servers appear to be available via IPv6...

Some IPv6 NTP Stratum 2 Servers

- 2001:12f0:4c1:6643::1
- 2001:12ff::8
- 2001:1418:1d7:1::1
- 2001:1470:ff80::12
- 2001:1af8:4201:1011::1
- 2001:41d0:1:ac7b::1
- 2001:4530:2::2:98
- 2001:4530:2::2:99
- 2001:470:1f04:a77::2
- 2001:470:9aad:123::1
- 2001:470:b407:123::1
- 2001:470:b409:123::1
- 2001:470:b8b9::2
- 2001:470:d:1c2:1:1:123:1
- 2001:638:504:2000::32
- 2001:638:504:2000::33
- 2001:6f8:128a:123::1
- 2001:6f8:1c00:3f::2
- 2001:738:5001::1
- 2001:7a8:810:260::c111:c0d3
- 2001:7b8:325:1::2
- 2001:828:100:2001:4::1
- 2002:58bf:b62::1
- 2002:8ac3:802d:1243::53
- 2801:82:0:1::2
- 2a01:238:4346:a200:cafe:babe:cafe:babe

8. IPv6 Programming

Porting IPv4 Code to IPv6

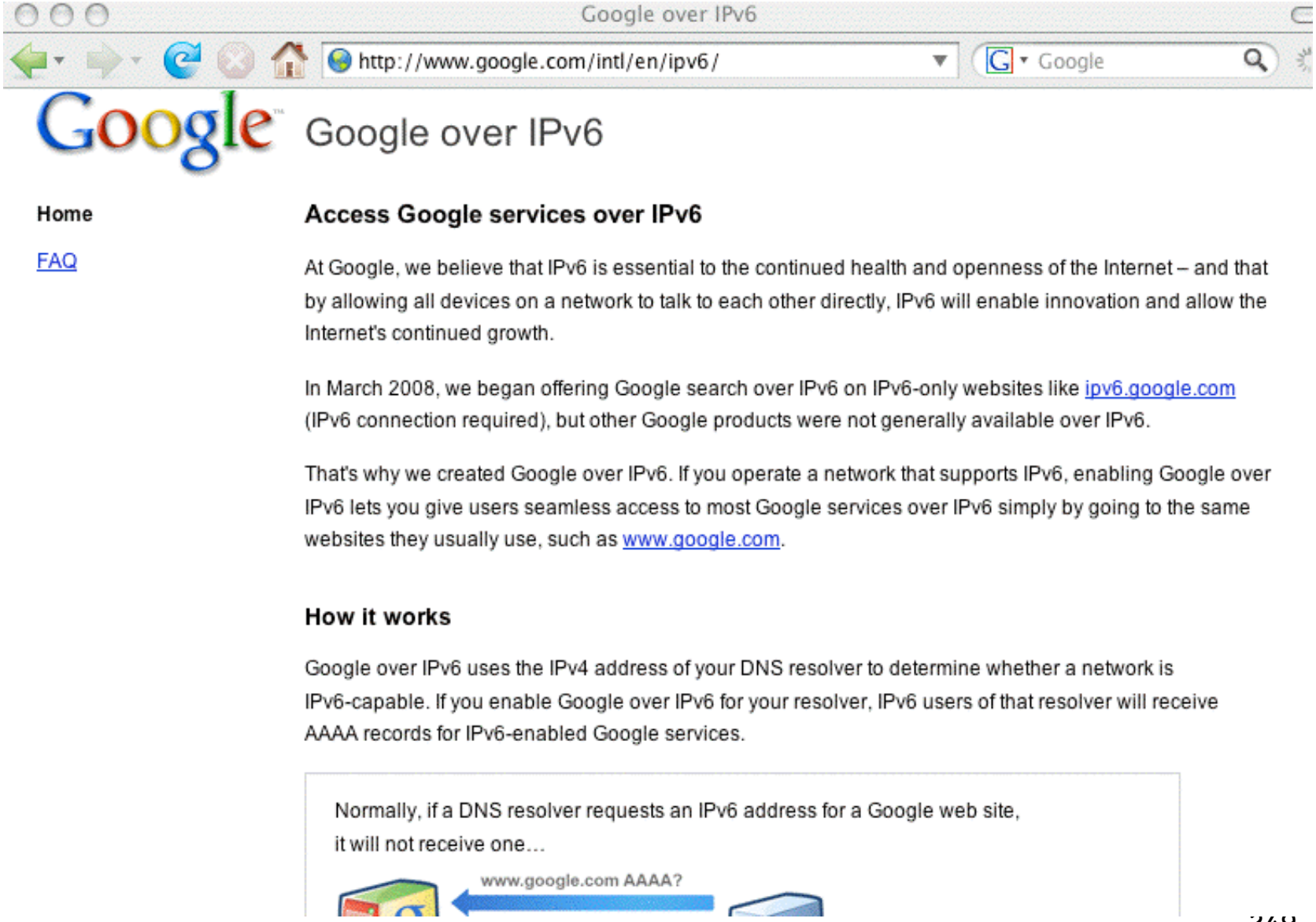
- The best advice I've found on porting IPv4 code to IPv6 is from http://owend.corp.he.net/ipv6/

  What other resources would people suggest?

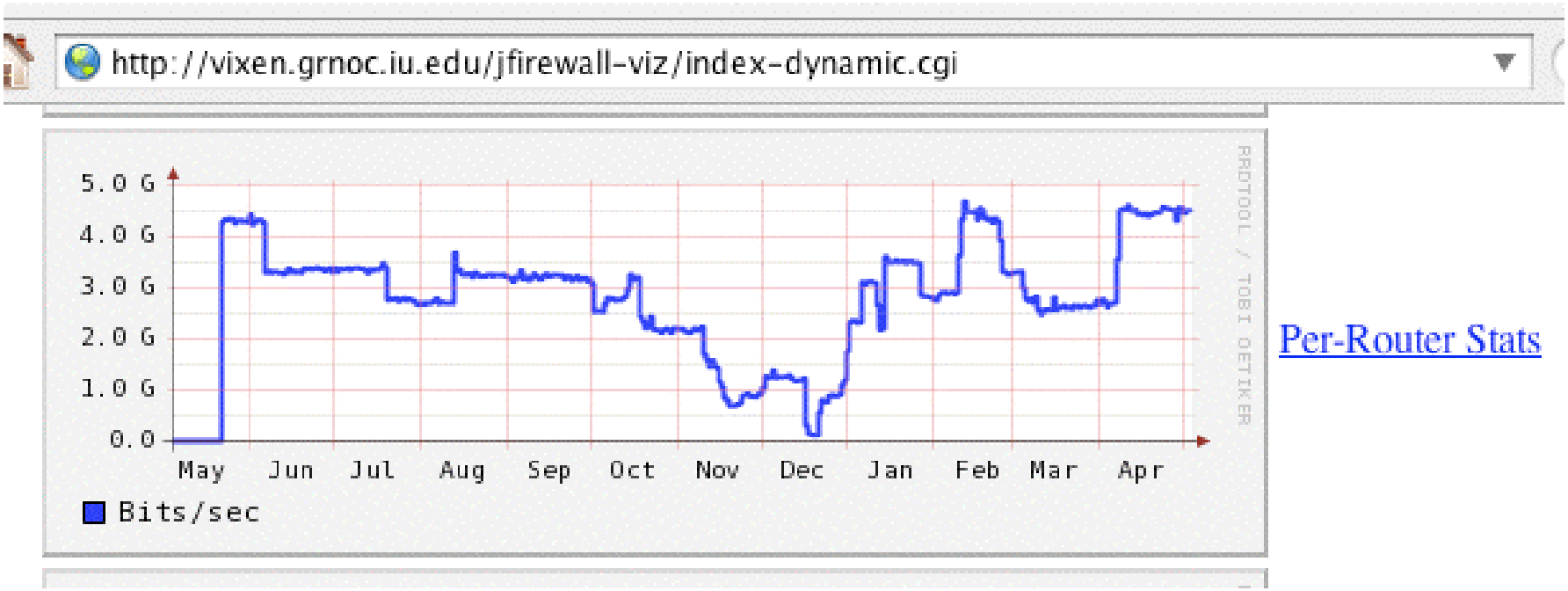
9. Who's Using IPv6?

Don't forget the list
of IPv6 web sites in Section 4.1!

Google Is Promoting Access to Google via IPv6



Internet2 Is Carrying Substantial IPv6 Traffic



Notes: This is over the last year and
the Y axis is from 0-5 Gbps

Source: http://vixen.grnoc.iu.edu/jfirewall-viz/index-dynamic.cgi

DREN Is *Widely* Using IPv6

DREN IPv6 Implementation Update

Internet2 Joint Techs, Winter 2010
2 Feb, 2010
Salt Lake City, UT

Ron Broersma
DREN Chief Engineer
High Performance Computing Modernization Program
ron@spawar.navy.mil

http://www.internet2.edu/presentations/jt2010feb/20100202-broersma.pdf

DREN Is *Widely* Using IPv6 (2)

Deployment progress

- ✔ WAN – dual stack everywhere, peering (unicast+multicast)
- ✔ LANs – all subnets fully support v6, renumber v4
- ✔ Infrastructure services – recursive DNS, NTP, SMTP, XMPP
- ✔ Support services – RADIUS, LDAP, Kerberos
- ✔ Public facing services – authoritative DNS, MX's, www, NTP
- ✔ Security "stack" – firewall, IDS, IPS, etc.

To Do: Get all the desktops, laptops, and servers running dual-stack

http://www.internet2.edu/presentations/jt2010feb/20100202-broersma.pdf

DREN Is *Widely* Using IPv6 (3)

Expanding internal IPv6 adoption

- Jan 2009 – only 5% of our systems (servers, desktops, laptops, etc.) were doing IPv6
  - Double from the year before
- Today: A major internal campaign has us now at 87.6%.
  - A totally volunteer and optional effort
  - We had to provide encouragement and incentives for over 500 independent projects and systems administrators

http://www.internet2.edu/presentations/jt2010feb/20100202-broersma.pdf

CERNET2 (China) Is IPv6 *ONLY*

Home

http://www.cernet2.edu.cn/en/char.htm

Google

Large scale Internet backbone over native IPv6

CNGI-CERNET2 is the largest Internet backbone over native IPv6 around the world, which is designed and implemented on the worldwide innovative concept of establishing large scale native IPv6 network. The success of CNGI-CERNET2 provides the solutions to the problems including the topologies design, routing design, and so on, and provides the environment for technique trials and applications demonstrations on China 's next generation Internet.

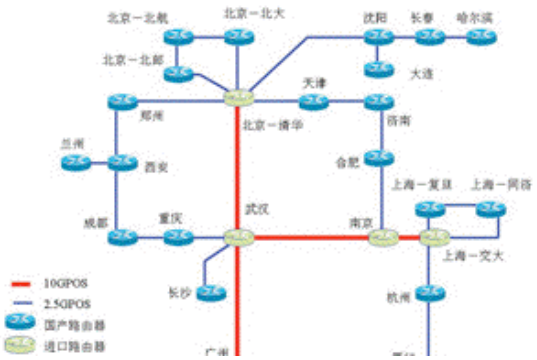
Importance of constructing native IPv6 network

The large amount of IPv6 technology research activities and experiments have been carried out, and the results indicate that the cost of the maintenance and management for IPv4/IPv6 dual stack network is pretty high, and the network itself is not secure. Moreover, it seems that the future network development can not indeed break away from IPv4's influence and there will be least substantial development of operational IPv6 network, if IPv4/IPv6 dual stack network is going on. Whereas it is a big global challenge to establish large scale Internet backbone over native IPv6.

Providing solutions to key technical problems

The success of CNGI-CERNET2 provides a series of the solutions to key technical problems of network engineering technology over native IPv6, including the topologies, routing design, IP address assignment and DNS registration, network testing and measurement and integrative network management, and so on.

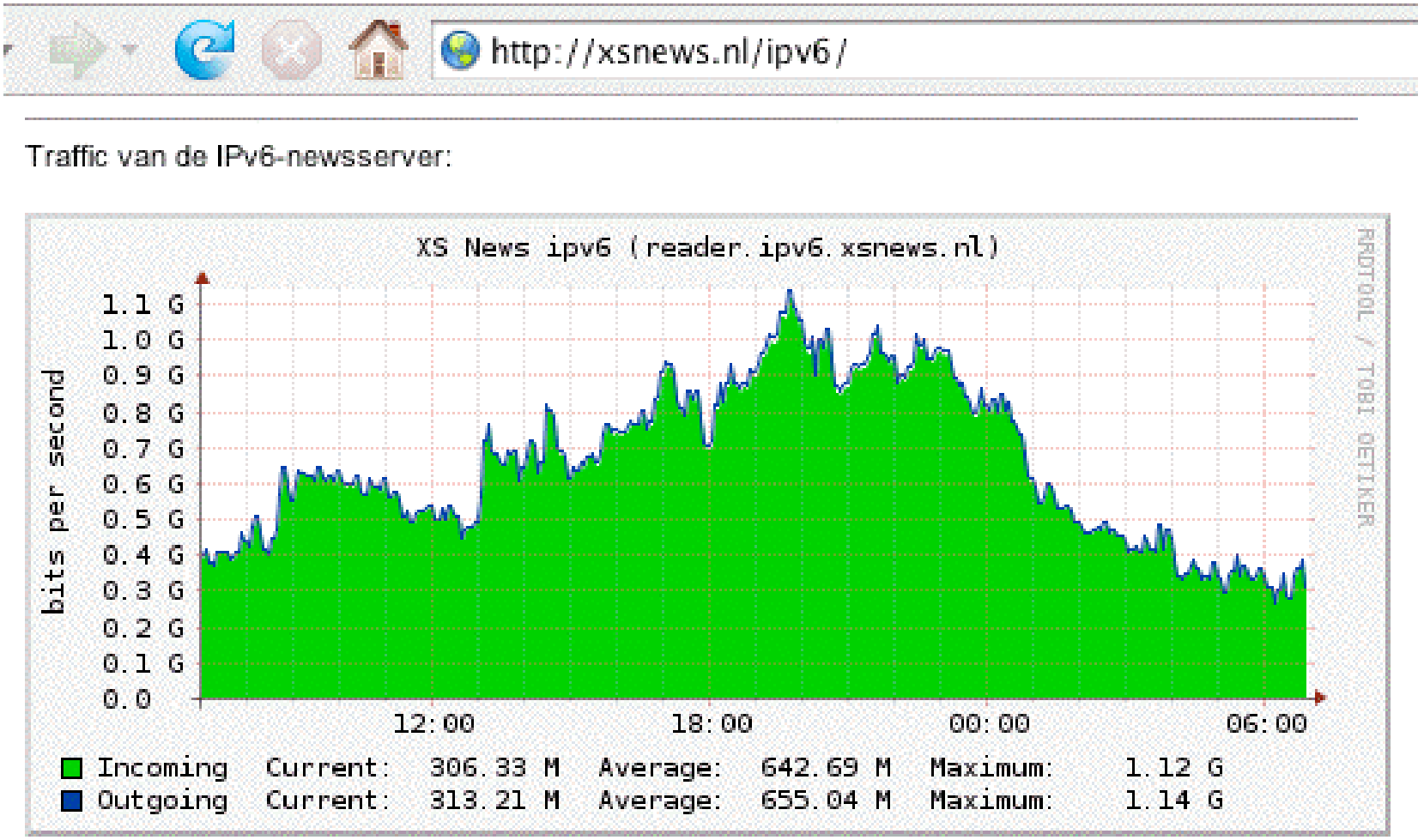
Multi-vendor IPv6 routers over large scale backbone



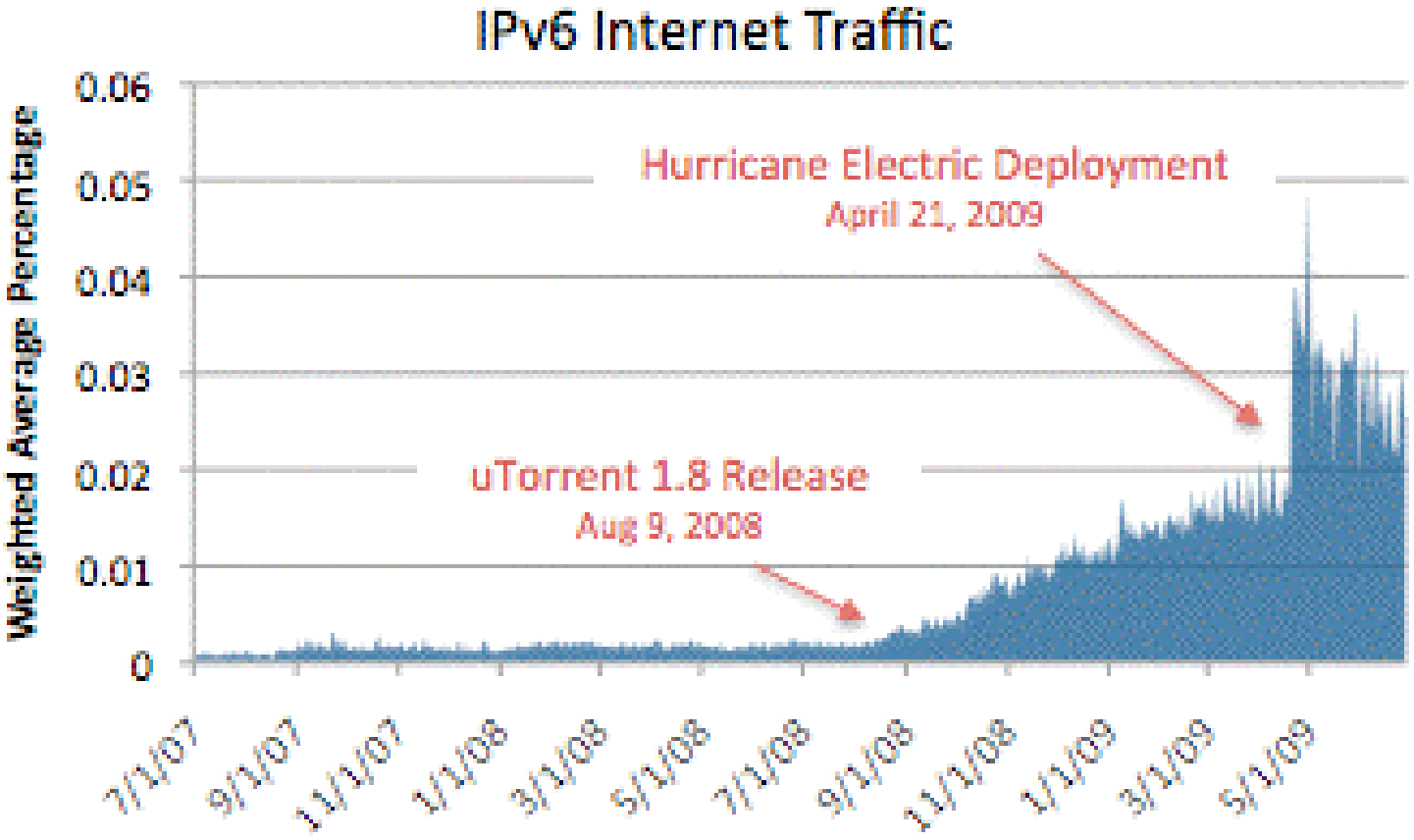
It was confirmed and implemented that the large scale Internet backbone over native IPv6 is built with multi-vendor IPv6 routers. This provides the solution to large scale inter-operability testing, and large scale inter-domain BGP routing, inter-operability and united network management between multi-vendor routers.

http://www.cernet2.edu.cn/en/char.htm

IPv6 Is Carrying A Lot of Usenet Traffic



IPv6 Is Also Being Used for P2P



See http://asert.arbornetworks.com/2009/09/who-put-the-ipv6-in-my-internet/

What About YOU?
You Should Be Getting Ready for IPv6!

- If you're not currently deploying IPv6 locally, or at least experimenting with IPv6 in a lab setting, the time has come for you to begin to do so.
- Deploying IPv6 support will be a team effort, so make sure your conversation includes:
  -- your network engineers
  -- your domain name server administrators
  -- your system administrators
  -- your support staff
  -- your security and abuse handling people
  -- your vendors
- Deployment can be incremental. You can take baby steps, you don't need to boil the ocean on day one.

Learning More About IPv6: Books

- A nice online reference: 6Net IPv6 Cookbook, www.6net.org/publications/deliverables/D3.1.2v2.pdf
- Traditional printed books about IPv6 are also available. If you go to Amazon's book section and search for IPv6, for example, you get 809 hits. That's a bit better than in the old days. :-)
- When considering which of those books might work for you, recognize that some are written for specific audiences (like programmers), and those sort of books may not meet your particular needs (unless you're a coder and you're trying to come up to speed for IPv6).
- Also recognize that IPv6 is rapidly evolving, so beware of any books that haven't been recently updated.

Thanks For the Chance To Talk!

- Are there any questions?