



JANET

IPv6 Hands-on Workshop

Module 2: **Systems and Services**

UKERNA, Lancaster University
and University of Southampton, 2006



Module Overview

- Brief introduction to IPv6 on end systems
- How to v6-enable core services
 - DNS
 - Email
 - Web
- (Very) brief introduction to APIs and application porting
- Network management and monitoring
 - Connectivity and traffic monitoring tools
 - Address and configuration management with DHCPv6
- Candidate road-map of how services could be enabled back home

Perspective on services?

- In considering IPv6 enabling various services, what is our perspective?
 - At Southampton, it was to create a network environment in which IPv4 and IPv6 nodes could exist
 - Thus we aim to offer key services (DNS, SMTP, etc) over either protocol, so nodes can use either
 - Complements the dual-stack network (routing) deployment
- A dual-stack services deployment does add some complexity, but as we'll see many services work well 'out of the box' for either protocol
 - This allows an early production deployment of some IPv6 services
 - If that's the approach you want to take

IPv6 status

- IPv6 has pervaded the IETF working groups
 - e.g. ipv6, v6ops, shim6, dnsop, dnsex, mip6, dhc, etc.
 - Statement at 61st IETF in Washington in 2005 to the effect of:
“We should not mention ‘IPv6’, we should just mention ‘IP’”
 - Final push for base protocol set to Internet Standard (i.e. send the message that “It’s done”)
- Ubiquitous in host platforms:
 - Solaris, Windows XP, Mac OS/X, FreeBSD, Linux, ...
- Becoming ubiquitous in router platforms:
 - Cisco, Juniper, 3Com, Hitachi, Extreme, FreeBSD, ...

*BSD/KAME

- Present by default in *BSD (“merged-in” support)
 - FreeBSD 4.0+, OpenBSD 2.7+, NetBSD 1.5+, BSD/OS 4.2+
 - Excellent and mature platform for host or router
 - As a PC-based router, can use GNU/Zebra on top of KAME kit
- Can add KAME “snap” kit for latest features
 - Available from www.kame.net
 - Look under `/usr/local/v6` or in `/usr/local/v6/etc/rc.net6`
 - IPv6-enabled by default
 - Includes IPv6 transport DNS lookups
- Includes multicast code
 - PIM-SM and MLD
 - PIM-SSM and MLDv2 (in more recent patches)
 - Being used on the m6bone for multicast
 - see <http://www.m6bone.net> and also <http://www.multicast.org.uk>

Solaris

- Sun Microsystems' Solaris 8 and onwards has IPv6 built-in
 - Prototype package for Solaris 7 in 1999
 - Solaris 9 added IPv6 IPsec, Tunnels over IPv6, IPv6oATM, X11 over IPv6, ...
 - Solaris 10/Express then added 6to4, RFC3484, Privacy addresses, and refined the sockets API
- Installing and Enabling
 - As an option during installation, or post-configure through config file munging and a reboot
 - By default, everything is stateless auto-configuration
 - Just touch `/etc/hostname6.<interface>` and you're off
- Files to watch for:
 - `/etc/hostname6.<interface>` existence - IPv6 auto-configuration no boot if so
 - `/etc/inet/ipnodes` - should contain at least 127.0.0.1 and `::1` (both 'localhost')
 - `/etc/nsswitch.conf` - should have an entry for "ipnodes: files dns"
 - `/etc/inet/ndpd.conf` - should be commented out if node is a host and not a router

Linux

- Out of the box support in most modern Linux distributions
 - Good features in
 - RedHat 8+ and the Fedora Core releases (e.g. keep IPv6 as module)
 - SUSE 8+
 - Modern Debians
 - Linux MIPv6 code available from MIPL project (<http://www.mobile-ipv6.org/>)
 - Kernel patch with userland tools ('current' for linux-2.6.16)
- USAGI Linux
 - Effort to improve original SGI “netdev” stack in mainstream kernel
 - Principally a collaboration between WIDE, KAME and TAHI
- See Peter Bieringer’s HOWTO
 - Application and services readiness status
 - <http://www.tldp.org/HOWTO/Linux+IPv6-HOWTO/>

Windows 2000/XP

- IPv6 is supported by Microsoft as of Windows XP
 - A “hotfix” is available for Windows 2000
 - Significant functionality improvements and tighter integration service pack series
 - Administration through the **netsh** utility
- To install, just run
 - ‘ipv6 install’ at a command prompt (pre XP SP1)
 - Add ‘Microsoft IPv6 Developer Edition’ component as a new protocol in the Network Connections Control Panel pane (XP SP1)
 - Add ‘Microsoft TCP/IP version 6’ as a new protocol in the Network Connections Control Panel pane (XP SP2)
- Then
 - ‘ipv6 if’ to see IPv6 features (on Windows 2000 or XP pre-SP1)
 - Use “netsh” utility, particularly the **netsh interface ipv6** context
 - Stack includes 6to4, Teredo and ISATAP transition tools
 - Also includes RFC3041 privacy extensions, enabled by default

Windows Vista

- Due 'soon'
- IPv6 included and enabled by default, including:
 - IPsec features
 - MLDv2 (IPv6 source specific multicast)
 - Teredo for tunneling IPv6 through IPv4 NATs
 - DHCPv6 client
 - IPv6 over PPP
- More info here:
 - <http://www.microsoft.com/technet/community/columns/cableguy/cg1005.msp>
- Check also Windows sockets API updates:
 - http://msdn.microsoft.com/library/default.asp?url=/library/en-us/winsock/winsock/ipv6_guide_for_windows_sockets_applications_2.asp

IPv6-enabling core services

- Goal of enabling all IPv4 services over IPv6
- Tools are there for the standards-based services:
 - DNS
 - Mail Transfer Agents
 - Mail Retrieval and User Agents
 - Web servers
 - “The Rest”
 - Database and Directories
 - Access
 - ...

DNS (2)

- The reverse DNS tree is delegated in a similar manner to IPv4, e.g.

```
0.63.78.152.in-addr.arpa      IN NS    ns0.ecs.soton.ac.uk.  
7.0.d.0.0.0.3.6.0.1.0.0.2.ip6.arpa  IN NS    ns0.ecs.soton.ac.uk.
```

- Managing long nibble-strings can be a bind
 - ISC BIND, for example, has \$ORIGIN syntactic sugar
 - Suggest to use \$ORIGIN throughout each subnet's DNS population
 - e.g. set \$ORIGIN, populate PTR records for a subnet, set \$ORIGIN for next subnet, populate...
 - Alternative is to use a zone per subnet and manage the nibble-strings in the name server configuration file

DNS Migration

- A dual-stack network suggests four entries per node
 - 2 forward under the same label (A, AAAA)
 - 1 reverse PTR under in-addr.arpa
 - 1 reverse PTR under ip6.arpa
- Many sites have scripted their DNS population
 - These scripts will need to be updated
 - All the 'corner cases' and local hacks, e.g. for special projects etc., will need to be re-evaluated and catered for
- Look out for unexpected readdressing issues!
 - Stateless Automatic Address Configuration uses layer 2 identifier to form the node's layer 3 address
 - So changing your NIC will mean your SLAAC address will change!

DNS IPv6 transport

- IPv6 transport on the query path
 - ... available on some of the root servers, trend toward all improving
 - ... available for .uk (Nominet) and .ac.uk (JANET NOSC)
 - Most name servers are not reachable over IPv6 transport yet
- Insufficient glue
 - Of the root servers that do have IPv6 transport, some do not return authoritative AAAA records when querying for “. NS” (i.e. the roots)
 - Well-behaved servers are only supposed to return address data in additional answers sections for domains in which they are authoritative

DNS IPv6 Transport (2)

- Just because you have IPv6 DNS transport locally, and your target authority has IPv6 DNS transport, doesn't mean your queries will be IPv6 all the way!
 - Iterative resolvers may require IPv4 from your first-hop name server
 - Recursive resolvers may require dual-stack between peers on the query path
- IPv6-only hosts will need the help of a dual-stack name server for resolving
 - Protocol Translation is possible solution but not recommended
 - This may be as simple as putting the IPv6 address of recursive dual-stack name server in resolv.conf

DNS – named.conf for BIND9

- Simple addition to enable transport
 - listen-on-v6 { any; };
 - “transfer-source-v6 *” to specify IPv6 source address for transfers
 - “query-source-v6 address * port *” to specify IPv6 source for queries
 - IPv6 addresses can be used in ACLs, e.g. to restrict zone transfers
- The rest of the server config is as for IPv4
- With a zone for each /64 you might have e.g.

```
zone “0.0.1.7.0.d.0.0.0.3.6.0.1.0.0.2.ip6.arpa” (  
    type master;          file “zones/0.0.1.7.0.d.0.0.0.3.6.0.1.0.0.2.ip6.arpa”;  
);
```
- If want to support ip6.int and don't have \$ORIGIN in the zone data:

```
zone “0.0.1.7.0.d.0.0.0.3.6.0.1.0.0.2.ip6.int” (  
    type master;          file “ zones/0.0.1.7.0.d.0.0.0.3.6.0.1.0.0.2.ip6.arpa”;  
);
```

IPv6 Mail Transfer

- MTA configuration
 - Add AAAA entry for MX servers in DNS
 - Configure MTA to listen on IPv6 sockets
 - Remember to adapt filter and relay rules in the MTA!
 - Sending host can then choose IPv4 or IPv6
- Caveat: no production RBLs with IPv6 literals as yet
 - All spam filtering should be done locally

Received: from tyholt.uninett.no ([IPv6:2001:700:1:4::1:0])

by jackdaw.ecs.soton.ac.uk (8.12.10/8.12.10) with ESMTP id j2NBH3ix000857

for <tjc@ecs.soton.ac.uk>; Wed, 23 Mar 2005 11:17:03 GMT

Received: from storhaugen.uninett.no (storhaugen.uninett.no [IPv6:2001:700:e000:0:290:27ff:fe22:7186])

by tyholt.uninett.no (8.12.10/8.12.10) with ESMTP id j2NBH3LL019094

for <tjc@ecs.soton.ac.uk>; Wed, 23 Mar 2005 12:17:03 +0100

MTA and DNS

- RFC3974 discusses various MX ordering options where MTAs are of varied connectivity
 - The simplest, and most desirable case, is where all MXes are dual stack for as long as there is IPv4 around
- Ideally all MXes would have A and AAAA in DNS

<code>example.ac.uk</code>	<code>IN MX 1</code>	<code>a.mx.example.ac.uk</code>
<code>example.ac.uk</code>	<code>IN MX 2</code>	
<code>b.mx.example.ac.uk</code>		
<code>a.mx.example.ac.uk</code>	<code>IN A</code>	<code>192.0.2.1</code>
	<code>IN AAAA</code>	<code>2001:db8:0:1::25</code>
<code>b.mx.example.ac.uk</code>	<code>IN A</code>	<code>192.0.2.129</code>
	<code>IN AAAA</code>	<code>2001:db8:0:8::25</code>

MTA = sendmail (1)

- Sendmail 8.10+ has IPv6 support
- In sendmail.mc, add a second listener for IPv6

```
DAEMON_OPTIONS(`Name=IPv4, Family=inet')dnl  
DAEMON_OPTIONS(`Name=IPv6, Family=inet6')dnl
```

- On linux, may need to hint that INET6 family is available at compile-time

```
APPENDDEF(`confENVDEF', `-DNETINET6')dnl
```

- Some (pre-BIND 9) servers mistakenly return SERVFAIL
 - AAAA lookup on labels that do have A records bound
 - Causes mail queuing and eventual expiry
 - Workaround in 8.12.1+ is query A first, but still prefer AAAA results
ResolverOptions=WorkAroundBrokenAAAA

MTA = sendmail (2)

- All MTA features that are IP-related feature for IPv6 as they do IPv4
- HOWEVER, 'syntactic sugar' required in config files
 - Prefix IPv6 address literals with "IPv6:"

e.g. in access file:

```
152.78.63          REJECT We don't accept mail from you
IPv6:2001:630:d0:7000  REJECT No relaying from your lab
```

Other MTAs and IPv6

- Exim
 - Support since version 4.30
 - HAVE_IPV6=yes in Local/Makefile at build-time
 - Change config literal separator from ':' to, e.g. ';'
 - Otherwise parsing literal addresses is tricky
- Postfix - since 2.2
- qmail - only with unsupported patches

```
local_interfaces = <; 127.0.0.1 ; \  
192.0.2.3 ; \  
::1 ; \  
2001:db8:1::25
```

Mail Retrieval and User Agents

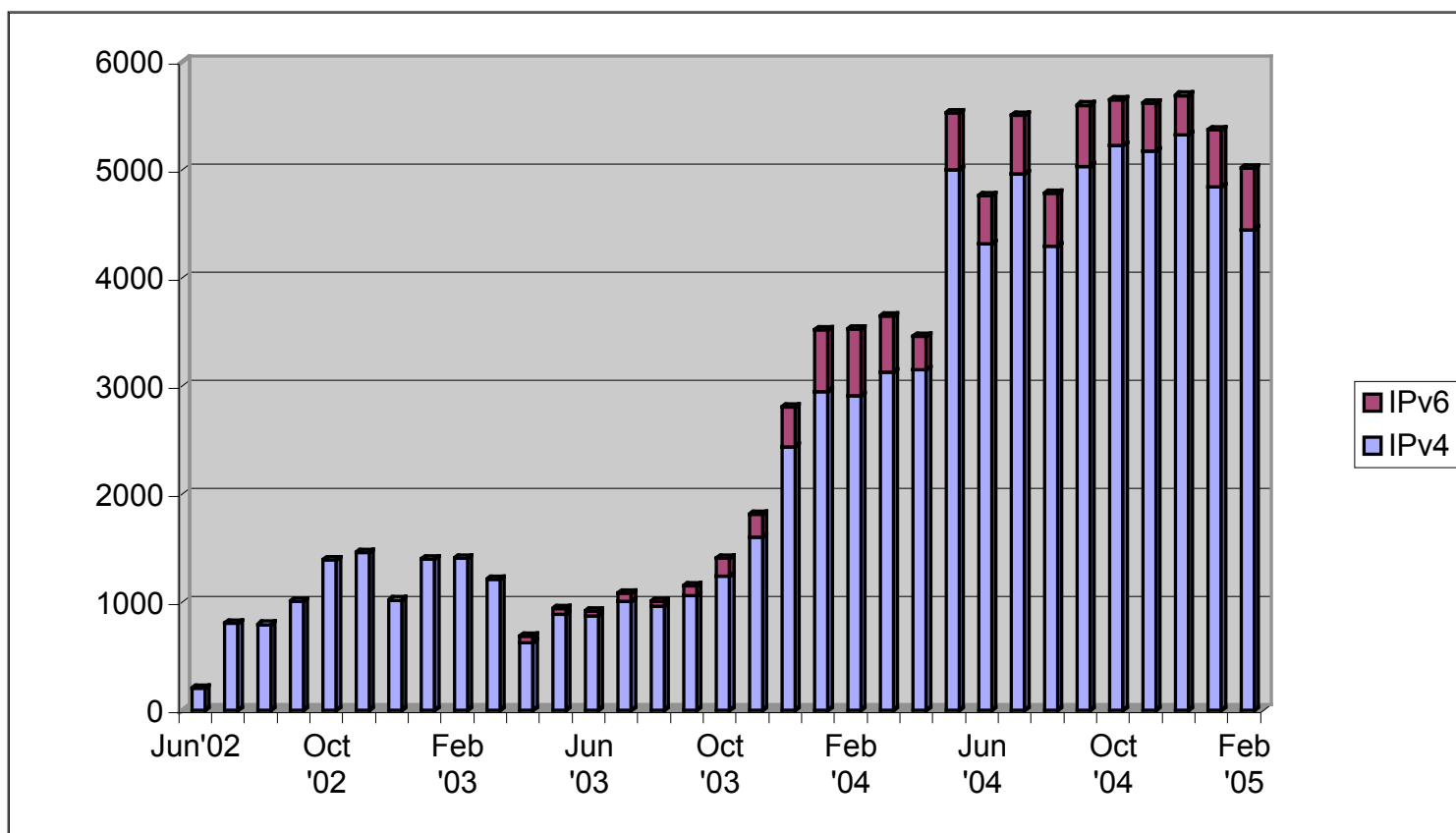
- Retrieval
 - Courier has native support in POP3 and IMAP4 daemons, since version 0.42.2
 - Cyrus IMAP has native support since 2.2.1-BETA, with 3rd party patch efforts on 2.1.15+ branch
 - No MS Exchange (yet)
- MUA
 - mutt, Mozilla Mail, KDE KMail and Ximian Evolution all have native IPv6 support from the maintainers
 - No support in Outlook or Outlook Express yet

Apache Web Server

- IPv6 code enabled by default in Apache 2
 - Both packaged and source-build versions
 - Patches available for Apache 1.3.11 through 1.3.37
 - BUT - the ASF recommendation is to migrate to Apache 2 regardless
- ‘Listen’ directive determines socket behaviour
 - With the build-time configure option “--enable-v4-mapped”
Listen 80 - Single socket for both (IPv4 mapped-addresses)
 - With the build-time configure option “--disable-v4-mapped”
Listen [::]:80 - will accept IPv6-only (unspecified address)
Listen 0.0.0.0:80 - will accept IPv4-only
 - --disable-v4-mapped is the default on *BSD platforms
- Access controls as per IPv4, even with address literals
 - **Allow from 2001:db8:1::/52**
Allow from 192.0.2.0/24

IPv6 web serving - An observation

- Snapshot of stats for www.ist-ipv6.org, a dual-stack web server that was hosted here at Southampton:



DHCPv6

- Several vendors working on DHCP but only a few implementations available at the moment
 - Cisco IOS <http://www.cisco.com/>
 - DHCPv6 <http://dhcpv6.sourceforge.net/> (old/ceased)
 - Dibbler <http://klub.com.pl/dhcpv6/> (work now restarted)
 - ISC DHCPv6 To be released under BSD license in 2007
 - Other vendors are working on their own implementations
- Prefix Delegation (RFC 3633)
 - Use DHCPv6 to configure downstream access router's IPv6 network prefix
 - Downstream router then apportions prefix to subnets/other routers
 - Helps to automate CPE router provisioning, etc.

Network Time Protocol - NTP

- NTP server and client available from <http://www.ntp.org/>
 - IPv6 addresses can be used all places where IPv4 is used, except for reference clock addresses which are always IPv4
 - -4 or -6 can specified in front of hostnames to force IPv4 or IPv6
 - Can also use IPv6 multicast
 - IANA has reserved the site-scope multicast address ff05::101
- The IPv6 enabled RIPE Test Traffic Measurement (TTM) nodes can also be used as IPv6 NTP servers
- Other commercial offerings exist, e.g. Meinberg NTP server

NNTP : Usenet News

- INN has native support since version 2.4 as a build-time configure option
- Currently 'mostly' there:
 - `innd` and `inndstart`, `auth_pass`, `nnrpd`, `innfeed`, and the `ident` auth program
 - but no support in `imapfeed` or other auxilliary tools (e.g. RADIUS auth)
- Configuration directives that refer to address literals have IPv6 counterparts
 - e.g. `bindaddress` = `bindaddress6`

Some other services

- Remote login access
 - OpenSSH was one of the first to offer native IPv6 support
 - Version 3.6.1p2 is perhaps as old as one would like to go
 - USAGI project has rtools and telnet daemons for linux
 - Solaris rtools and telnet daemons are 6-capable where the OS is
- Directories
 - OpenLDAP has native support in versions 2.0+
- Web proxies
 - squid requires 3rd party patch (from KAME project)
 - wwwoffle has native support
- IRC has support built in
- Instant Messaging
 - AIM, MSN, YIM, etc. are IPv4-only
 - Jabber has native support
 - jabberd can bridge to other IM protocols, so v6-only clients can talk to AIM etc.
- ... increasingly a case of “it just works out of the box” (thankfully!)

IPv6 APIs - C

- Use Berkeley Sockets API
 - RFC3493 basic extensions: a new socket address structure to carry IPv6 addresses, new address conversion functions, and some new socket options
 - RFC3542 advanced API: RAW sockets, direct header access, improvements for backwards compatible code
 - Coding practice the same, but with slight API changes
 - e.g., for AF/IP-independent code use
 - getaddrinfo() – map host name to address
 - getnameinfo() – map address to host name
 - See
 - <http://www.kame.net/newsletter/19980604>
 - http://www.sun.com/software/solaris/ipv6/porting_guide_ipv6.pdf
 - <http://jungla.dit.upm.es/~ecastro/IPv6-web/ipv6.html>
- Use Windows Sockets API
 - Older IPv6 socket APIs currently supported (RFC and RFC)
 - See http://msdn.microsoft.com/library/default.asp?url=/library/en-us/winsock/winsock/ipv6_guide_for_windows_sockets_applications_2.asp

IPv6 APIs - Java

- JDK 1.4.2 (Java 2 Standard Edition) supports IPv6
 - Basic Sockets API, URL Literal manipulation, but no Advanced Sockets (Java doesn't do raw sockets for IPv4, either)
 - Solaris 8+ and Linux kernels 2.1.2+ supported in JDK 1.4.2
 - System properties control JVM stack behaviour

```
java.net.preferIPv4Stack=<true|false>
java.net.preferIPv6Addresses=<true|false>
```
 - Class hierarchy changed so that common InetAddress has two children, one for each protocol family
 - The rest of the API remains unaffected thanks to the object-oriented abstractions
- JDK 1.5.0 improves the support further
 - Proper Win32 support
 - See http://java.sun.com/j2se/1.5.0/docs/guide/net/ipv6_guide/

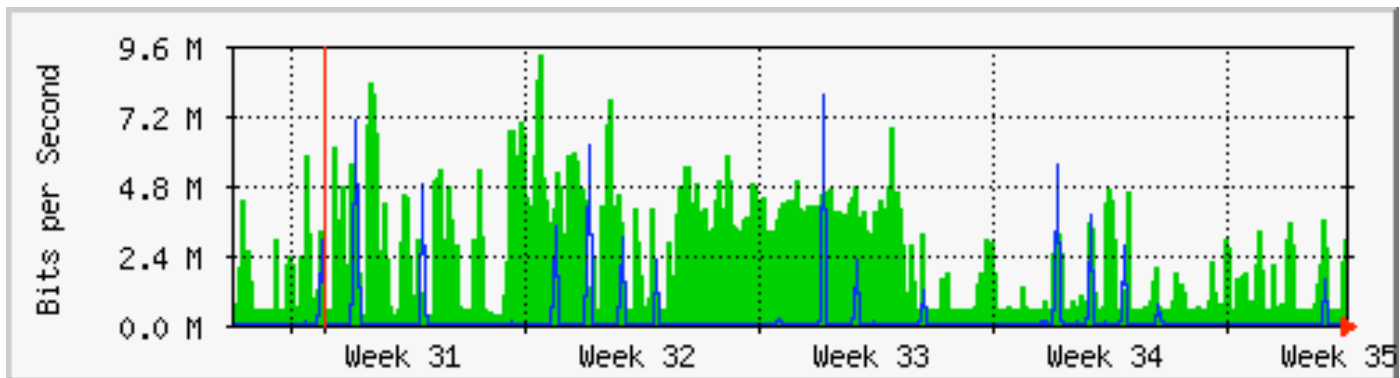
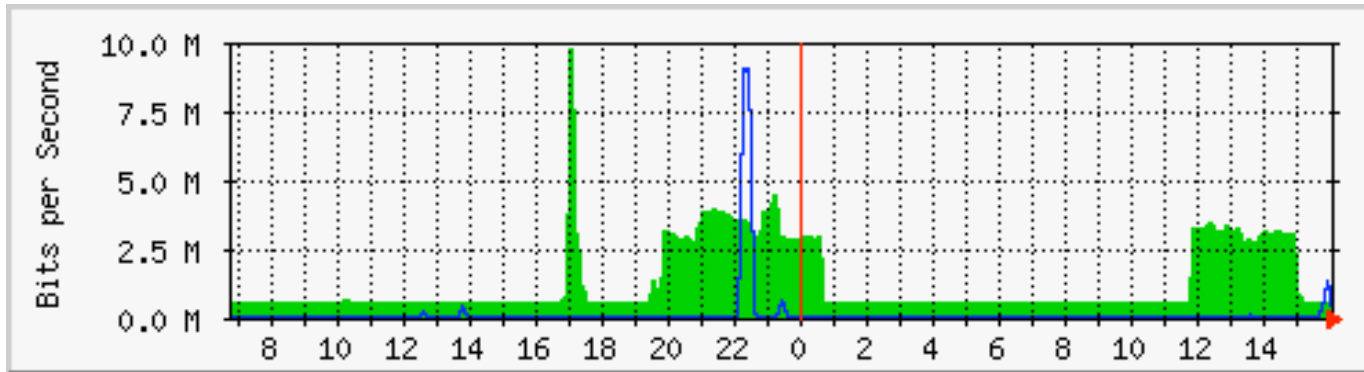
Adopter-inspired developments

- Various streaming tools
 - ECS-TV
 - <http://www.zepler.tv/>
(locally scoped IPv6 multicast, using videolan client package)
 - Surge
 - <http://www.ipv6.ecs.soton.ac.uk/virginradio/>
(re-broadcast - with permission - digital radio using icecast)
 - <http://www.surgeradio.co.uk/listen/advanced.html>
(University radio station, includes IPv6 multicast)
- Conferencing tool ports (e.g. Gnomemeeting, OpenH323)
- Globus-based weather station system
- IPv6 Multicast-based peer-to-peer game engine
- Network IDS utility
 - ... inspired by new technology deployment and inclusion in teaching

Network Management and Monitoring

- Over 40 tools used in the EU FWK5 project, 6NET
 - See <http://tools.6net.org/> for a list and comments on respective IPv6 capabilities and caveats
- SNMP
 - MIBs still being finalised
 - Limited implementations support the emerging MIBs
 - Need expect-style scripts to get IPv6-specific data for dual-stack interfaces, e.g. to prime MRTG

MRTG: IPv6 traffic levels



External IPv6 traffic at Soton-ECS: top is for a recent 24 hour period, bottom is a 1 month view

Test and diagnosis tools

- IPv6 versions of IPv4 familiar diagnostic tools are stable
 - ping (ping6 on some OSes)
 - traceroute (traceroute6 on some OSes)
 - Ethereal, tcpdump and other libpcap-based applications
- For exterior routing diagnoses
 - ASpathtree display graphically the BGP4+ routing paths managed by the Cisco/Juniper/Zebra routers of a backbone
 - Looking glass, e.g. JANET's IPv6 looking glass
- RIPE NCC supports other IPv6 tools
 - See their IPv6 WG pages: <http://www.ripe.net/ripe/wg/ipv6/>
- Most route views will look similar to those you know for IPv4

Firewalls and Packet Filters

- Few commercial options
 - Checkpoint Firewall-1 NG with Application Intelligence
 - Nokia IP380
 - ... examples with IPv6-Ready certification
- IOS access lists - same principle as IPv4
 - Using 'ipv6 access-list' command, e.g.

```
ipv6 access-list name permit tcp \
2001:0db8:0300:0201::/64 eq 22
```
 - Also can be a subcommand, e.g.

```
ipv6 access-list name
permit tcp 2001:db8:300:201::/64 eq 22
deny 2001:db8:0:2::/64 any
```

Firewalls and Packet Filters

- JUNOS access lists – same principle as IPv4
 - The JUNOS policy framework combines routing policies with firewall filters to control the flow of traffic through the router.
 - Using ‘set firewall family inet6 filter’ command, e.g

```
[edit]
```

```
    set firewall family filter inet6 filter control-ssh term  
    name from \
```

```
        next-header tcp destination-port ssh address  
        2001:0db8:0300:0201::/64
```

```
    set firewall family filter inet6 filter control-ssh term  
    name then accept
```

- As a sub command from the firewall filter level, e.g.

```
[edit firewall family inet6 filter control-ssh]
```

```
    set term name from next-header tcp destination-port ssh
```

```
    set term name from address 2001:0db8:0300:0201::/64
```

```
    set term name then accept
```

Firewalls and Packet Filters

- BSD pf
 - IPv6 and IPv4 addresses can be used interchangeably

```
pass out quick on $if proto tcp from any to \ 2001:db8::22
port ssh keep state
```
 - Resolves hostnames into all addresses at load-time only
- Linux (ip6tables)
 - Handled separately - iptables for IPv4, ip6tables for IPv6
 - Identical syntax, MANGLE and FILTER tables

```
ip6tables -A FORWARD -d 2001:db8::22 -p tcp --dport 22 \
-i eth0 -j ACCEPT
```
 - ip6tables lagging behind, e.g. connection/state tracking, have to match on SYN flags for NEW and ESTABLISHED, not RELATED

Firewalls and Packet Filters

- Labels v. Literals
 - In all service configurations, it may make sense to use labels where possible and rely on DNS resolution at invocation time to affect the relevant protocol (v4 or v6)
 - BUT for firewalls, this may be particularly hazardous
- Common firewall rules that are useful
 - Permitting IP-41, so that IPv6-in-IPv4 and 6to4 tunnels work
 - Not filtering ICMPv6 as harshly as ICMP, so that Path MTU discovery works (mandated for all IPv6 nodes)
 - See draft-ietf-v6ops-icmpv6-filtering-recs-02

Intrusion Detection

- Need to be able to detect:
 - Attacks in the application space, e.g. web server exploits
 - These are IP independent
 - Look for same patterns whether over IPv4 or IPv6 transport
 - Specific DoS type attacks reliant on the IP version
 - e.g. Maliciously crafted IPv6 hop-by-hop options
- IPv6 IDS components will be desirable
- Snort is the de facto standard for open source IDS
 - But no official IPv6 snort (yet)

QoS

- Differentiated services (per-hop)
 - There is the 8-bit Type of Service (ToS) byte in IPv4
 - In IPv6, there is the 8-bit Traffic Class field
 - Can use same Diffserv Code Points (DSCPs) across IP versions
 - JANET currently working towards production DiffServ QoS
- Flow Label field
 - RFC3697
 - Its use currently remains undefined
 - By default, set the Flow Label field to zero if unused
 - In principle it could be used to distinguish distinct IPv6 flows

A lot to consider!

- This session has contained a lot of specific examples, some of which may not be relevant to you
 - The aim was to illustrate that IPv6-enabled services can be deployed now, if you wish, alongside IPv4 services
- At Southampton our IPv6-enabled services include:
 - DNS (all three servers)
 - SMTP (all external MX relays)
 - NTP (two servers)
 - Web (including core presence: www.ecs.soton.ac.uk)
 - Firewalls (BSD pf, now moving to integrated platform)
 - All network routing on common hardware
- And to date we've not broken existing IPv4 services doing so

Summary

- Very general introduction to status of various OS implementations
- Sample set of core services and the requirements to 'sixify' them
- (Very) brief introduction to APIs and application porting
- Pointers toward network monitoring and diagnosis tools
- Filtering and firewalling considerations
- Introduction to stateful configuration services

- Next up: Services hands-on lab...