



IPv6 experience

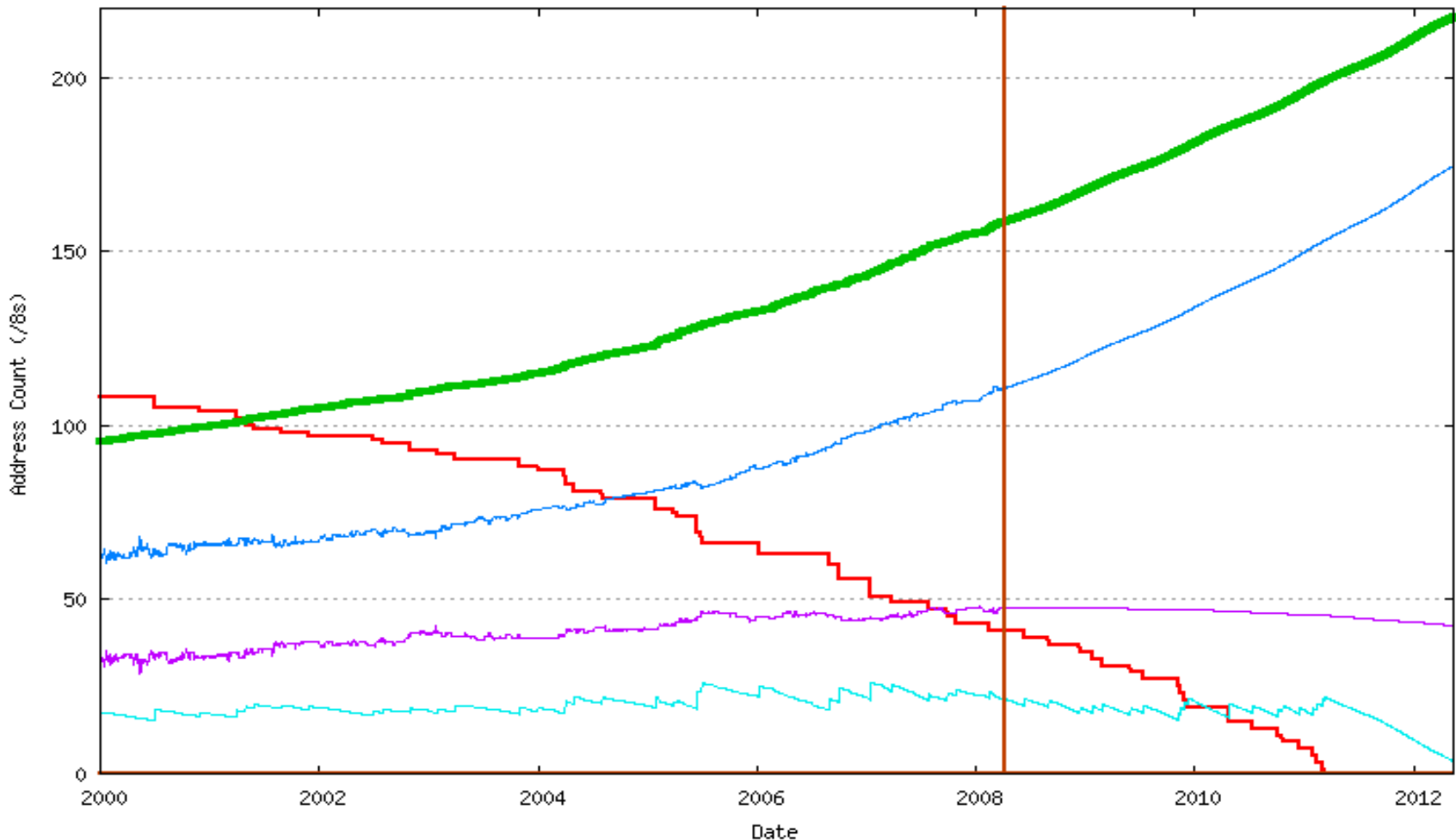
Mattias Wadenstein
System Integrator, NDGF
HEPiX Spring 2008 at CERN
Geneva, 2008-05-05

- What is IPv6
- Why should we care?
- The testbed
- The experience
 - What worked
 - What needed work
- Conclusions

- Same, same
 - Just like IPv4 (which we all use)
 - “The stuff underneath TCP”
 - What you get when you do a dns lookup on a hostname

- But different..
 - 128-bit instead of 32-bit
 - 2001:6b0:e:2018:21d:e0ff:fe38:c561 vs 130.239.18.137
 - Requires different syscalls for low-level (C, etc) programming (getaddrinfo instead of gethostbyname, etc)
 - Different DNS resource records (A for IPv4 AAAA for IPv6, reverse zones also different)

- From Geoff Huston (<http://ipv4.potaroo.net>)



- Even if most of us have plenty of IPv4
 - Prestige, status, technical excellence, fun, etc
 - User laptop on a conference only getting a IPv6 address (and a broken webproxy or NAT for IPv4 content)
 - Political/funding reasons (IPv6 might be an important checkbox item)
 - IPv4 address space might become worth \$\$\$, without IPv6 you can't sell off unused space

- NDGF not really involved
- But HPC2N kind of is
- The real work done at the Academic Computer Club at Umeå University
 - Some overlap of HPC2N and ACC sysadmins
 - As well as overlap with ACC and CS dept
- Trying it out in a not-quite-production environment before rolling it out

- ACC services
- Multiuser machines and workstations
 - AIX, Solaris, Linux (Ubuntu 6.06/7.04/7.10/8.04)
 - A bit less than 1k users
- Public services
 - Free software mirror <ftp.acc.umu.se>
 - Debian, Ubuntu, Mozilla, Gnome, Gimp – 1Gbit/s avg
 - IRC servers (Freenode, OFTC, GIMPnet)
- Web, mail, dns, etc, etc

- We don't do networking
 - “Can you enable IPv6 on this segment?” - “OK”
 - This means I can't tell you how to setup routers
- The big decision
 - IPv6 has autoconfiguration based on NIC MAC
 - Or static allocation
 - Or dhcpv6
- Dual-stack everything – without breakage
 - We chose static allocation

- OS support
 - But see “what needed work” for details
- Mainstream software
- Most “odd” software
- Assuming that you run versions that aren't ancient
 - Our webserver is still on apache 1.3...

- DNS
 - Local zone management software needed to be extended to understand IPv6 for both forward and reverse zones
 - Tell bind to listen to the appropriate IPv6 interface, by default it seems to just listen on the IPv4 interface
- Log parsing scripts
 - Or anything else that needs to identify an IP address

- Some odd software
 - In this case not so much, but some software still can't handle a IPv6 address
 - This probably applies to lots of grid/HEP/HPC software though
 - Starting early makes it possible to enable it system by system, and not be rushed if some systems can't be dual-stacked quickly

- AIX sometimes forgets its IPv6 interface
 - Might be a NIC driver bug (non-IBM card)
- Linux NFS only speaks IPv4
 - Not a problem as long as you dual-stack, it will just use IPv4 instead
- Routing problems
 - Only noticed by a small minority, but really annoying when you want to have production-like availability of your services
 - Spontaneous loss of default route

- Static IPv6 addressing on Linux
 - The Linux kernel is very eager to do `addrconf`
 - You can turn it off by `sysctl`
 - But!
 - The `sysctl` is only accessible once the `ipv6` module is loaded and the interface is up
 - And at that point, Linux is doing the `addrconf..`,

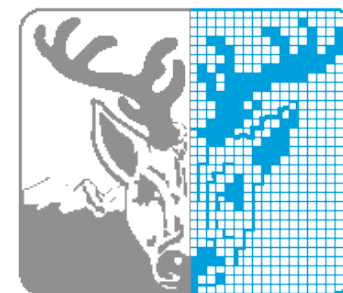
- The solution, hack /etc/network/interfaces:

```
iface eth0 inet6 static
pre-up modprobe ipv6
up /sbin/sysctl -q -w net.ipv6.conf.eth0.autoconf=0
address 2001:6b0:e:2018::137
netmask 64
```

?



ACC



HPC2N