A Case Study of IPv6 Deployment in tcd.ie

David Malone

<dwmalone@{maths.tcd,cnri.dit}.ie>

22 May 2003

What is IPv6?

1. Current version of IP is version 4.
2. IPv6 is an evolution of IPv4.
3. Not backwards or forwards compatible.

Major changes

- Bigger addresses.
- Better extensibility.
- Built in autoconfig.
- Mandatory IPsec.
- More integrated multicast.

Addresses

IPv4 addresses: 32 bit `134.226.81.11` IPv6 addresses:

- 128 bit addresses `340282366920938463463374607431768211456`,
- Written in 8 hex quads,
- Several shortcuts allowed.

Examples

- 2001:0770:0010:0300:0000:0000:86e2:510b
- 2001:770:10:300:0:0:86e2:510b
- 2001:770:10:300::86e2:510b
- 2001:770:10:300::134.226.81.11

Special Addresses

::	Unspec
::1	localhost
fe80:: block	link-local
fec0:: block	site-local
ff00:: block	multicast

Stage 1: Getting Started

```
% ifconfig -a
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
        inet6 ::1 prefixlen 128
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x1
        inet 127.0.0.1 netmask 0xff000000
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        inet6 fe80::203:93ff:fe46:17a6%en0 prefixlen 64 scopeid 0x4
        inet 10.0.0.1 netmask 0xff000000 broadcast 10.255.255.255
        ether 00:03:93:46:17:a6
```

Pinging

```
% ping6 ::1
PING6(56=40+8+8 bytes) ::1 --> ::1
16 bytes from ::1, icmp_seq=0 hlim=64 time=0.392 ms
% ping6 -I en0 fe80::230:65ff:fe03:d972
16 bytes from fe80::230:65ff:fe03:d972%en0, icmp_seq=0 hlim=64 time=1.373 ms
% ping6 -I en0 ff02::1
PING ff02::1(ff02::1) from fe80::2b0:d0ff:fed7:741d en0: 56 data bytes
64 bytes from ::1: icmp_seq=1 ttl=64 time=0.062 ms
64 bytes from fe80::2b0:d0ff:fe05:fc06: icmp_seq=1 ttl=64 time=0.194 ms (DUP!)
64 bytes from fe80::206:5bff:fe68:249b: icmp_seq=1 ttl=64 time=0.224 ms (DUP!)
64 bytes from fe80::202:b3ff:fe65:604b: icmp_seq=1 ttl=64 time=0.256 ms (DUP!)
64 bytes from fe80::2b0:d0ff:fef4:c6c5: icmp_seq=1 ttl=64 time=0.334 ms (DUP!)
64 bytes from fe80::203:93ff:fe46:17a6: icmp_seq=1 ttl=64 time=0.384 ms (DUP!)
...
```

Enabling

FreeBSD Add `ipv6_enable="YES"` to `/etc/rc.conf`

Redhat Add `NETWORKING_IPV6="yes"` to `/etc/sysconfig/network`

Solaris Create `/etc/hostname6.`*`ifname`*.

Stage 2: IPv6 connectivity

- Use tunnel.
- Protocol 41.
- 2001:618:400:e::/64 address space from BT.

```
% ifconfig gif0 create
% ifconfig gif0 tunnel 134.226.10.51 193.113.58.80
% ifconfig gif0 inet6 2001:618:400::1:DC4:1467 2001:618:400::1:DC4:1466 prefixlen 128
% route add -inet6 default 2001:618:400::1:DC4:1466
```

Rtadvd

- Routers drive autoconfiguration.
- Configure prefix 2001:618:400:e:: get address 2001:618:400:e:2a0:c9ff:feb1:34e7
- Use `rtadvd fxp0` on FreeBSD.
- Router tells nodes prefix, basic routes, lifetimes, MTU...

Problems with routing

- Scenic routing between cs.tcd.ie and maths.tcd.ie.
- Got /48 from BT.
- Tunnel BT to CS, Tunnel CS to Maths.
- 2001:618:409:100:: —- ISS, 2001:618:409:200:: — CS, 2001:618:409::300 — maths.
- 2001:618:409:400:: — 2001:618:409:ffff:: are reserved.

HEAnet

- Initially testing with 3ffe:8037: addresses, tunnel to maths.
- Shortly after HEAnet allocated 2001:770::.
- TCD allocated 2001:770:10:, old addressing plan renumbered.
- All set, what now?

Stage 3: Services

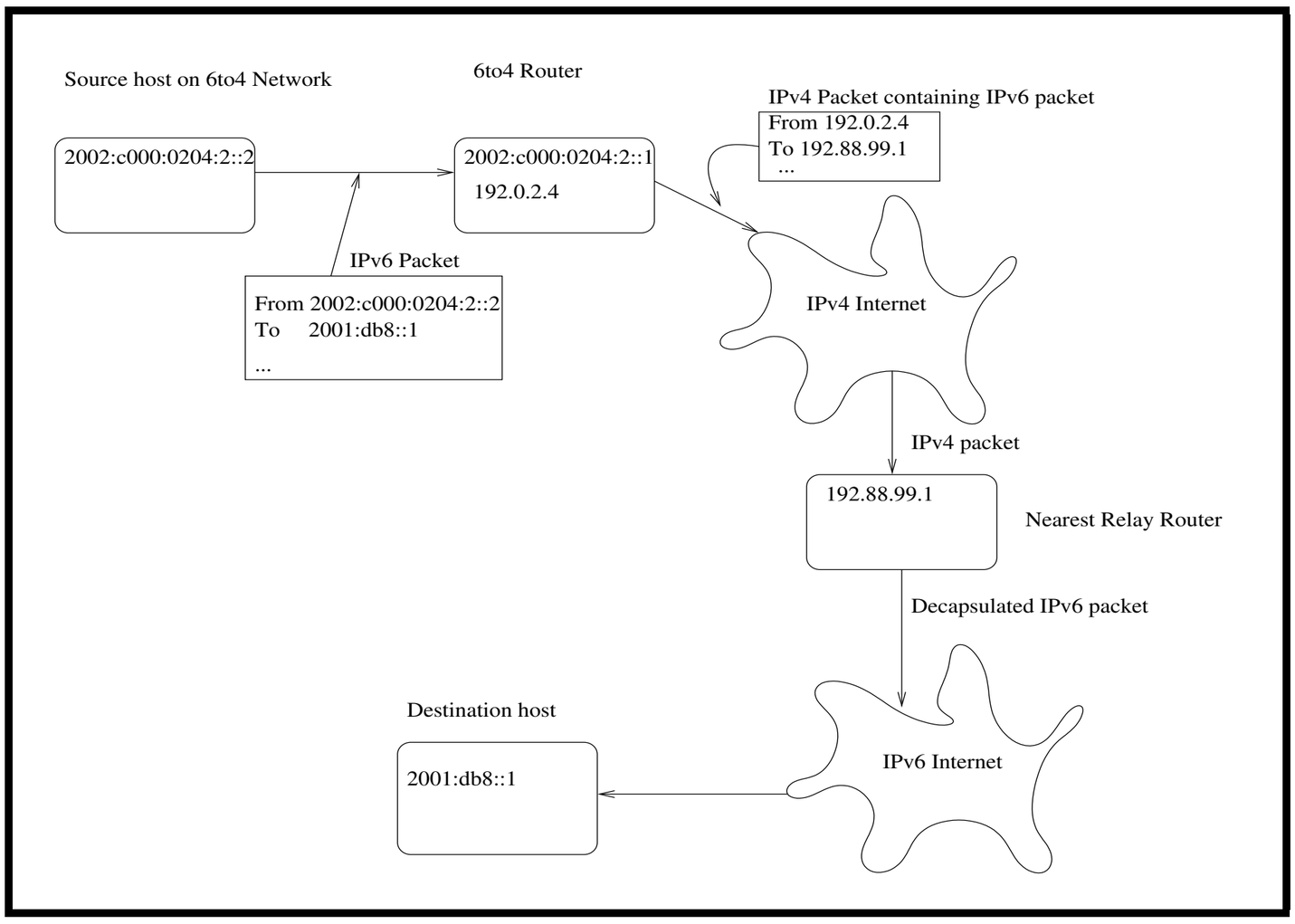
- OpenSSH has good IPv6 support.
- To be useful need DNS info:
- A record maps salmon →134.226.81.11.
- PTR record maps 11.81.226.134.in-addr.arpa → salmon
- AAAA record maps salmon.ipv6 → 2001:770:10:300::...
- PTR record maps ...0.7.7.0.1.0.0.2.ip6.arpa → salmon.ip6

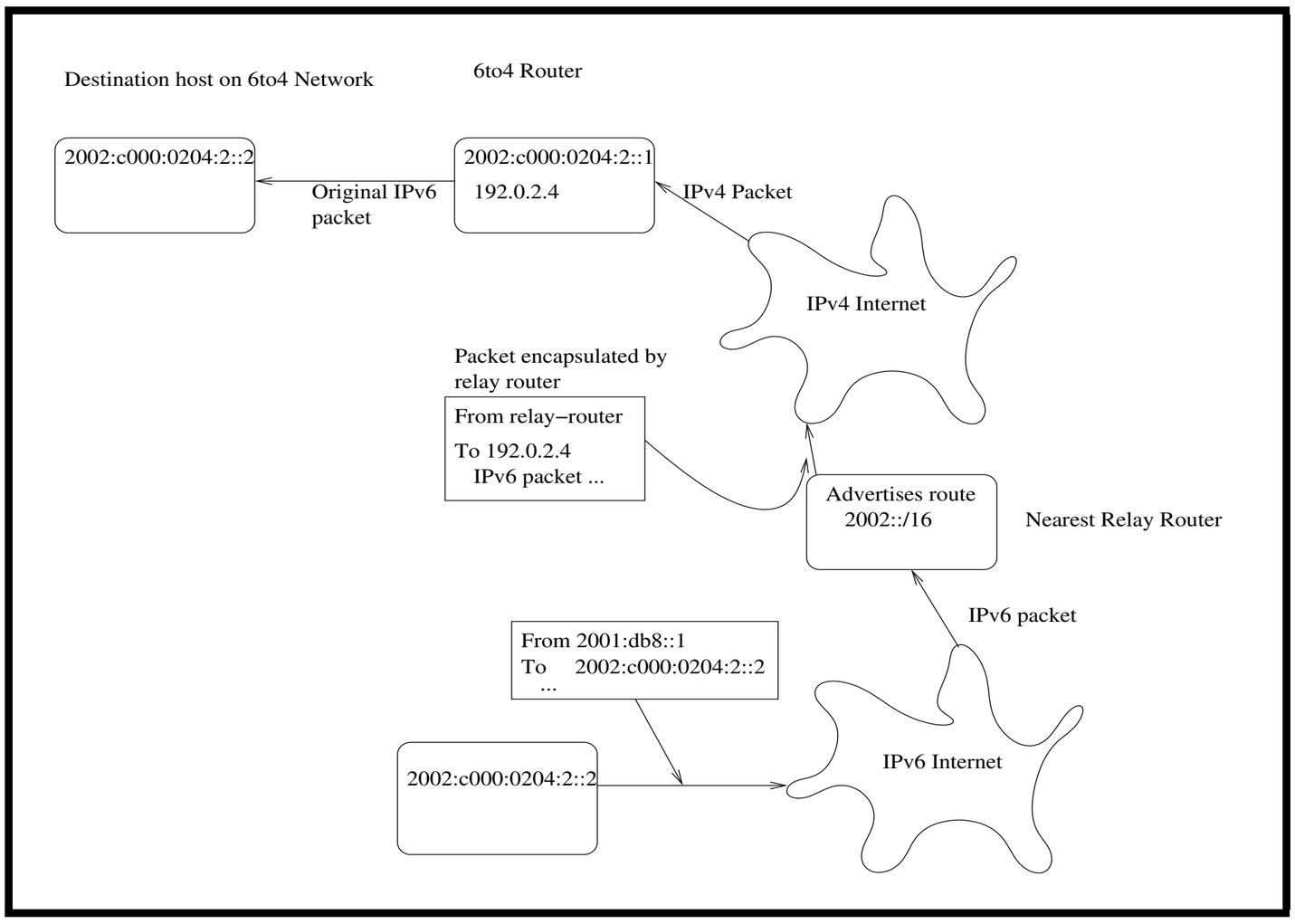
Testing

- After updating known hosts, seems good.
- Further tests to before AAAA for normal names.
- Get IPv6 at home and set default domain to ipv6.maths.tcd.ie.
- All still good, add AAAA for normal names.

Connecting at home

- 6to4 — very easy, easy config.
- www.xxx.yyy.zzz → 2002:WWXX:YYZZ::
- Eircom and HEAnet offer relays.
- Do ask your ISP.


Source host on 6to4 Network
6to4 Router
IPv4 Packet containing IPv6 packet
From 192.0.2.4
To 192.88.99.1
...
2002:c000:0204:2::2
2002:c000:0204:2::1
192.0.2.4
IPv6 Packet
From 2002:c000:0204:2::2
To 2001:db8::1
...
IPv4 Internet
IPv4 packet
192.88.99.1
Nearest Relay Router
Decapsulated IPv6 packet
Destination host
2001:db8::1
IPv6 Internet


Destination host on 6to4 Network
6to4 Router
2002:c000:0204:2::2
Original IPv6 packet
2002:c000:0204:2::1
192.0.2.4
IPv4 Packet
IPv4 Internet
Packet encapsulated by relay router
From relay−router
To 192.0.2.4
IPv6 packet ...
Advertises route
2002::/16
Nearest Relay Router
IPv6 packet
From 2001:db8::1
To 2002:c000:0204:2::2
...
2002:c000:0204:2::2
IPv6 Internet

6to4 Script for FreeBSD

```
#!/bin/sh

IPV4=$1
PARTS=‘echo $IPV4 | tr . ’ ’‘
PREFIX48=‘printf "2002:%02x%02x:%02x%02x" $PARTS‘

STF_IF="stf0"
STF_NET6="$PREFIX48":0000
STF_IP6="$STF_NET6"::1

ifconfig $STF_IF inet6 $STF_IP6 prefixlen 16 alias
route add -inet6 default 2002:c058:6301::
```

Or set `stf_interface_ipv4addr` in `rc.conf`

6to4 Script for Linux

```
#!/bin/bash

IPV4=$1
PARTS=‘echo $IPV4 | tr . ’ ’‘
PREFIX48=‘printf "2002:%02x%02x:%02x%02x" $PARTS‘

STF_NET6="$PREFIX48":0000
STF_IP6="$STF_NET6"::1

/sbin/ip tunnel add tun6to4 mode sit remote any local $IPV4
/sbin/ip link set dev tun6to4 up
/sbin/ip -6 addr add $STF_IP6/16 dev tun6to4
/sbin/ip -6 route add 2000::/3 via ::192.88.99.1 dev tun6to4
```

Or set `IPV6TO4INIT` in `/etc/sysconfig/network-scripts/ifcfg-`*`if`*.

Web Services

Upgrade to Apache2

- Add IPv6 addrs to `.htaccess` files.
- Update log processing scripts.
- PHP problems? Run as CGI.

DNS & Bind 9

Having AAAA records and doing DNS over v6 different.

named.conf:

```
+       listen-on { any; };
+       listen-on-v6 { any; };
        query-source address * port 53;
+       query-source-v6 address * port 53;
```

zone file:

```
 ; Master nameserver for maths.tcd.ie.  This must be an A record to be used
 ; in any NS records.
 ns             IN      A       134.226.81.11
+               IN      AAAA    2001:770:10:300::86e2:510b
```

Mail

IPv6 support in sendmail, postfix (patch), qmail (patch), exim.

Maths use MMDF — challenge by Dave Wilson.

- Receive mail from network.
- Look up addressees for MX.
- Send mail to network.

Converting Apps

- Sockets API pretty agnostic:
  `s/AF_INET/AF_INET6/`
- Need to look up A and AAAA records.
- New functions getaddrinfo and getnameinfo.

```
struct addrinfo hints, *res, *res0;
int s;
memset(&hints, 0, sizeof(hints));
hints.ai_family = PF_UNSPEC;
hints.ai_socktype = SOCK_STREAM;
getaddrinfo("www.kame.net", "http", &hints, &res0);
for (res = res0; res; res = res->ai_next) {
        s = socket(res->ai_family, res->ai_socktype,
            res->ai_protocol);
        if (connect(s, res->ai_addr, res->ai_addrlen) < 0)
                continue;
        break;
}
freeaddrinfo(res0);
```

Meanwhile

- Other services: NNTP, FTP, ident.
- Growing number of autoconf machines show up (OS X, Linux, *BSD).
- TCD to HEAnet via Ethernet. Second FreeBSD router for native link.
- Protocol redundancy during SQL slammer.

Stage 4: Future

- Paul Reilly working on www.tcd.ie (PHP OK?)
- Use FreeBSD vlans support to connect all TCD vlans.
- Need to IPv6 finger, ntp and *web proxy*. (Bad doubleclick).
- Internal services: NFS, X11, phone, samba, ...