

Michael Sinatra

Forcing the Issue

Motivations for IPv6

- If we believe the scenarios about what will happen after IPv4 free-pool run-out, the resultant horse-trading for IPv4 space will lead to massive fragmentation and IPv4 “haves” and “have-nots.”
 - Dale Finkelson: We should endeavor to make our resources available to everyone, including those in developing countries.
 - We do not want to provide only for the haves.

Motivations for IPv6

- Forcing the Issue:
 - Many believe that the time is now to begin the serious transition to IPv6. (ARIN, ICANN, IETF, etc.)
 - Vendors still sometimes say “we’d support (more) v6 features, but nobody seems to be using it.
 - We felt that it was time to stop talking and start doing and see what a larger-scale v6 implementation really requires.
- Our own address space supply was of lesser importance (but still a factor).

Application service example

- A small scale XMPP server running ejabberd.
- Only two users; largely experimental.
- UCB does plan a wider-scale XMPP service.

ejabberd

```
{listen,  
  [{5222, ejabberd_c2s, [inet6, {access, c2s}, {shaper,  
    c2s_shaper}, {max_stanza_size, 65536},  
    starttls_require, {certfile,  
      "/usr/local/etc/ejabberd/ssl.pem"}]}],  
  
  [...]
```

ejabberd

```
{listen,  
  [{5222, ejabberd_c2s, [inet6, {access, c2s}, {shaper,  
    c2s_shaper}, {max_stanza_size, 65536},  
    starttls_require, {certfile,  
      "/usr/local/etc/ejabberd/ssl.pem"}]}],  
  
  {5222, ejabberd_c2s, [{access, c2s}, {shaper,  
    c2s_shaper}, {max_stanza_size, 65536},  
    starttls_require, {certfile,  
      "/usr/local/etc/ejabberd/ssl.pem"}]}],  
  [...]
```

- Does not work!

ejabberd

```
{listen,  
  [{15222, ejabberd_c2s, [inet6, {access, c2s}, {shaper,  
    c2s_shaper}, {max_stanza_size, 65536},  
    starttls_require, {certfile,  
      "/usr/local/etc/ejabberd/ssl.pem"}]}],  
  
  {5222, ejabberd_c2s, [{access, c2s}, {shaper,  
    c2s_shaper}, {max_stanza_size, 65536},  
    starttls_require, {certfile,  
      "/usr/local/etc/ejabberd/ssl.pem"}]}],  
  [...]
```

ejabberd - pf.conf

```
rdr pass on dc0 proto tcp from any to \  
    2607:f140:1:8001::5454 port 5222 -> \  
    2607:f140:1:8001::5454 port 15222  
  
rdr pass on dc0 proto tcp from any to \  
    2607:f140:1:8001::5454 port 5223 -> \  
    2607:f140:1:8001::5454 port 15223  
  
rdr pass on dc0 proto tcp from any to \  
    2607:f140:1:8001::5454 port 5269 -> \  
    2607:f140:1:8001::5454 port 15269
```

DNS stuff

- For many of us, DNS was the first--after routing--thing we got going on our experimental IPv6 networks.
- But we didn't always operationalize it in the same way as we did with our IPv4 networks.
- Even integrating IPv6 DNS so that we could support it in the same way as IPv4 posed some challenges.
 - Time to bite the bullet and get OSPFv3 running for our anycast configuration.
 - Anycasting the IPv6 nameservice alongside the IPv4 service on existing servers worked out pretty well.
 - Then we tried to register the new service with EDUCAUSE.

DNS stuff

```
;; ADDITIONAL SECTION:
```

```
ns.v6.berkeley.edu.      172800  IN      A        128.32.136.6  
ns.v6.berkeley.edu.      172800  IN      AAAA     2607:f140:ffff:fffe::6
```

Error

Your request cannot be processed for the following reason(s):

* According to RFC2373, section 2.2 part 1, each part of an IPv6 IP address must contain at least one numeral (with the exception of reductions). The IP address you entered, (2607:F140:FFFF:FFFE::6), does not meet this criteria.

* According to RFC2373, section 2.2 part 1, each part of an IPv6 IP address must contain at least one numeral (with the exception of reductions). The IP address you entered, (2607:F140:FFFF:FFFE::6), does not meet this criteria.

DNS stuff

```
foo.berkeley.edu.  IN      AAAA   2607:F140:0000:0000:0000:0000:0000:0006
```

Error

Your request cannot be processed for the following reason(s):

* The IP address you entered (2607:F140:0000:0000:0000:0000:0000:0006) is in a restricted IP address range. Please use a different IP address.

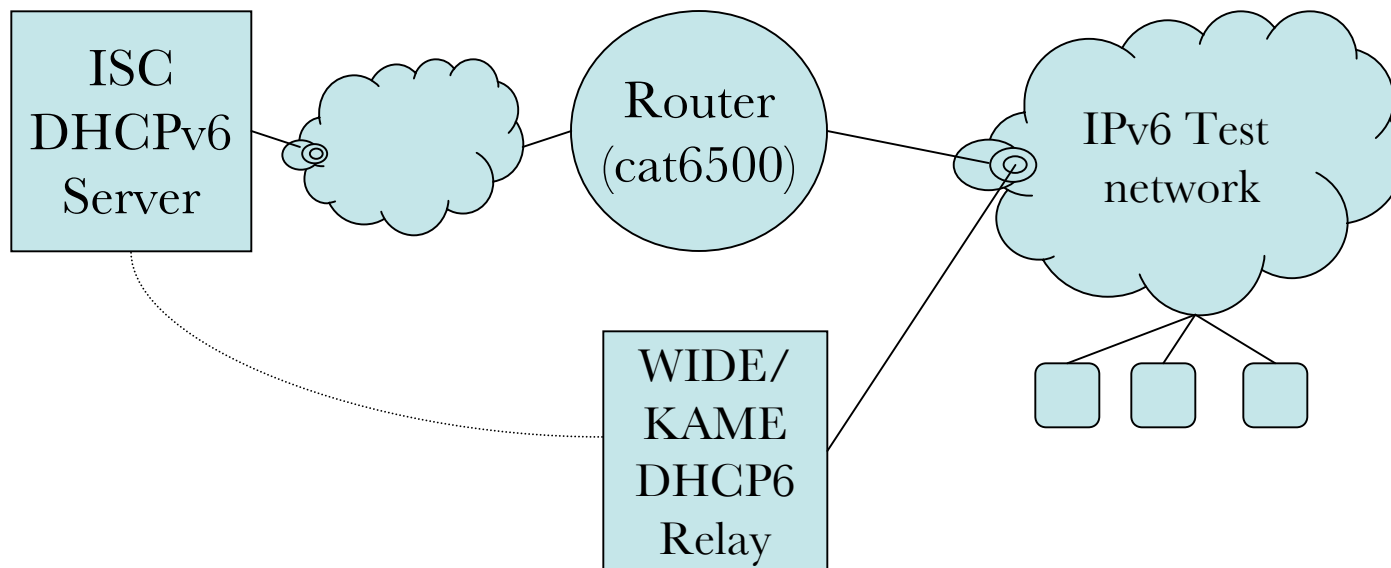
- The phrase "restricted IP address" is a hyperlink, and when I click on that, I get:

The .edu registry can only accept IPv6 addresses that are in the Aggregatable Global Unicast Addresses space. Please see <http://www.iana.net/assignments/ipv6-address-space> for more information on the IPv6 address space. This document is based on information from RFC2373 and RFC2374.

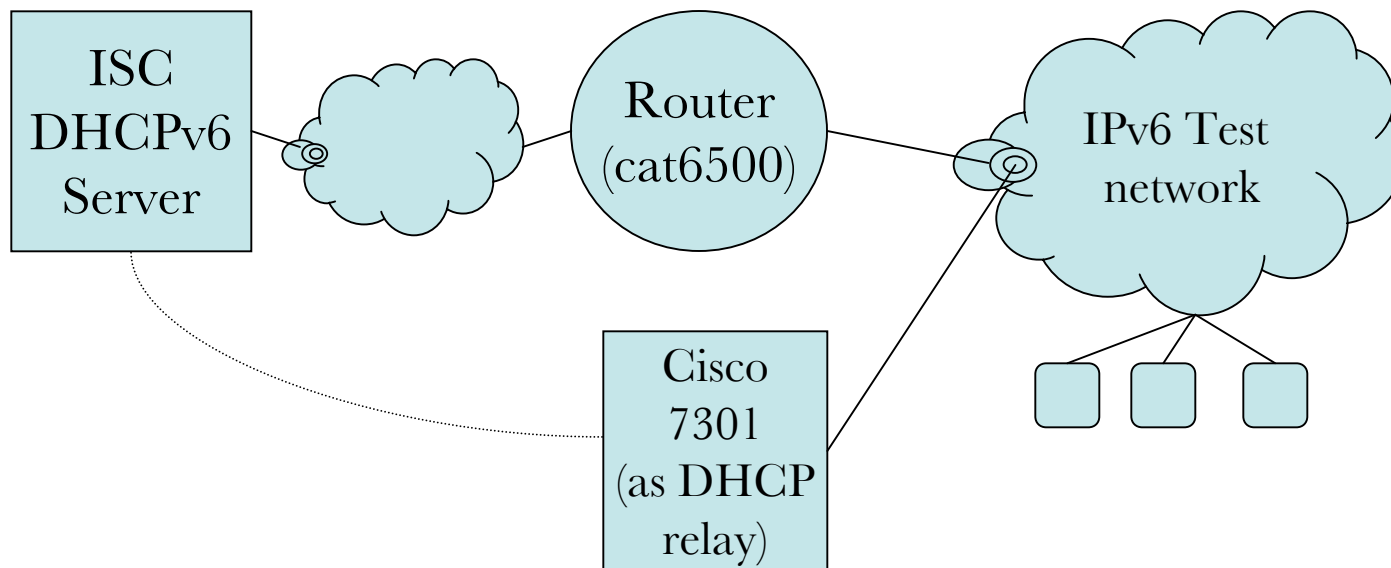
DHCPv6!!!

- We've been hearing that DHCPv6 is not ready for prime time. But there are reasons for using DHCPv6:
 - Many people are scared of stateless autoconfig.
 - We already have enterprise IPv4 DHCP services.
 - We may be using fancy switch features like DHCP snooping and IP source guard, ARP verification, etc.
 - The only way to figure out how non-ready-for-prime-time it is is to try it out.

DHCPv6 testbed: pillows are lumpy and someone is hogging the covers



DHCPv6 testbed: pillows are lumpy and someone is hogging the covers



DHCPv6!!!

- Well, it “works”:
 - Clients get addresses via the relay
 - Configuration for the test subnet:

```
subnet6 2607:f140:1:8fff::/64 {  
    range6 2607:f140:1:8fff:1000::/96;  
}
```

- Pretty simple, huh?
- Haven’t yet gotten DDNS to work and I can’t yet match a host using the DUID, in the same way that one can with DHCPv4 and the MAC address. (I am probably doing something wrong.)

DHCPv6!!!

- Client support:
 - Vista: Very good, except it defaults to temporary privacy addresses and autoconfigures even if “managed-config” flag is set in the router. (Must explicitly set the “no-autoconfig” flag, but must still send RAs.)
 - MacOS: No. :-(
 - Linux: RedHat/Fedora has DHCPv6 option right in the installer, but I haven’t gotten it to work. It successfully polls the DHCPv6 server for IA-NA but doesn’t configure an address.
 - Ubuntu: After installing, add the dibbler-client package and it “just works.”
 - FreeBSD (and probably other open-source BSDs): WIDE/KAME DHCP6 client (dhcp6c) works great, but you must explicitly configure it to get an address.

DHCPv6!!!

- Client support:
 - Solaris 10 8/07 x86: It's supposed to have support, but it doesn't appear to "just work" out of the box with the default installer. I need to do more examination of Solaris 10, but I understand it should work.
 - Will try latest Solaris Express and/or OpenSolaris next.

DHCPv6 - Conclusion

- Not ready for prime time, but almost ready for late-night. (Will be appearing on Letterman next week, and then will be opening for George Carlin at Bally's in Las Vegas February 14-24.)
- We will probably need to change registration process. May not simply be able to register MAC addresses and have DHCP only serve "known" clients like we do with IPv4.
- Lack of relay support is current show-stopper.

Overall conclusions

- Even simple conversions can get tricky. There are many details to consider in IPv6 conversion of an application and there are devils in all of them.
- Cisco IOS IPv6 support in the Cat 6500 was making me cranky. Feature parity in the different release trains is uneven. Most competition doesn't have this problem, and even other Cisco platforms don't.
- However, Cisco has listened to my comments, and they have heard from other customers. They are considering aligning feature parity on the Cat 6500. They also are looking into the DHCP relay issue.

Overall conclusions

- Most vendors willing to listen and help out.
- Same is true of open-source developers.
- Our campus has been very receptive to trying out IPv6 on departmental networks. Overall, they're very interested.
- Most or all of these issues can be solved and made to work. But that will take time. It's high time we all got started...