An update on IPv6 in FreeBSD

Bjoern A. Zeeb
Zabbadoz.CoM

EuroBSDCon
Saturday, 2011-10-08
Agenda

1. FreeBSD and World IPv6 Day.

2. "IPv6-only" validation work.

3. IPv6 in FreeBSD 9.
FreeBSD and World IPv6 Day
What did we do?

• Created FreeBSD and PC-BSD v6-only snapshots.

• Created web pages.

• Had an IRC channel #freebsd-w6d.

• Did not sleep though were not too excited anymore after an hour or two.
IPv6 in FreeBSD

Introduction
FreeBSD has shipped tightly integrated IPv6 support: first release to include "out-of-the-box" IPv6 support in the FreeBSD community, including pa

Latest news
- August 3, 2011: 9.0-BETA1 IPv6-only snapshot see IPv6Only wiki page.
- June 9, 2011: Thanks to everyone having joined (unspectacular) statistics of www.freebsd.org after the event for you.
- June 7, 2011: New set of IPv6-only snapshot nameserver support in rtssl (B) and rtssold (B), here and find download links and netinstall detail.
- June 6, 2011: The FreeBSD Foundation and IPv6 support the efforts of World IPv6 Day to accelerate and find more information on FreeBSD.
- June 6, 2011: New set of IPv6-only snapshot and find download links here.
- June 6, 2011: Hiroki Sato commits RFC 6106 (RDNS and DNSSL) support.

IPv6 in FreeBSD
FreeBSD is a widely used, open source operating system for decades of research, as well as a reference implementation that shipped IPv6 support in March 2000 as part of the FreeBSD Project.

IPv6 and the FreeBSD Project
The FreeBSD Project has been an early adopter of IPv6, and the FreeBSD project is currently working on implementing IPv6 connectivity on a variety of platforms. FreeBSD is used by critical infrastructure providers as well as many of the world's busiest and most reliable web sites as the best IPv6 support provider. To read more about some of their products, see the FreeBSD Foundation Testimonials.

www.freebsd.org/ipv6
What happened?

• Nice day on IRC.

• 2 questions on filtering IPv6 frags (pf, ipfw).

• Question on RFC 4941 privacy extensions.

• One possibly IPv6 related 7.4 panic.

• Question on v4-mapped addresses and traceroute6. We drop v4-mapped addresses from the wire early. Some MPLS networks tend to send them.
What happened?

- IPv6-only snapshots had downloads.
- ... lots where by IPv4.
- Mirrors in DE and JP.
- http://www.freebsd.org/ipv6/ipv6only.html
- http://www.pcbsd.org/IPv6
- Press Release, hit NANOG.
"IPv6-only"(*) validation work

(*) no-INET or no-IPv4 support
What about IPv6-Only Services?

Is an IPv6-only service a viable option today?

Not really.
- Only ~4% of the existing client base would successfully connect to an IPv6-only service
Why am I am talking about this now then?

"If you are not part of the solution, you are part of the problem."

(Donn Lee, APNIC32)
Reasons

• It's interesting and fun (and pesky ;-) )

• The top 100 websites seem a boring target to a developer and skype is not the world.

• Someone had an idea for April Fools' Day.

• Two errors are not the same error.

• It will be too late for feature parity in another 15 years.
Reasons (cont.ed)

- Internet Service Providers.
- Mobile.
- Inter-Company connects and road runners / tele workers (VPN connections).
- Web Hosters.
- Management overhead.
- Trying to be ready and not late.
What is "ipv6-only"?

- **Kernel without INET support:**
  ```
  include GENERIC
  ident GENERIC-IPV6ONLY
  makeoptions MKMODULESENV+="WITHOUT_INET_SUPPORT="
  nooptions INET
  nodevice gre
  ```

- **src.conf(5) options:**
  ```
  WITHOUT_INET=
  WITHOUT_INET_SUPPORT=
  ```

- **feature_present(3):**
  ```
  kern.features.inet: 1
  kern.features.inet6: 1
  ```
IPv6-only user space

• Few user space utilities that can compile out IPv4 specific support. More utilities that just work in a no-IPv4-support environment.

• FreeBSD 9 bsdinstall and pc-sysinstall.

• ifconfig and improved rc(1) framework.

• jls.

• (Patches for) ftp.
Network Configuration
Would you like to try stateless address autoconfiguration (SLAAC)?

< Yes >  < No >  

FreeBSD Installer

Resolver Configuration

< OK >  < Cancel >  

em0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
options=9b<RXCSUM,TXCSUM,VLAN_MTU,VLAN_HWTAGGING,VLAN_HWCUM>
ether 00:0c:29:0e:31:f2
inet6 fe00::20c:29ff:fe0e:31f2%em0 prefixlen 64 scopeid 0x2
inet6 2001:db8::20c:29ff:fe0e:31f2 prefixlen 64 autoconf
nd6 options=23<PERFORMNUD,ACCEPT_RTADV,AUTO_LINKLOCAL>
media: Ethernet autoselect (1000baseT <full-duplex>)
status: active

plip0: flags=8818<POINTOPOINT,SIMPLEX,MULTICAST> metric 0 mtu 1500
options=29<PERFORMNUD,IFDISABLED,AUTO_LINKLOCAL>
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> metric 0 mtu 16384
options=3<RXCSUM,TXCSUM>
inet6 ::1 prefixlen 128
inet6 fe00::1%lo0 prefixlen 64 scopeid 0x4
nd6 options=21<PERFORMNUD,AUTO_LINKLOCAL>
"ipv6-only" problems

- Software not working with IPv6 at all.
- Software working on Dual-Stack (DS) with IPv4 addresses but not on IPv6-only.
- Software working on DS without IPv4 addresses but not on IPv6-only.
- Software giving confusing error messages on IPv6-only.
Not working with v6 or working in DS with v4

- Samples:
  `cvsupd, cvs pserver, openssl s_client, bsnmpclient, ..`

- Problems:
  - `gethostbyname()` is a bad idea.
  - `bsnmpclient.c` actually uses `getaddrinfo()`, but:
    ```c
    hints.ai_family = AF_INET;
    ```
Working in Dual-Stack without v4 addresses

• Sample: ftp client

• Problem: AF_INET socket call to get socket buffer size on start:

    /*
     * Get the default socket buffer sizes if we don't already have them.
     * It doesn't matter which socket we do this to, because on the first
     * call no socket buffer sizes will have been modified, so we are
     * guaranteed to get the system defaults.
     */
    s = socket(AF_INET, SOCK_STREAM, 0);
    if (s == -1)
        err(1, "Can't create socket to determine default socket sizes");
Confusing error messages on v6-only

● Sample:
  %host people.freebsd.org
  people.freebsd.org is an alias for freefall.freebsd.org.
  freefall.freebsd.org has address 69.147.83.40
  freefall.freebsd.org has IPv6 address 2001:4f8:fff6::28
  ...

● Problem:
  %fetch http://people.freebsd.org/~bz/example.diff
  fetch: ... : Protocol not supported

● But:
  %fetch -6 http://people.freebsd.org/~bz/example.diff
  fetch: ... : Operation not permitted
What has improved?

• (new) NFS seems to work on only IPv6.
• pjd fixed High Available STorage daemon.
• bsdinstall now works with IPv6(-only).
• ifconfig problem was relaxed for rescue.
• ntp noise was fixed.
• A couple of more fixes (base and ports) to come after the release cycle.
Infrastructure

• Dedicated machines for building i386 and amd64.

• Dedicated machine for distributing.

• Pondering freebsd-update support for future.
How to test or use?

- You can start just removing your IPv4 addresses but it will be a different experience.
- Use jails.
- Use virtual or dedicated machines and a real no-INET kernel.

➡ Test the FreeBSD base system, test ports and other 3rd party software.
Test your own software.
How to get? Feedback?

- Compile yourself.

- We provide information and snapshots:
  - http://www.freebsd.org/ipv6/
  - http://wiki.freebsd.org/IPv6Only
  - http://noinet.nyi.freebsd.org/

- Report success or problems.

- Send patches.
  - e.g. gethostbyname() sweep on the tree.
  - Test ports & submit changes (upstream).
Some new features in FreeBSD 9.0

- Lots of general improvements.
- Secure Neighbor Discovery (SeND). Kernel hooks and private socket to intercept/re-inject packets. net-mgmt/send version 0.3 port.
- ipfw fwd ipv6 support. Allows IPv6 captive portal, transparent proxying, ... on v6. IPv6 policy routing.
rtadvd(8) / rtsold(8)

- Massively cleaned up and improved.
- Fixed RFC 4191 Route Info.
- RFC 6106 IPv6 Router Advertisement Options for DNS Configuration (RDNSS, DNSSL) support.
rtadvd(8)

- "noifprefix" option (RAs without prefix possible).
- SIGHUP to reload config.
- Improved handling of dynamic interfaces.
- Can start with non-existent interfaces.
- rtadvctl(8).
rtsold(8)

- Update resolv.conf(5) nameserver and search list using resolvconf(8) script (-R option).
- Possible to use and accept RAs even when IPv6 forwarding is enabled (-F option).
- Used by installer to learn information automatically if available (as we do with DHCP for IPv4).
ifconfig(8) options (in 8.x)

- ifconfig inet6 ...., show up in nd6: line.
- accept_rtadv, -accept_rtadv
- defaultif, -defaultif
- ifdisabled, -ifdisabled
- nud, -nud
- prefer_source, -prefer_source
ifconfig(8) options (new in 9.0)

- **no_radr, -no_radr**: Add routers learnt from RAs to default router list or not. Defaults from `net.inet6.ip6.no_radr sysctl`.

- **auto_linklocal, -auto_linklocal**: Create link-local addresses when interface becomes available. Defaults from `net.inet6.ip6.auto_linklocal sysctl`.
more IPv6 sysctls of interest.

- net.inet6.ip6.use_tempaddr


- net.inet6.ip6.norbit_raif: Suppress R-Bit in NA when accepting RAs on same interface even when forwarding.
rc.conf options

• A lot more fine-grained.

• Some backward compatibility in place.

• `ipv6_enable="YES"` is gone!

• `ipv6_activate_all_interfaces="NO"`
  If there is no `_ipv6=".."` config, the interface will stay disabled for IPv6.
  Set to "YES" to restore historic default on.
rc.conf options (cont.ed)

- **Use inet6 for configuring IPv6!**
  
  ```
  ifconfig_IF_ipv6="inet6 2001:db8::1/64"
  ```

- **Add more addresses (aliases) using:**
  
  ```
  ifconfig_IF_alias0="inet6 2001:db8::2/128"
  ```

  This is the normal alias<N> configuration and can be mixed with IPv4.
rc.conf options (cont.ed)

- `ipv6_privacy="NO"`
  Set to "YES" to use and prefer RFC 4941 privacy addresses with SLAAC.

- `ipv6_ipv4mapping="NO"`
  Set to "YES" to enable support for v4-mapped v6 addresses (`::ffff:a.b.c.d`).

If you use Java, you want to read:
http://diario.behrens.de/2008/10/12/java_and_ipv6_on_bsd.html
rc.conf options (cont.ed)

- `ip6addrctl_policy="AUTO"`
  - `ipv4_prefer`: IPv4 preferred
  - `ipv6_prefer`: IPv6 preferred
  - `AUTO`: try `/etc/ip6addrctl.conf`, if not exists, depending on `ipv6_activate_all_interfaces`:
    if "YES" prefer IPv6 else prefer IPv4.

- `ipv6_cpe_wanif="NO"`
  Set to interface name to set interface flags and sysctls correctly. Per IF options if set override.
Interface configuration

- You usually do not want to accept RAs on your internal interfaces of the router where you advertise yourself or on servers when only doing static configuration. On 9 usually no need for:
  \[\text{ifconfig}_\text{IF}_\text{ipv6}="\text{inet6} \ldots -\text{accept}_\text{rtadv}\"

- Double check on 8.x or add .. if needed:
  \[\text{ipv6}_\text{ifconfig}_\text{IF}="\text{inet6} \ldots -\text{accept}_\text{rtadv}\"\]
Summary

• Greatly improved and more fine-grained controllable for a lot more use cases at costs of backward compatibility.

• Most things now work with a single knob, out of the box, and as expected.

• Still ahead on industry with e.g. RFC 6106 support by default.

• Downstream projects like PC-BSD and pfsense support most/all as well with upcoming versions.
Outlook

- Outstanding 6rd patch.
- Outstanding NAT64 integration with pf.
- Carp and IPv6 rtadvd support.
- "ipv6-only" changes continue.
- More feature parity where missing.
Outlook (cont.ed)

- Pondering DHCPv6 integration.
- IPv6 performance.
- IPv6 compliance.
- Call for documentation improvements.
Questions?

Feedback:
bz@FreeBSD.org
Thanks and happy IPv6ing!

(IPv6 smiley)