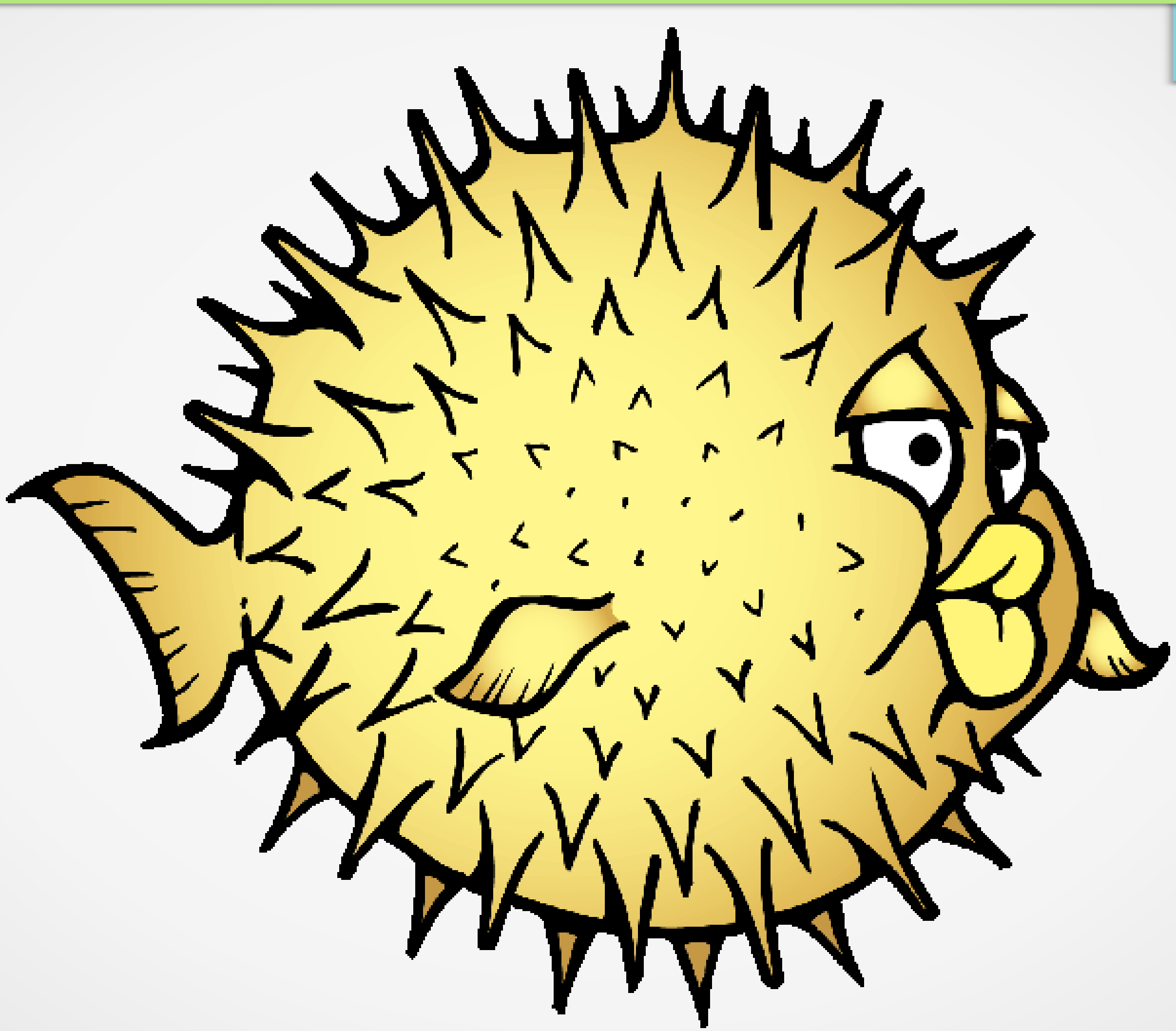
OpenBSD: a crash course



Brian Callahan <bcallah@openbsd.org>

Overview

- What is OpenBSD?
- The now
- The future
- **Why, in 2014, is OpenBSD the choice for you?**

Overview

- Ask questions during and after
- If you're not asking questions, I'm not doing my job right

Who am I?

- OpenBSD developer
  - My first commit was one year ago, yesterday.
- Ports
  - mips64el (loongson)
- Built OpenBSD/octeon 5.4 release binaries and snapshots
- “MIPS on OpenBSD” - NYC*BUG, April 2013
- Not a CS person
- Admin at devio.us (OpenBSD shell provider)
- More about me at gopher://gopher.anthrobsd.net/

What is OpenBSD?

What is OpenBSD?

- It's that firewall OS... right...?
- Crazy, paranoid, security people...
- I read an email from them once, they're mean

What is OpenBSD?

- Welcome to OpenBSD: The proactively secure Unix-like operating system.
  - Default /etc/motd
- The OpenBSD project produces a **FREE**, multi-platform 4.4BSD-based UNIX-like **operating system**. Our efforts emphasize **portability**, **standardization**, **correctness**, **proactive security** and **integrated cryptography**.
- **Only two remote holes in the default install, in a heck of a long time!**
  - This is important, way beyond the obvious!
- For developers, by developers

What is OpenBSD?

- ***FREE***
  - BSD-licensed
  - As per goals.html, acceptable copyright is “ISC or Berkeley style preferred, GPL acceptable as a last recourse but not in the kernel, NDA never acceptable”
  - As of 2003, the ISC license is preferred license for new code

The ISC License

- src/share/misc/license.template

```
It is important to specify the year of the copyright. Additional years
should be separated by a comma, e.g.
  Copyright (c) 2003, 2004

If you add extra text to the body of the license, be careful not to
add further restrictions.

/*
 * Copyright (c) YYYY YOUR NAME HERE <user@your.domain>
 *
 * Permission to use, copy, modify, and distribute this software for any
 * purpose with or without fee is hereby granted, provided that the above
 * copyright notice and this permission notice appear in all copies.
 *
 * THE SOFTWARE IS PROVIDED “AS IS” AND THE AUTHOR DISCLAIMS ALL WARRANTIES
 * WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTEES OF
 * MECHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
 * ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
 * WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
 * ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF
 * OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
 */
```

Operating system

- OpenBSD, like all the BSDs, is a full operating system
  - Kernel *and* userland, all from one team
- No NDA code
  - If a vendor won't release documentation and no one wants to reverse engineer it, then it's not going to be supported
- No blobs!
- Use our code
  - It's there so that it can be used, use what you need (or use the whole damn thing!)

What is OpenBSD?

- Portability
- 21 supported platforms over 14 architectures
- Supported platforms:
  - alpha, amd64, armish, armv7 (beagle), aviion, hp300, hppa, i386, landisk, loongson, luna88k, macppc, mvme68k, mvme88k, octeon, sgi, socppc, sparc, sparc64, vax, zaurus
- Architectures:
  - alpha, amd64, arm, hppa, i386, m68k, m88k, mips64, mips64el, powerpc, sh, sparc, sparc64, vax

What is OpenBSD?

- 2 “current porting efforts”
  - hppa64, solbourne
- 8 discontinued platforms
  - amiga (3.2), arc (2.3), cats (4.0), mac68k (5.1), palm (5.3), pegasos (3.5), pmax (2.7), sun3 (2.9)
- No cross compiling!
  - All supported platforms are *self-hosting*

What is OpenBSD?

- Standardization
  - Adherence to POSIX, ANSI
  - ls –this-flag-is-so-long-I-forgot-what-it-does
  - Documentation!
- Correctness
  - Hacks discouraged
  - Do it the right way or don't do it
  - Auditing of the source tree
    - No NDAs!
    - No blobs!
  - Documentation!

What is OpenBSD?

- Proactive security
  - What services are on after install?
    - SSH (root login disabled if you add a user at install)
      - OpenSSH
        - THE implementation of SSH
    - NTP (if you turn it on)
      - OpenNTPD
        - Coded with security in mind
  - No NDAs!
  - No blobs!
  - Documentation!
  - Cryptography
- In today's world of script kiddies and other nonsense, do you really want to *start* you machine's life with a bunch of security problems?

No blobs

- Theme of the 3.9 release (lyrics.html#39)
  - Blobs can be 'de-supported' at any time.
  - Blobs cannot be supported by developers.
  - Blobs cannot be fixed by developers.
  - Blobs cannot be improved.
  - Blobs cannot be audited.
  - Blobs are specific to an architecture, this less portable.
  - Blobs are often quite massively bloated.
- **We have no idea that blobs aren't doing something bad!**
- And once it's in your kernel, it's already too late

Documentation

- Extensive man pages and FAQ.
- Documentation is a serious task
  - A mistake in the documentation is a **bug**
  - New programs must have a man page for import
- Good documentation lets users solve problems quickly
- Good documentation lets programmers write better code
  - Positive feedback loop

What is OpenBSD?

- Proactive security
  - W^X
  - Guard pages
  - Privilege separation
  - Privilege revocation
  - chroot jailing
  - ProPolice
  - strlcpy() and strlcat()
  - swap encryption, in small chunks

Proactive security

- Privilege separation (Theo de Raadt, EuroBSDcon 2013)
  - First priv-sep program was postfix
  - Next, OpenSSH
- Now...
  - ping ping6 pflogd isakmpd iked dhclient systrace aucat ssh sshd authpf bgpd dhcpd dhcrelay dvmrpd ftp-proxy hostapd identd iscsid ldapd ldpd mrinfo mtrace nsd (taught them) ntpd ospf6d ospfd popa3d (since removed from our tree) portmap rbootd relayd ripd rtadvd sasyncd slowcgi smtpd spamd spamlogd tcpdump tftp-proxy tftpd traceroute traceroute6 ypldap
- If an OpenBSD developer writes a daemon which doesn't use this technique they'd be laughed straight into another project...

Recent shout-outs

- 30c3: X Security – It's worse than it looks (Ilja van Sprundel)
  - “The OpenBSD guys did this at least five years ago, maybe longer, and the Xorg guys should just steal the OpenBSD code and just be done with it. The OpenBSD guys know how to do privilege separation, just get their code, use it.” (51:45)
    - In reference to (the lack of) privilege separation in X.
  - His slide says:
    - The whole X server runs as root
There has been some talk of implementing priv sep / priv drop
Afaik noone has done this for xorg
OpenBSD did it!

What is OpenBSD?

- Integrated cryptography
- Why?
  - Because we can!
  - OpenBSD is based in Canada
- OpenSSH (obviously)
- Pseudo Random Number Generators
  - October 1, 2013 (markus@)
  - replace rc4 with ChaCha20; inspired by Nick Mathewson's work on libottery; feedback and ok djm@
    - rc4 possibly reliably crackable

What is OpenBSD?

- Released every six months
  - May 1 and November 1
- Every release has its own theme, including artwork and an original song
  - 5.4: *The Sound of Music*
- Officially support current release and previous release
  - 5.4 and 5.3
  - When 5.5 is released (May 1, 2014), 5.3 is no longer supported
- Upgrade is supported from release to release. Big jumps need a reinstall.

What is OpenBSD?

- Let's take a look at a real OpenBSD release set.
- **OpenBSD = Quality**
  - This goal of quality permeates everything we do, from our code to our t-shirts.

OpenBSD is funded by YOU (really)

- The OpenBSD Project is funded by CD sales and sales of other merchandise
  - T-shirts (they're fantastic, buy some)
  - Books
  - Posters!!!
  - And, of course, our donors
    - See donations.html

What is OpenBSD?

- **“Maybe we are an Operating System Idea Incubator Organization”** (Y2038: Going long long on time_t to cope with 2,147,483,647+1, Theo de Raadt, EuroBSDcon 2013)
- Basically, we incubate ideas
- First OS to ship IPsec, IPv6, crypto, SSH, strong random, stack protector, and other security mitigation methods
- Ultimately, we're not afraid to break things in big ways, clean it all up, then push everyone else along
  - **We are changing the world**

OpenBSD beneficiaries

- OB2E (mwl): The OpenBSD Community (xxxiv - xxxv)
  - Four tiers
    - Users, Contributors, Committers, Coordinator
- My list: OpenBSD beneficiaries
  - Also four tiers
    - Developers, Direct users, Indirect users, the World

OpenBSD beneficiaries

OpenBSD beneficiaries

- Developers
  - Theo de Raadt, founder and project lead
  - About 80 active developers
  - The names that show up on the source-changes@ and ports-changes@ mailing lists
  - New developers by invitation
  - For developers, by developers

OpenBSD beneficiaries

- Direct users
  - Non-developers using OpenBSD
    - Own machines
    - Company machines
    - “the users”
  - New developers generally come from here
    - Send patches
    - Lots of good work generally earns an invitation

Direct users

- What does OpenBSD offer me?

Direct users

- Consistency
  - No matter on which of the 21 platforms you install OpenBSD, you can expect the OS to look and feel and behave the same
  - My current inventory: Lenovo ThinkPad EDGE E420 (amd64), Asus EeePC 1000H (i386), Dell Dimension 4550 (i386), Gateway Solo 2150 (i386), Lemote Fuloong (loongson), 2x Lemote Yeeloong (loongson), Apple iBook G3 (macppc), Apple PowerBook G4 (macppc), Portwell CAM-0100-7616 (octeon), UBNT EdgeRouter LITE (octeon), SGI O2 (sgi), 2x Sun Netra X1 (sparc64)
  - I don't have time to research a different OS/distro for each machine, worry about updating each one in a different manner, wonder if each machine actually has all the latest security updates
  - A single, consistent, well documented way to do everything

Easy prototyping

- I know how to install OpenBSD on my laptop... therefore I already know how to install OpenBSD on my BeagleBoard Black
  - I know how to install packages on my laptop (pkg_add)... therefore I already know how to install OpenBSD on my Lemote Fuloong
- My “patented” method of OpenBSD installation:
  - Boot the installer, place a shot glass glass on your Enter key, go get a drink, done
    - Not quite, but pretty damn close

From single use server to desktop

- Scales well for all needs
  - Need a small arm box to run Tor?
  - Have an old sparc64 lying around to turn into your home network file sharing server or firewall?
  - Old laptop for mom?
  - Need to run a full desktop? We got you covered there too
    - OpenBSD the first *BSD with Gnome3 *with updates*
    - KDE4.11
    - Xfce4.10
    - Firefox, Chromium
    - LibreOffice
    - The list goes on and on...
- And you still get all the security and other benefits

OpenBSD beneficiaries

- Indirect users
  - Not using OpenBSD, but consciously using some subproject of OpenBSD (or maybe not so consciously)
    - OpenSSH, CARP, mandoc, pf
- OpenSSH?
  - Everyone
- pf?
  - FreeBSD, NetBSD, DragonFly BSD, Mac OS X, Apple iOS, QNX (Blackberry)
- mandoc?
  - FreeBSD, NetBSD, DragonFly BSD, Minix3, Linux

Indirect users

- Other things in the OpenBSD tree you may use...
  - tmux, cwm, mg

OpenBSD beneficiaries

- The World
  - People who have no idea their devices run our code, or benefit from our work
- vBSDcon (October 25-27, 2013)
  - henning@ and reyk@ - “Inspecting Packets with OpenBSD and pf”
    - pf is in your iPhone and your Blackberry (slide 48)

The now

strlcpy() and strlcat()

- strlcat/strlcpy
  - Added to OpenBSD on July 1, 1998 (2.4)
  - Introduced in a USENIX paper in 1999 by Todd C. Miller and Theo de Raadt
  - 1996: team audit of the OpenBSD tree “found rampant misuse of strncpy() and strncat()”
    - http://www.openbsd.org/papers/strlcpy-paper.pdf
    - http://www.openbsd.org/papers/strlcpy-slides.pdf
  - Designed to provide an easier time writing and auditing string code to avoid overflows and truncations

strlcpy() and strlcat()

- Yeah, ok, but who's actually using this stuff?

strlcpy() and strlcat()

- OpenBSD, FreeBSD, NetBSD, DragonFly BSD
- Linux kernel (Android), Solaris, Mac OS X (iOS)
- CFEngine, Chromium, ClamAV, CUPS, FFmpeg, GLib2, Git, libevent (use tmux?), Mono, Mozilla, nmap, OpenSSH, Perl, PostgreSQL, rsync, samba, SDL, sqlite3, Tor, VLC, X11, ZFS
- Tons more examples
  - http://marc.info/?l=openbsd-tech&m=138733933417096 “On the matter of strlcpy/strlcat acceptance by industry”

Security theatre

- Snowden and the NSA leaks
- GCHQ in the UK
- Jacob Applebaum's 30c3 talk “To Protect and Infect, Part 2”
  - (42:00) Graphic lists “Windows, Linux, FreeBSD or Solaris” as targets for SWAP (replaces host protected area of the hard drive)
    - Grain of salt, please... but...
- RC4 crackable?
  - We moved to ChaCha20

The future

- The future is now
- The future is coming

The future is now

- Dropping a lot of old compat headers
  - <sgtty.h> and <sys/timeb.h> (among others) are gone on -current
  - As is libcompat
  - How secure do you think un-updated code from the 1980s is?

The future is now

- Radeon KMS
  - August 11, 2013 (jsg@)
  - Add a port of the TTM and Radeon DRM code from Linux 3.8.13. Includes kernel modesetting, framebuffer console and support for newer hardware.

    Firmware needs to be present for acceleration and in some cases modesetting to work. It can be installed via fw_update or manually via pkg_add.

    With lots of help from kettenis@ some macppc bits from mpi@ and some ttm refcount/queue bits from FreeBSD.

    Thanks to M:Tier and the OpenBSD Foundation for sponsoring this work.

Radeon KMS

- Available now in -current, will be in 5.5
- Intel KMS already in 5.4, with improvements in -current
- Firmware for Radeon KMS will be downloaded and installed on first boot
- Works for Radeon cards through and including the HD6xxx series

Autoinstaller

- October 27, 2013 (uwe@)
- Unattended installation using DHCP and a response file

For a completely unattended installation bsd.rd has to be netbooted, a DHCP server must be running and provide “next-server”, which will be used to fetch “http://<next-server>/install.conf”. The format of the response file is a list of “<key> = <value>” pairs where <key> is a substring of the interactive question (case-insensitive) and <value> is what would be entered interactively.

…

This is a starting point, it still a bit rough.

ok krw@, many improvements by halex@

Autoinstaller

- Call for testing underway now!
  - See undeadly.org

64-bit (long long) time_t

- August 13, 2013 (guenther@)

  Switch time_t, ino_t, clock_t, and struct kevent's ident and data members to 64bit types. Assign new syscall numbers for (almost all) the syscalls that involve the affected types, including anything with time_t, timeval, itimerval, timespec, rusage, dirent, stat, or kevent arguments. Add a d_off member to struct dirent and replace getdirentries() with getdents(), thus immensely simplifying and accelerating telldir/seekdir. Build perl with -DBIG_TIME.

  …

  Much assistance in fixing userland issues from deraadt@ and tedu@ and build assistance from todd@ and otto@

64-bit time_t

- We were not the first
  - NetBSD moved to long long before we did
  - **BUT** they did not engage their ports upstreams
    - How are you going to make the world better if you don't engage the world?

64-bit time_t

- Available on -current now, will be in 5.5
- time_t is long long on all archs
- Two steps
  - Base
  - Ports
    - This is important too!
    - Not just for 64-bit time_t, but for all these major changes
- Now we can be sure that January 19, 2038 won't be bothersome
  - But you will need this before 2038...

The future is coming

- signify(1)
  - December 31, 2013 (tedu@)
  - add signify, a tool to sign and verify signatures.

    man page and error message help from espie

    other feedback from deraadt djm mikeb
- Will be used to sign packages (soon) and the base system (less soon)

Your own OpenBSD adventure

I'm new, how do I start?

- Run OpenBSD
- Join the mailing lists
  - misc@, ports@, tech@, source-changes@, ports-changes@
- Read the mailing lists
  - Really, there are archives all over the place
  - You don't want to be that person who asks something that's been answered 100 times before
- Read the FAQ and man pages
  - Again, really. It's amazing.
  - Errors in documentation are **bugs**

Say I want to get involved...

- Do everything from the previous slide
- Then
  - Use OpenBSD some more
  - You'll find things to improve
  - Improve them!
  - And send in your patches
- I had a MIPS laptop and a desire to use it...

New user challenge

- If you have even the slightest inkling for OpenBSD...
  - Go home, install it on a machine, any machine.
  - In two weeks time, send in a patch
    - Could be anything, a typo in a man page (or website)
    - Port update
    - Something more substantial?

OpenBSD and you in 2014

- A consistent, security minded OS that is literally future proofing the world
- Technologies you're probably using today, even if you didn't know it before
- Write better code with good documentation
- And, of course, a really kick ass firewall

You have questions?

- I have answers. You go first.

Where to reach me

Questions/Comments/Flames/Occasion Encouragement

bcallah@openbsd.org

ADN: @bcallah

The End

- To the bar!