Adventures in HardenedBSD

Shawn Webb
shawn.webb@hardenedbsd.org
@lattera

History Lesson

- 30 Jun 2013 – Shawn Webb ASLR blog post
  - Joined forces with Oliver Pinter soon after
- 24 Aug 2013 – Oliver posts patch
  - Very incomplete and still WIP
  - Posted to show progress
- April-ish 2014 – Initial SEGVGUARD implementation
- 17 May 2014 – BSDCan ASLR presentation
- 28 Sep 2014 – EuroBSDCon ASLR presentation
  - Help from the community! ARM port working!
- 15 Oct 2014 – [lin]procfs hardening
- 08 Dec 2014 – Shared library load order randomization

History Lesson

- 09 Feb 2015 – FreeBSD introduces kernel-level privilege escalation:
  - coredump with `/bin/sh -c ...` passed to devd
  - devd trusted blindly:

    > “So a file named "a.out; /bin/id; meh" or so should result in execution of aforementioned /bin/id.”

- 10 Feb 2015 – HardenedBSD disables functionality

History Lesson

- 18 Mar 2015 – NOEXEC
- 24 Apr 2015 – Integriforce
- 07 May 2015 – First RPI2 builds
  - First non-Intel hardware
  - Proves HardenedBSD portable and stable
- 23 Jun 2015 – True stack randomization

History Lesson

- 04 Jul 2015 – VDSO randomization
  - Did this on vacation with a migraine with an angry wife. (I love you!)
- 07 Sep 2015 – arm64 support
  - Only tested with qemu as FreeBSD's reference arm64 system costs $2700 USD bare.
- 31 Dec 2015 – Secure binary updater
  - Far superior than freebsd-update

History Lesson

- 25 Jan 2016 – KLD syscall hardening
  - Why should jails access KLD info?
- 25 Feb 2016 – Prevent RTLD from creating executable per-thread stacks
  - Lolwut?!?
- 26 Mar 2016 – LibreSSL in base
- 15 Apr 2016 – PIE base support
  - We can have our PIE and SEGV it, too!

History Lesson

- 17 Apr 2016 – RELRO + BIND_NOW
- 05 Jun 2016 – Integriforce in hbsd-update
- 06 Jun 2016 – OPNsense publishes build with HardenedBSD ASLR enabled
  - No shared object load order randomization
- 11 Jun 2016 – RISC-V support
- This history lesson doesn't include ports

Where Are We Today?

- ASLR
  - 100% complete
  - Strongest ASLR in BSDlandia
  - PIEified base, some ports with PIE
    - PIEified base for: amd64, arm64, i386
  - Shared object load order randomization

Where Are We Today?

- FreeBSD ASR
  - [open hbsd site]

Where Are We Today?

- NOEXEC
  - 90% complete
  - Need to support TEXTRELs
- OPNsense Integration of ASLR
  - 90% complete
  - Needs PIEified base and ports
- hbsd-update
  - 100% complete
  - Secure binary updating for base

Where Are We Today?

- SEGVGUARD
  - Technically 100% complete
  - Going to go through an overhaul
- Various hardening features
  - IPv6 hardening, [lin]procfs hardening, KLD hardening
- secadm
  - 100% complete
  - Integriforce with whitelisting mode

Importance of Sanity

- I've learned a few lessons throughout this process.
  - Trolls will be trolls, don't lose sleep over them
    - People will ignore their deceitful claims
  - New motto: “Proving the trolls wrong, one commit at a time”
- Build a proactive, vibrant, and uplifting community
  - Actively seek out ways to help others

Importance of Sanity

- Focus on technology, not drama or politics
- Focus on what you're doing right
- Focus on awesomeness
- In the end, do what makes you passionate
  - Became burnt out on upstreaming ASLR
  - Needed to go back to my passion: implementing defensive infosec in HardenedBSD
    - And prodding @_pronto_ to port SystemD to NodeJS

Short-Term Goals

- Finish up NOEXEC
- FS EXATTR support for exploit mitigation toggles
- secadm in base
- Integriforce installed with the installer and installworld?
- Documentation
- 11.0-RELEASE

Long-Term Goals

- Full audit of the linuxulator
  - Linuxulator currently creates RWX mappings
    - `68572 0x80158e000 0x80158f000 rwx 1 0 2 0 C--- vn /compat/linux/lib64/libc-2.12.so`
  - VDSO cannot be randomized
- Trusted Path Execution
- PaX UDEREF
- RAP?
- OpenBSD's Tor Browser Bundle

Long-Term Goals

- System call hardening
- sysctl hardening
- LibreSSL in base as default crypto lib
- Further work with OPNsense
- RPI3 fully supported (unofficial goal)
- Shipping an appliance with HardenedBSD

Row6

- Special appliance
  - Proprietary and enhanced version of HardenedBSD
  - 4x16TB flash arrays in a 1U chassis
  - NVMe disk for OS, 4x16TB for data
  - Super freaking ultra fast
  - Super freaking ultra low-power consumption
    - 280 Watts idle, 480 watts full load
  - 2x10 core Xeon with hyperthreading, 256GB RAM
  - 8x16TB in a 1U being designed and tested

Special Thank You

- NYCBUG
  - And attendees!
- Open source community in general
- Those who spend time innovating and uplifting others

exit(0);

Questions? Preguntas? Chistes? Bromas?