Sad news

- A NetBSD developer Eric Schnoebelen passed away a few days ago
  - schnoebe@n.o
About me

- maya@NetBSD.org
- Bugs, pkgsrc, drivers (graphics, wireless, ...)


NetBSD 9 and 8.1

- We are going to make netbsd-9 branch in the next month.
- Also 8.1 will be released.
Security

- Kernel ASLR: randomize the kernel image and the kernel memory areas
  - GENERIC: randomizes by default **all** the dynamic memory areas (direct map, PTE space, etc)
  - GENERIC_KASLR: adds randomization on the kernel image
  - Most advanced KASLR implementation to date

- Audited Network Stack
  - More security and more robustness in the network components, cleaner code, safety measures

- Kernel Heap Hardening
  - More difficult to exploit use-after-frees and double-frees on kernel pools
  - More difficult to exploit buffer overflows on kernel pools
  - WIP

- Sanitizer/Instrumentation Support (next slide): quality assurance, detected dozens of bugs and weaknesses, thanks to advanced kernel support
Sanitizer/Instrumentation Support

- Allow to detect several kinds of bugs: buffer overflow, undefined behavior, etc
- Kernel: NetBSD is one of the few OSes to have extensive kernel support
  - KASAN: detect kernel memory corruptions
  - KUBSAN: detect kernel undefined behavior
  - KCOV: ease fuzzing
  - KLEAK: detect kernel memory disclosures (developed in and for NetBSD)
- Userland:
  - ASAN
  - UBSAN + micro-UBSAN (in libc)
  - TSAN
  - MSAN
- SyzBot fuzzing: 24h/24 fuzzing of the NetBSD kernel
Syzkaller fuzzing the NetBSD kernel in Google Cloud Engine (GCE)

NetBSD
fixed bugs (9)

<table>
<thead>
<tr>
<th>Name</th>
<th>Active</th>
<th>Uptime</th>
<th>Build</th>
<th>Kernel</th>
<th>Syzkaller</th>
<th>Corpus</th>
<th>Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>netbsd/ci2-netbsd</td>
<td>now</td>
<td>13m</td>
<td>18m</td>
<td>779fde7b</td>
<td>427ea437</td>
<td>2895</td>
<td>4265</td>
</tr>
</tbody>
</table>

Instances:

upstream (27):

<table>
<thead>
<tr>
<th>Title</th>
<th>Repro</th>
<th>Bisected</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>netbsd test error: timed out</td>
<td></td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>ASan: Unauthorized Access in file ctor</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>lock error in / 500.ADDR1 do_sys_accept</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>assert failed: pg-&gt;wire_count l= 0</td>
<td>0</td>
<td></td>
<td>27</td>
</tr>
<tr>
<td>assert failed: iff-&gt;ff_excise</td>
<td></td>
<td>syz</td>
<td>4</td>
</tr>
<tr>
<td>lock error in / 81.ADDR1 do_sys_accept</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>assert failed: usec &gt;= 0 &amp;&amp; usec &lt; ADDR</td>
<td></td>
<td>syz</td>
<td>5</td>
</tr>
<tr>
<td>panic: event init: unable to alloc ADDR</td>
<td></td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>
• Comprehensive virtualization solution for NetBSD
• CPUs supported: x86 AMD, x86 Intel, more can be added
• Fully MP-safe, fine-grained locking, up to 128 VMs with 256 VCPUs
• NVMM **does not implement an emulator**, but provides *libnvmm*, a virtualization API that allows to add NVMM support in already-existing emulators (like Qemu, VirtualBox, etc)
• Advanced features: *libnvmm* can emulate certain guest operations on behalf of the emulator
• Security: the complex emulation machinery **runs in userland**, and not in the kernel! This means limited attack surface for the host
- NVMM support in QEMU, fully functional
ARM improvements - NetBSD -current

- AArch64 support
- Single kernel for multiple SoC families - GENERIC (armv7), GENERIC64 (aarch64)
- ACPI support - ARM Server Base System Architecture (SBSA), Server Base Boot Requirement (SBBR)
- UEFI bootloader
- Automatic root device detection on live images (armv7.img, arm64.img)
- Performance event monitoring support - tprof(4)
- Multiple cluster / big.LITTLE support
- GICv3 interrupt controller driver
- MSI and MSI-X support added to GIC (v2m) and GICv3 (ITS) drivers
- QEMU ARM Virtual Machine ("virt") and virtio-mmio support
- Loadable kernel module support
- COMPAT_NETBSD32 support
- Kernel address sanitizer (kASan)
- New SoCs
  - Allwinner A10, A13, A64, A83T, GR8, H5, H6, R8
  - Amlogic S805 (converted to FDT), S905
  - NVIDIA Tegra X1
  - Rockchip RK3328, RK3399
  - Samsung Exynos 5422
- Cloud providers
  - Amazon AWS EC2 a1
  - Scaleway
aarch64 TODO

- COMPAT_LINUX
- kernel preemption
- interrupt affinity (intrctl)
- Thumb mode support for COMPAT_NETBSD32
- TLB ASID randomization
- DTrace
The Attic Museum

- Removal of old unmaintained/buggy components:
  - vm86, ipkdb, NDIS, NATM, ISDN, compat_svr4, compat_ibcs2, and the list goes on
  - Reduces the maintenance burden, simplifies the kernel code, sometimes also reduces the attack surface (security)
  - Cleanup still ongoing...

- See the full list
  - https://wiki.netbsd.org/attic_museum/
**Miscellaneous**

- DRM/KMS update to Linux 4.4 (Intel up to Kaby Lake)
- x86: Kernel support for 16TB of physical memory, 32TB of virtual memory
- ZFS update
- DTrace update
GSoc

- NetBSD will participate Google Summer of Code in this year, too. Yay!
- Submit your application!
Resources

Papers and slides include this talk will be available in the following page

http://www.netbsd.org/gallery/presentations/#2019