Linux Security in 10 Years

Brad Spengler / grsecurity
Security Strategy

- Raise TCO
  - Total cost of ‘own’ership (Dave Aitel)
- Aim for skilled attackers
  - APT these days
- Create unpredictable & hostile environment
  - ASLR
  - Infoleak removal
  - No RWX in memory or on disk
Often, Access Control only plays a role post-exploitation
  o A “last line” of defense

Post-exploitation, an attacker wants permanence

Develop more complex exploit that plays along with published SELinux policies?

Attack that new perf_counter system call completely unmonitored by SELinux?
Access Control Won’t Save You

- Vmsplice
- Tee/splice
- Perf_counter
- Move_pages
- ELF loader
- Aout loader
- Brk
- Mremap
- Pipe
- Etc...
Kernel in the TCB

- Lots of attention paid to hardening userland
- Nearly no mainline attention to the kernel
- What will attackers target?

Enlightenment
  - Disables SELinux, TOMOYO, IMA, AppArmor, all other LSMs
  - Grants full root, full capabilities, works in Xen
  - Upcoming LXC/OpenVZ support, since:
    - “If you are inside a user_namespace your capabilities will only be good for manipulating other objects [...] that you have created after you entered the user namespace”
Only public exploits produce a change in public perception of security

Kernel security wasn’t suddenly horrible in 2009, I simply showed how horrible it’s always been

Unlike with Tavis v. Microsoft, I received no threats from Linux vendors
  - Although...

In the end, stronger SELinux protections, stronger mmap_min_addr, much higher user awareness
Remove infoleaks
  - Symbol information
  - Slabinfo
  - PAX_USERCOPY

Remove RWX from kernel

Protect sensitive data
  - Constify function pointers!
  - IDT/GDT/syscall table/etc
  - Vsyscall shadow table (see sgrakkyu’s remote SELinux-disabling exploit)
 Protect against invalid userland memory accesses in general

 Make refcount overflows unexploitable
  ○ Currently equivalent to use-after-free

 kmalloc(sizeof(somestruct) * attacker_len)
  ○ See recent ethtool get_rxnfc() vulnerability

 Basically, secure the kernel! Your super fine-grained security systems will thank you
Payoffs

- **PAX_UDEREF**
  - Found likely oldest Linux bug ever (>= v0.01)
  - vgaarb direct userland dereference
  - NVIDIA direct userland dereference

- **PAX_KERNEEXEC**
  - Enlightenment won’t run (nor (all?) other memory-corruption based public exploits)

- **PAX_USERCOPY**
  - Found heap-based ~64kb infoleak

- **PAX_MEMORY_SANITIZE**
  - Found use-after-free in CONFIG_NO_BOOTMEM
ASLR is a simple, useful technique
  - Ineffective in several cases (ones mainline doesn’t handle properly already, and others)
  - Statistics-based security

Deterministic control flow integrity
  - So long ret2libc/ROP/any other name

The syscall table is protected – how about those page tables?
Into the lion’s den!