seccomp update

v4.3 – v4.8

https://outflux.net/slides/2016/lss/seccomp.pdf

Linux Security Summit, Toronto 2016
Kees Cook <keescook@chromium.org>
(pronounced “Case”)
What is seccomp?

• Programmatic kernel attack surface reduction
• Used by:
  – Chrome
  – Android (minijail)
  – vsftpd
  – OpenSSH
  – Systemd (“SystemCallFilter=…”)
  – LXC (blacklisting)
  – … and you too! (easiest via libseccomp)
Architecture support

- x86: v3.5
- s390: v3.6
- arm: v3.8
- mips: v3.15
- arm64: v3.19, AKASHI Takahiro
- powerpc: v4.3, Michael Ellerman
- tile: v4.3, Chris Metcalf
- um: v4.5, Mickaël Salaün
- parisc: v4.5, Helge Deller
Regression tests

- tools/testing/selftests/seccomp/seccomp_bpf.c
- v4.3: support for s390, Kees Cook
- v4.3: support added for powerpc, Michael Ellerman
- v4.5: support added for um, Mickaël Salaün
  - Removed requirement for PTRACE_GETREGSET
- v4.5: support added for parisc, Helge Deller
  - Included new parisc support for PTRACE_GETREGSET anyway
- v4.7: support added for mips, Matt Redfearn
- v4.8: new tests for ptrace behavior
- Tile support missing? I just noticed this today...
R.I.P. split-phase internals

• Added: v3.19, Andy Lutomirski
• Splits per-architecture calls to seccomp into 2 phases: non-trace actions, tracing actions
• Speeds up simple filters on architectures with high-cost syscall slow path
• Only used on x86
• But x86 sped up slow path
• And experiments with ARM split-phase didn't gain much
• So... due to complexity, removed: v4.8
ptrace ordering

• v4.8: run ptrace ahead of seccomp
• No change in attack surface
• Makes “normal” tracing more sensible
• Reruns filters after SECCOMP_RET_TRACE
Other changes

- v4.4: CRIU support to dump/load filters, Tycho Andersen
- v4.5: fix NNP flag setting when filters added on processes already with a filter, Jann Horn
Wanted: deep argument inspection

• seccomp must not access userspace memory
  – check would race with syscall usage
  – double-read would result in poor performance

• Possible ugly solutions
  – flag an LSM to perform checks at LSM hook time
  – cached argument copying requires teaching syscall infrastructure about the cache
Wanted: discoverable logging

• Most logging needs are already addressed by using the existing audit hook
  – Requires a preexisting global audit rule
• Instead of a heavy-weight monitoring process, something easy that can be examined by a non-admin
• With ptrace reordered, need may evaporate
Questions?

https://outflux.net/slides/2016/lss/seccomp.pdf

@kees_cook

keescook@chromium.org
keescook@google.com
kees@outflux.net