Kernel Security Anti-Patterns: Low Hanging Fruit

http://outflux.net/slides/2013/lss/fruit.pdf

Linux Security Summit, New Orleans 2013

Kees Cook <keescook@google.com>
(pronounced “Case”)

chrome

Overview

- Anti-pattern awareness
- Finding anti-patterns
- Format strings
- String manipulation
- Double-reads
- USB
- Keeping anti-patterns fixed

Anti-pattern Awareness

- Plenty of known general anti-patterns
  - Busy waiting, hard coding, …
  - http://en.wikipedia.org/wiki/Anti-pattern
- Security anti-patterns are less well known
- Document security anti-patterns for kernel?
  - We've got scripts/checkpatch.pl

chrome

Finding Anti-patterns

- Actually go look when you see something ugly
  - printk(buffer);
  - strncpy(destination, source, strlen(source));
  - read, alloc, read again
  - complex parsing of binary structures (USB!)

chrome

Format strings

- printk(buffer); → printk("%s", buffer);
- Lots of stuff accidentally pass strings that are ultimately parsed as format strings
  - CVE-2013-2851
  - CVE-2013-2852
- Use gcc to help
  - -Wformat -Wformat-security -Werror=format-security
  - Dumb about const char *
- %n is dangerous with limited real utility

chrome

String manipulation

- strncpy(destination, source, strlen(source));
  - Unlike snprintf, does **not** NULL terminate
  - Want to always end with NULL? strlcpy
  - Want to never end with NULL? memcpy
  - Regardless, check destination size
  - ISCSI unauth remote stack overflow CVE-2013-2850
- Never used unchecked copy_from/to_user
  - Various graphics drivers
  - Always verify userspace reads (yay SMAP)

chrome

Double-reads

```
struct something {
    unsigned int size;
    unsigned char data[];
};
unsigned int tmp, pos;
struct something *kernel_data;
copy_from_user(&tmp, user_data, sizeof(tmp));
kernel_data = malloc(tmp);
copy_from_user(kernel_data, user_data, tmp);
for (pos = 0; pos < kernel_data->size; pos++) {
    do_something(kernel_data->data[pos]);
}
```


chrome

USB

- HID Report Descriptors
  - Mistakes are similar to double-read
  - 12 CVEs found in a week
  - Verification done with a Facedancer
- Future
  - Mass-storage
  - Webcam

chrome

Keeping Anti-patterns Fixed

- Remove dangerous functions or side effects
  - Remove %n again
- Strong gcc defaults
  - Future: gcc plugins from PaX
- Coccinelle
  - Tests can run from the tree: scripts/coccinelle/
- Smatch
  - Show Dan Carpenter things to catch

Questions?

http://outflux.net/slides/2013/lss/fruit.pdf

keescook@{chromium.org,google.com}

kees@outflux.net