unistd.h, Vulnerabilities

CS 2130: Computer Systems and Organization 1

April 28, 2023

Announcements

- Homework 10 due Monday at 11pm
  - Limited number of submissions, test your code before submitting
- Review session in lab next week
- Final Exam: May 4, 7-10pm, Chem 402
  - Cumulative, see practice tests
- Remember to fill out course evaluations

write

`write:`

- Argument checking
- `syscall`
- Return value checking
- `ret`

Using `write`

Memory

Common Memory Problems (from reading)

- Memory leak
- Uninitialized memory
- Accidental cast-to-pointer
- Wrong use of ‘sizeof‘
- Unary operator precedence mistakes
- Use after free
- Stack buffer overflow
- Heap buffer overflow
- Global buffer overflow
- Use after return
- Uninitialized pointer
- Use after scope

Vulnerabilities

Vulnerabilities

Vulnerability: a program for which something like this could happen (security holes)

- Ex: stack buffer overflow possibility
- Not necessarily malicious (like when we talked about backdoors)

Exploit: a way to use a vulnerability or backdoor that has been created

- Ex: the magic long word to type into our program

Vulnerabilities

Anytime you can modify memory the programmer did not expect you to be able to modify, there’s something you can do to give yourself power or rights the programmer didn’t mean to give you

Vulnerabilities

What should you do when you find a vulnerability?

Good Practices

Good practices when finding a vulnerability:

1. Tell the owner
2. Wait (a reasonable amount of time for a fix)
3. Publish