

Exam 1 Review

Stephen Checkoway
University of Illinois at Chicago
CS 487 – Fall 2017

Format

- Fifty minutes
- No notes
- Work alone (copying or sharing answers *will* result in failing the course)
- Three questions
 - Multiple choice
 - Short answer
 - Attack construction

Topics

- Threat models
- Example attacks
- Memory layout
- Stack
- Buffer overflows
- Constructing shell code
- Integer overflow
- Format string attacks
- Code-reuse attacks
- Defenses
- Malware
- Finding vulnerabilities
- Passwords & authentication
- Access control
- Web & browser

Threat models

- Who are the attackers?
- What are their capabilities?
- What is their motivation?
- What is their level of access?

Example attacks

- Goto fail
- Shellshock
- Samy worm

Memory layout

- Stack (including argv and envp)
- Heap
- Libraries
- Code
- Data

Stack

- Grows down (on most architectures)
- Stack pointer
- Frame pointer
- Return address (pushed to stack or stored in a register)
- Function arguments (on stack or in registers)
- Local variables

Buffer overflows

- Overwrite control data or code pointers
 - On the stack
 - On the heap
- Overwriting data used for control

Constructing shell code

- Want to call `execve`
 - `eax`: `0xb`
 - `ebx`: pointer to `"/bin/sh"`
 - `ecx`: pointer to NULL-terminated array of pointers to arguments
 - `edx`: pointer to NULL-terminated array of pointers to environment variables
- Avoiding zero bytes
 - Sometimes you need to, sometimes you don't

Integer overflow

- Truncations
- Using the same data as both signed and unsigned
- Comparing signed and unsigned

Format string

- Using %n and %x
- %hhn
- Where do you put shell code?

Code-reuse attacks

- Return-to-libc
- Chaining return-to-libc calls
- Return-oriented programming (ROP)
- Constructing gadgets

Defenses

- Stack cookies (a.k.a. stack canaries)
- Data execution prevention (DEP)
- Address space layout randomization (ASLR)

Malware

- Infection type
 - virus
 - worm
 - trojan
 - etc
- Attack
 - wiper
 - dropper
 - bot
 - ransomware

Finding vulnerabilities

- White box vs. black box
- Manual vs. automated
- Fuzzing
- Reverse engineering

Passwords & authentication

- What makes a good password
 - Length, mostly
- Salt
- Rainbow tables
- Password managers
- One-time passwords
- Two-factor authentication

Access control

- Difference between authentication and authorization
- Mandatory access control (MAC)
- Discretionary access control (DAC)
- Role-based access control (RBAC)

Web & browser

- Threats to the web server
 - Code injection (e.g., SQL injection)
- Threats to the browser
 - Running untrusted code in a sandbox
- Threats to one page from another
 - Same origin policy (SOP)
- Cross-origin attacks
 - CSRF
 - XSS
 - Defenses