Secure Programming

Joe Pletcher
CIS330
Set User ID

- Or, suid
- Every program you run on a machine runs as your user
- Your user cannot do certain things
  - Read particular files, bind to ports <1024, etc
Suid: passwd

- So, on a UNIX system, all passwords are stored in /etc/shadow
- You cannot read or write /etc/shadow
- You can change your password
  - This obviously involves both reading and writing /etc/shadow
- Wut?
Suid: in general

- Certain programs are marked suid, this lets them run as a different user
- `find / -user root -perm -4000 -print`
- This is our list of targets. Why?
Hijacking suid

- If we can control suid binaries, we can do anything as that user
- This user is usually root
- This means we can do anything
- owned/pwned/0wned/whatever
- This means we win
How?

- So, how do we hijack control of suid binary?
- A bunch of ways!
  - Buffer overflows, heap overflows, format string vulnerabilities, many others
  - We’re going to talk about buffer overflows
Stack overflows

• So, we declare all our variables with a fixed size

• char foo[20];

• This allocates 20 bytes on the stack for foo

• gets(foo) < Bad. Why?
Bad ideas

• gets
• strcpy
• strcat
• Example!
Resources

- Hacking: The Art of Exploitation 2nd Ed
- Smashing the Stack for Fun and Profit
- This is just a start:
  - Heap overflows, SQLi, XSS, ASLR, Canaries