BUILDING ON BASICS | SEC204

Overview

- Defuse the Bomb !!!
- Building on basics

DEFUSE THE BOMB

DEFUSE THE BOMB EXERCISE

- Ok, lets recap on everything we have done so far and play the ‘Defuse the Bomb’ game!
  - Ubuntu: Desktop/Labs/bomb_lab
  - Hints to solve the lab: http://csapp.cs.cmu.edu/public/bomblab.pdf
  - Original source of exercise:
    Bryant and O’Hallaron, Computer Systems: A Programmer’s Perspective, Carnegie Melon University, http://csapp.cs.cmu.edu/public/labs.html
- `$strings bomb`
- `$objdump -d bomb`
- `$gdb bomb`

BUILDING ON BASICS

Outline

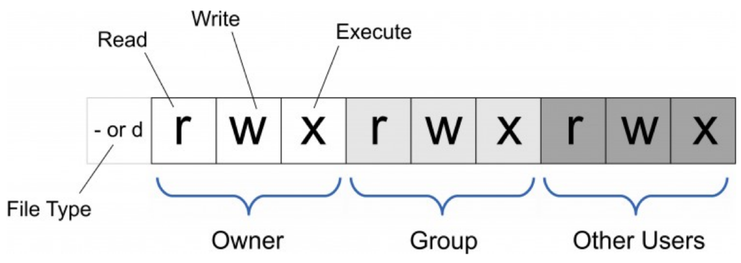
- File Access
- File Permissions
- User IDs
- Pseudo-random numbers

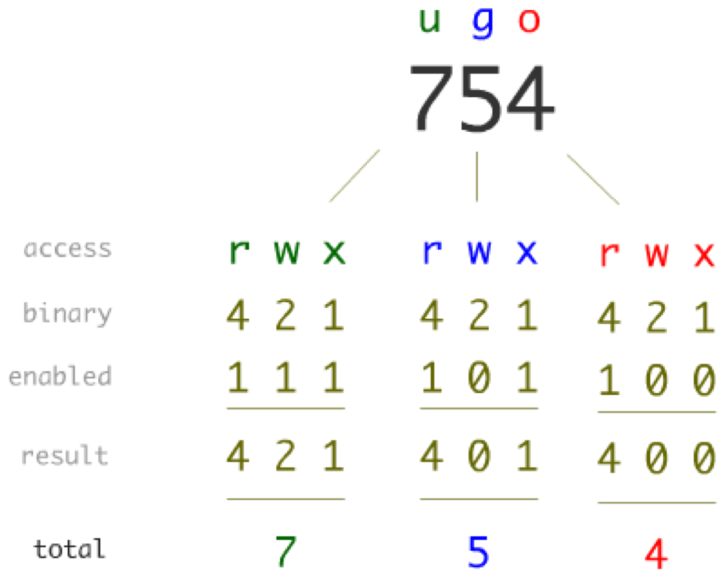
FILE ACCESS

- To access files in C, we use file descriptors or filestreams
- File descriptors (fd) are unique integer numbers used to reference open files
  - No other open file will have the same file descriptor
- They use 4 common functions. For all, the error return code is -1
  - open() – opens file. If successful, it will return the fd
  - close() – takes as input argument the fd
  - read() – input arguments: fd, pointer to data to read, the number of bytes to read
  - write() – input arguments: fd, pointer to data to write, the number of bytes to write

FILE PERMISSIONS

- Read, Write, Execute
- Weight 4, 2, 1 respectively
- Can be set for file owner, group, and others
- Change file permissions with chmod
  - chmod 754 filename.c





USER IDs

- Real User ID
  - The owner of the process
- Effective UID
  - The uid used by the O/S to make access control decisions
- Saved UID
  - Stores previous uid so it can be restored later
  - Usually set to EUID when a SETUID program starts
- Lets run the notetaker.c and notesearch.c from the hacking VM

```
$ gcc -o notetaker notetaker.c
$ sudo chown root:root ./notetaker
$ sudo chmod u+s ./notetaker
$ ls -al ./notetaker
$ ./notetaker “this is a test of multiuser notes”
$ ls -l /bar/notes
$ cat /var/notes
$ sudo cat /var/notes


$ gcc -o notesearch notesearch.c
$ sudo chown root:root ./notesearch
$ sudo chmod u+s ./notesearch
$ ./notesearch
$ sudo su jose
$ ./notesearch
$ ./notetaker “This is another note for the
reader user”
$ ./notesearch
```

PSEUDO RANDOM NUMBERS

- True random number generation is an unsolved problem
- Pseudo random numbers can be generated with rand()
  - The generator must be seeded with a maximum value, RAND_MAX
- Lets run the rand_example.c from the hacking VM

```
$ gcc rand_example.c
$ ./a.out
$ ./a.out
```

FURTHER READING

- Hacking: The art of exploitation, section 0x280, pg 81-114