Mason Competitive Cyber

Intro to Reverse Engineering

News since last meeting

- Mac OSX High Sierra has a HUGE vulnerability
  - Click login several times with username root and empty password

- FCC votes to overturn net neutrality
  - Bad for anyone not being paid by an ISP

- An archive of social media posts scraped by the US military were accidentally exposed
  - Contracter left AWS S3 buckets set to “public”
  - Last week, AWS made changes
    - Added obvious “public” tags in S3
    - Big bold red letters when creating public buckets

Recent CTFs

- RC3 CTF 2017
  - MasonCC0 got 2nd
  - forensics 100

- TUCTF 2017
  - writeup on competitivecyber.club

Upcoming CTFs & Events

- Juniors CTF
  - This weekend
  - Beginner friendly
  - December 1st-3rd

- GRIMM visiting next week
  - register online
  - competitivecyber.club/articles/grimmco/

Reverse Engineering

- Taking a program, figuring out how it works
  - Don’t get source code

- You need linux

- In CTFs, the program is almost always an ELF

```
chmod +x filename
./filename
```

- RE vs. PWN

Reverse Engineering

- Run it

  ```
  ./filename
  ```

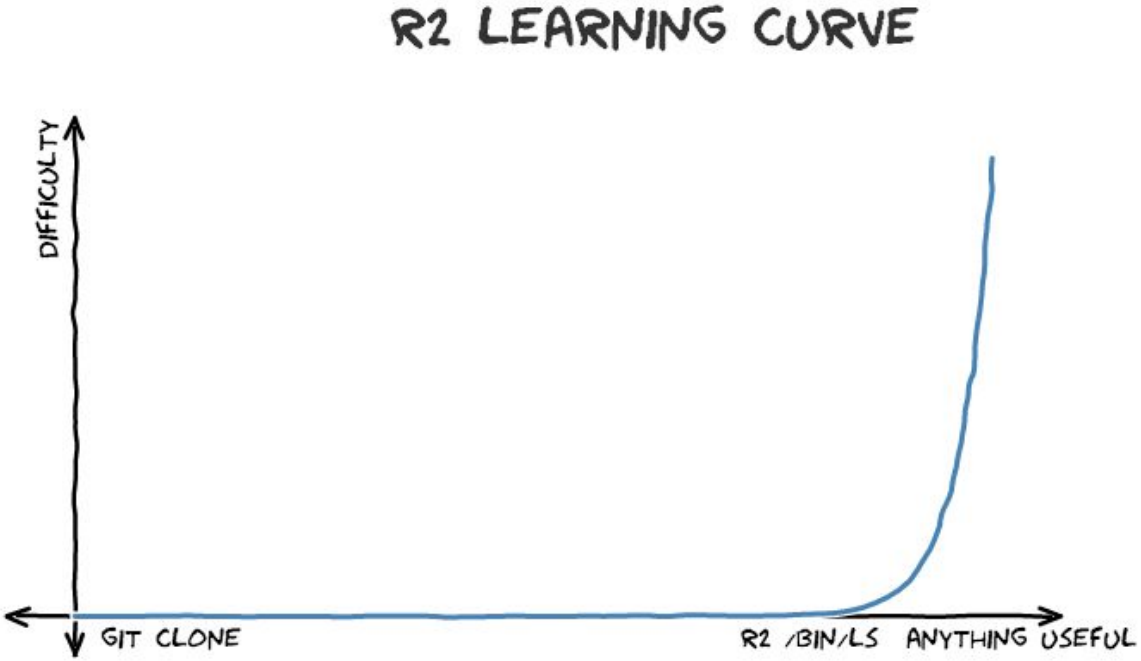
- Strings
  - works if password is hardcoded as a group of printable characters

    ```
    strings filename
    strings filename | grep -i ctf
    ```

- Disassemble it
  - Learn some Assembly

Reverse Engineering Tools

- IDA Pro
  - Costs $$$
  - GUI
  - Windows, OSX, or Linux
- Binary Ninja
  - Same as above
- Objdump
- **Radare2**

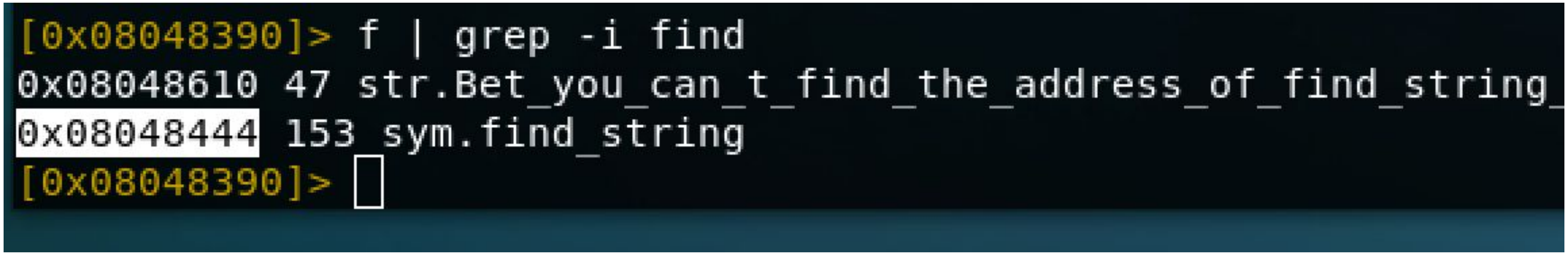


Radare2

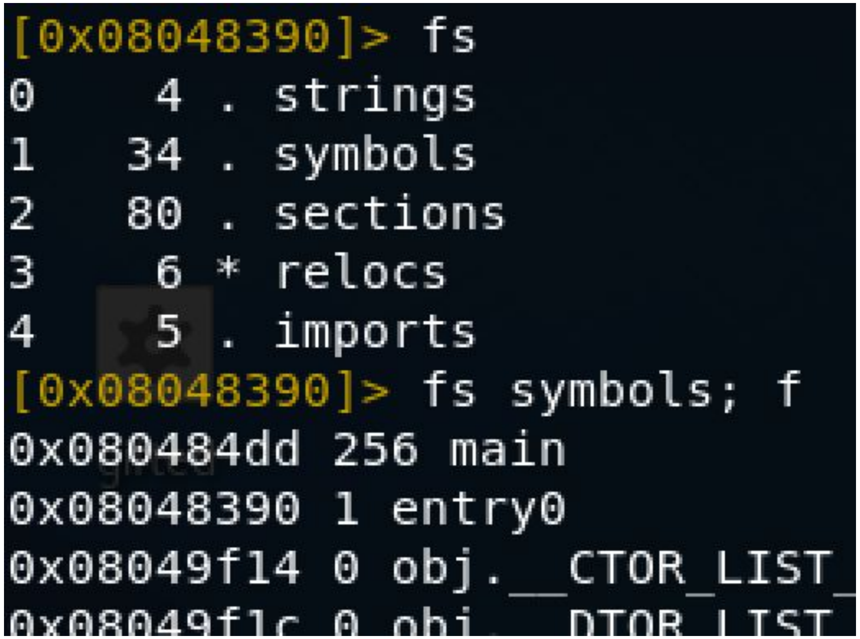
Command	**Use**
r2 filename	starts radare2
s	seek memory address
pdf @ offset	print disassembled function
ahi	assign as string

add ? to incomplete command to get list of commands and their uses

R2 Baby’s Second RE



```
[0x08048390]> f | grep -i find
0x08048610 47 str.Bet_you_can_t_find_the_address_of_find_string_
0x08048444 153 sym.find_string
[0x08048390]> 
```



```
[0x08048390]> fs
0     4 . strings
1    34 . symbols
2    80 . sections
3     6 * relocs
4     5 . imports
[0x08048390]> fs symbols; f
0x080484dd 256 main
0x08048390 1 entry0
0x08049f14 0 obj.__CTOR_LIST_
0x08049f1c 0 obj.__DTOR_LIST
```

R2 Megabeets

- r2 megabeets
  - starts at address 0x08048370
  - ie views other entry points

- fs <flag space>; f
  - show flag spaces
  - print flags it contains

- pd $r @ main
  - disassemble “main” function

    argument taken

R2 Megabeets

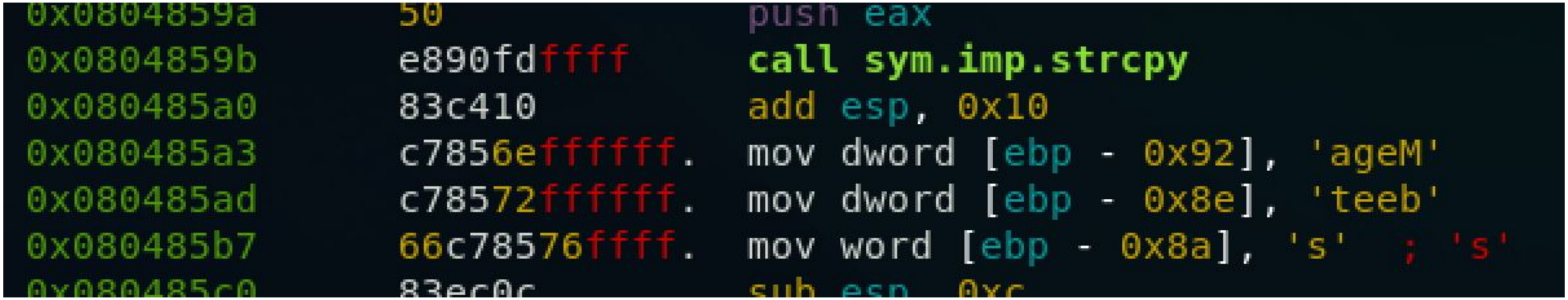
- pd $r @ sym.beet
  - disassemble “beet” function
  - found sym.beet in main

    argument copied to buffer

    compared to output of sym.rot13

- ahi s @@=0x080485a3 0x080485ad 0x080485b7
  - assign values from beet as string

- pd $r @ sym.beet



```
0x0804859a      50              push eax
0x0804859b      e890fdffff      call sym.imp.strcpy
0x080485a0      83c410          add esp, 0x10
0x080485a3      c7856effffff.   mov dword [ebp - 0x92], 'ageM'
0x080485ad      c78572ffffff.   mov dword [ebp - 0x8e], 'teeb'
0x080485b7      66c78576ffff.   mov word [ebp - 0x8a], 's'  ; 's'
0x080485c0      83ec0c          sub esp, 0xc
```

Proud Sponsors

Thank you to these organizations who give us their support:

BATTELLE

It can be done™