Mason CC Advanced Track

Python and Ruby CTF Scripting, Regex

Recent CTFs

- How we placed in recent CTF
- Something we learned from it
  - Description
  - Could make separate slides going into detail about specific problem from CTF

Upcoming CTFs & Events

- DefCamp CTF Quals
  - I personally don’t much care for it, but hey
- HackerGround
  - MD, similar to the other recent Annapolis one
  - In person, probably not terrible networking, I like TopGolf Loudon

Upcoming CTFs & Events

WARGAMES SCREENING

October 21
go.gmu.edu/wargames
Presented by Mason CC
and Patriot Hackers

ELMENDORF AIR FORCE BASE
LORING AIR FORCE BASE
GRAND FORKS AIR FORCE BASE
CITY
DENVER
NORAD
27
ALBUQUERQUE
ALASKAN AIR DEFENSE ZONE
DEFCON 1 2 3 4
MODERN
C++
WARGAMES
IS IT A GAME, OR IS IT REAL?

TopGolf Loudon

- Weird place for a CTF, but hell why not

Club Updates

- Todo: This slide

What are we doing?

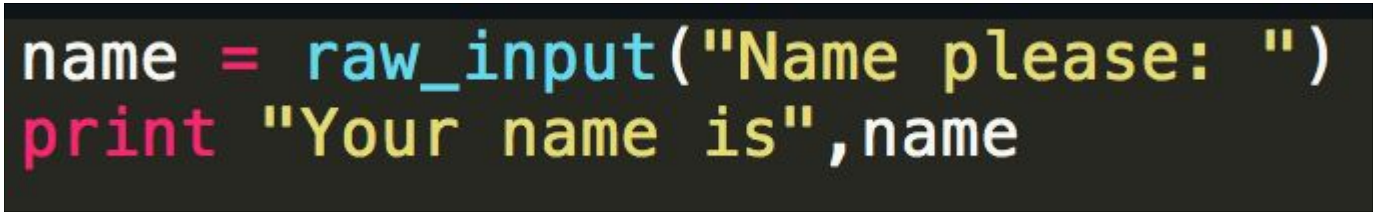
- Python and Ruby CTF Scripting, Regex
- Regex, Python, and Ruby
- Plenty of real world applicability
  - Python, Ruby, Regular Expressions, etc
  - If you’ve never used libraries or handled objects, this is a good introduction

Similarities

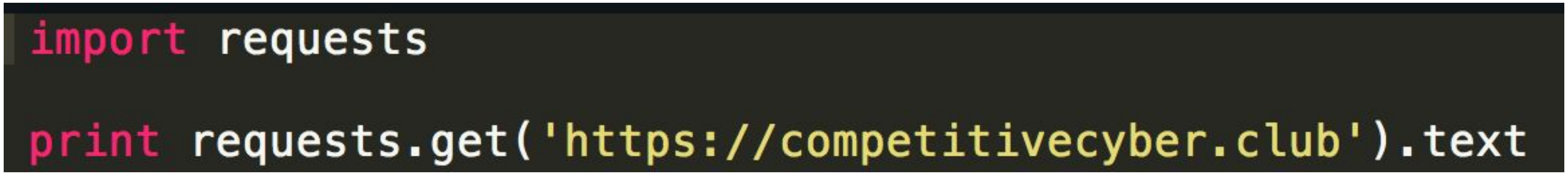
- Both are commonly interpreted
  - .pyc is a common counterexample
- Both are easy to start - procedural is possible
- Both generally expect an external application
  - Kinda goes with the compilation bit

Python Differences

- Very tab/intent oriented/dependent
- 2.7 v 3 clash
  - Dependencies issues
  - Syntax (i.e. print, raw_input, etc)
- Better community
- Pretty fast



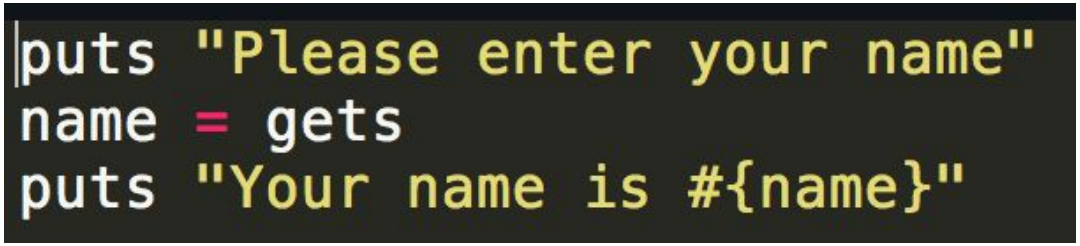
```
name = raw_input("Name please: ")
print "Your name is",name
```



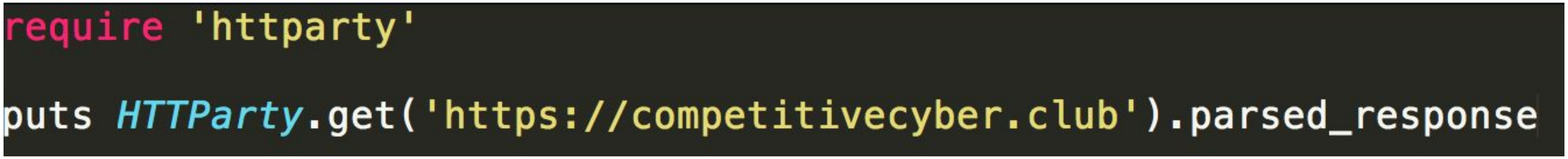
```
import requests

print requests.get('https://competitivecyber.club').text
```

Ruby Differences

- Relatively unique syntax
- Uses terms like **if .. end**, not indentation dependent
- **Bad** memory collection
- Relatively clean for higher level tasks like requests
- Better string interpolation IMO
- Gem-crazy



```
puts "Please enter your name"
name = gets
puts "Your name is #{name}"
```



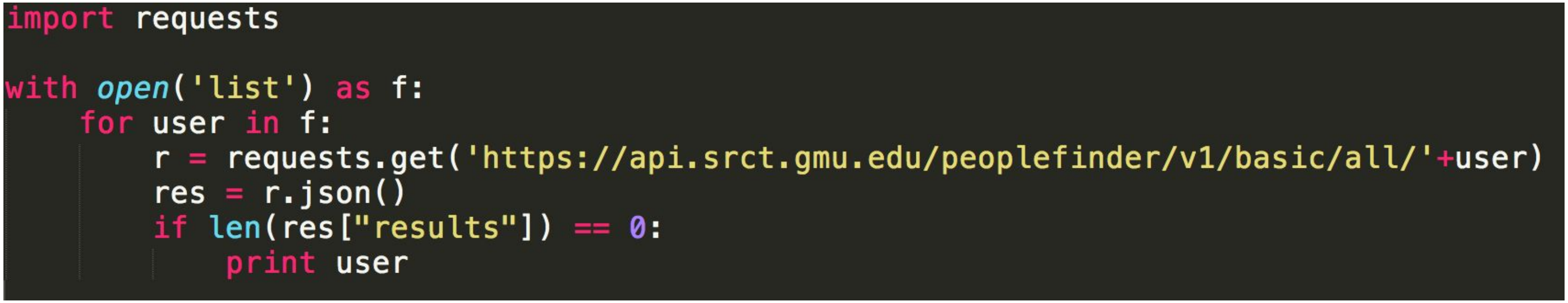
```
require 'httparty'

puts HTTParty.get('https://competitivecyber.club').parsed_response
```

CTF Context

- Automation, some CTFs will require it of you
- **pwntools in python**
- Long list of gems (dependencies) in Ruby
  - This is standard in Ruby, people are freaks for gems

Non-CTF Example

- Script to determine viable students
- From VSE Ice Cream Social



```
import requests

with open('list') as f:
    for user in f:
        r = requests.get('https://api.srct.gmu.edu/peoplefinder/v1/basic/all/'+user)
        res = r.json()
        if len(res["results"]) == 0:
            print user
```

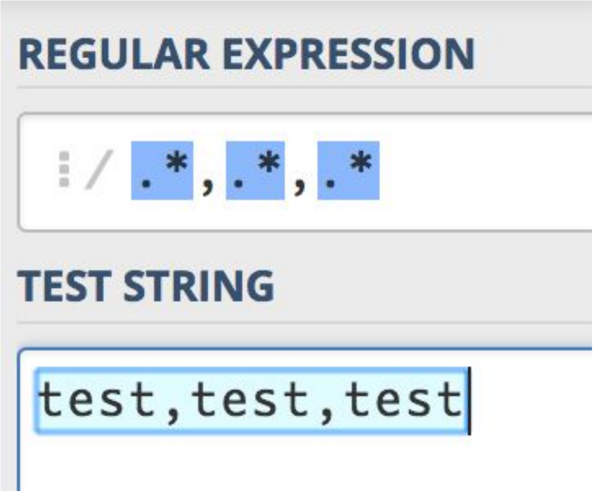
Active Projects

- Paul’s cipher script
  - Mike already had one, didn’t publish because it was in Ruby
- Chris’s RE script
  - Basic angr wrapper
- Mike’s WIP decompression scripts
  - Finds blocks of data with compression patterns and brute forces against an expected format, kinda hacky

Regular Expressions

- https://go.gmu.edu/regexgolf
- regex101.com
- Basically a set of rules for pattern matching strings
- Really useful in CTFs and in general
- People like rolling security rules in regex
- Example:
- Has named groups, partial matches, etc

/.*,.*,.*/g



REGULAR EXPRESSION

```
/ .*,.*,.*
```

TEST STRING

```
test,test,test
```

Example Scripting Challenge

- Use *netcat* to interactively play with it
- **Key terms: sockets, parsing, splitting, optionally regex**
- **bsides.michaelbailey.co:1337**