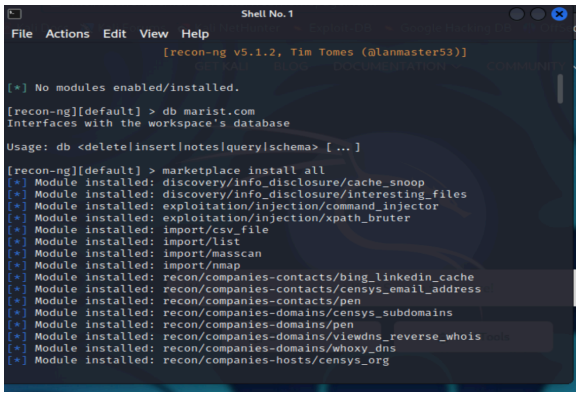
Lab 03- Open Source Intelligence Gathering
Julianna Russo
Network Security
Professor Cannistra
10/30/2023

Description:

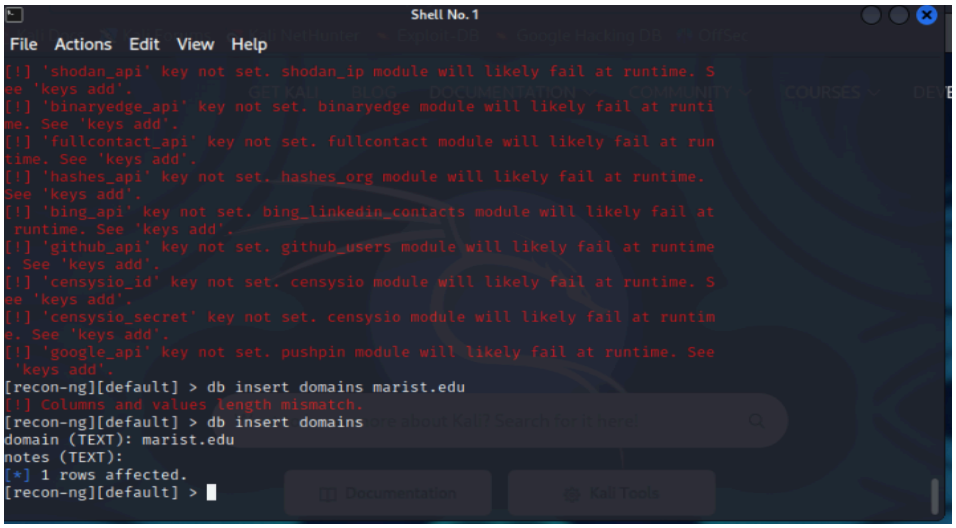
This lab focuses on using passive discovery tools to gather data from open-source intelligence(OSINT). This public information can be utilized and is extremely valuable when targeting a specific organization from an offensive security standpoint. I used Marist as my target and got permission to do so. I used the tools Recon-ng, The Harvester, Shodan.io, NMAP, and Spiderfoot to perform these techniques, all within my Kali Linux VM.

Recon-ng

1. The first is to install all modules from the marketplace, this will allow us to gather the information from the databases, the command used is marketplace install all



2. Insert the target domain



```
Shell No. 1
File  Actions  Edit  View  Help
   Version: 1.1

Description:
  Uses the HackerTarget.com API to find host names. Updates the 'hosts' table with the results.

Options:
  Name    Current Value  Required  Description
  ------  -------------  --------  -----------
  SOURCE  default        yes       source of input (see 'info' for details)

Source Options:
  default        SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
  <string>       string representing a single input
  <path>         path to a file containing a list of inputs
  query <sql>    database query returning one column of inputs

[recon-ng][lab3ns][hackertarget] > options list

  Name    Current Value  Required  Description
  ------  -------------  --------  -----------
  SOURCE  default        yes       source of input (see 'info' for details)

[recon-ng][lab3ns][hackertarget] > options set SOURCE default
SOURCE => default
[recon-ng][lab3ns][hackertarget] >
```

- Use options set SOURCE default and running the module will allow me to check that what I have works so far, then I can unset the source using the unset command. Then I can reenter the options set command to specify the domain name

```
Shell No. 1
File  Actions  Edit  View  Help
[*] Region: None
[*] ----------------------------------------------------------------
[*] Country: None
[*] Host: www.fdrlibrary.marist.edu
[*] Ip_Address: 148.100.100.30
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] ----------------------------------------------------------------
[*] Country: None
[*] Host: sgavoting.it.marist.edu
[*] Ip_Address: 148.100.49.60
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] ----------------------------------------------------------------

SUMMARY
-------
[*] 383 total (383 new) hosts found.
[recon-ng][lab3ns][hackertarget] >
[recon-ng][lab3ns][hackertarget] >
```

- After unsetting and resetting the source to marist.edu and running it, i was able to get 383 sources and their IP addresses
- The most common IP address is 148.100.x.x

```
SUMMARY

[*] 383 total (0 new) hosts found.
[recon-ng][lab3ns][hackertarget] > modules load reporting/html
[recon-ng][lab3ns][html] > info

     Name: HTML Report Generator
   Author: Tim Tomes (@lanmaster53)
  Version: 1.0

Description:
  Creates an HTML report.

Options:
  Name     Current Value                                  Required  Description
  ------   -------------                                  --------  -----------
  CREATOR                                                 yes       use creator name in the report footer
  CUSTOMER                                                yes       use customer name in the report header
  FILENAME /home/kali/.recon-ng/workspaces/lab3ns/results.html  yes  path and filename for report output
  SANITIZE True                                           yes       mask sensitive data in the report

[recon-ng][lab3ns][html] >
```

This contains the HTML report of the the domain

Unexpected/Errors:

```
Shell No. 1
File  Actions  Edit  View  Help
[*] Module installed: reporting/csv
[*] Module installed: reporting/html
[*] Module installed: reporting/json
[*] Module installed: reporting/list
[*] Module installed: reporting/proxifier
[*] Module installed: reporting/pushpin
[*] Module installed: reporting/xlsx
[*] Module installed: reporting/xml
[*] Reloading modules ...
[!] Module 'recon/companies-contacts/censys_email_address' disabled. Dependen
cy required: 'me 'CensysIPv4' from 'censys.search' (/usr/lib/python3/dist-pac
kages/censys/search/__init__.py)'.
[!] 'bing_api' key not set. bing_linkedin_cache module will likely fail at ru
ntime. See 'keys add'.
[!] Module 'recon/companies-hosts/censys_tls_subjects' disabled. Dependency r
equired: 'me 'CensysIPv4' from 'censys.search' (/usr/lib/python3/dist-package
s/censys/search/__init__.py)'.
[!] Module 'recon/companies-hosts/censys_org' disabled. Dependency required:
'me 'CensysIPv4' from 'censys.search' (/usr/lib/python3/dist-packages/censys/
search/__init__.py)'.
[!] 'whoxy_api' key not set. whoxy_dns module will likely fail at runtime. Se
e 'keys add'.
[!] Module 'recon/companies-domains/censys_subdomains' disabled. Dependency r
equired: 'me 'CensysCertificates' from 'censys.search' (/usr/lib/python3/dist
-packages/censys/search/__init__.py)'.
[!] 'spyse_api' key not set. spyse_subdomains module will likely fail at runt
ime. See 'keys add'.
```

When I tried to enter the marketplace command I got this, which was concerning but I figured out that I didn't install what I needed to, so after I installed everything I got the tool to run smoothly.

Value:

This tool is valuable, I don't think it is more valuable than the other tools I used in this lab, but it gave me a good idea of what the target contained. This tool gave me the different URLs and their IP addresses, in a neat format. I was able to figure out the most common IP address for this target, which is 148.100.x.x

The Harvester:

- To set the target you have to enter the command theHarvester -d <target> -l -b < search engine>
  I used "all" instead of a specific search engine because I wasn't getting all the information just using google or bing.

```
kali@kali: ~
File  Actions  Edit  View  Help
[*] ASNS found: 9
--------------------
AS131965
AS14618
AS16625
AS201133
AS209242
AS32244
AS396982
AS54641
AS6124

[*] Interesting Urls found: 21
------------------------------
http://linuxone.cloud.marist.edu/
http://maristpoll.marist.edu/313-tolerance-for-religious-rights/
http://maristpoll.marist.edu/wp-content/misc/Yahoo%20News/20170417_Summary%20Yahoo%20News-Marist%20Poll_Weed%20and%20The%20American%20Family.pdf
http://mokusei.net/2023/auth.it.marist.edu/
http://temp-dropbox.it.marist.edu/
http://think.marist.edu/KellyClough
http://www.fdrlibrary.marist.edu/_resources/images/morg/md0906.pdf
https://dropbox.it.marist.edu/
https://login.marist.edu/cas/login
https://login.marist.edu/cas/login?
https://login.marist.edu/cas/login?service=https%3A%2F%2Fsgavoting.it.marist.edu%2F
https://maristpoll.marist.edu/polls/npr-pbs-newshour-marist-national-poll-extreme-weather-october-5-2021/
https://microsoft.marist.edu/
https://myaccount.marist.edu/react/
https://remotesupport.marist.edu/
https://www.fdrlibrary.marist.edu/
https://www.marist.edu/
https://www.marist.edu/-/2018-10-11-music-fall-concert-series
https://www.marist.edu/admission/undergraduate/apply
https://www.marist.edu/management
https://www.marist.edu/page-not-found

[*] LinkedIn Links found: 0
----------------------------

[*] IPs found: 210
```

```
kali@kali: ~
File  Actions  Edit  View  Help
54.243.238.66
54.243.250.147
54.81.232.47
54.85.246.96
54.87.189.131
54.88.35.99
64.225.35.74
64.91.243.43
72.52.5.200
88.221.168.234

[*] Emails found: 6
--------------------
admission@marist.edu
emily.harris@marist.edu
graduate@marist.edu
health.services@marist.edu
helpdesk@marist.edu

[*] Hosts found: 13460
----------------------
148-100-128-100.foxnet.marist.edu
148-100-128-101.foxnet.marist.edu
148-100-128-102.foxnet.marist.edu
148-100-128-104.foxnet.marist.edu
148-100-128-105.foxnet.marist.edu
148-100-128-108.foxnet.marist.edu
148-100-128-109.foxnet.marist.edu
148-100-128-11.foxnet.marist.edu
148-100-128-110.foxnet.marist.edu
148-100-128-111.foxnet.marist.edu
148-100-128-112.foxnet.marist.edu
148-100-128-113.foxnet.marist.edu
148-100-128-114.foxnet.marist.edu
148-100-128-115.foxnet.marist.edu
148-100-128-116.foxnet.marist.edu
148-100-128-117.foxnet.marist.edu
148-100-128-119.foxnet.marist.edu
148-100-128-120.foxnet.marist.edu
148-100-128-122.foxnet.marist.edu
148-100-128-126.foxnet.marist.edu
```

Using the theHarvester -d marist.edu -l 100 -b all command I was able to get

- 9 ASNS

- 21 interesting URLS
- 210 IPs
- 6 emails
- 13460 Hosts

Emails
admission@marist.edu
Emily.harris@marist.edu
graduate@marist.edu
health.services@marist.edu
helpdesk@marist.edu

The structure of emails for Marist is typically first name/first word. lastname/second word @ marist.edu

ASNS found

ASNS
- AS131965
- AS14618
- AS16625
- AS201133
- AS209242
- AS32244
- AS396982
- AS54641
- AS6124

Unexpected:

When I was first trying to get theHarvester to target marist.edu I wasn't getting very far with my results, that was because I was searching with either google or bing when I needed to be searching with all. I was surprised at how much information I was able to extract. I found more

information about the domain using the Harvester than Recon-ng, and the information I found was more unique.

Value:

This tool was probably my favorite out of all the tools in this lab, mainly because of how much information I was able to access with relative ease. This tool was able to give me over a thousand hosts, over 200 IP addresses, a few emails, and ASNS. I was able to figure out the format of the emails used by the target, and the IP address format for this target.

Shodan.io

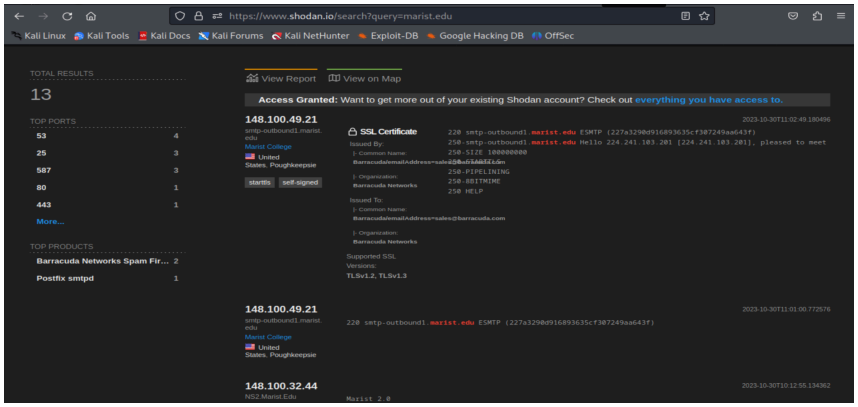
1. First I had to install Shodan on my vm, but it turns out that it already was

```
┌──(kali㉿kali)-[~]
└─$ easy_install shodan
easy_install: command not found

┌──(kali㉿kali)-[~]
└─$ sudo apt install python3-pip
[sudo] password for kali:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
python3-pip is already the newest version (23.2+dfsg-1).
python3-pip set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.

┌──(kali㉿kali)-[~]
└─$ sudo pip3 install shodan
Requirement already satisfied: shodan in /usr/lib/python3/dist-packages (1.28.0)
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager
. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv

┌──(kali㉿kali)-[~]
└─$
```

Input my API key to be able to use Shodan, which I had to create an account to get

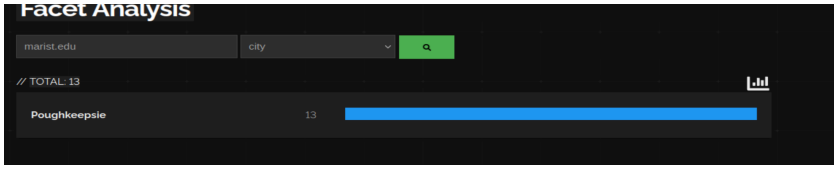
```
┌──(kali㉿kali)-[/home/kali]
└─PS> sudo pip3 install shodan
[sudo] password for kali:
Requirement already satisfied: shodan in /usr/lib/python3/dist-packages (1.28.0)
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager
. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv

┌──(kali㉿kali)-[/home/kali]
└─PS> shodan init GFuykZYgZfJrFXnEDCYcrpUeRY5ElayE
Successfully initialized

┌──(kali㉿kali)-[/home/kali]
└─PS>
```

I was denied access when I tried to gather information using the marist.edu domain, so instead of running it on the command line, I used the website instead and found success.

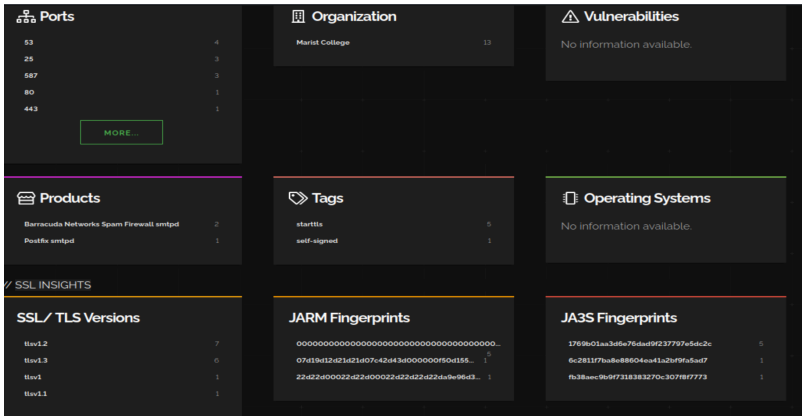


There are 13 IP addresses found by Shodan.io

All 13 IP addresses were found to be located in Poughkeepsie

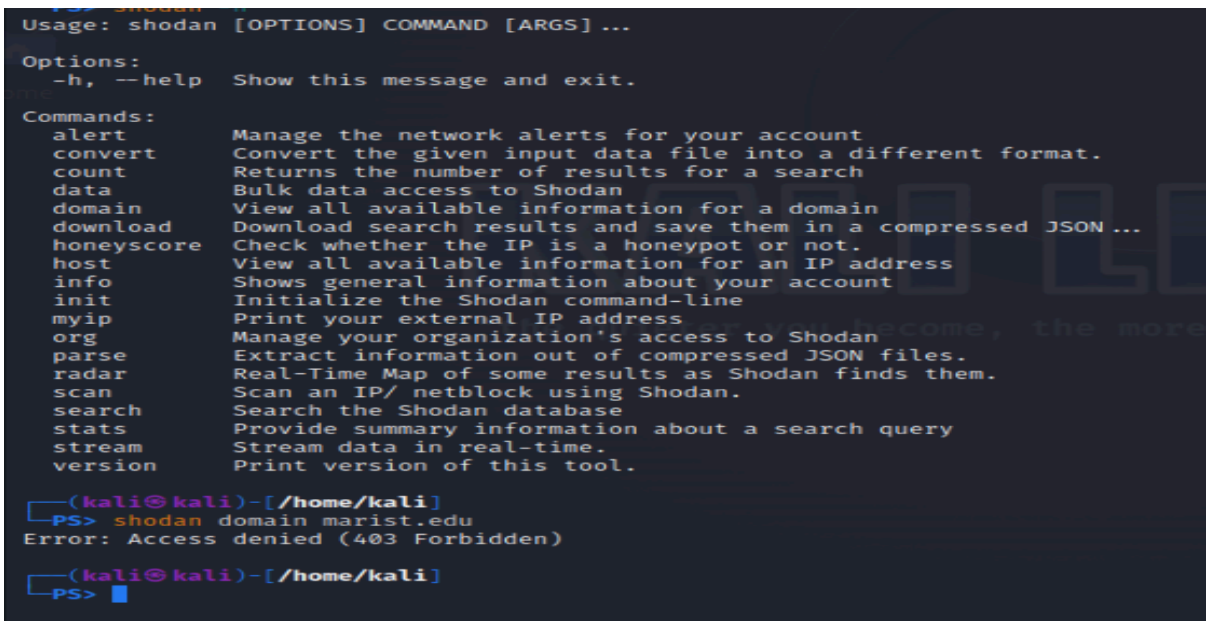


There are 13 Ports



This is a general summary of some of the information Shodan.io gathered for the marist.edu domain. This would be a very good example of what to give to a client in a passive directory report due to the simple format and easy-to-understand data. I like the vulnerabilities portion because it would make it clear to the user if there were any vulnerabilities.

Unexpected:



```
Usage: shodan [OPTIONS] COMMAND [ARGS]...

Options:
  -h, --help  Show this message and exit.

Commands:
  alert       Manage the network alerts for your account
  convert     Convert the given input data file into a different format.
  count       Returns the number of results for a search
  data        Bulk data access to Shodan
  domain      View all available information for a domain
  download    Download search results and save them in a compressed JSON...
  honeyscore  Check whether the IP is a honeypot or not.
  host        View all available information for an IP address
  info        Shows general information about your account
  init        Initialize the Shodan command-line
  myip        Print your external IP address
  org         Manage your organization's access to Shodan
  parse       Extract information out of compressed JSON files.
  radar       Real-Time Map of some results as Shodan finds them.
  scan        Scan an IP/ netblock using Shodan.
  search      Search the Shodan database
  stats       Provide summary information about a search query
  stream      Stream data in real-time.
  version     Print version of this tool.

┌──(kali㉿kali)-[/home/kali]
└─PS> shodan domain marist.edu
Error: Access denied (403 Forbidden)

┌──(kali㉿kali)-[/home/kali]
└─PS>
```

-

When I tried to access the target domain I was denied access, but when I was able to resolve this issue by going through Firefox on my VM, once I did this I was able to access the target domain. I wasn't that surprised with the results of this tool, since I had gotten pretty similar results using the other tools in this lab.

Value:

I believe this tool would be most valuable when working closely with clients since the information found using this tool is formatted in a very user-friendly way. I especially like how clearly the information is labeled, especially the vulnerabilities table,

since that is what the clients are probably most worried about. It is important to make sure the client is able to understand the information that is given to them, otherwise what needs to get done, might not get done.

Nmap

- To get Nmap to gather information from marist.edu I used the command nmap marist.edu and it returned

```
SEE THE MAN PAGE (https://nmap.org/book/man.html) FOR MORE OPTIONS AND EXAMPLES
┌──(kali㉿kali)-[~]
└─$ nmamp marist.edu
Command 'nmamp' not found, did you mean:
  command 'nmap' from deb nmap
Try: sudo apt install <deb name>

┌──(kali㉿kali)-[~]
└─$ nmap marist.edu
Starting Nmap 7.94 ( https://nmap.org ) at 2023-10-30 14:28 EDT
Nmap scan report for marist.edu (148.100.2.7)
Host is up (0.011s latency).
Not shown: 994 filtered tcp ports (no-response), 4 filtered tcp ports (host-unreach)
PORT     STATE SERVICE
80/tcp   open  http
443/tcp  open  https

Nmap done: 1 IP address (1 host up) scanned in 56.87 seconds

┌──(kali㉿kali)-[~]
└─$
```

The ports retrieved were
80/tcp : Open: HTTP
443/tcp: Open: https
The IP address retrieved: 148.100.2.7

Unexpected:

I had trouble with this tool, I wasn't able to retrieve that much information using this tool, but I think that might be due to this tool mostly being used for active discovery techniques. But I was able to get information able the ports the target used.

Value:

I didn't get much information from this tool, but it is good for checking for vulnerable ports. The value of this tool is that it can quickly and easily get port information.

Spiderfoot

```
┌──(kali㉿kali)-[~/spiderfoot/spiderfoot]
└─$ python3 ./sf.py -l 127.0.0.1:5001
python3: can't open file '/home/kali/spiderfoot/spiderfoot/./sf.py': [Errno 2] No such file or directory

┌──(kali㉿kali)-[~/spiderfoot/spiderfoot]
└─$ git clone https://github.com/smicallef/spiderfoot
Cloning into 'spiderfoot'...
remote: Enumerating objects: 26180, done.
remote: Counting objects: 100% (3600/3600), done.
remote: Compressing objects: 100% (243/243), done.
remote: Total 26180 (delta 3448), reused 3370 (delta 3357), pack-reused 22580
Receiving objects: 100% (26180/26180), 16.00 MiB | 8.54 MiB/s, done.
Resolving deltas: 100% (21211/21211), done.

┌──(kali㉿kali)-[~/spiderfoot/spiderfoot]
└─$ ls
correlation.py  dicts    helpers.py  logger.py  spiderfoot  target.py  threadpool.py
db.py           event.py  __init__.py  plugin.py  static      templates  __version__.py

┌──(kali㉿kali)-[~/spiderfoot/spiderfoot]
└─$ cd spiderfoot

┌──(kali㉿kali)-[~/spiderfoot/spiderfoot/spiderfoot]
└─$ ls
correlations            docker-compose.yml  docs                modules          setup.cfg  sf.py      spiderfoot  VERSION
docker-compose-dev.yml  Dockerfile          generate-certificate  README.md      sfcli.py   sfscan.py  test
docker-compose-full.yml Dockerfile.full     LICENSE             requirements.txt  sflib.py   sfwebui.py  THANKYOU

┌──(kali㉿kali)-[~/spiderfoot/spiderfoot/spiderfoot]
└─$ pip
^[[A
Usage:
  pip <command> [options]

Commands:
  install                     Install packages.
  download                    Download packages.
  uninstall                   Uninstall packages.
  freeze                      Output installed packages in requirements format.
  inspect                     Inspect the python environment.
  list                        List installed packages.
```

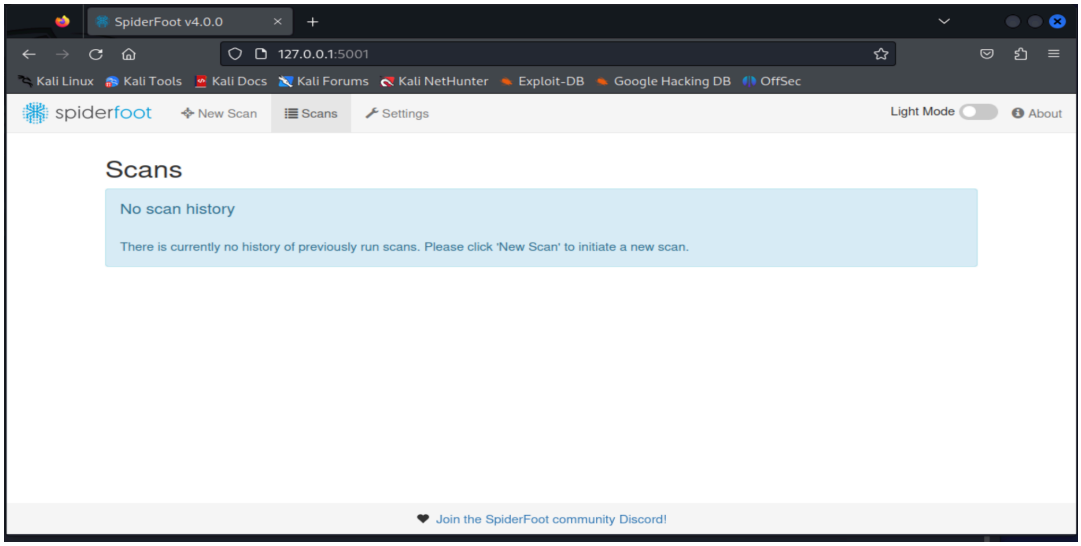
The first step is to get the files from GitHub, and clone them into my vm. This tool used a lot of different commands to achieve what was needed,

Next I had to install the requirements using the pip install -r requirements.txt command

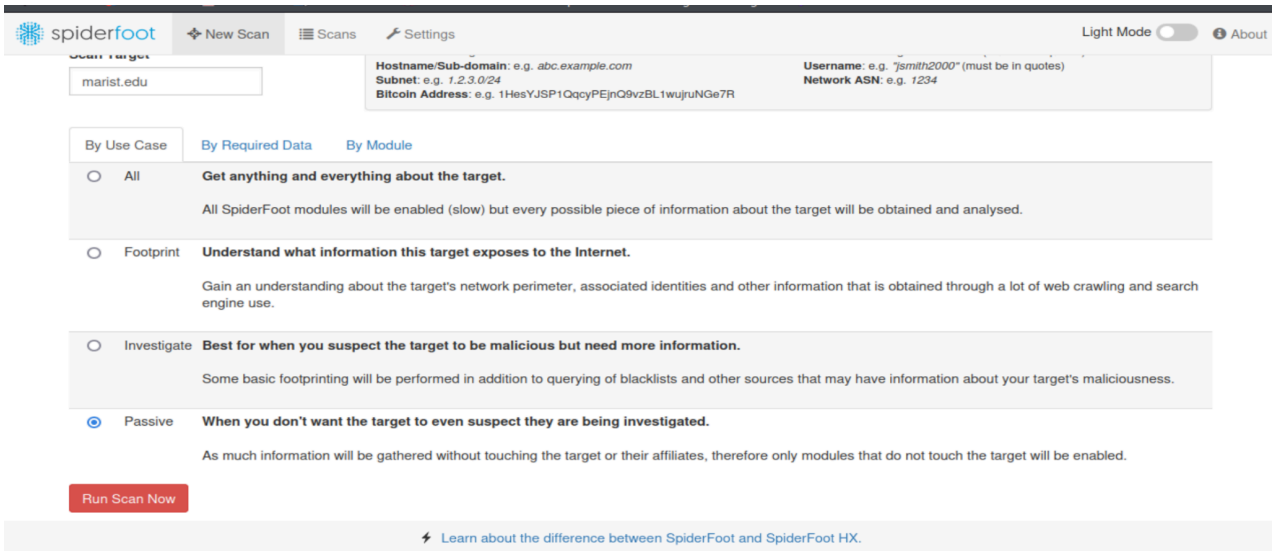
```
┌──(kali㉿kali)-[~/spiderfoot/spiderfoot/spiderfoot]
└─$ pip install -r requirements.txt
\Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: adblockparser<1,≥0.7 in /usr/lib/python3/dist-packages (from -r requirements.txt (line 1)) (0.7)
Requirement already satisfied: dnspython<3,≥2.3.0 in /usr/lib/python3/dist-packages (from -r requirements.txt (line 2)) (2.4.1)
Collecting ExifRead<3,≥2.3.2 (from -r requirements.txt (line 3))
  Downloading ExifRead-2.3.2-py3-none-any.whl (38 kB)
Requirement already satisfied: CherryPy<19,≥18.8.0 in /usr/lib/python3/dist-packages (from -r requirements.txt (line 4)) (18.8.0)
Requirement already satisfied: cherrypy-cors<2,≥1.6 in /usr/lib/python3/dist-packages (from -r requirements.txt (line 5)) (1.6)
Collecting Mako<2,≥1.2.4 (from -r requirements.txt (line 6))
  Downloading Mako-1.2.4-py3-none-any.whl (78 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 78.7/78.7 kB 5.7 MB/s eta 0:00:00
Requirement already satisfied: beautifulsoup4<5,≥4.11.2 in /usr/lib/python3/dist-packages (from -r requirements.txt (line 7)) (4.
12.2)
Requirement already satisfied: lxml<5,≥4.9.2 in /usr/lib/python3/dist-packages (from -r requirements.txt (line 8)) (4.9.3)
Requirement already satisfied: netaddr<1,≥0.8.0 in /usr/lib/python3/dist-packages (from -r requirements.txt (line 9)) (0.8.0)
Requirement already satisfied: pysocks<2,≥1.7.1 in /usr/lib/python3/dist-packages (from -r requirements.txt (line 10)) (1.7.1)
Requirement already satisfied: requests<3,≥2.28.2 in /usr/lib/python3/dist-packages (from -r requirements.txt (line 11)) (2.31.0)
Collecting ipwhois<1.2.0,≥1.1.0 (from -r requirements.txt (line 12))
  Downloading ipwhois-1.1.0-py2.py3-none-any.whl (74 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 74.5/74.5 kB 2.9 MB/s eta 0:00:00
Collecting ipaddr<3,≥2.2.0 (from -r requirements.txt (line 13))
  Using cached ipaddr-2.2.0-py3-none-any.whl
Collecting phonenumbers<9,≥8.13.6 (from -r requirements.txt (line 14))
  Obtaining dependency information for phonenumbers<9,≥8.13.6 from https://files.pythonhosted.org/packages/11/96/34328bd2275f27ad
6dca46607b339ae4fc5fd4cd75037a62d11bb6942a0e/phonenumbers-8.13.23-py2.py3-none-any.whl.metadata
  Downloading phonenumbers-8.13.23-py2.py3-none-any.whl.metadata (11 kB)
Requirement already satisfied: pygexf<0.3,≥0.2.2 in /usr/lib/python3/dist-packages (from -r requirements.txt (line 15)) (0.2.2)
Collecting PyPDF2<2,≥1.28.6 (from -r requirements.txt (line 16))
  Downloading PyPDF2-1.28.6-py3-none-any.whl (87 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 87.1/87.1 kB 6.2 MB/s eta 0:00:00
Collecting python-whois<0.8,≥0.7.3 (from -r requirements.txt (line 17))
  Downloading python-whois-0.7.3.tar.gz (91 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 91.5/91.5 kB 6.9 MB/s eta 0:00:00
  Preparing metadata (setup.py) ... done
Requirement already satisfied: secure<0.4.0,≥0.3.0 in /usr/lib/python3/dist-packages (from -r requirements.txt (line 18)) (0.3.0)
Collecting pyOpenSSL<22,≥21.0.0 (from -r requirements.txt (line 19))
  Downloading pyOpenSSL-21.0.0-py2.py3-none-any.whl (55 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 55.1/55.1 kB 4.0 MB/s eta 0:00:00
Requirement already satisfied: python-docx<0.9,≥0.8.11 in /usr/lib/python3/dist-packages (from -r requirements.txt (line 20)) (0.
```

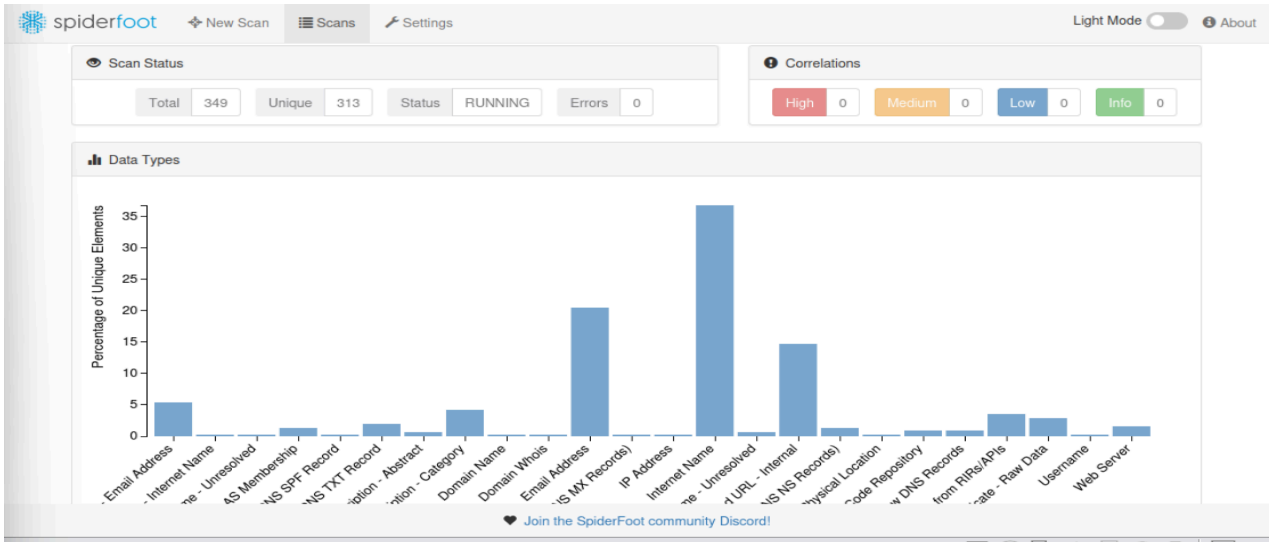
The way to get to the correct URL is to use the command python3 ./sf.py -l 127.0.0.1:5001, which will allow you to open up firefox(or any browser) and spiderfoot will pop up

```
┌──(kali㉿kali)-[~/spiderfoot/spiderfoot/spiderfoot]
└─$ python3 ./sf.py -l 127.0.0.1:5001
2023-10-30 15:28:32,543 [INFO] sf : Starting web server at 127.0.0.1:5001 ...

2023-10-30 15:28:32,544 [WARNING] sf :
*********************************************************************
Warning: passwd file contains no passwords. Authentication disabled.
Please consider adding authentication to protect this instance!
Refer to https://www.spiderfoot.net/documentation/#security.
*********************************************************************

**************************************************************
 Use SpiderFoot by starting your web browser of choice and
 browse to http://127.0.0.1:5001/
**************************************************************
```



To create a new scan you have to click new scan, then enter the name of your scan, and your target, make sure to click passive scan, since this is a lab focused on passive techniques, and I don't have permission to practice active directory techniques from Marist.





Type	Unique Data Elements	Total Data Elements	Last Data Element
Affiliate - Email Address	17	24	2023-10-30 15:40:52
Affiliate - Internet Name	1	1	2023-10-30 15:37:55
Affiliate - Internet Name - Unresolved	1	1	2023-10-30 15:37:55
BGP AS Membership	4	8	2023-10-30 15:38:32
DNS SPF Record	1	1	2023-10-30 15:37:54
DNS TXT Record	6	6	2023-10-30 15:37:54
Description - Abstract	2	2	2023-10-30 15:40:26
Description - Category	13	13	2023-10-30 15:40:26
Domain Name	1	2	2023-10-30 15:40:40
Domain Whois	1	1	2023-10-30 15:39:42
Email Address	64	67	2023-10-30 15:39:38
Email Gateway (DNS MX Records)	1	1	2023-10-30 15:37:54
IP Address	1	1	2023-10-30 15:40:40
Internet Name	115	123	2023-10-30 15:40:50
Internet Name - Unresolved	2	2	2023-10-30 15:40:49

\

Domain Whois	1	1	2023-10-30 15:39:42
Email Address	64	67	2023-10-30 15:39:38
Email Gateway (DNS MX Records)	1	1	2023-10-30 15:37:54
IP Address	1	1	2023-10-30 15:40:40
Internet Name	115	123	2023-10-30 15:40:50
Internet Name - Unresolved	2	2	2023-10-30 15:40:49
Linked URL - Internal	46	52	2023-10-30 15:39:38
Name Server (DNS NS Records)	4	4	2023-10-30 15:37:54
Physical Location	1	3	2023-10-30 15:38:32
Public Code Repository	3	4	2023-10-30 15:40:28
Raw DNS Records	3	3	2023-10-30 15:37:54
Raw Data from RIRs/APIs	11	11	2023-10-30 15:40:53
SSL Certificate - Raw Data	9	9	2023-10-30 15:43:10
Username	1	1	2023-10-30 15:40:18
Web Server	5	9	2023-10-30 15:38:32

This is a graph of the unique names found in the domain.
The information found consists of the unique names in the domain, the public code repository, some usernames, emails of users, names and information of affiliates, and more useful information.

Unexpected:

There wasn't that much that surprised me about this lab, I was surprised about how much information I was able to gather just by entering the domain name on the website.

Value:

This is another tool that is great to use when working closely with a client, the information this tool retrieved was formatted in an easy way to understand for someone who isn't as proficient in cybersecurity technology. This information can be used to prepare for an active discovery penetration test, especially since there are filters on this tool that will specifically do an active discovery.

Topology

Syntax

Marketplace install all	Recon-ng, installs all the modules
workspaces <create|list|load|remove>	Recon-ng creates/delets a workspace
db <delete|insert|notes|query|schema>	Recon-ng: creates/delets/modifies a database
Db insert domains	Recon-ng: inserts a domain
Show domains	Recon-ng: shows the domains in the system
Modules search hack	Recon-ng:searching for a module that contains the work hack to find th epath of the hackertarget module
Options set	Recong-ng: sets the option
theHarvester -d marist.edu -l 100 -b all	The Harvester: gets the information from a target domain, from all search engines
sudo pip3 install shodan	Shodan.io: installs shodan
Shodan init <API KEY>	Shodan.io: asks for the API key so user can access information
Nmap marist.edu	NMAP: this targets the marist.edu domain
pip install -r requirements.txt	Spiderfoot: installs the requirements text
git clone https://github.com/smicallef/spiderfoot	Spiderfoot: retrieves the information found in github that is required for spiderfoot to work
python3 ./sf.py -l 127.0.0.1:5001	Spiderfoot: sets the local host and starts the webserver

Conclusion

The tools in this lab are valuable for not just passive discovery, but it is great practice and preparation for active discovery attacks too. To be able to tell what is vulnerable in an Open Source intelligence resource means that there are probably more vulnerabilities that cna be exposed during a penetration test. Since this information is public, it can be utilized and is extremely valuable when targeting a specific organization from an offensive security standpoint. One way that this organization can better protect its information is to change the format of its email addresses and usernames, repetitive formats make it easier for hackers to gain access to systems.

Resources

(2021). Retrieved from https://www.geeksforgeeks.org/recon-ng-installation-on-kali-linux/

(N.d.). Retrieved from https://account.shodan.io/

S, B. R. B. (2020). Retrieved from https://medium.com/@bibinrajbs/using-recon-ng-in-kali-2020-cc76aa3a4a6d

Shubham. (2021). Retrieved from https://hackeracademy.org/how-to-use-the-harvester-tool-for-information-gathering/

Sohvaxus. (n.d.-a). Retrieved from https://sohvaxus.github.io/content/theharvester.html

Sohvaxus. (n.d.-b). Retrieved from https://sohvaxus.github.io/content/theharvester.html

Veiga, M. S. da. (2019). Retrieved from https://medium.com/hacker-toolbelt/spiderfoot-osint-reconnaissance-59f3a53bbfc2

Wagle, G. (2020). Penetration testing - reconnaissance. Retrieved from https://medium.com/@hiregovindawagle/penetration-testing-reconnaissance-e1fe7849f6b2