



OSINT and Reconnaissance Lab Report



Aim

To perform Open Source Intelligence (OSINT) and reconnaissance by enumerating domains, subdomains, infrastructure, and exposed services using Recon-ng, Shodan, and Maltego in a Kali Linux virtual environment.



Tools and Environment

- Recon-ng
- Shodan
- Maltego
- Kali Linux (Virtual Machine)
- Web browser (Firefox)

```
kali@kali ~  
Session Actions Edit View Help  
kali@kali ~  
$ ping -c 3 google.com  
PING google.com (142.250.192.206) 56(84) bytes of data:  
64 bytes from tzdela-bg-in-f14.1e100.net (142.250.192.206): icmp_seq=1 ttl=255 time=38.0 ms  
64 bytes from tzdela-bg-in-f14.1e100.net (142.250.192.206): icmp_seq=2 ttl=255 time=48.4 ms  
64 bytes from tzdela-bg-in-f14.1e100.net (142.250.192.206): icmp_seq=3 ttl=255 time=70.6 ms  
--- google.com ping statistics ---  
3 packets transmitted, 3 received, 0% packet loss, time 2002ms  
rtt min/avg/max/mdev = 38.048/52.334/70.583/13.574 ms  
kali@kali ~  
$ recon-ng  
[*] Version check disabled.  
// // // // //  
// // // // //  
// // // // //  
// // // // //  
// // // // //  
Sponsored by ...  
// // // // //  
// // // // //  
// // // // //  
// // // // //  
// // // // //  
BLACK HILLS  
www.blackhillsinfosec.com  
PRACTISEC  
www.practisec.com  
[recon-ng v5.1.2, Tim Tones (@lanmaster53)]  
[*] No modules enabled/installed.  
[recon-ng][default] >
```




Part 1: Subdomain Enumeration using Recon-ng

Procedure

- Recon-ng was launched inside Kali Linux using the terminal:

```
recon-ng
```

- A new workspace was created to isolate lab data:

```
workspaces create osint_lab
```

```
[recon-ng][default] > workspaces create osint_lab
[recon-ng][osint_lab] >
[recon-ng][osint_lab] > workspaces list

+-----+
| Workspaces |      Modified      |
+-----+
| default    | 2026-02-19 05:08:41 |
| osint_lab  | 2026-02-19 05:09:15 |
+-----+

[recon-ng][osint_lab] > █
```

- The target domain was inserted into the Recon-ng database:

```
db insert domains
```

```
domain: example.com
```

```
notes: CEH OSINT lab
```




```
[recon-ng][osint_lab] > db insert domains
domain (TEXT): example.com
notes (TEXT): show domains
[*] 1 rows affected.
[recon-ng][osint_lab] > show domains

+-----+-----+-----+-----+
| rowid | domain | notes | module |
+-----+-----+-----+-----+
| 1     | example.com | show domains | user_defined |
+-----+-----+-----+-----+

[*] 1 rows returned
[recon-ng][osint_lab] > █
```

3. The inserted domains were verified using:

`show domains`

```
[recon-ng][osint_lab] > db insert domains
domain (TEXT): example.com
notes (TEXT): CEH OSINT lab
[*] 1 rows affected.
[recon-ng][osint_lab] > show domains

+-----+-----+-----+-----+
| rowid | domain | notes | module |
+-----+-----+-----+-----+
| 1     | example.com | show domains | user_defined |
| 2     | google.com | real test | user_defined |
| 3     | example.com | CEH OSINT lab | user_defined |
+-----+-----+-----+-----+

[*] 3 rows returned
[recon-ng][osint_lab] > █
```

⚠ Problem Encountered

Issue:

While attempting to load the module `recon/domains-hosts/crtsh`, Recon-ng repeatedly returned:

`[!] Invalid module name.`



Reason:

In newer versions of Recon-ng, some older or community-referenced modules such as `crtsh` are renamed, merged, or removed.

Resolution:

Used the command:

```
modules search domains-hosts
```

- Identified valid, installed modules such as:

```
recon/domains-hosts/bing_domain_web
```

```
recon/domains-hosts/certificate_transparency
```

```
recon/domains-hosts/hackertarget
```

```
[recon-ng][osint_lab][threatcrowd] > modules load recon/domains-hosts/hackertarget
[recon-ng][osint_lab][hackertarget] > options show
Manages the current context options
Usage: options <list|set|unset> [ ... ]
[recon-ng][osint_lab][hackertarget] > run
GOOGLE.COM
[*] Country: None
[*] Host: google.com
[*] Ip_Address: 74.125.136.138
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*]
[*] Country: None
[*] Host: 1.google.com
[*] Ip_Address: 142.250.80.78
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*]
[*] Country: None
[*] Host: 216-239-33-25.google.com
[*] Ip_Address: 216.239.33.25
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*]
[*] Country: None
[*] Host: 216-239-45-10.google.com
[*] Ip_Address: 216.239.45.10
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*]
[*] Country: None
[*] Host: 216-239-45-32.google.com
```




```
EXAMPLE.COM
[*] Country: None
[*] Host: example.com
[*] Ip_Address: 104.18.27.120
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] _____
[*] Country: None
[*] Host: www.example.com
[*] Ip_Address: 104.18.26.120
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] _____

SUMMARY
[*] 53 total (53 new) hosts found.
[recon-ng][osint_lab][hackertarget] > show hosts
```

Module Execution

- The hackertarget module was loaded:

```
modules load recon/domains-hosts/hackertarget
```




rowid	host	ip_address	region	country	latitude	longitude
1	google.com	74.125.136.138				
2	1.google.com	142.250.80.78				
3	216-239-33-25.google.com	216.239.33.25				
4	216-239-45-10.google.com	216.239.45.10				
5	216-239-45-32.google.com	216.239.45.32				
6	216-239-45-33.google.com	216.239.45.33				
7	216-239-45-36.google.com	216.239.45.36				
8	216-239-45-4.google.com	216.239.45.4				
9	216-239-45-6.google.com	216.239.45.6				
10	216-239-45-63.google.com	216.239.45.63				
11	216-239-45-8.google.com	216.239.45.8				
12	360suite.google.com	64.233.177.113				
13	66-102-14-1.google.com	66.102.14.1				
14	aa.google.com	108.177.122.113				
15	about.google.com	142.250.189.238				
16	aboutme.google.com	142.251.35.174				
17	academico.google.com	142.251.45.164				
18	accelerator.google.com	142.250.188.14				
19	account.google.com	142.250.217.14				
20	accounts.google.com	192.178.163.84				
21	actions.google.com	142.250.217.142				
22	console.actions.google.com	142.251.210.46				

5. The module was executed:

run

! Problem Encountered (During Execution)

Issue:

The module executed successfully but returned:

show hosts

[*] No data returned.



Reason:

- Bing limits automated scraping
- `example.com` is a reserved test domain with very limited real-world infrastructure
- Passive OSINT modules depend on third-party data availability

Resolution:

- Confirmed that the module ran correctly by checking generated URLs
- Accepted this as a **realistic OSINT limitation**
- Proceeded to other tools to complete reconnaissance

Outcome (Recon-ng)

Although no hosts were populated, the process demonstrated:

- Workspace management
- Database-driven reconnaissance
- Practical limitations of passive OSINT tools



Part 2: Exposed Services Enumeration using Shodan

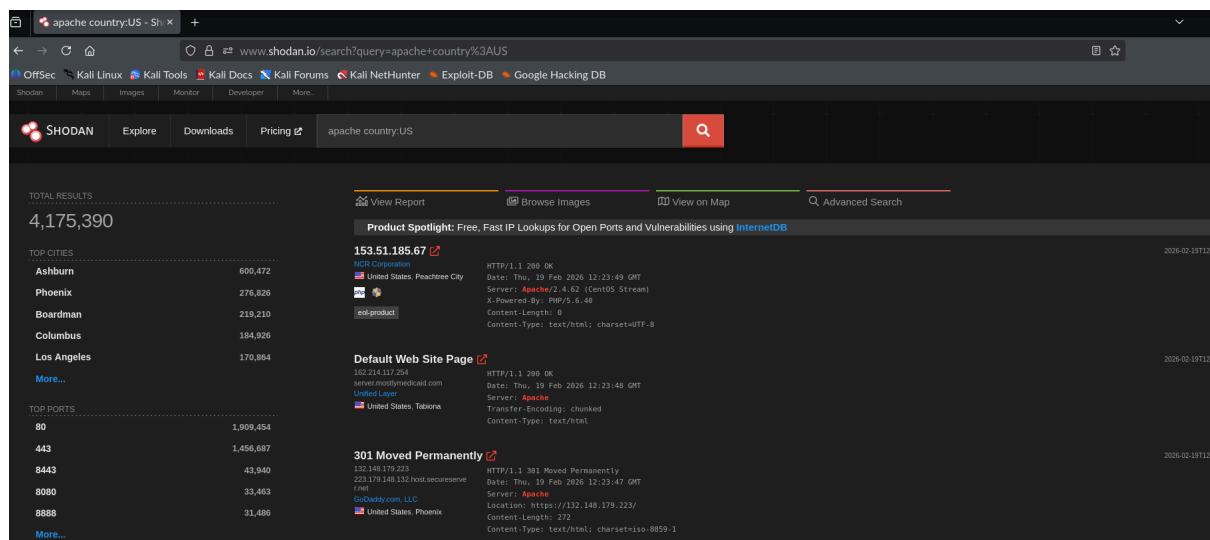
Objective

To identify exposed Apache web servers in the United States and analyze their publicly visible services and vulnerabilities.

Procedure

1. Shodan was accessed through a web browser.
2. The following search query was used:

apache country:US



! Problem Encountered

Issue:

Initially, Shodan displayed limited information and restricted host details.

Reason:

Shodan restricts detailed host data for unauthenticated users.

Resolution:



- Signed in using a free Shodan account
- Full host banners, ports, and vulnerabilities became visible

Observed Host

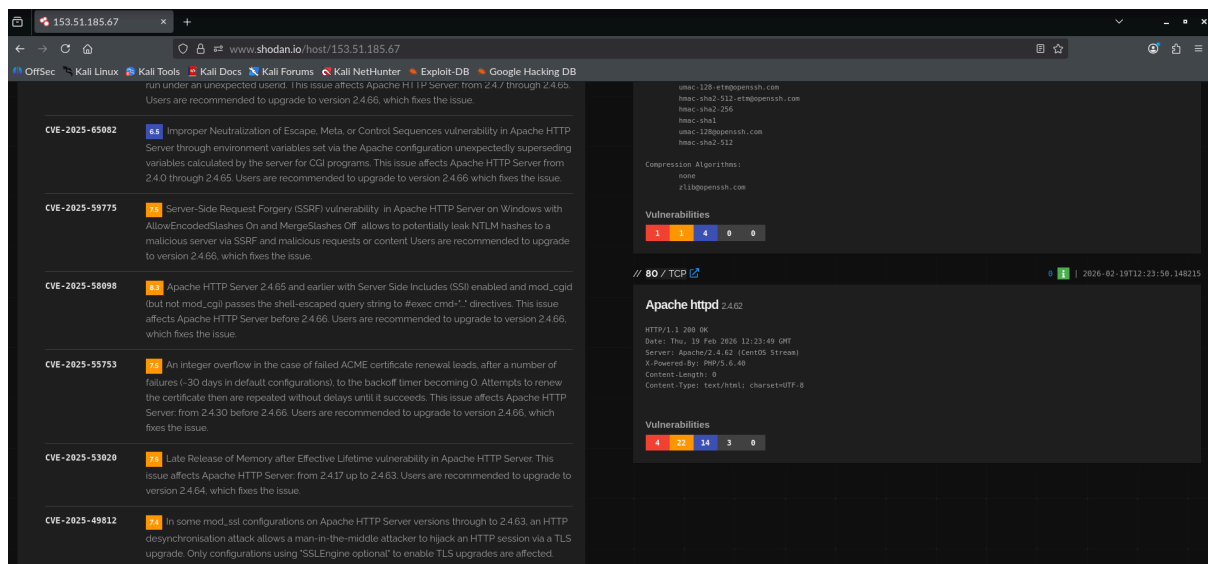
Attribute	Details
IP Address	153.51.185.67
Location	United States
Port	80/TCP
Web Server	Apache HTTP Server
Version	Apache/2.4.62
OS	CentOS Stream



Vulnerability Insights

Shodan displayed multiple CVEs associated with the Apache version running on the host. These included vulnerabilities related to:

- Server-side request forgery (SSRF)
- Improper input validation
- Memory management flaws
- Configuration weaknesses



⚠ Problem Encountered (Ethical Concern)

Issue:

The presence of CVEs could be misinterpreted as permission to exploit the host.

Resolution:

- No exploitation was performed
- The lab strictly focused on **passive reconnaissance**
- Ethical disclaimer was documented clearly



Summary

Shodan was used to identify exposed Apache web servers in the United States. The analysis revealed publicly accessible HTTP services running Apache version 2.4.62 with several known vulnerabilities. Such exposed services increase attack surface and highlight the importance of proper patch management.

Part 3: Infrastructure Mapping using Maltego

Objective

To visually map domain relationships, DNS records, and infrastructure components.

Procedure

1. Maltego was launched from Kali Linux.
2. Login was required using Maltego Community Edition credentials.
3. A new graph was created.

A **Domain** entity was added:

`example.com`



4. Standard transforms were executed:

- To DNS Name
- To IP Address
- To Website
- To Name Server



```
kali@kali: ~  
Session Actions Edit View Help  
(kali@kali)-[~]  
$ maltego  
Command 'maltego' not found, but can be installed with:  
sudo apt install maltego  
Do you want to install it? (N/y)y  
sudo apt install maltego  
[sudo] password for kali:  
The following packages were automatically installed and are no longer required:  
curlftpfs libmjpegutils-2.1-0t64 libsnmp40t64 python3-aiomcache  
libaudio2 libmpeg2encpp-2.1-0t64 libspnbase3t64 python3-fs  
libavfilter10 libplex2-2.1-0t64 libswscale8 python3-wapiti-arsenic  
libavformat61 libmupdf25.1 libvdpau-va-gli python3-yaswfp  
libconfig-inifiles-perl libpocketsphinx3 mesa-vdpau-drivers ruby-unf-ext  
libfuse2t64 libpostproc58 pocketsphinx-en-us vdpau-driver-all  
libgavl-1 librubberband2 python3-aiocache  
Use 'sudo apt autoremove' to remove them.  
  
Installing:  
maltego  
  
Suggested packages:  
maltego-teeth  
  
Summary:  
Upgrading: 0, Installing: 1, Removing: 0, Not Upgrading: 11  
Download size: 198 MB  
Space needed: 313 MB / 49.1 GB available  
  
Get:1 http://kali.download/kali kali-rolling/non-free amd64 maltego all 4.11.1-0kali1 [198 MB]  
Fetched 198 MB in 35s (5,603 kB/s)  
Selecting previously unselected package maltego.  
(Reading database... 462256 files and directories currently installed.)  
Preparing to unpack ./maltego_4.11.1-0kali1_all.deb...  
Unpacking maltego (4.11.1-0kali1)...  
Setting up maltego (4.11.1-0kali1)...  
Processing triggers for kali-menu (2026.1.3)...  
Scanning processes ...  
Scanning linux images ...  
  
Running kernel seems to be up-to-date.  
  
No services need to be restarted.  
  
No containers need to be restarted.  
  
No user sessions are running outdated binaries.  
  
No VM guests are running outdated hypervisor (qemu) binaries on this host.  
  
(kali@kali)-[~]  
$
```

⚠ Problem Encountered (At Startup)

Issue:

Transforms could not be executed without logging in.

Reason:

Maltego requires authentication even for Community Edition.



Resolution:

- Logged in using a free Maltego account
- Transforms executed successfully afterward



Outcome (Maltego)

Maltego generated a visual graph illustrating:

- Domain-to-DNS relationships
- IP address associations



- Infrastructure layout

This visualization helped correlate data collected from Recon-ng and Shodan.

Conclusion

This lab demonstrated the end-to-end OSINT reconnaissance workflow using industry-standard tools. Recon-ng provided a structured, database-driven approach to passive reconnaissance, Shodan revealed exposed services and vulnerabilities across the internet, and Maltego offered visual correlation of infrastructure data. The experiment also highlighted realistic challenges such as API restrictions, module availability issues, authentication requirements, and ethical boundaries, all of which are critical considerations in real-world cybersecurity operations.