

**Name:** Ayilara Busari Dare

## **Assignment 1 & 2**

### **Lab 1: Reconnaissance (Information Gathering)**

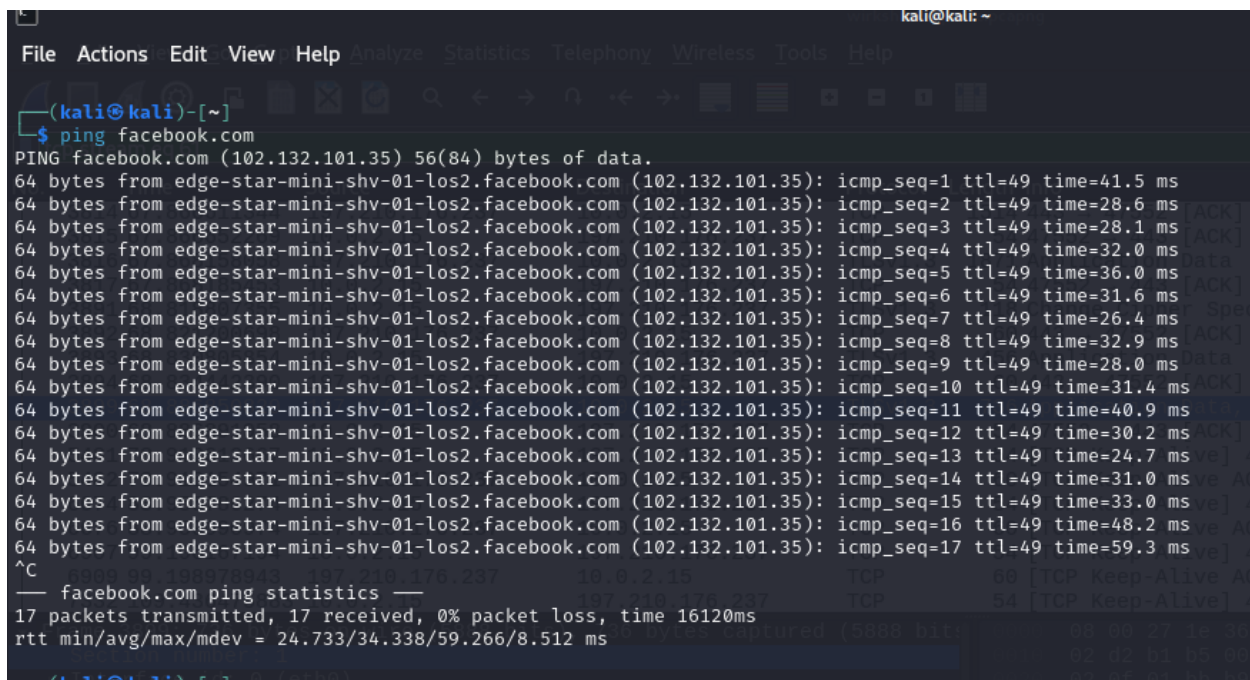
#### **Exercise 1:**

Use the ping command to find the IP addresses of the following domains:

- facebook.com
- twitter.com
- amazon.com

#### **Record Your Answers:**

1. Ping facebook.com: the IP address return by 102.132.101.35, The facebook.com ping statistics is 372 packets transmitted, 367 received, 1.34409% packet loss, time 373125ms rtt min/avg/max/mdev = 39.404/975.674/3740.778/618.839 ms, pipe 4. This was cut while taking long time.
2. Ping tiwtter.com: The return IP address is 104.244.42.1, twitter.com ping statistics 649 packets transmitted, 640 received, 1.38675% packet loss, time 652000ms rtt min/avg/max/mdev = 176.468/1185.798/4370.263/772.553 ms, pipe 5
3. ping Amazon.com: return IP address 54.239.28.8, statistics 650 packets transmitted, 592 received, 8.92308% packet loss, time 661579ms rtt min/avg/max/mdev = 205.048/690.887/4826.043/1116.637 ms, pipe 5



```
kali@kali: ~  
File Actions Edit View Help Analyze Statistics Telephony Wireless Tools Help  
$ ping facebook.com  
PING facebook.com (102.132.101.35) 56(84) bytes of data:  
64 bytes from edge-star-mini-shv-01-los2.facebook.com (102.132.101.35): icmp_seq=1 ttl=49 time=41.5 ms  
64 bytes from edge-star-mini-shv-01-los2.facebook.com (102.132.101.35): icmp_seq=2 ttl=49 time=28.6 ms  
64 bytes from edge-star-mini-shv-01-los2.facebook.com (102.132.101.35): icmp_seq=3 ttl=49 time=28.1 ms  
64 bytes from edge-star-mini-shv-01-los2.facebook.com (102.132.101.35): icmp_seq=4 ttl=49 time=32.0 ms  
64 bytes from edge-star-mini-shv-01-los2.facebook.com (102.132.101.35): icmp_seq=5 ttl=49 time=36.0 ms  
64 bytes from edge-star-mini-shv-01-los2.facebook.com (102.132.101.35): icmp_seq=6 ttl=49 time=31.6 ms  
64 bytes from edge-star-mini-shv-01-los2.facebook.com (102.132.101.35): icmp_seq=7 ttl=49 time=26.4 ms  
64 bytes from edge-star-mini-shv-01-los2.facebook.com (102.132.101.35): icmp_seq=8 ttl=49 time=32.9 ms  
64 bytes from edge-star-mini-shv-01-los2.facebook.com (102.132.101.35): icmp_seq=9 ttl=49 time=28.0 ms  
64 bytes from edge-star-mini-shv-01-los2.facebook.com (102.132.101.35): icmp_seq=10 ttl=49 time=31.4 ms  
64 bytes from edge-star-mini-shv-01-los2.facebook.com (102.132.101.35): icmp_seq=11 ttl=49 time=40.9 ms  
64 bytes from edge-star-mini-shv-01-los2.facebook.com (102.132.101.35): icmp_seq=12 ttl=49 time=30.2 ms  
64 bytes from edge-star-mini-shv-01-los2.facebook.com (102.132.101.35): icmp_seq=13 ttl=49 time=24.7 ms  
64 bytes from edge-star-mini-shv-01-los2.facebook.com (102.132.101.35): icmp_seq=14 ttl=49 time=31.0 ms  
64 bytes from edge-star-mini-shv-01-los2.facebook.com (102.132.101.35): icmp_seq=15 ttl=49 time=33.0 ms  
64 bytes from edge-star-mini-shv-01-los2.facebook.com (102.132.101.35): icmp_seq=16 ttl=49 time=48.2 ms  
64 bytes from edge-star-mini-shv-01-los2.facebook.com (102.132.101.35): icmp_seq=17 ttl=49 time=59.3 ms  
^C  
— facebook.com ping statistics —  
17 packets transmitted, 17 received, 0% packet loss, time 16120ms  
rtt min/avg/max/mdev = 24.733/34.338/59.266/8.512 ms  
$
```

## Exercise 2:

Run the whois command for the following domains:

- github.com
- linkedin.com
- apple.com Answer

## These Questions:

1. What is the registration expiration date for github.com? is Registrar Registration Expiration Date: 2026-10-09T00:00:00+0000
2. Who is the registrar for linkedin.com? the Registrar is MarkMonitor, Inc.
3. What country is the registrant of apple.com from? The country is United States

```
(kali㉿kali)-[~]
$ whois github.com
Domain Name: GITHUB.COM
Registry Domain ID: 1264983250_DOMAIN_COM-VRSN
Registrar WHOIS Server: whois.markmonitor.com
Registrar URL: http://www.markmonitor.com
Updated Date: 2024-09-07T09:16:32Z
Creation Date: 2007-10-09T18:20:50Z
Registry Expiry Date: 2026-10-09T18:20:50Z
Registrar: MarkMonitor Inc.
Registrar IANA ID: 292
Registrar Abuse Contact Email: abusecomplaints@markmonitor.com
Registrar Abuse Contact Phone: +1.2086851750
Domain Status: clientDeleteProhibited https://icann.org/epp#clientDeleteProhibited
Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibited
Domain Status: clientUpdateProhibited https://icann.org/epp#clientUpdateProhibited
Name Server: DNS1.P08.NSONE.NET
Name Server: DNS2.P08.NSONE.NET
Name Server: DNS3.P08.NSONE.NET
Name Server: DNS4.P08.NSONE.NET
Name Server: NS-1283.AWSDNS-32.ORG
Name Server: NS-1707.AWSDNS-21.CO.UK
Name Server: NS-421.AWSDNS-52.COM
Name Server: NS-520.AWSDNS-01.NET
DNSSEC: unsigned
URL of the ICANN Whois Inaccuracy Complaint Form: https://www.icann.org/wicf/
>>> Last update of whois database: 2024-11-07T09:48:55Z <<<
For more information on Whois status codes, please visit https://icann.org/epp
```

## Exercise 3:

Use nslookup to look up DNS information for the following domains:

- bbc.co.uk
- netflix.com

Answer These Questions:

1. What is the IP address for bbc.co.uk? The IP address is 192.168.1.1

2. What are the name servers (NS) for netflix.com? netflix.com

```
(kali㉿kali)-[~]
$ nslookup netflix.com
Server:      192.168.72.171
Address:     192.168.72.171#53
Non-authoritative answer:
Name:   netflix.com
Address: 3.251.50.149
Name:   netflix.com
Address: 54.74.73.31
Name:   netflix.com
Address: 54.155.178.54
Name:   netflix.com
Address: 2a05:d018:76c:b684:8ab7:ac02:667b:e863
Name:   netflix.com
Address: 2a05:d018:76c:b683:a2cd:4240:8669:6d4
Name:   netflix.com
Address: 2a05:d018:76c:b685:e8ab:afd3:af51:3aed
```

## Lab 2: Website Enumeration and Information Gathering

### Exercise 1:

Run the whatweb command to detect technologies for the following targets:

- example.com
- stackoverflow.com
- github.com

### Record Your Findings:

1. example.com: the domain and country names are http://example.com [200 OK] Country[EUROPEAN UNION][EU], it is developed on HTML5, HTTPServer[ECAcc
2. a. stackoverflow.com: http://stackoverflow.com [301 Moved Permanently] Cookies[\_\_cf\_bm,\_cfuvid], Country[UNITED STATES][US], HTTPServer[cloudflare], HttpOnly[\_\_cf\_bm,\_cfuvid], IP[104.18.32.7], RedirectLocation[https://stackoverflow.com/], Title[301 Moved Permanently], UncommonHeaders[x-dns-prefetch-control,cf-ray]  
b. https://stackoverflow.com/ [200 OK] Cookies[\_\_cf\_bm,\_\_cflb,\_cfuvid,prov], Country[UNITED STATES][US], Email[apple-touch-icon@2.png], HTML5, HTTPServer[cloudflare], HttpOnly[\_\_cf\_bm,\_\_cflb,\_cfuvid,prov], IP[104.18.32.7], JQuery[3.7.1], Open-Graph-Protocol, OpenSearch[opensearch.xml],

Script[application/json,text/uri-list,true], StackExchange, Strict-Transport-Security[max-age=15552000], Title[Stack Overflow - Where Developers Learn, Share, & Build Careers], UncommonHeaders[cf-ray,cf-cache-status,content-security-policy,feature-policy,x-request-guid,x-dns-prefetch-control], X-Frame-Options[SAMEORIGIN]

3. GitHub.com: <https://github.com/> [200 OK] Content-Language[en-US], Cookies[\_gh\_sess,\_octo,logged\_in], Country[UNITED STATES][US], HTML5, HTTPServer[GitHub.com], HttpOnly[\_gh\_sess,logged\_in], IP[140.82.121.4], Open-Graph-Protocol[object][1401488693436528], OpenSearch[/opensearch.xml], Script[application/javascript,application/json,text/javascript], Strict-Transport-Security[max-age=31536000; includeSubdomains; preload], Title[GitHub · Build and ship software on a single, collaborative platform · GitHub], UncommonHeaders[x-content-type-options,referrer-policy,content-security-policy,x-github-request-id], X-Frame-Options[deny], X-XSS-Protection[0]

```
(kali@kali)-[~]
$ whatweb github.com
http://github.com [301 Moved Permanently] Country[UNITED STATES][US], IP[140.82.121.3], RedirectLocation[https://github.com/]
https://github.com/ [200 OK] Content-Language[en-US], Cookies[_gh_sess,_octo,logged_in], Country[UNITED STATES][US], HTML5, HTTPServer[GitHub.com], HttpOnly[_gh_sess,logged_in], IP[140.82.121.3], Open-Graph-Protocol[object][1401488693436528], OpenSearch[/opensearch.xml], Script[application/javascript,application/json,text/javascript], Strict-Transport-Security[max-age=31536000; includeSubdomains; preload], Title[GitHub · Build and ship software on a single, collaborative platform · GitHub], UncommonHeaders[x-content-type-options,referrer-policy,content-security-policy,x-github-request-id], X-Frame-Options[deny], X-XSS-Protection[0]

(kali@kali)-[~]
$
192.168.1.15 197.210.170.237 TCP 80 47552 - 443 [ACK] Seq=518 Ack=4518 Win=31800 Len=0
192.168.1.15 197.210.170.237 TLSv1.3 118 Change Cipher Spec, Application Data
192.168.1.15 197.210.170.237 TCP 80 443 - 47552 [ACK] Seq=4518 Ack=582 Win=65535 Len=0
192.168.1.15 197.210.170.237 TLSv1.3 456 Application Data
192.168.1.15 197.210.170.237 TCP 80 443 - 47552 [ACK] Seq=4518 Ack=584 Win=65535 Len=0
```

## Perform Aggressive Scanning Using whatweb

### Exercise 2:

Perform an aggressive scan on the following targets:

- google.com
- facebook.com

### Record Your Findings:

1. google.com: WhatWeb report for <http://www.google.com/> , Status: 200 OK , Title: Google, IP: 216.58.223.228, Country : UNITED STATES, US, HTTP Headers: HTTP/1.1 301 Moved Permanently Location: <http://www.google.com/> Content-Type: text/html; charset=UTF-8 Content-Security-Policy-Report-Only: object-src 'none';base-uri 'self';script-src 'nonce-CaTw9kek-UhTchm8rKwGIA' 'strict-dynamic' 'report-sample' 'unsafe-eval' 'unsafe-inline' https: http;;report-uri https://csp.withgoogle.com/csp/gws/other-hp

Date: Sun, 03 Nov 2024 14:27:16 GMT

Expires: Tue, 03 Dec 2024 14:27:16 GMT

Cache-Control: public, max-age=2592000

Server: gws

Content-Length: 219

X-XSS-Protection: 0

X-Frame-Options: SAMEORIGIN.

Connection: close

HTML version 5, detected by the doctype declaration

- facebook.com: WhatWeb report for [https://web.facebook.com/?\\_rdc=1&\\_rdr&\\_fb\\_noscript=1](https://web.facebook.com/?_rdc=1&_rdr&_fb_noscript=1) Status : 200 OK Title <None> IP: 163.70.147.22 Country : FRANCE, FR, HTML version 5, detected by the doctype declaration, Password Field, Script, X-XSS-Protection and HTTP Headers

```
(kali@kali)-[~]
$ whatweb --aggression 3 -v google.com
WhatWeb report for http://google.com
Status : 301 Moved Permanently
Title : 301 Moved
IP : 216.58.223.238
Country : UNITED STATES, US
Summary : HTTPServer[gws], RedirectLocation[http://www.google.com/], UncommonHeaders[content-security-policy-report-only], X-Frame-
tion[0]
Detected Plugins:
[ HTTPServer ]
  HTTP server header string. This plugin also attempts to
  identify the operating system from the server header.
  String : gws (from server string)
[ RedirectLocation ]
  HTTP Server string location. used with http-status 301 and
  302
  String : http://www.google.com/ (from location)
[ UncommonHeaders ]
  Uncommon HTTP server headers. The blacklist includes all
  the standard headers and many non standard but common ones.
  Interesting but fairly common headers should have their own
```

## Lab 3: Subdomain Hunting

### Exercise 1:

Run the sublist3r command for the following domains:

- github.com
- google.com

### Record Your Findings:

- Subdomains for github.com: Total Unique Subdomains found is 95, here are some list  
www.github.com  
atom-installer.github.com  
branch.github.com

brandguide.github.com  
camo.github.com  
central.github.com  
cla.github.com  
classroom.github.com  
cloud.github.com  
f.cloud.github.com  
codespaces.github.com  
codespaces-dev.github.com  
codespaces-ppe.github.com  
communication.github.com  
www.communication.github.com  
m.communication.github.com  
res.communication.github.com  
t.communication.github.com  
community.github.com  
docs.github.com  
docs-front-door.github.com  
dodgeball.github.com  
workspaces-ppe.github.com

## 2. Subdomains for google.com: Total Unique Subdomains Found: 97

www.google.com  
accounts.google.com  
freezone.accounts.google.com  
adwords.google.com  
qa.adz.google.com  
answers.google.com  
apps-secure-data-connector.google.com  
audioads.google.com  
checkout.google.com  
mtv-da-1.ad.corp.google.com  
ads-compare.eem.corp.google.com  
da.ext.corp.google.com  
m.guts.corp.google.com  
m.gutsdev.corp.google.com

login.corp.google.com  
mtv-da.corp.google.com  
mygeist.corp.google.com  
mygeist2010.corp.google.com  
proxyconfig.corp.google.com  
reseed.corp.google.com  
twdsalesgsa.twd.corp.google.com  
uberproxy.corp.google.com

```
File Actions Edit View Help
tcp.stream eq 6
No. Time Source Destination Protocol Length Info
3814 67.866511344 197.210.176.237 197.210.176.237 TCP 1514 44
3815 67.866522269 197.210.176.237 197.210.176.237 TCP 54 47
3816 67.866533194 197.210.176.237 197.210.176.237 TLSv1.3 1871 Ap
3817 67.866544119 197.210.176.237 197.210.176.237 TCP 54 47
[-] Enumerating subdomains now for google.com 197.210.176.237
[-] Searching now in Baidu.. 197.210.176.237 TLSv1.3 118 Ch
[-] Searching now in Yahoo.. 197.210.176.237 TCP 60 44
[-] Searching now in Google.. 197.210.176.237 TLSv1.3 456 Ap
[-] Searching now in Bing.. 197.210.176.237 TCP 60 44
[-] Searching now in Ask.. 197.210.176.237 TLSv1.3 60 Ap
[-] Searching now in Netcraft.. 197.210.176.237 TCP 54 47
[-] Searching now in DNSdumpster.. 197.210.176.237 TCP 54 [T
[-] Searching now in Virustotal.. 197.210.176.237 TCP 60 [T
[-] Searching now in ThreatCrowd.. 197.210.176.237 TCP 54 [T
[-] Searching now in SSL Certificates.. 197.210.176.237 TCP 60 [T
[-] Searching now in PassiveDNS.. 197.210.176.237 TCP 54 [T
[!] Error: Virustotal probably now is blocking our requests 197.210.176.237 TCP 60 [T
[-] Total Unique Subdomains Found: 97 197.210.176.237 TCP 54 [T
www.google.com 736 bytes on wire (5888 bits), 736 bytes captured (5888 bit)
accounts.google.com 0 (eth0)
freezone.accounts.google.com on type: Ethernet (1)
adwords.google.com Nov 6, 2024 06:57:06.703279863 EST
qa.adz.google.com 4 11:57:06.703279863 UTC
answers.google.com Time: 1730894226.703279863
audioads.google.com on this packet: 0.0000000000 seconds]
checkout.google.com previous captured frame: 0.040073816 seconds]
mtv-da-1.ad.corp.google.com is displayed frame: 0.052408439 seconds]
ads-compare.eem.corp.google.com
```

## Exercise 2:

Perform a directory discovery scan on the following targets:

- <http://example.com>



- <http://example.org>

**Record Your Findings:**

1. Directories, for example.com: ENERATED WORDS: 4612

Scanning URL: <http://example.com/>

2. Directories for example.org: The Generated words is 4612 and DOWNLOADED: 1234 - FOUND: 0

**Exercise 3:**

Use theHarvester to gather information on the following domain:

- example.com

**Record Your Findings:**

- Emails and Information Gathered: No IPs found; Emails found: 7

anything@mailexample.com

dana@example.com

email@example.com

mail@example.com

someone@example.com

username@example.com

what-is-@example.com

Hosts found: 8

.example.com

WWW.example.com

a.example.com

api.example.com

app.example.com

mail.example.com

ns1.example.com

subdomain.example.com



```
Kali Linux
File Actions Edit View Help
Read proxies.yaml from /home/kali/.theHarvester/proxies.yaml
*****
* Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Ha
* theHarvester
* theHarvester 4.5.1
* Coded by Christian Martorella
* Edge-Security Research
* cmartorella@edge-security.com
*
*****
[*] Target: example.com

Created default api-keys.yaml at /home/kali/.theHarvester/api-keys.yaml
Searching 0 results.
[*] Searching Bing.

[*] No IPs found.

[*] Emails found: 7
anything@mailexample.com
dana@example.com
email@example.com
mail@example.com
someone@example.com
username@example.com
what-is-@example.com

[*] Hosts found: 8
```

**Lab 4: Basic Port Scanning**

**Exercise 1:**

Perform a basic port scan on your OWASP VM IP address and record your findings:

- **Open Ports:** 9 open ports were found and the ports are  
22/tcp , 80/tcp, 139/tcp, 143/tcp, 443/tcp, 445/tcp, 5001/tcp, 8080/tcp and 8081/tcp

```
kali-linux-2024.1-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Kali Linux
File Actions Edit View Help
(kali@kali)-[~]
$ nmap 192.168.72.135
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-05 08:08 EST
Nmap scan report for 192.168.72.135
Host is up (0.014s latency).
Not shown: 991 filtered tcp ports (no-response)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
139/tcp   open  netbios-ssn
143/tcp   open  imap
443/tcp   open  https
445/tcp   open  microsoft-ds
5001/tcp  open  complex-link
8080/tcp  open  http-proxy
8081/tcp  open  blackice-icecap
Nmap done: 1 IP address (1 host up) scanned in 4.98 seconds
(kali@kali)-[~]
$
```

## Exercise 2:

Perform an aggressive scan on your OWASP VM IP address and record your findings:

- **Service Versions:** are OpenSSH 5.3p1 Debian 3ubuntu4 (Ubuntu Linux; protocol 2.0), Apache httpd 2.2.14 ((Ubuntu) mod\_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch proxy\_html/3.0.1 mod\_python/3.3.1 Python/2.6.5 mod\_ssl/2.2.14 OpenSSL...), netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP), Courier Imapd (released 2008), Apache httpd 2.2.14 ((Ubuntu) mod\_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch proxy\_html/3.0.1 mod\_python/3.3.1 Python/2.6.5 mod\_ssl/2.2.14 OpenSSL...), netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP), object Java Object Serialization, Apache Tomcat/Coyote JSP engine 1.1 and Jetty 6.1.25
- **Operating System:** Linux (software version 3.1.0.22)

```
File Machine View Input Devices Help
kali@kali: ~
File Actions Edit View Help
$ sudo nmap -sV -O 192.168.72.135

[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-05 08:12 EST
Nmap scan report for 192.168.72.135
Host is up (0.0019s latency).
Not shown: 991 filtered tcp ports (no-response)
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 5.3p1 Debian 3ubuntu4 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http         Apache httpd 2.2.14 ((Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL...)
139/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
143/tcp   open  imap         Courier Imapd (released 2008)
443/tcp   open  ssl/http     Apache httpd 2.2.14 ((Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL...)
445/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
5001/tcp  open  java-object  Java Object Serialization
8080/tcp  open  http         Apache Tomcat/Coyote JSP engine 1.1
8081/tcp  open  http         Jetty 6.1.25
1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port5001-TCP:V=7.94SVN:XI=7&D=11/5%Time=672A19C4%P=x86_64-pc-linux-gnu%r
SF:(NULL,4,"\\xac\\xed\\x05");
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: bridge|general purpose|switch
Running (JUST GUESSING): Oracle Virtualbox (96%), QEMU (91%), Bay Networks embedded (86%)
OS CPE: cpe:/o:oracle:virtualbox cpe:/a:qemu:qemu cpe:/h:baynetworks:baystack_450
Aggressive OS guesses: Oracle Virtualbox (96%), QEMU user mode network gateway (91%), Bay Networks BayStack 450 switch (software version 3.1.0.22) (86%)
No exact OS matches for host (test conditions non-ideal).
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 23.05 seconds
```

Exercise 3:

Conduct a vulnerability scan on your OWASP VM IP address and record your findings:

• Vulnerabilities: Different Vulnerabilities were identified and this show between tow port, port 22 and 80tcp, such vulnerabilities are:

- 1. **Cross-domain and Client Access policies**, the state is vulnerable, Description is overly permissive configurations enables Cross-site Request Forgery attacks and may allow third parties to access sensitive data meant for the user.
- 2. **http-vuln-cve2011-3192**: this is VULNERABLE to Apache byterange filter DoS, IDs: CVE:CVE-2011-3192 BID:49303. The Apache web server is vulnerable to a denial-of-service attack when numerous overlapping byte ranges are requested.
- 3. http-vuln-cve2014-3704: ERROR: Script execution failed (use -d to debug)  
ssl-dh-params:  
VULNERABLE: Diffie-Hellman Key Exchange Insufficient Group Strength State:  
VULNERABLE, Transport Layer Security (TLS) services that use Diffie-Hellman groups of insufficient strength, especially those using one of a few commonly shared groups, may be susceptible to passive eavesdropping attacks.
- 4. http-vuln-cve2011-3192: VULNERABLE:

Apache byterange filter DoS

State: VULNERABLE

IDs: CVE:CVE-2011-3192 BID:49303

The Apache web server is vulnerable to a denial-of-service attack when numerous overlapping byte ranges are requested.

## 5. ssl-ccs-injection:

VULNERABLE: SSL/TLS MITM vulnerability (CCS Injection)

State: VULNERABLE

Risk factor: High

OpenSSL before 0.9.8za, 1.0.0 before 1.0.0m, and 1.0.1 before 1.0.1h does not properly restrict processing of Change CipherSpec messages, which allows man-in-the-middle attackers to trigger use of a zero length master key in certain OpenSSL-to-OpenSSL communications, and consequently hijack sessions or obtain sensitive information, via a crafted TLS handshake, aka the "CCS Injection" vulnerability.

## 6. Host script results:

smb-vuln-regsvcs-dos:

VULNERABLE: Service regsvcs in Microsoft Windows systems vulnerable to denial of service

State: VULNERABLE

The service regsvcs in Microsoft Windows 2000 systems is vulnerable to denial of service caused by a null deference pointer. This script will crash the service if it is vulnerable. This vulnerability was discovered by Ron Bowes while working on smb-enum-sessions.

```
File Actions Edit View Help
kali@kali: ~
$ nmap --script vuln 192.168.72.135
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-05 08:47 EST
Nmap scan report for 192.168.72.135
Host is up (0.017s latency).
Not shown: 991 filtered tcp ports (no-response)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
| http-cross-domain-policy:
| VULNERABLE:
| Cross-domain and Client Access policies.
| State: VULNERABLE
| A cross-domain policy file specifies the permissions that a web client such as Java, Adobe Flash, Adobe Reader,
| etc. use to access data across different domains. A client access policy file is similar to cross-domain policy
| but is used for M$ Silverlight applications. Overly permissive configurations enables Cross-site Request
| Forgery attacks, and may allow third parties to access sensitive data meant for the user.
| Check results:
| /crossdomain.xml:
| <?xml version="1.0"?>
| <!DOCTYPE cross-domain-policy SYSTEM "http://www.macromedia.com/xml/dtds/cross-domain-policy.dtd">
| <cross-domain-policy>
|   <allow-access-from domain="*" />
| </cross-domain-policy>
| Extra information:
| Trusted domains:*
|
| References:
| http://acunetix.com/vulnerabilities/web/insecure-clientaccesspolicy-xml-file
| https://www.owasp.org/index.php/Test_RIA_cross_domain_policy_%28OTG-CONFIG-008%29
| http://gurusekalra.blogspot.com/2013/08/bypassing-same-origin-policy-with-flash.html
| https://www.adobe.com/devnet/articles/crossdomain_policy_file_spec.html
```

#### Exercise 4:

- Perform a vulnerability scan on your OWASP VM and record your findings: using Nikto
- Vulnerabilities Found: here are the vulnerabilities found
  - Server: Apache/2.2.14 (Ubuntu) mod\_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch proxy\_html/3.0.1 mod\_python/3.3.1 Python/2.6.5 mod\_ssl/2.2.14 OpenSSL/0.9.8k Phusion\_Passenger/4.0.38 mod\_perl/2.0.4 Perl/v5.10.1
  - Server may leak inodes via ETags, header found with file /, inode: 286483, size: 28067, mtime: Thu Jul 30 22:55:52 2015. See: <http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2003-1418>
  - The anti-clickjacking X-Frame-Options header is not present. See: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options>
  - The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: <https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/>
  - /cgi-bin/: Directory indexing found.
  - crossdomain.xml contains a full wildcard entry. See: <http://jeremiahgrossman.blogspot.com/2008/05/crossdomainxml-invites-cross-site.html>
  - images: IP address found in the 'location' header. The IP is "127.0.1.1". See: [https://portswigger.net/kb/issues/00600300\\_private-ip-addresses-disclosed](https://portswigger.net/kb/issues/00600300_private-ip-addresses-disclosed)
  - images: The web server may reveal its internal or real IP in the Location header via a request to with HTTP/1.0. The value is "127.0.1.1". See: <http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2000-0649>
  - index: Uncommon header 'tcn' found, with contents: list.
  - index: Apache mod\_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. The following alternatives for 'index' were found: index.css, index.html. See: <http://www.wisec.it/sectou.php?id=4698ebdc59d15>, <https://exchange.xforce.ibmcloud.com/vulnerabilities/8275>
  - favicon.ico: identifies this app/server as: owasp.org. See: <https://en.wikipedia.org/wiki/Favicon>
  - Python/2.6.5 appears to be outdated (current is at least 3.9.6).
  - mod\_perl/2.0.4 appears to be outdated (current is at least 2.0.11).
  - proxy\_html/3.0.1 appears to be outdated (current is at least 3.1.2).
  - Apache/2.2.14 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.

- mod\_python/3.3.1 appears to be outdated (current is at least 3.5.0).
- PHP/5.3.2-1ubuntu4.30 appears to be outdated (current is at least 8.1.5), PHP 7.4.28 for the 7.4 branch.
- mod\_mono/2.4.3 appears to be outdated (current is at least 3.12).
- mod\_ssl/2.2.14 appears to be outdated (current is at least 2.9.6) (may depend on server version).
- OpenSSL/0.9.8k appears to be outdated (current is at least 3.0.7). OpenSSL 1.1.1s is current for the 1.x branch and will be supported until Nov 11 2023.
- Phusion\_Passenger/4.0.38 appears to be outdated (current is at least 6.0.7).
- Perl/v5.10.1 appears to be outdated (current is at least v5.32.1).

```

kali@kali: ~
File Actions Edit View Help

(kali@kali)~$ nikto -h http://192.168.72.135
- Nikto v2.5.0

+ Target IP:      192.168.72.135
+ Target Hostname: 192.168.72.135
+ Target Port:    80
+ Start Time:     2024-11-05 09:22:48 (GMT-5)

+ Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL/0.9.8k Phusion_Passenger/4.0.38 mod_perl/2.0.4 Perl/v5.10.1
+ /: Server may leak inodes via ETags, header found with file /, inode: 286483, size: 28067, mtime: Thu Jul 30 22:55:52 2015. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2003-1418
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ /cgi-bin/: Directory indexing found.
+ /crossdomain.xml contains a full wildcard entry. See: http://jeremiahgrossman.blogspot.com/2008/05/crossdomainxml-invites-cross-site.html
+ /images: IP address found in the 'location' header. The IP is "127.0.1.1". See: https://portswigger.net/kb/issues/00600300_private-ip-addresses-disclosed
+ /images: The web server may reveal its internal or real IP in the Location header via a request to with HTTP/1.0. The value is "127.0.1.1". See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2000-0649
+ /index: Uncommon header 'tcn' found, with contents: list.
+ /index: Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. The following alternatives for 'index' were found: index.css, index.html. See: http://www.wisec.it/sectou.php?id=4698ebdc59d15,https://exchange.xforce.ibmcloud.com/vulnerabilities/8275
+ /favicon.ico: identifies this app/server as: owasp.org. See: https://en.wikipedia.org/wiki/Favicon
+ Python/2.6.5 appears to be outdated (current is at least 3.9.6).
+ mod_perl/2.0.4 appears to be outdated (current is at least 2.0.11).
+ proxy_html/3.0.1 appears to be outdated (current is at least 3.1.2).
+ Apache/2.2.14 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ mod_python/3.3.1 appears to be outdated (current is at least 3.5.0).
+ PHP/5.3.2-1ubuntu4.30 appears to be outdated (current is at least 8.1.5), PHP 7.4.28 for the 7.4 branch.
+ mod_mono/2.4.3 appears to be outdated (current is at least 3.12).
+ mod_ssl/2.2.14 appears to be outdated (current is at least 2.9.6) (may depend on server version).
```

## Lab 5: Wireshark

### Exercise 1:

- Explore the Wireshark GUI. Identify and list the main components you see, including where to find the Statistics menu.

Files, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, other futures include shark tail for start capture packets, stop, filters and the types of network to be capture.

### Exercise 2:

- Capture network traffic using both Wireshark and tshark. Compare the two methods and note any differences in the user experience.

Both Wireshark and tshark were able to capture packet in real time and provide details of the use network communications or browser, however there are both have differences such that Wireshark is user interface friendly which is easy to navigate, search or filter for anything you are looking for, and the capture packets were grouped under headings while in Tshark majority of mention were not there although it was also in arranged in such a way that you will notice source ip, destination ip and protocols.

### Analyzing Captured Packets

1. Analyze the captured packets in the Packet List Pane.

The pane was examined by applying filter to search for different packet list.

2. Apply display filters to isolate specific types of traffic. Common filters include:

- Filter for HTTP traffic: http, when filter http, many things were observed such as Request method: GET, Request URI show the site visited and Request Version: HTTP/1.1, host: ww5.owaspbwa.org\r\n, user-Agent: Mozilla/5.0 on linux used to access the website, cookies: expiry\_partner=; caf\_ipaddr=102.89.68.183, and county = NG
- Filter for DNS traffic: dns, one dns show that facebook.com was access between source ip 192.168.72.171 and destination ip 10.0.2.15.
- Filter for specific IP addresses: ip.addr == 192.168.72.171, This show request and response, Transaction ID: 0x0f49 and encrypted.
- Filter for TCP packets: tcp, Frame 22: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface eth0, id 0  
Section number: 1, Interface id: 0 (eth0), Encapsulation type: Ethernet (1)  
Arrival Time: Nov 6, 2024 07:04:28.401917592 EST, UTC Arrival Time: Nov 6, 2024 12:04:28.401917592 UTC, Epoch Arrival Time: 1730894668.401917592, Frame Length: 54 bytes (432 bits), Capture Length: 54 bytes (432 bits).  
Ethernet II, Src: PCSSystemtec\_1e:36:4a (08:00:27:1e:36:4a), Dst: 52:54:00:12:35:02 (52:54:00:12:35:02). Internet Protocol Version 4, Src: 10.0.2.15, Dst: 142.250.200.142  
  
Transmission Control Protocol, Src Port: 37454, Dst Port: 443, Seq: 518, Ack: 6201, Len: 0. The analyse protocol header was carried out and anomalies or suspicious patterns found.

### Exercise 3:

- Use filters to analyze different types of traffic. Record the following:
  - Number of HTTP packets captured: 8 http packets capture
  - Number of DNS packets captured: 375 of dns packets were capture
  - Specific IP addresses you identified in the traffic: 216.58.223.226

### Exercise 4:

- Select a packet and list the following information:



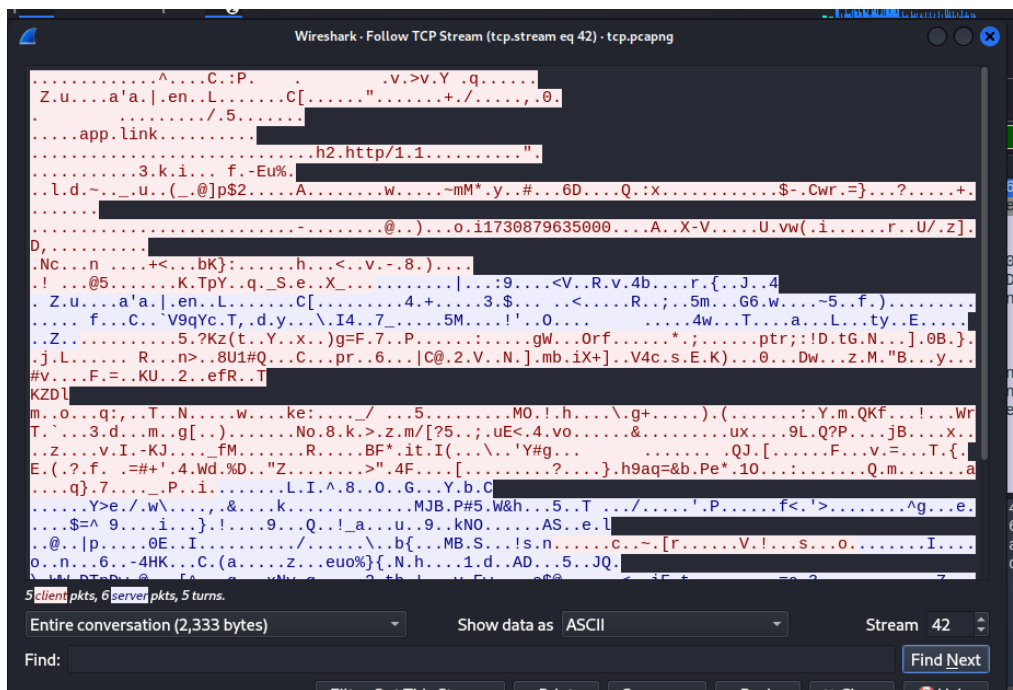
- Source IP: 10.0.2.15
- Destination IP: 192.168.72.171
- Protocol: DNS
- Any TCP Flags observed: Yes, Flags: 0x0100 Standard query

## Advanced Packet Analysis Techniques

### Exercise 5:

- Follow a TCP stream for a specific session and summarize the data exchanged between the client and server.

The TCP stream was followed there is data exchange between 5 clients and 6 server. No plain text observed, everything are encrypted.



### Exercise 6:

- Take a screenshot of the Protocol Hierarchy and analyze the data. Which protocol is most prevalent in your capture?

Wireshark - Protocol Hierarchy Statistics - tcp.pcapng

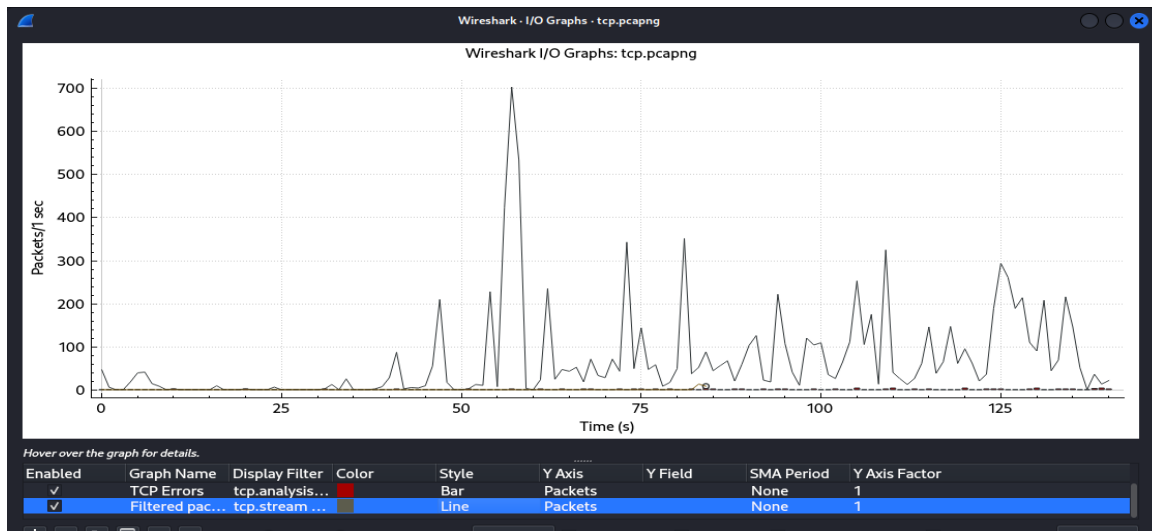
Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits
Frame	100.0	21	100.0	3523	69 k	0	0	0
Ethernet	100.0	21	9.3	326	6,427	0	0	0
Internet Protocol Version 4	100.0	21	11.9	420	8,281	0	0	0
Transmission Control Protocol	100.0	21	78.8	2777	54 k	10	224	4,416
Transport Layer Security	52.4	11	66.2	2333	46 k	11	2333	46 k

Display filter: tcp.stream eq 42

Close Copy Protocols Help

## Exercise 7:

- Create an IO Graph showing TCP traffic. Describe any noticeable patterns you observe:



IO graph show TCP traffic base on packet transmit over time (packet/1sec). about 50 packets were over a period of 3 seconds and 700 packets over period of 55 sec, there are up and down graph which show the number of transmitted per sec.

## Exercise 8:

- Save your capture file and describe a scenario where you would need to review this data later. What specific findings do you hope to extract?

The capture file saved as capturefile.pcapng and it can be used for further analysis while investigate any wrong or malicious activities.

## **Practical Applications of Wireshark**

### **Exercise 9:**

- Describe a real-world scenario where you would use Wireshark to troubleshoot a network issue. What specific symptoms would you investigate?

In a real world scenario Wireshark can be used to capture real time and troubleshoot a network, if a network is slow than usual and I will analyze the total number packet that travel per seconds, this give me insight of what problem might likely to be, whether is a deliberate action to flood the network so that it can be slow or network poor transmission.

### **Exercise 10:**

- Identify at least two potential security threats in your captured traffic. What indicators led you to suspect these activities? No suspicious activities were found.

## **Lab 6: Advanced Packet Analysis Techniques**

### **Exercise 1:**

- Describe the purpose of the SYN and ACK flags in the TCP handshake. How do these flags indicate the status of a connection?

The purpose of the SYN and ACK flags in the TCP handshake is a method established communication between two devices on a network, Syn will ask for permission and Ack acknowledged the response for readiness for connection and communication.

### **Exercise 2:**

- Choose an HTTP packet and summarize its request method, status code, and any notable headers. What can you infer about the transaction?

The request Method: GET, Request Version: HTTP/1.1

Status code: 2.0

User-Agent: Mozilla/5.0

### **Exercise 3:**

- Identify a DNS query and its corresponding response. What information does the response provide, and how is it structured? 0... .... = Response: Message is a query

### **Exercise 4:**

- Create a custom filter that captures only TCP traffic from your machine to a specific target IP. Document the filter syntax and the packets captured. `tcp and ip.src == 10.0.2.15`

### **Exercise 5:**

- Write a filter that captures traffic on a specific port (e.g., HTTP port 80) and analyze the results. What packets were captured? `http and tcp.port == 80`

## Exercise 6:

- Analyze your capture for any anomalies or indicators of potential vulnerabilities. Document your findings and suggest possible remediation steps. No vulnerability observed

## Exercise 7:

- Capture HTTPS traffic and identify the initial handshake packets. What information is exchanged during this handshake, and how does it contribute to security?

Nothing see to analyze due encryption.

## Exercise 8:

- Prepare a brief report summarizing your findings during the assessment. Include potential risks and recommended actions.

Use of filter bar to check a specific protocol. Ip address and port, the analysis was carryout to Identified network protocols used, to detect anomalies or unusual traffic patterns, inspect packet contents for sensitive information. And verify packet integrity and authenticity. During this analysis no potential risks were identified.

## Exercise 9:

- Create a capture report that includes your objectives, methods, key findings, and any recommendations for improving network security.

