NAME: Aminu Ibrahim Jalo **DATE:03/11/24**

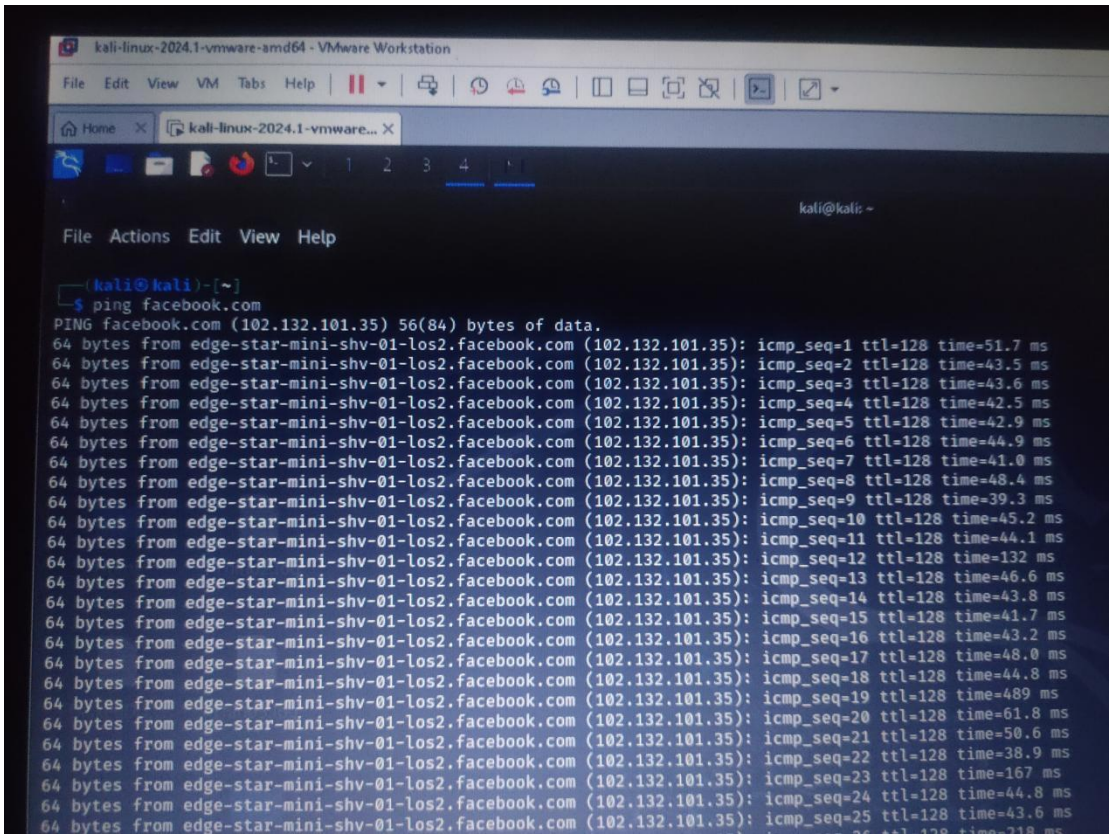
ID: IDEAS/24/7357

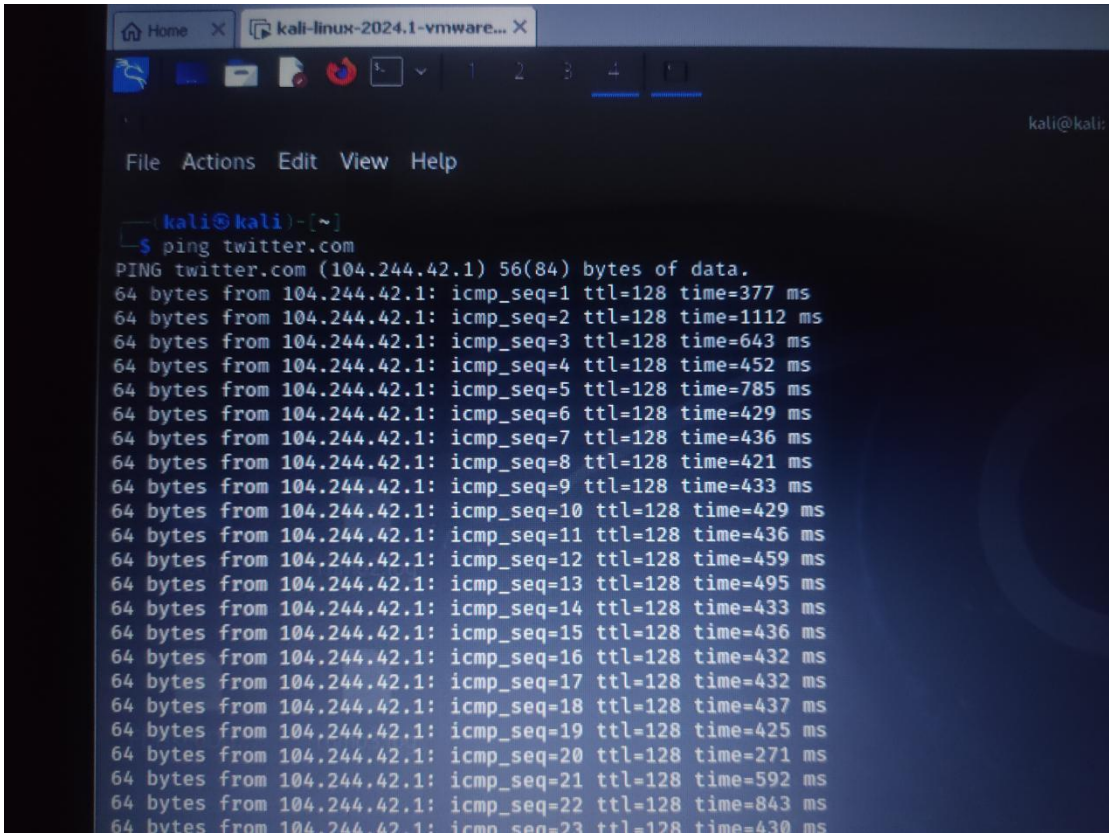
COURSE: INT 302 Assignment

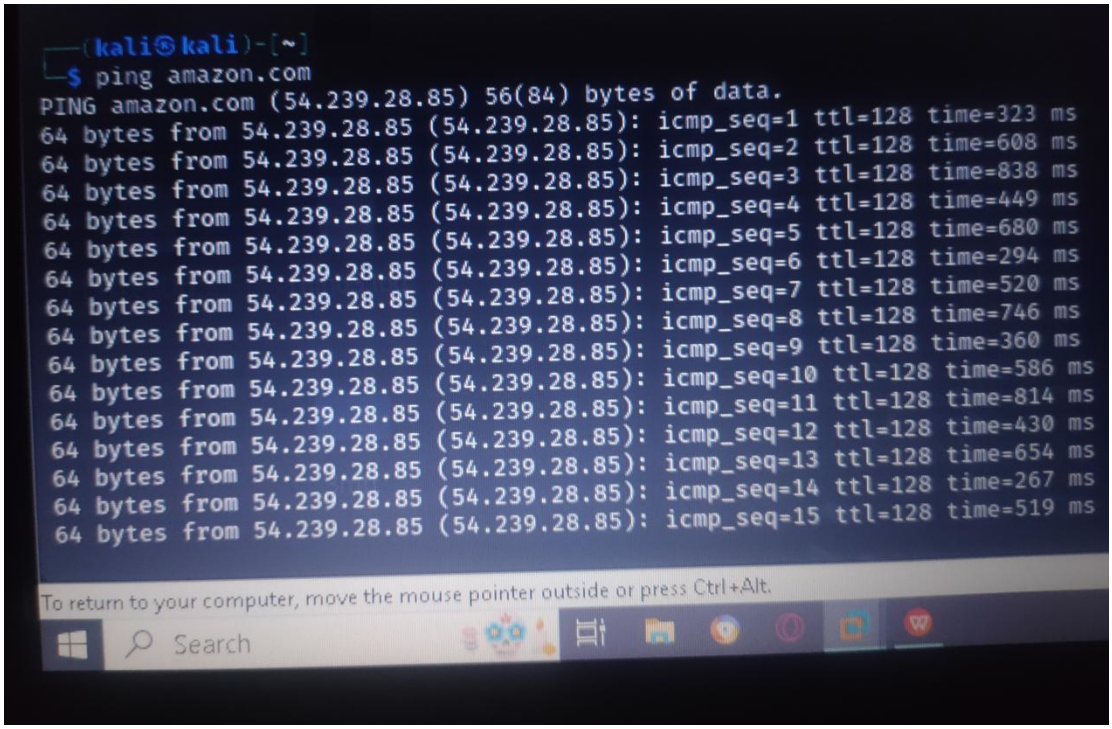
LAB 1:

Reconnaissance (Information Gathering)

Step 1: Get the IP Address of a Domain Using ping
Record Your Answers:
1. facebook.com: 102.132.101.35
2. twitter.com:104.244.42.1
3. amazon.com:54.239.28.85



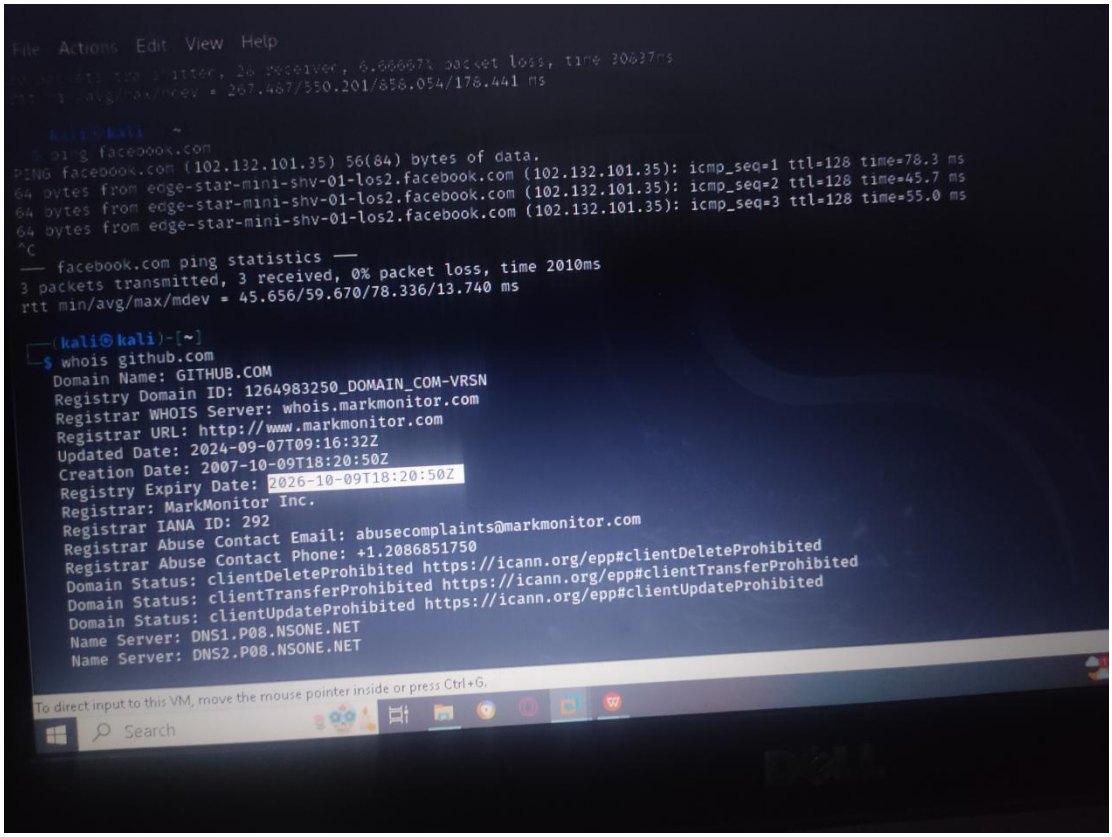




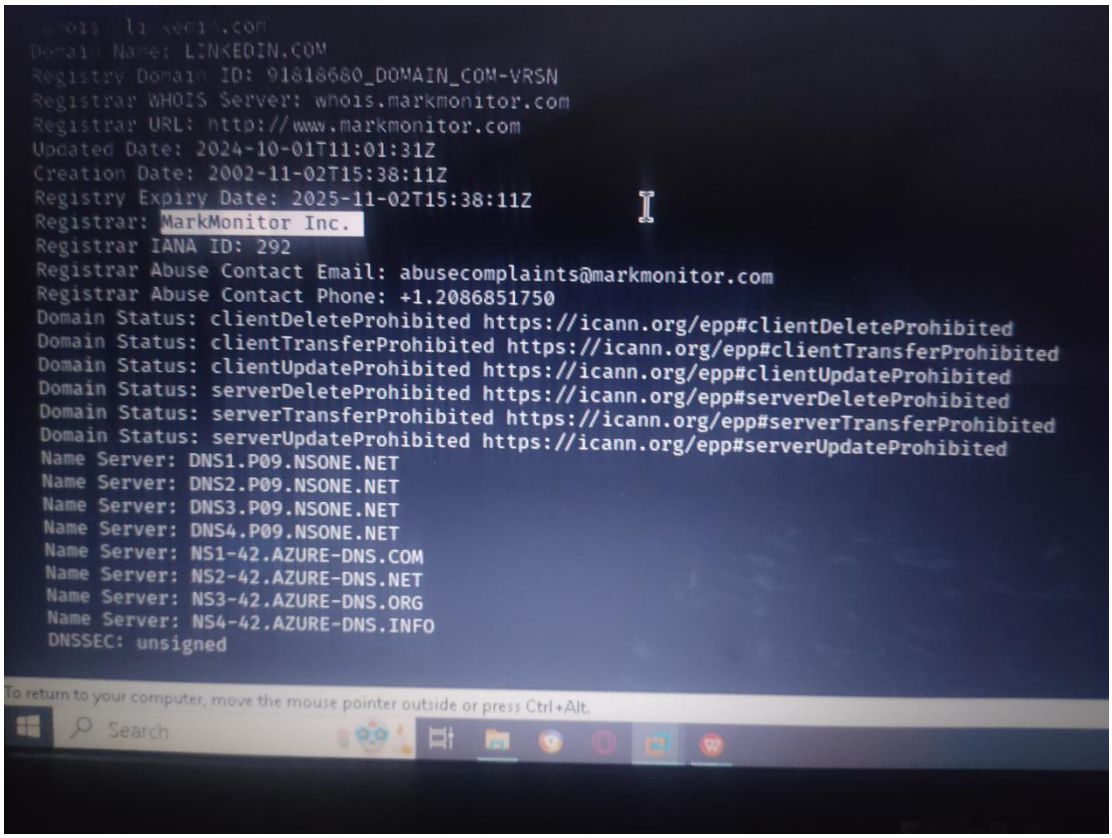
Step 2: Retrieve Domain Registration Details Using whois
Answer These Questions:

1. What is the registration expiration date for github.com?

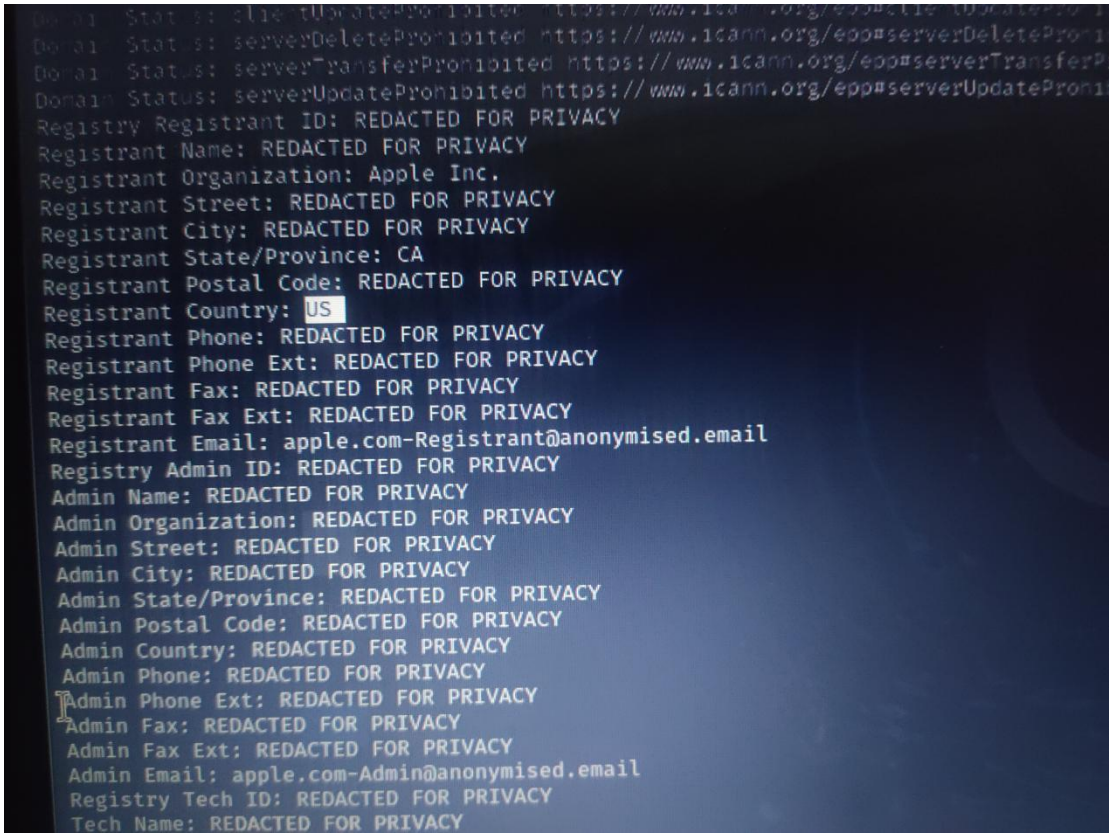
2026-10-09T18:20:50Z



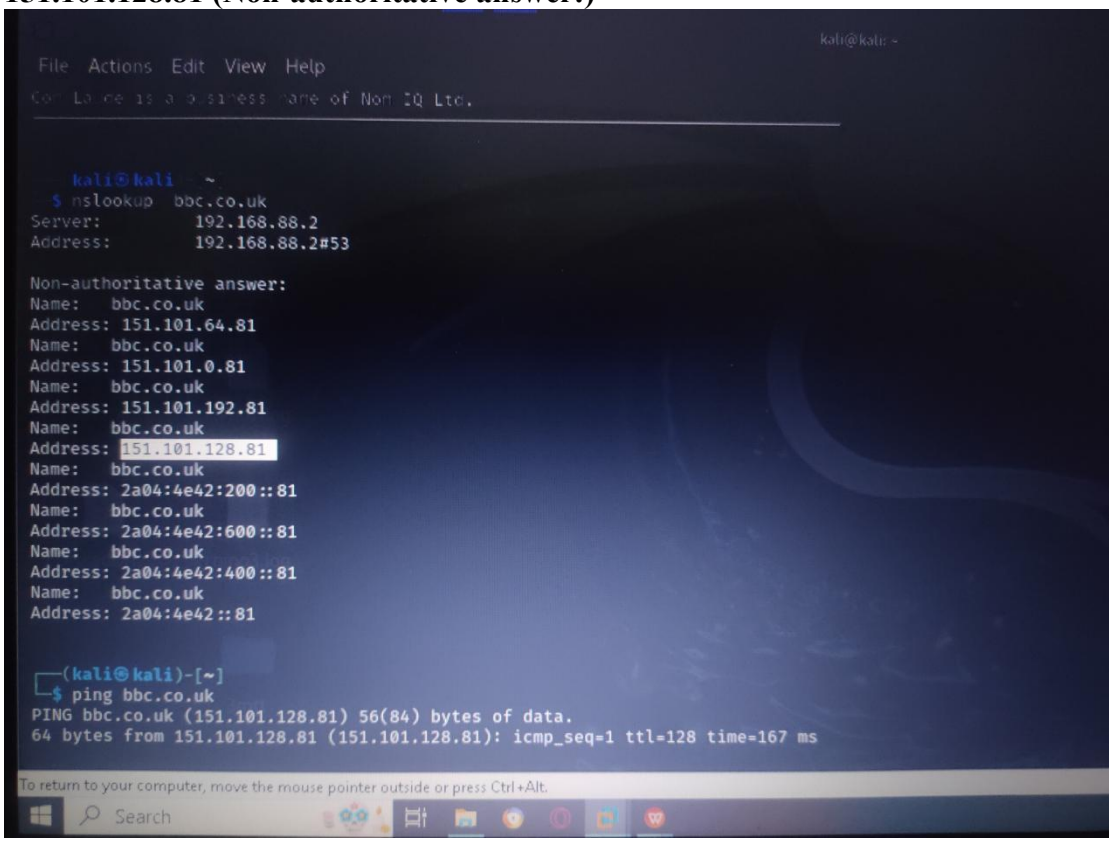
2. Who is the registrar for linkedin.com?
MarkMonitor Inc.



3. What country is the registrant of apple.com from?
US

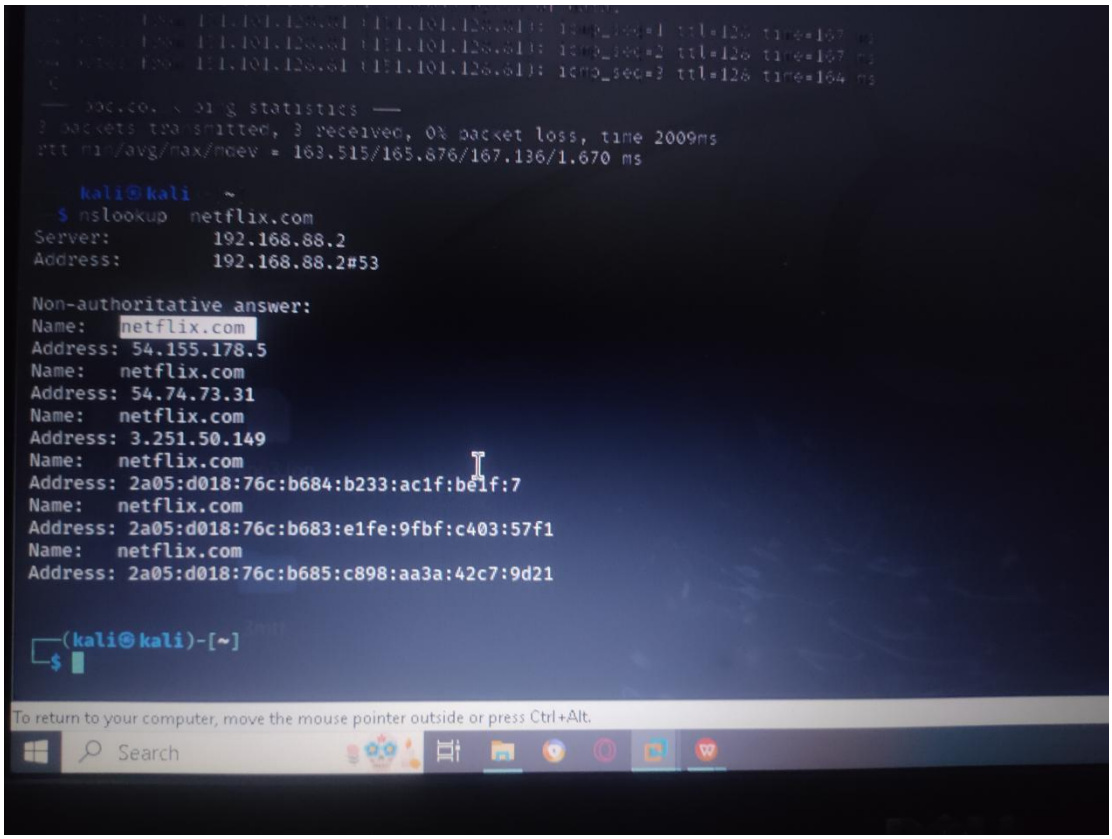


Step 3: Perform a DNS Lookup Using nslookup
Answer These Questions:
1. What is the IP address for bbc.co.uk?
151.101.128.81 (Non-authoritative answer:)



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

netflix.com



Lab 2:

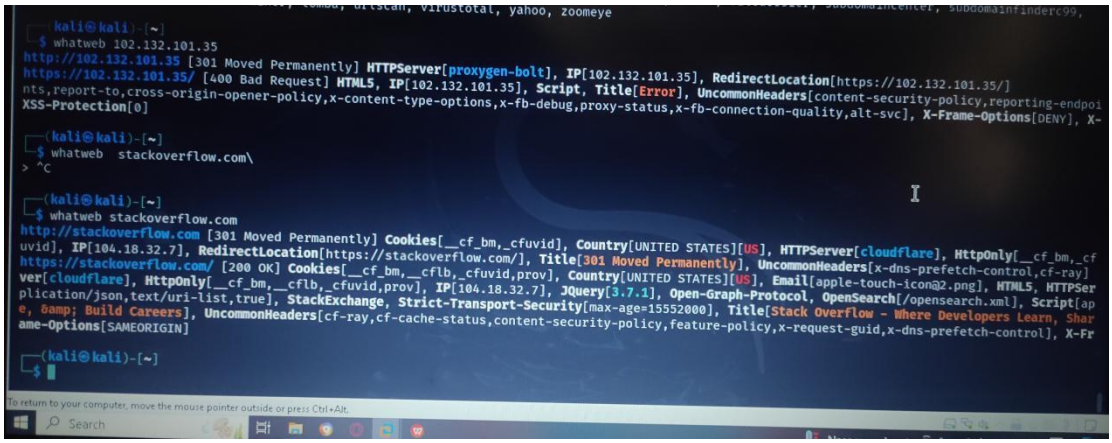
Website Enumeration and Information Gathering

Exercise 1:

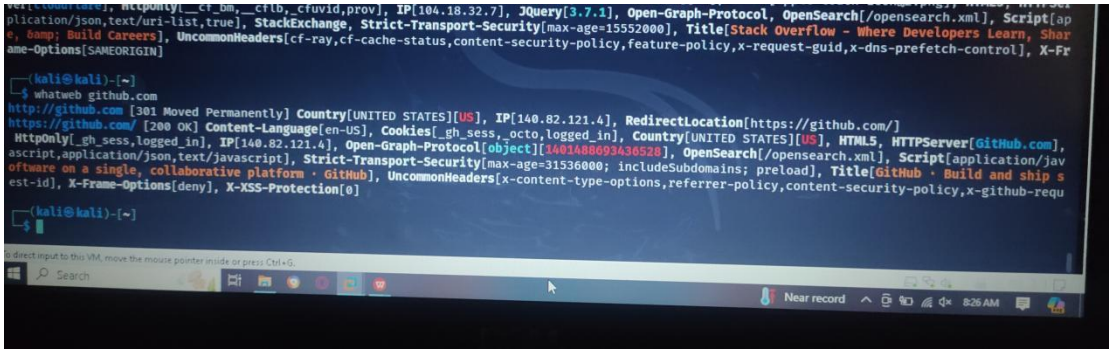
Run the whatweb command to detect technologies for the following targets:

•102.132.101.35

• stackoverflow.com



• github.com

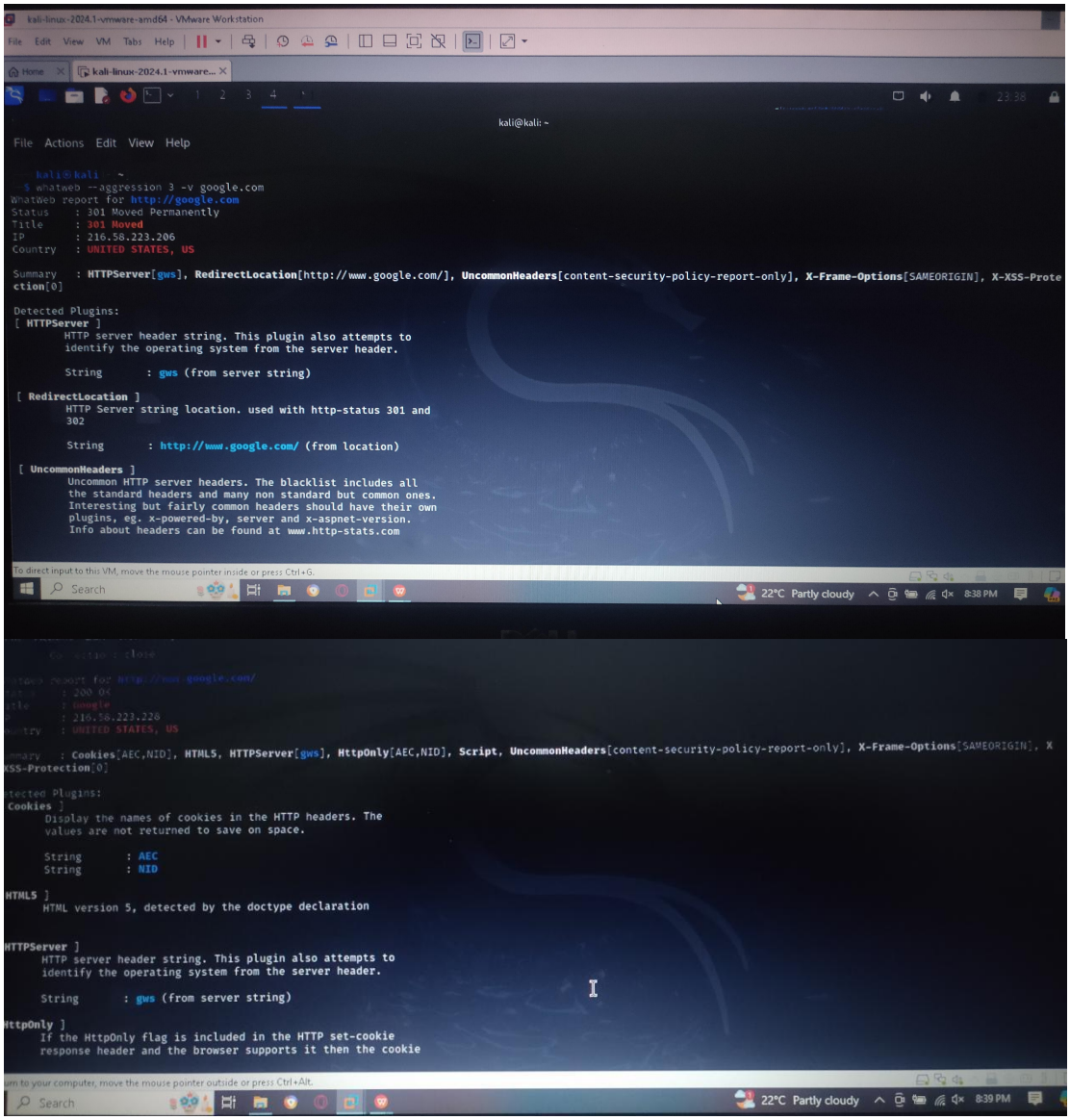


Step 2: Perform Aggressive Scanning Using whatweb
Exercise 2:
Perform an aggressive scan on the following targets:
• google.com

• facebook.com

Record Your Findings:
1. **google.com:**
WhatWeb report for http://www.google.com/
Status : 301 Moved Permanently
Title : 301 Moved
IP : 216.58.223.206
Country : UNITED STATES, US
Summary : HTTPServer[gws], RedirectLocation[http://www.google.com/], UncommonHeaders[content-security-policy-report-only], X-Frame-Options[SAMEORIGIN], X-XSS-Protection[0]

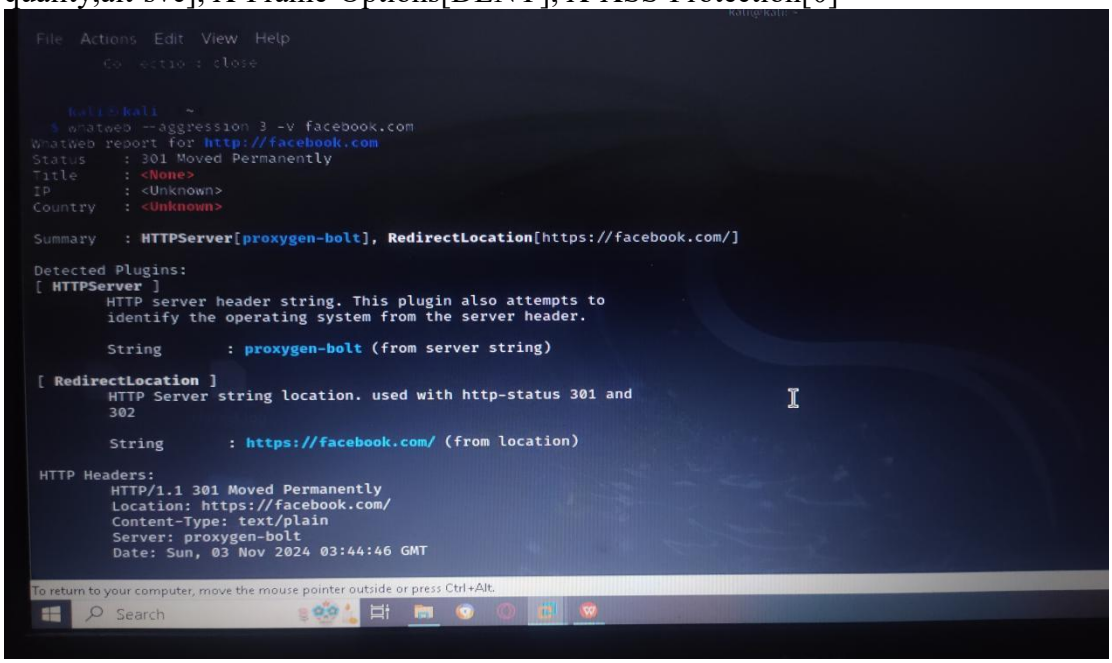
WhatWeb report for http://www.google.com/
Status : 200 OK
Title : Google
IP : 216.58.223.228
Country : UNITED STATES, US
Summary : Cookies[AEC,NID], HTML5, HTTPServer[gws], HttpOnly[AEC,NID], Script, UncommonHeaders[content-security-policy-report-only], X-Frame-Options[SAMEORIGIN], X-XSS-Protection[0]

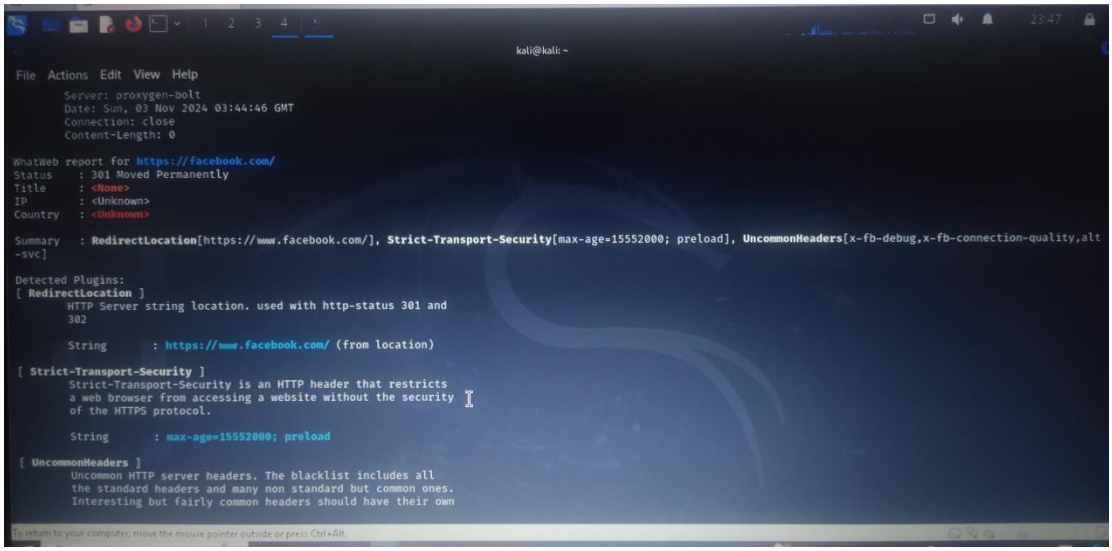


2. **facebook.com**:
WhatWeb report for http://facebook.com
Status : 301 Moved Permanently
Title : <None>
IP : <Unknown>
Country : <Unknown>
Summary : HTTPServer[proxygen-bolt], RedirectLocation[https://facebook.com/]
WhatWeb report for https://facebook.com/
Status : 301 Moved Permanently
Title : <None>
IP : <Unknown>
Country : <Unknown>
Summary : RedirectLocation[https://www.facebook.com/], Strict-Transport-Security[max-age=15552000; preload], UncommonHeaders[x-fb-debug,x-fb-connection-quality,alt-svc]

WhatWeb report for https://www.facebook.com/
Status : 302 Found

Title : <None>
IP : <Unknown>
Country : <Unknown>
Summary : RedirectLocation[https://web.facebook.com/?_rdc=1&_rdr], Strict-Transport-Security[max-age=15552000; preload], UncommonHeaders[reporting-endpoints,report-to,cross-origin-opener-policy,x-fb-zr-redirect,x-fb-debug,x-fb-connection-quality,alt-svc]
WhatWeb report for https://web.facebook.com/?_rdc=1&_rdr
Status : 200 OK
Title : <None>
IP : <Unknown>
Country : <Unknown>
Summary : Cookies[fr,sb], HTML5, HttpOnly[fr,sb], Meta-Refresh-Redirect[/?_rdc=1&_rdr&_fb_noscript=1], PasswordField[pass], Script[application/ld+json,text/javascript], Strict-Transport-Security[max-age=15552000; preload], UncommonHeaders[reporting-endpoints,report-to,content-security-policy,document-policy,permissions-policy,cross-origin-resource-policy,cross-origin-opener-policy,x-content-type-options,x-fb-debug,x-fb-connection-quality,alt-svc], X-Frame-Options[DENY], X-XSS-Protection[0]
WhatWeb report for https://web.facebook.com/?_rdc=1&_rdr&_fb_noscript=1
Status : 200 OK
Title : <None>
IP : <Unknown>
Country : <Unknown>
Summary : Cookies[fr,noscript,sb], HTML5, HttpOnly[fr,sb], PasswordField[pass], Script[application/ld+json,text/javascript], Strict-Transport-Security[max-age=15552000; preload], UncommonHeaders[reporting-endpoints,report-to,content-security-policy,document-policy,permissions-policy,cross-origin-resource-policy,cross-origin-opener-policy,x-content-type-options,x-fb-debug,x-fb-connection-quality,alt-svc], X-Frame-Options[DENY], X-XSS-Protection[0]





Lab 3:
Subdomain Hunting

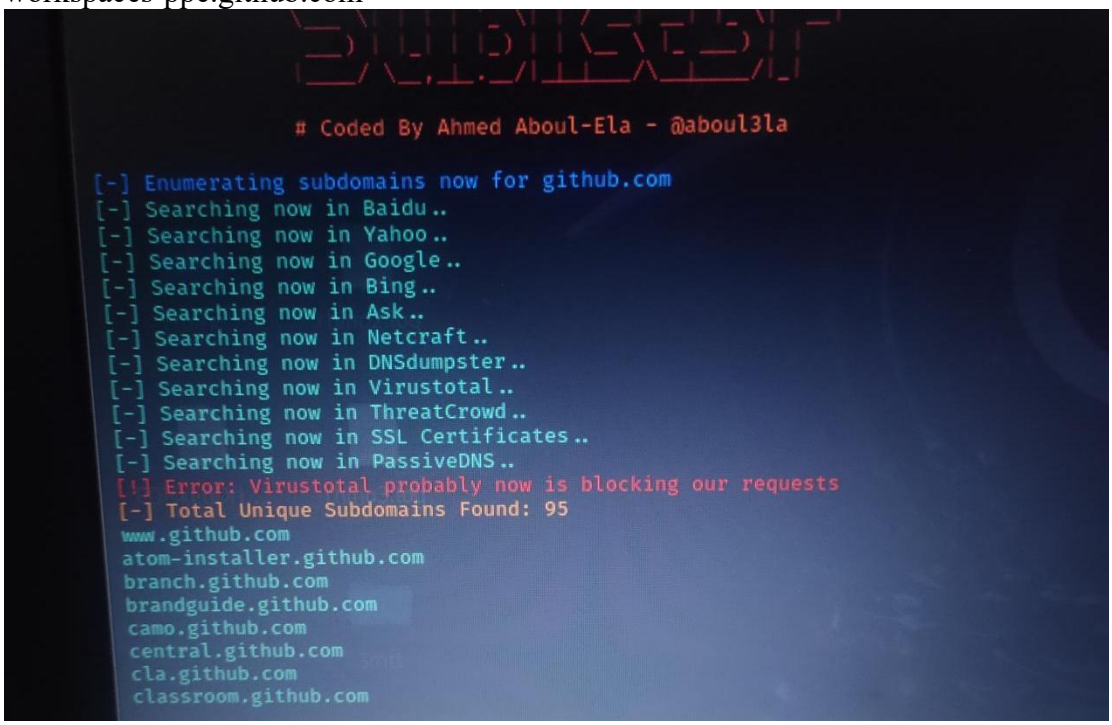
Exercise 1:
Record Your Findings:

1. Subdomains for github.com:
Enumerating subdomains now for github.com
[-] Searching now in Baidu..
[-] Searching now in Yahoo..
[-] Searching now in Google..
[-] Searching now in Bing..
[-] Searching now in Ask..
[-] Searching now in Netcraft..
[-] Searching now in DNSdumpster..
[-] Searching now in Virustotal..
[-] Searching now in ThreatCrowd..
[-] Searching now in SSL Certificates..
[-] Searching now in PassiveDNS..

[-] Total Unique Subdomains Found: 95
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
edu.github.com
education.github.com
emails.github.com
enterprise.github.com
support.enterprise.github.com
www.support.enterprise.github.com
examregistration.github.com
examregistration-api.github.com
examregistration-uat.github.com
examregistration-uat-api.github.com
fast.github.com
garage.github.com
gist.github.com
graphql.github.com
www.graphql.github.com
graphql-stage.github.com
www.graphql-stage.github.com
help.github.com
helpnext.github.com
hq.github.com
vpn-ca.iad.github.com
id.github.com
import.github.com
import2.github.com
importer2.github.com
jira.github.com
www.jira.github.com
jobs.github.com
lab.github.com
lab-sandbox.github.com
learn.github.com
mac-installer.github.com
maintainers.github.com
www.maintainers.github.com
octostatus-production.github.com
offer.github.com
partnerportal.github.com
www.partnerportal.github.com
pkg.github.com
porter.github.com
porter2.github.com
proxima-review-lab.github.com
raw.github.com

registry.github.com
render.github.com
render-lab.github.com
www.render-lab.github.com
review-lab.github.com
octocaptcha.review-lab.github.com
rs.github.com
schrauger.github.com
api.security.github.com
www.api.security.github.com
skyline.github.com
www.skyline.github.com
slack.github.com
smtp.github.com
www.smtp.github.com
staging-lab.github.com
api.stars.github.com
www.api.stars.github.com
status.github.com
stg.github.com
styleguide.github.com
ws.support.github.com
www.ws.support.github.com
talks.github.com
visualstudio.github.com
www.visualstudio.github.com
vscode-auth.github.com
workspaces.github.com
workspaces-dev.github.com
workspaces-ppe.github.com

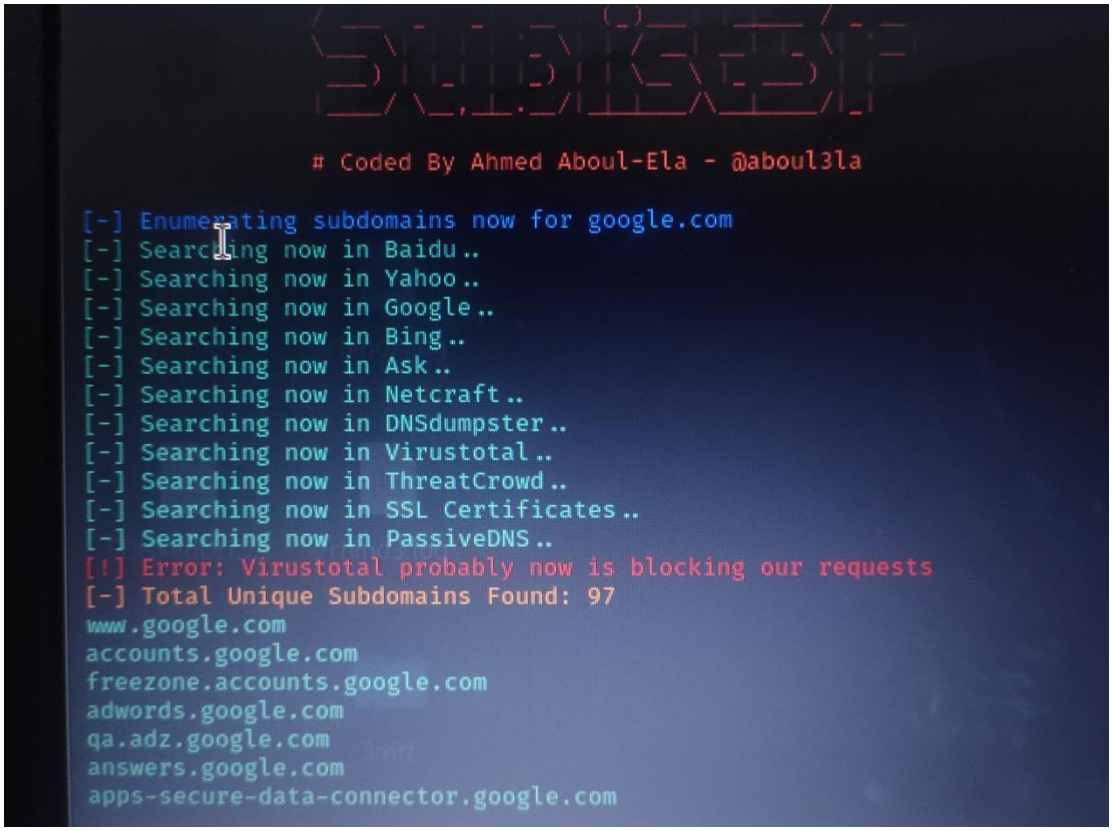


2Subdomains for google.com:
Enumerating subdomains now for google.com
[-] Searching now in Baidu..
[-] Searching now in Yahoo..
[-] Searching now in Google..
[-] Searching now in Bing..
[-] Searching now in Ask..
[-] Searching now in Netcraft..
[-] Searching now in DNSdumpster..
[-] Searching now in Virustotal..
[-] Searching now in ThreatCrowd..
[-] Searching now in SSL Certificates..
[-] Searching now in PassiveDNS..

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
uberproxy-nocert.corp.google.com
uberproxy-san.corp.google.com
ext.google.com
cag.ext.google.com
cod.ext.google.com
da.ext.google.com
eggroll.ext.google.com
fra-da.ext.google.com
glass.ext.google.com
glass-eur.ext.google.com
glass-mtv.ext.google.com
glass-twd.ext.google.com
hot-da.ext.google.com

hyd-da.ext.google.com
ice.ext.google.com
meeting.ext.google.com
mtv-da.ext.google.com
soaproxyprod01.ext.google.com
soaproxytest01.ext.google.com
spdy-proxy.ext.google.com
spdy-proxy-debug.ext.google.com
twd-da.ext.google.com
flexpack.google.com
www.flexpack.google.com
accounts.flexpack.google.com
gaiastaging.flexpack.google.com
mail.flexpack.google.com
plus.flexpack.google.com
search.flexpack.google.com
freezone.google.com
www.freezone.google.com
accounts.freezone.google.com
gaiastaging.freezone.google.com
mail.freezone.google.com
news.freezone.google.com
plus.freezone.google.com
search.freezone.google.com
gmail.google.com
hosted-id.google.com
jmt0.google.com
aspmx.l.google.com
alt1.aspmx.l.google.com
alt2.aspmx.l.google.com
alt3.aspmx.l.google.com
alt4.aspmx.l.google.com
gmail-smtp-in.l.google.com
alt1.gmail-smtp-in.l.google.com
alt2.gmail-smtp-in.l.google.com
alt3.gmail-smtp-in.l.google.com
alt4.gmail-smtp-in.l.google.com
gmr-smtp-in.l.google.com
alt1.gmr-smtp-in.l.google.com
alt2.gmr-smtp-in.l.google.com
alt3.gmr-smtp-in.l.google.com
alt4.gmr-smtp-in.l.google.com
vp.video.l.google.com
m.google.com
freezone.m.google.com
mail.google.com
freezone.mail.google.com
misc.google.com
misc-sni.google.com
mtalk.google.com

mx.google.com
ics.prod.google.com
sandbox.google.com
cert-test.sandbox.google.com
ecc-test.sandbox.google.com
services.google.com
talk.google.com
upload.google.com
dg.video.google.com
upload.video.google.com
wifi.google.com
onex.wifi.google.com



Step 3: Information Gathering Using theHarvester

Exercise 3:

Use theHarvester to gather information on the following domain:

• tiktok.com

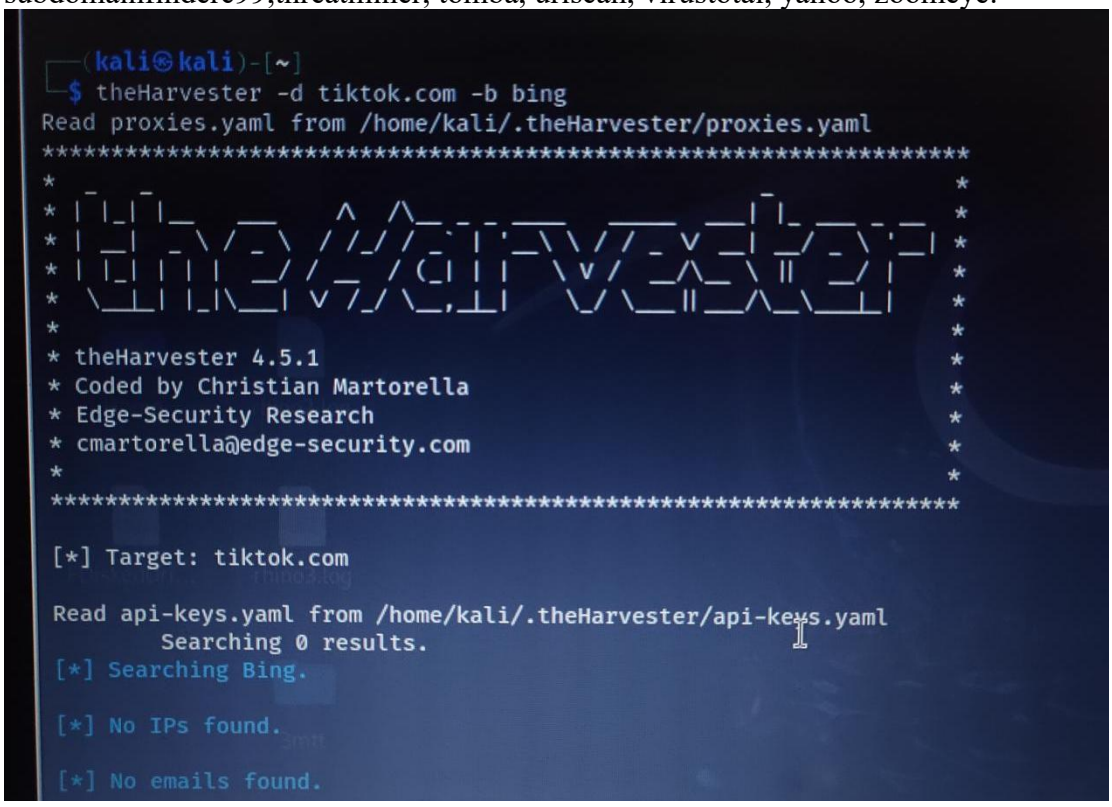
Findings:

• Emails and Information Gathered:

Target: tiktok.com
Searching Bing.
No IPs found.
No emails found.
Hosts found: 0

usage: theHarvester [-h] -d DOMAIN [-l LIMIT] [-S START] [-p] [-s] [--screenshot SCREENSHOT] [-v] [-e DNS_SERVER] [-t] [-r [DNS_RESOLVE]] [-n]
[-c] [-f FILENAME] [-b SOURCE]

-b SOURCE, --source SOURCE
anubis, baidu, bevigil, binaryedge, bing, bingapi, bufferoverun, brave, censys, certspotter, criminalip, crtsh, dnsdumpster, duckduckgo, fullhunt, github-code, hackertarget, hunter, hunterhow, intelx, netlas, onyphe, otx, pentesttools, projectdiscovery, rapiddns, rocketreach, securityTrails, sitedossier, subdomaincenter, subdomainfinderc99,threatminer, tomba, urlscan, virustotal, yahoo, zoomeye.



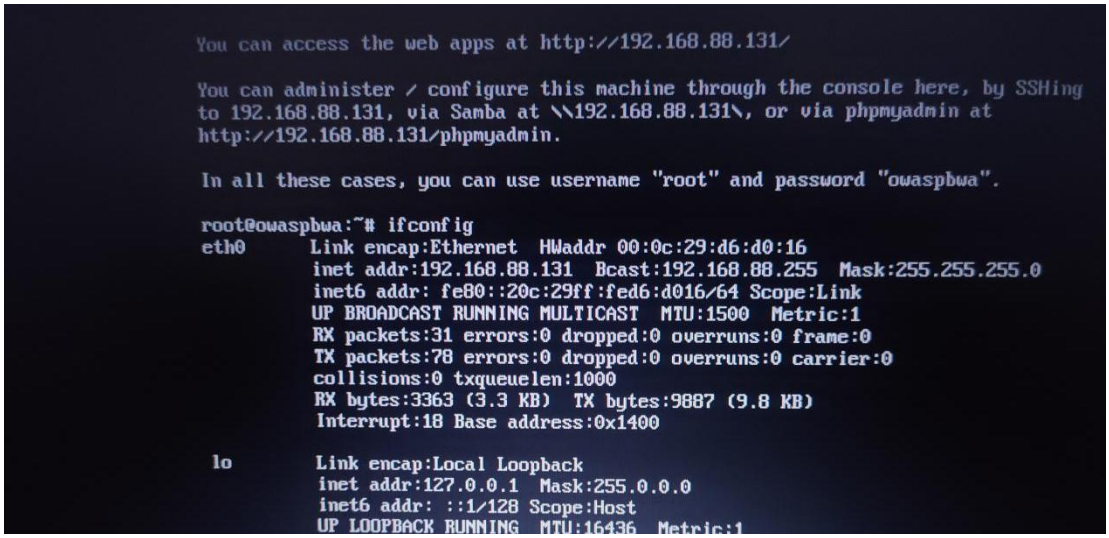
Lab 4:

Basic Port Scanning

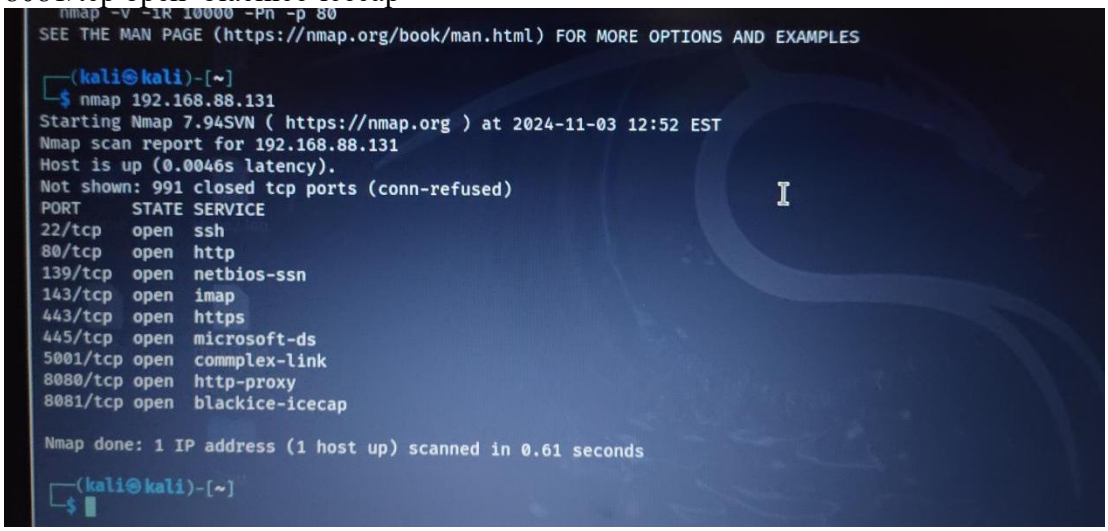
Step 1: Gather the IP Address of Your OWASP VM

Record the IP Address:

• OWASP VM IP Address:**192.168.88.131**



Step 2: Basic Port Scanning with nmap
Exercise 1:
Perform a basic port scan on your OWASP VM IP address and record your findings:
• Open Ports:
Host is up (0.0046s latency).
Not shown: 991 closed tcp ports (conn-refused)
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 commplex-link
8080/tcp open http-proxy
8081/tcp open blackice-icecap

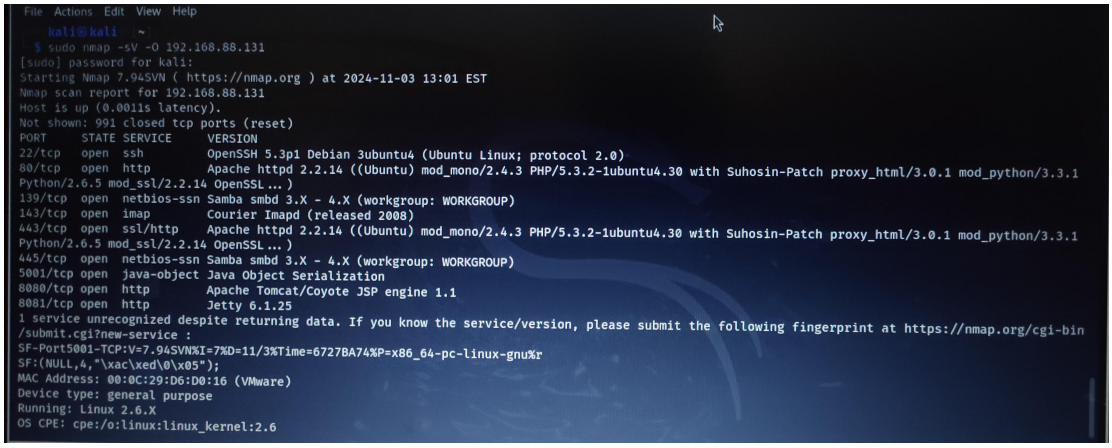


Step 3: Aggressive Scanning with nmap
nmap -sV -O 192.168.88.131
requires root privileges

Exercise 2:
Perform an aggressive scan on your OWASP VM IP address and record your findings:
• **Service Versions:**
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

• **Operating System:**
Linux 2.6.17 - 2.6.36



Step 4: Vulnerability Scanning with nmap
Exercise 3:
Conduct a vulnerability scan on your OWASP VM IP address and record your findings:
• Vulnerabilities:
Nmap scan report for 192.168.88.131
Host is up (0.083s latency).
Not shown: 991 closed tcp ports (conn-refused)
PORT STATE SERVICE
22/tcp open ssh
80/tcp open http
| 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.
| Disclosure date: 2011-08-19
| References:
| https://seclists.org/fulldisclosure/2011/Aug/175
| https://www.tenable.com/plugins/nessus/55976
| https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-3192
|_ https://www.securityfocus.com/bid/49303
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
| http-internal-ip-disclosure:
|_ Internal IP Leaked: 127.0.1.1
|_http-dombased-xss: Couldn't find any DOM based XSS.
|_http-sql-injection: ERROR: Script execution failed (use -d to debug)
| 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 acces 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://gursevkalra.blogspot.com/2013/08/bypassing-same-origin-policy-with-
flash.html
|       http://sethsec.blogspot.com/2014/03/exploiting-misconfigured-
crossdomainxml.html
|       https://www.adobe.com/devnet-
docs/acrobatetk/tools/AppSec/CrossDomain_PolicyFile_Specification.pdf
|       http://acunetix.com/vulnerabilities/web/insecure-clientaccesspolicy-xml-file
|       https://www.owasp.org/index.php/Test_RIA_cross_domain_policy_%28OTG-
CONFIG-008%29
|_      https://www.adobe.com/devnet/articles/crossdomain_policy_file_spec.html
|_http-trace: TRACE is enabled
| http-csrf:
| Spidering limited to: maxdepth=3; maxpagecount=20; withinhost=192.168.88.131
|   Found the following possible CSRF vulnerabilities:
|
|     Path: http://192.168.88.131:80/shepherd/login.jsp
|     Form id:
|_    Form action: login
| http-enum:
|_  /wordpress/: Blog
139/tcp  open  netbios-ssn
143/tcp  open  imap
443/tcp  open  https
|_http-aspnet-debug: ERROR: Script execution failed (use -d to debug)
| ssl-poodle:
|   VULNERABLE:
|   SSL POODLE information leak
|     State: VULNERABLE
|     IDs:  CVE:CVE-2014-3566  BID:70574
```

```
|       The SSL protocol 3.0, as used in OpenSSL through 1.0.1i and other
|       products, uses nondeterministic CBC padding, which makes it easier
|       for man-in-the-middle attackers to obtain cleartext data via a
|       padding-oracle attack, aka the "POODLE" issue.
|     Disclosure date: 2014-10-14
|     Check results:
|       TLS_RSA_WITH_AES_128_CBC_SHA
|     References:
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-3566
|       https://www.securityfocus.com/bid/70574
|       https://www.imperialviolet.org/2014/10/14/poodle.html
|_      https://www.openssl.org/~bodo/ssl-poodle.pdf
| http-dombased-xss:
| Spidering limited to: maxdepth=3; maxpagecount=20; withinhost=192.168.88.131
|   Found the following indications of potential DOM based XSS:
|
|     Source: document.write("Hello, " +
document.URL.substring(pos,document.URL.length)
|     Pages: http://192.168.88.131:443/dom-xss-example.html
|
|     Source: document.write('<FORM METHOD="GET"
ACTION="'+location.href+'">Enter your name:<input name="name"><input
type="submit" value="Submit"></form>')
|_    Pages: http://192.168.88.131:443/dom-xss-example.html
| http-csrf:
| Spidering limited to: maxdepth=3; maxpagecount=20; withinhost=192.168.88.131
|   Found the following possible CSRF vulnerabilities:
|
|     Path: https://192.168.88.131:443/dom-xss-example.html
|     Form id:
|     Form action: '+location.href+'
|
|     Path: https://192.168.88.131:443/wordpress/
|     Form id: searchform
|     Form action: https://192.168.88.131/wordpress/
|
|     Path: https://192.168.88.131:443/shepherd/login.jsp
|     Form id:
|     Form action: login
|
|     Path: https://192.168.88.131:443/AppSensorDemo/login.jsp
|     Form id:
|     Form action: Login
|
|     Path: https://192.168.88.131:443/phpBB2/
|     Form id:
|     Form action: login.php?sid=32d25fb12d3c41d612e16c6aea0ef072
|
|     Path: https://192.168.88.131:443/WackoPicko/
|     Form id: query2
```

```
|     Form action: /WackoPicko/pictures/search.php
|
|     Path: https://192.168.88.131:443/WackoPicko/
|     Form id:
|_    Form action: /WackoPicko/pic' + 'check' + '.php
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
| 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 ChangeCipherSpec 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.
|
|     References:
|       http://www.openssl.org/news/secadv_20140605.txt
|       http://www.cvedetails.com/cve/2014-0224
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-0224
|_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.
|     Check results:
|       WEAK DH GROUP 1
|             Cipher Suite: TLS_DHE_RSA_WITH_AES_128_CBC_SHA
|             Modulus Type: Safe prime
|             Modulus Source: mod_ssl 2.2.x/1024-bit MODP group with safe prime
modulus
|             Modulus Length: 1024
|             Generator Length: 8
|             Public Key Length: 1024
|     References:
|_      https://weakdh.org
|_http-vuln-wnr1000-creds: ERROR: Script execution failed (use -d to debug)
445/tcp  open  microsoft-ds
5001/tcp open  commplex-link
8080/tcp open  http-proxy
| http-slowloris-check:
|   VULNERABLE:
|   Slowloris DOS attack
|     State: LIKELY VULNERABLE
|     IDs:  CVE:CVE-2007-6750
```

```
|       Slowloris tries to keep many connections to the target web server open and hold
|       them open as long as possible.  It accomplishes this by opening connections to
|       the target web server and sending a partial request. By doing so, it starves
|       the http server's resources causing Denial Of Service.
|
|     Disclosure date: 2009-09-17
|     References:
|       http://ha.ckers.org/slowloris/
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2007-6750
| http-cookie-flags:
|   /manager/html/upload:
|     JSESSIONID:
|       httponly flag not set
|   /manager/html:
|     JSESSIONID:
|_      httponly flag not set
| http-enum:
|   /examples/: Sample scripts
|   /manager/html/upload: Apache Tomcat (401 Unauthorized)
|   /manager/html: Apache Tomcat (401 Unauthorized)
|_  /docs/: Potentially interesting folder
8081/tcp open  blackice-icecap

Host script results:
|_samba-vuln-cve-2012-1182: Could not negotiate a connection:SMB: ERROR:
Server returned less data than it was supposed to (one or more fields are missing);
aborting [14]
| smb-vuln-regsvc-dos:
|   VULNERABLE:
|   Service regsvc in Microsoft Windows systems vulnerable to denial of service
|     State: VULNERABLE
|       The service regsvc 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.
|_
|_smb-vuln-ms10-061: Could not negotiate a connection:SMB: ERROR: Server
returned less data than it was supposed to (one or more fields are missing); aborting
[14]
|_smb-vuln-ms10-054: false

Nmap done: 1 IP address (1 host up) scanned in 558.24 seconds
```

Step 5: Web Vulnerability Scanning with nikto

Exercise 4:

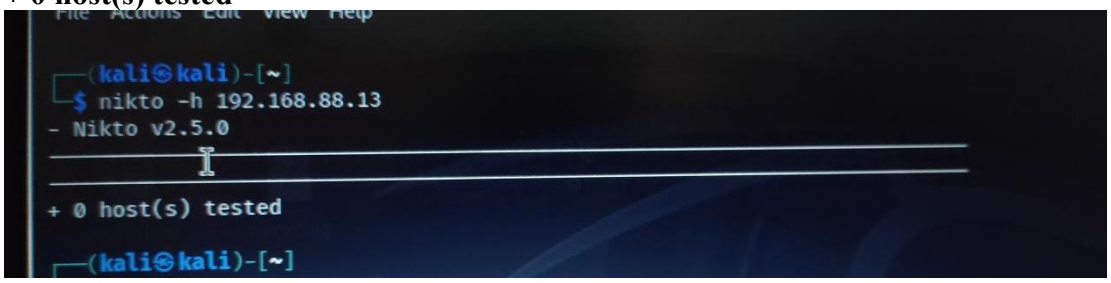
Perform a vulnerability scan on your OWASP VM and record your findings:

• Vulnerabilities Found:

nikto -h 192.168.88.13

- Nikto v2.5.0

+ 0 host(s) tested



Lab 5:
Wireshark

Exercise 1:

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

Step 3: Analyzing Captured Packets

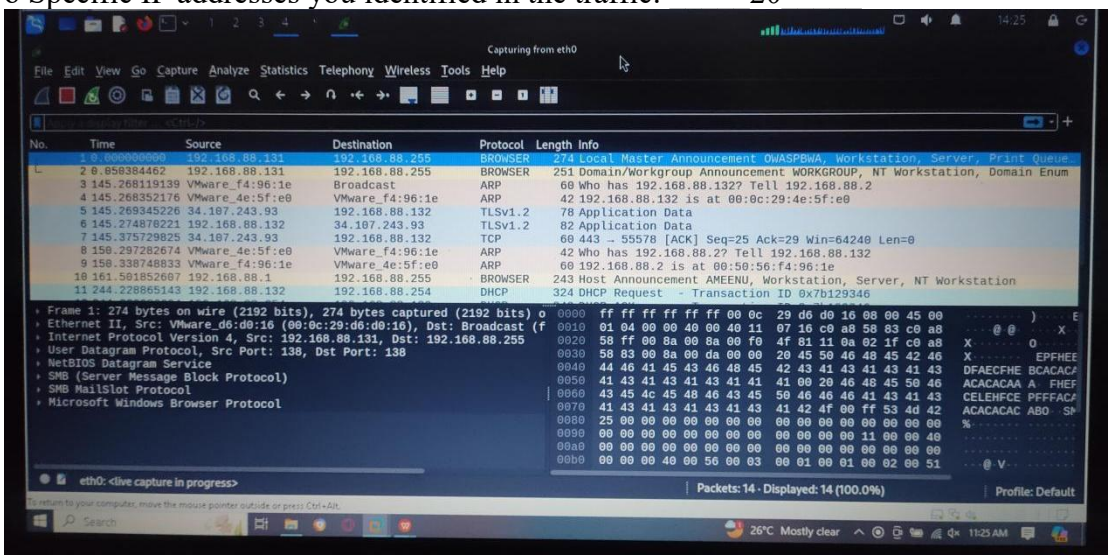
Exercise 3:

Use filters to analyze different types of traffic. Record the following:

o Number of HTTP packets captured: ___0_____

o Number of DNS packets captured: _____0_____

o Specific IP addresses you identified in the traffic: 20



Step 4: Understanding Packet Details

Exercise 4:

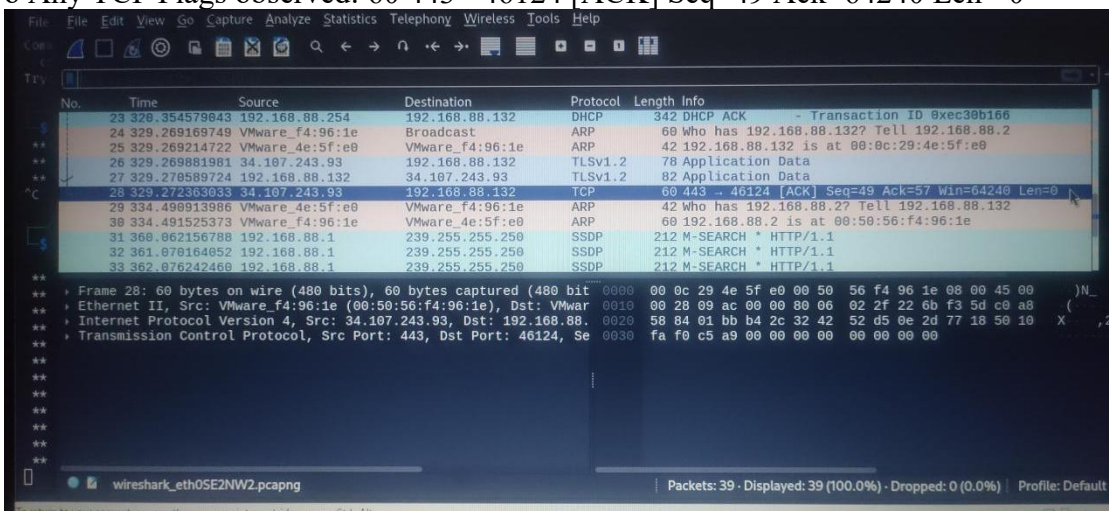
• Select a packet and list the following information:

o Source IP: 34.107.243.93

o Destination IP:192.168.88.132

o Protocol: TCP

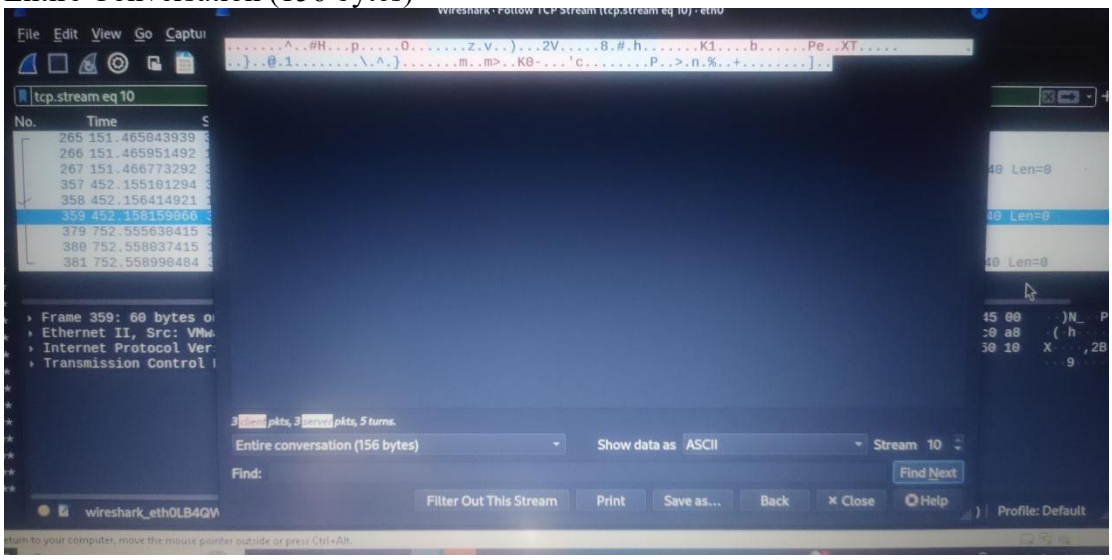
o Any TCP Flags observed: 60 443 - 46124 [ACK] Seq=49 Ack=64240 Len =0



Step 5: Advanced Packet Analysis Techniques

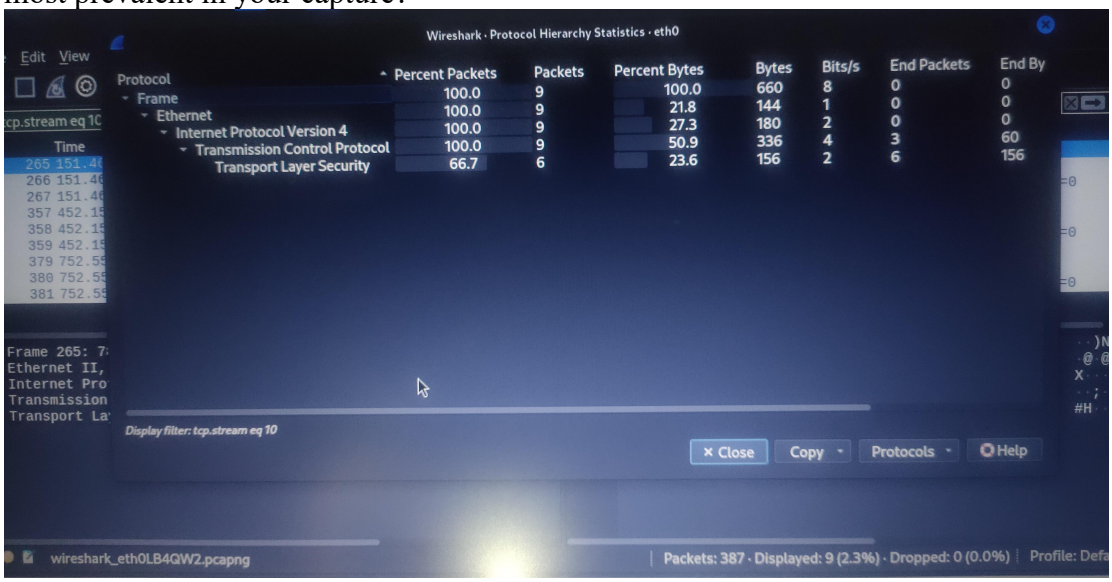
Exercise 5:
• Follow a TCP stream for a specific session and summarize the data exchanged between the client and server.

.....^..#H...p.....O.....z.v..)...2V.....8.#.h.....K1....b.....Pe..XT.....
...}..@.1.....\.^.}.....m..m>..K0-...'c.....P..>.n.%..+.....]..
3 client pkts, 3 server pkts, 5 turns.
Entire Conversation (156 bytes)



2. Protocol Hierarchy:
o Navigate to Statistics > Protocol Hierarchy to see a breakdown of captured protocols.
o This will help you identify which protocols are most common in your capture

Exercise 6:
• Take a screenshot of the Protocol Hierarchy and analyze the data. Which protocol is most prevalent in your capture?



Frame is most prevalent in your capture

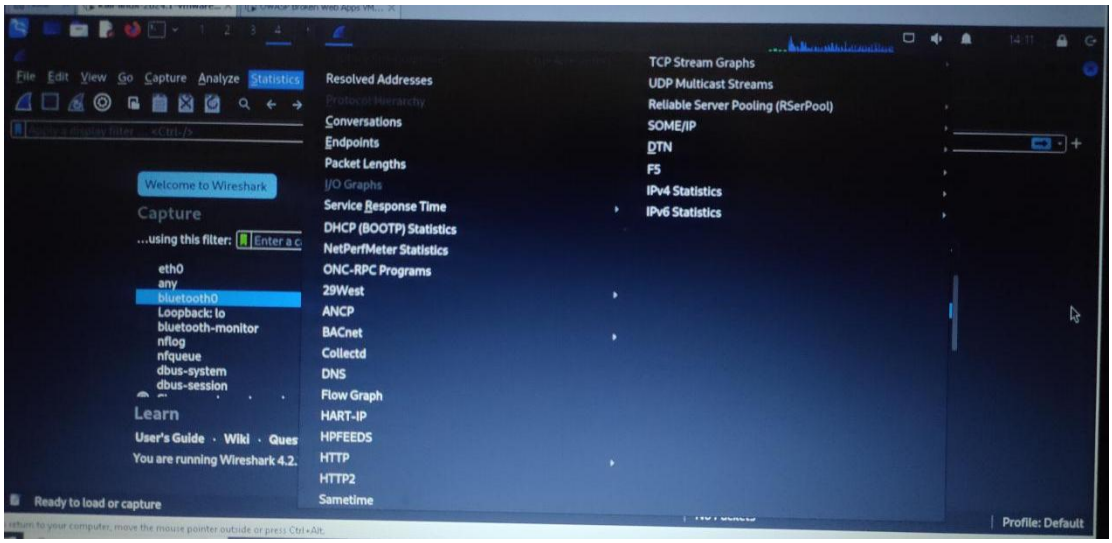
3. **IO Graphs**:

o Access **Statistics** > **IO Graphs** to visualize traffic over time.

Exercise 7:

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

o This can help identify spikes in traffic, indicating potential issues or security events.



Step 6: Exporting Captured Data

1. Save your captured packets for further analysis or reporting

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?

Step 7: Practical Applications of Wireshark

1. **Detecting Network Issues**:

Exercise 9:

•Describe a real-world scenario where you would use Wireshark to troubleshoot a network issue.

-Wireshark could be use in filtering traffic, for instance; there is an incident happened in an organization and the investigator can use wirshark in filter what happened on the network.

Also it can be use to check malicious attacks by considering the multiple attempts by an actor.

What specific symptoms would you investigate? Malicious attack

2. **Security Analysis**:

o Use Wireshark to identify potential security threats, such as unauthorized access attempts, malware communications, or data exfiltration.

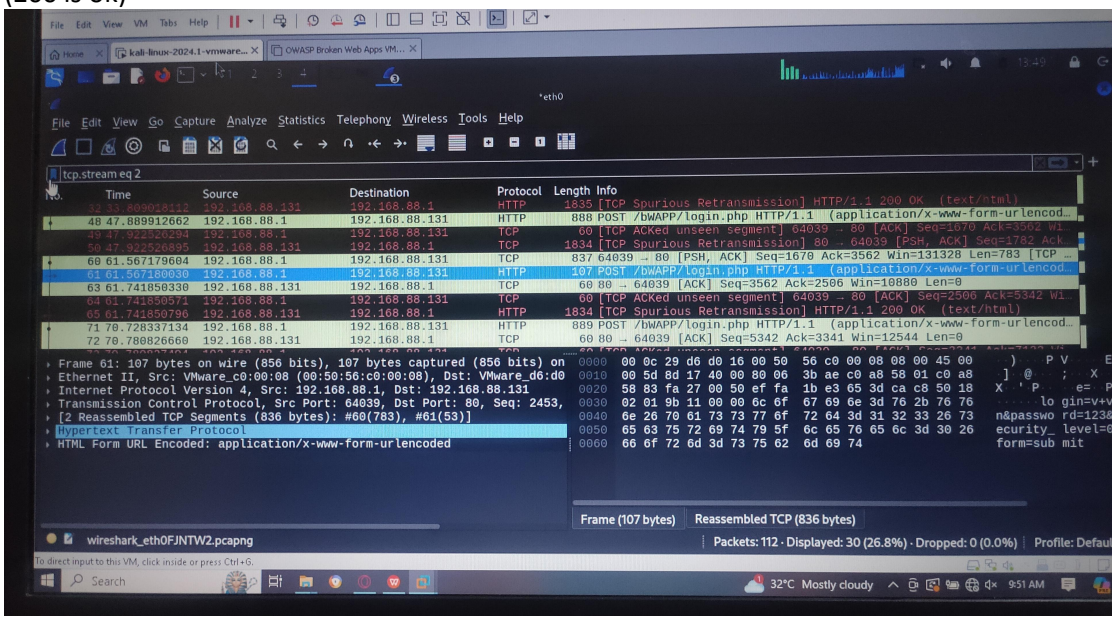
o Investigate any suspicious packets and document your findings.

-There is no any suspicious packets due to no other tempering by a third party.

Exercise 10:

• Identify at least two potential security threats in your captured traffic. What indicators led you to suspect these activities?

-I tried but no any good feedback, I used my OWASP IP Address on my window browser while my wirshark is on, I tried to log in to bwap with wrong and correct loggins and immidietly checked my wireshark filters if the http post will show **fail** , unfortunately it kept showing **win** and I followed through the http stream but it still figured ab successful output (200 is ok)



Lab 6:

Advanced Packet Analysis Techniques

Step 1: Dissecting Protocols

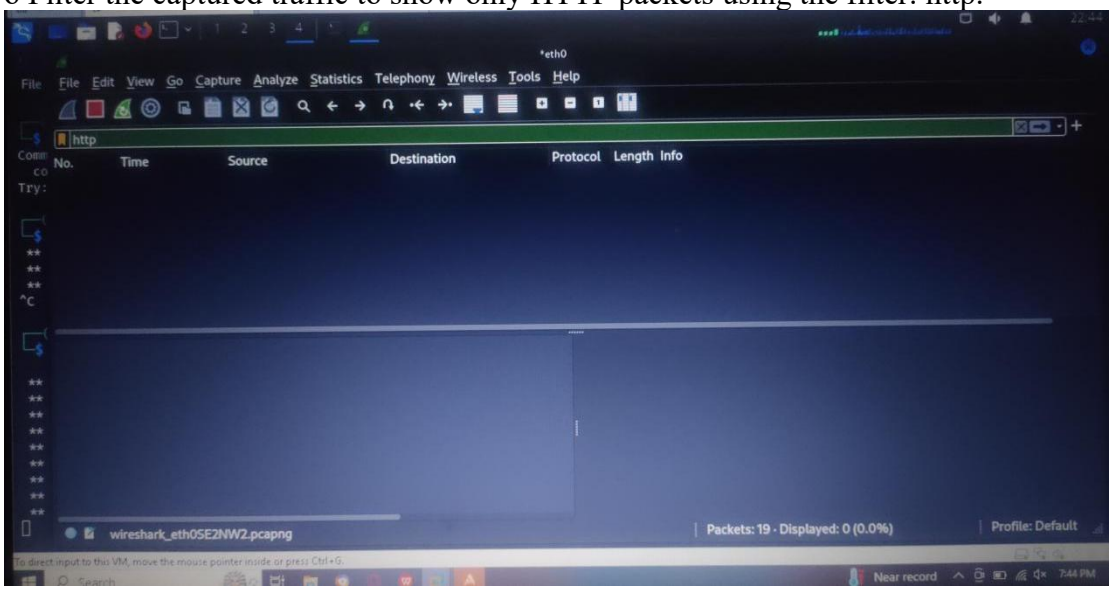
Exercise 1:

• Describe the purpose of the SYN and ACK flags in the TCP handshake. How do these flags

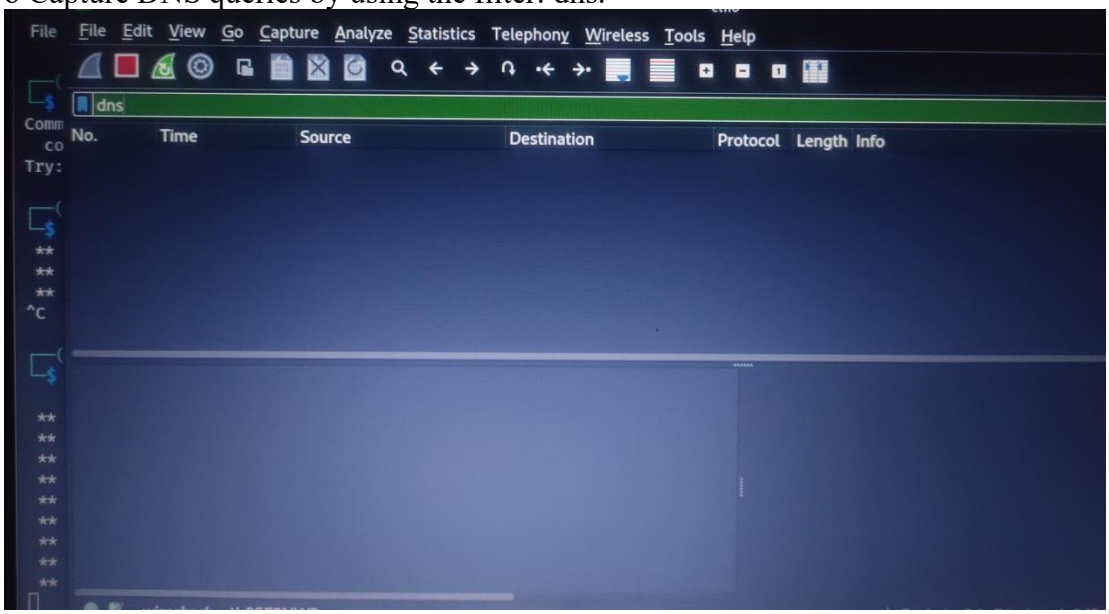
indicate the status of a connection?

2. **HTTP Analysis**:

o Filter the captured traffic to show only HTTP packets using the filter: http.



o Examine the headers of an HTTP request and response
It filtered 0
Exercise 2:
• Choose an HTTP packet and summarize its request method, status code, and any notable headers. What can you infer about the transaction? _____
3. **DNS Analysis**:
o Capture DNS queries by using the filter: dns.



o Examine the DNS response packets to see the resolved IP addresses for the queried domains.
Exercise 3:
• Identify a DNS query and its corresponding response. What information does the response provide, and how is it structured?
Step 2: Creating Custom Filters
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. _____
2. **Using Filter Expressions**:
o Utilize the Wireshark display filter expression dialog to construct complex filters.
o Practice filtering packets based on multiple criteria, such as source/destination IP, protocol type, and port numbers.
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? _
Step 3: Identifying Vulnerabilities
Exercise 6:
• Analyze your capture for any anomalies or indicators of potential vulnerabilities. Document your
findings and suggest possible remediation steps. _____
2. **Security Protocols**:
o Examine traffic from secure protocols (HTTPS) and identify how encryption affects packet analysis.
o Use the ssl filter to analyze SSL/TLS handshake packets.

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? _
Step 4: Practical Applications and Reporting
Exercise 8:
• Prepare a brief report summarizing your findings during the assessment. Include potential risks
and recommended actions. _____
2. **Creating a Capture Report**:
o Document your analysis steps, findings, and any relevant screenshots or packet details.
o Prepare a presentation summarizing your lab experience and learnings.
Exercise 9:•
Create a capture report that includes your objectives, methods, key findings, and any recommendations for improving network security. _