Dancing Notes

Objective:

- The objective was to identify and exploit vulnerabilities within the **Dancing** machine on Hack The Box and retrieve a flag to submit.

Tools Used:

- **Nmap:** A powerful network scanning tool used for network discovery and security auditing.
- **Samba Client:** A command-line tool used to connect to SMB shares.

Methodology:

Information Gathering (Reconnaissance):

Ping the machine: I started off by pinging the machine to check if it was reachable by pinging its IP address:

```
ping -c 6 10.129.189.181
```

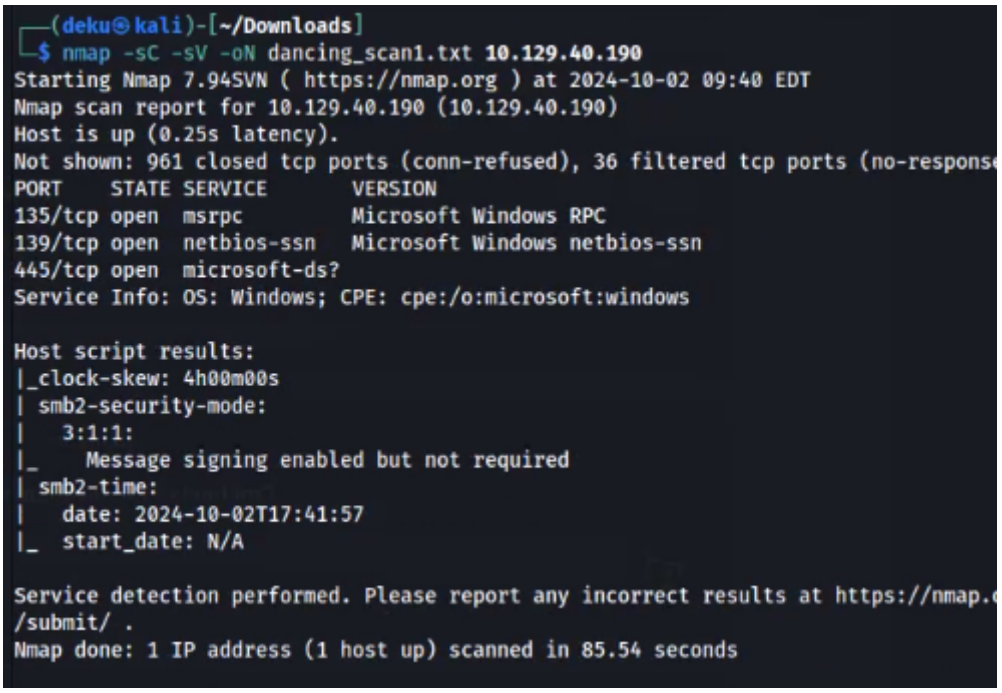
- **ping:** Sends ICMP echo request packets to the specified IP address to check if the host is reachable.
- **-c 6:** Sends 6 ping requests and then stops.
- **10.129.189.181:** The IP address of the target machine.

The target machine responded, indicating that it was up and reachable.

Network Scanning:

Nmap scan: After confirming that the machine was reachable, I proceeded with an Nmap scan to identify open ports and running services:

```
sudo nmap -sC -sV -oN dancing_scan.txt 10.129.189.181
```



- **sudo:** Runs the command with superuser privileges, which are often required for network operations.
- **nmap:** The command-line tool used for network discovery and security auditing.
- **-sC:** Runs default scripts to assist with service detection and vulnerability enumeration.
- **-sV:** Enables version detection, allowing Nmap to determine the version of services running on open ports.
- **-oN dancing_scan.txt:** Saves the scan results to a file named "dancing_scan.txt" for further analysis.
- **10.129.189.181:** The IP address of the target machine.

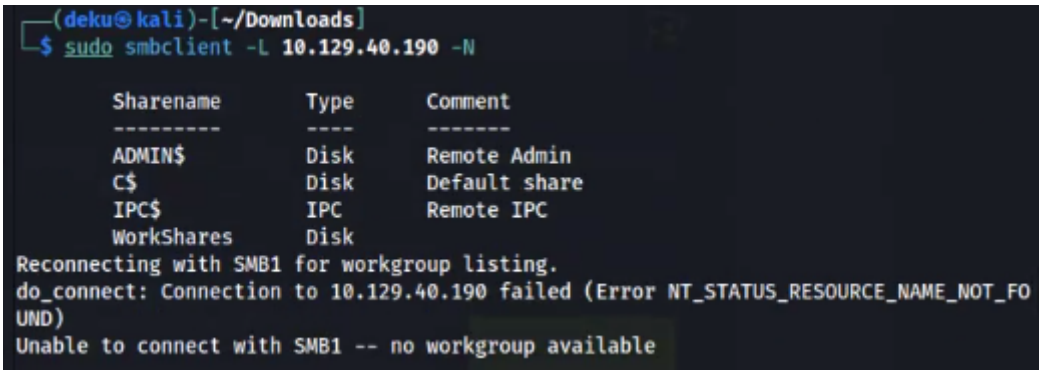
Nmap Results:

- The scan revealed open ports and services running on the machine, including potential SMB services.

SMB Enumeration:

List SMB Shares: To gather more information about shared resources on the target machine, I executed the following command:

```
sudo smbclient -L 10.129.189.181 -N
```



- **smbclient:** A command-line tool used to access SMB/CIFS resources on servers.
- **-L:** Lists all shares available on the specified server.
- **10.129.189.181:** The IP address of the target machine.
- **-N:** Connects without a password (anonymous access).

This command revealed available SMB shares on the target machine, indicating that there were accessible shares.

Accessing a Share: I then attempted to connect to the `WorkShares` SMB share:

```
sudo smbclient //10.129.189.181/WorkShares -N
```

- **//10.129.189.181/WorkShares:** The SMB share I wanted to access.
- **-N:** Again, this option connects without a password.

Upon connecting to the `WorkShares` share, I navigated to a user directory:

```
cd James.P
```

- **cd James.P:** Changes the directory to `James.P` within the `WorkShares` share.

Next, I listed the files in the directory using:

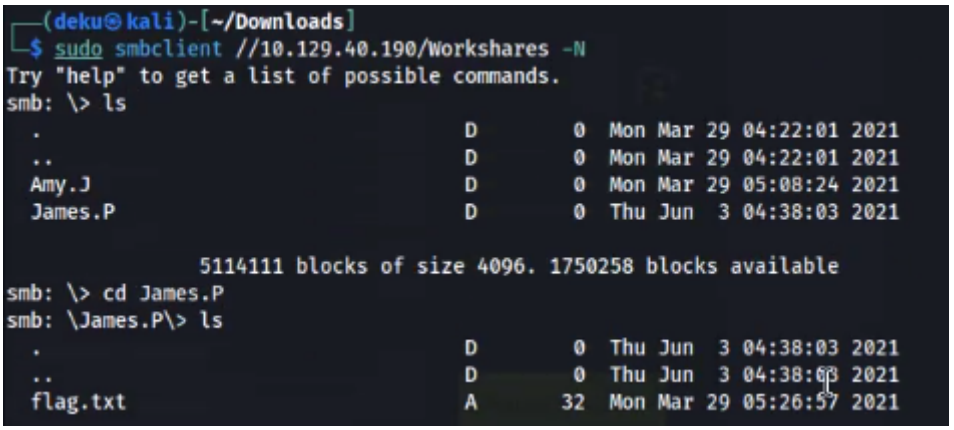
```
ls
```

- **ls:** Lists the files and directories within the current directory.

Finally, I retrieved the contents of the `flag.txt` file using:

```
more flag.txt
```

- **more flag.txt:** Displays the contents of the `flag.txt` file.



Conclusion:

- The **Dancing** challenge highlighted the importance of securing SMB shares and properly configuring file permissions. This challenge provided practical experience in identifying and exploiting vulnerabilities associated with file sharing.

Recommendations for Securing SMB Shares:

To protect against unauthorized access to SMB shares, the following measures are recommended:

1. **Restrict Access:** Limit access to SMB shares to only authorized users and devices.
2. **Disable Guest Access:** Disable anonymous access to prevent unauthorized users from connecting to the shares.
3. **Strong Authentication:** Enforce strong passwords for all accounts accessing the SMB shares.
4. **Regular Audits:** Conduct regular audits of shared resources and access logs to ensure compliance with security policies.
5. **Use Encryption:** Implement encryption for SMB connections to secure data in transit.