Vulnversity

Vulnversity

Learh about active recon, web app attacks and privilege escalation

Reconnaissance

Reconnaissance

```
 nmap -T4  -sV 10.10.231.79
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-22 14:57 PKT
Nmap scan report for 10.10.231.79
Host is up (0.26s latency).
Not shown: 994 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 3.0.3
22/tcp   open  ssh         OpenSSH 7.2p2 Ubuntu 4ubuntu2.7 (Ubuntu Linux; protocol 2.0)
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
3128/tcp open  http-proxy  Squid http proxy 3.5.12
3333/tcp open  http        Apache httpd 2.4.18 ((Ubuntu))
Service Info: Host: VULNUNIVERSITY; OSs: Unix, 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 59.51 seconds
```

enum4linux targetip (great script)

vsftpd 3.0.3 Potential Exploits

https://www.exploit-db.com/exploits/49719

http://www.securityspace.com/smysecure/catid.html?id=1.3.6.1.4.1.25623.1.0.108045

OpenSSH 7.2p2 Potential Exploits

https://www.exploit-db.com/exploits/40136

https://www.rapid7.com/db/modules/auxiliary/scanner/ssh/ssh_enumusers/

https://hackerone.com/reports/476439

Samba smb Potential Exploits

As this is Intentionally Vulnerable Machine you may check for E**ternal Blue** by yourself

Squid Proxy Potential Exploits

https://www.rapid7.com/db/modules/exploit/linux/proxy/squid_ntlm_authenticate/

https://www.rapid7.com/db/vulnerabilities/squid-proxy-gopher-bo/

https://www.cybersecurity-help.cz/vdb/squid-cache_org/squid/3.5.12/

Apache httpd 2.4.18 Potential Exploits

https://www.rapid7.com/db/vulnerabilities/apache-httpd-cve-2019-0211/

https://www.exploit-db.com/exploits/46676

https://hackerone.com/reports/520903

Quick Summary about Nmap

nmap flag	**Description**
-sV	Attempts to determine the version of the services running
-p <x> or -p-	Port scan for port <x> or scan all ports
-Pn	Disable host discovery and just scan for open ports

nmap flag	Description
-A	Enables OS and version detection, executes in-build scripts for further enumeration
-sC	Scan with the default nmap scripts
-v	Verbose mode
-sU	UDP port scan
-sS	TCP SYN port scan

Nmap Basic Port Scan Summary

Option	Purpose
-p-	all ports
-p1-1023	scan ports 1 to 1023
-F	100 most common ports
-r	scan ports in consecutive order
-T<0-5>	-T0 being the slowest and T5 the fastest
--max-rate 50	rate <= 50 packets/sec
--min-rate 15	rate >= 15 packets/sec
--min-parallelism 100	at least 100 probes in parallel

Nmap Post Port Scan Summary

Option	Meaning
-sV	determine service/version info on open ports
-sV --version-light	try the most likely probes (2)
-sV --version-all	try all available probes (9)
-O	detect OS
--traceroute	run traceroute to target

Option	Meaning
--script=SCRIPTS	Nmap scripts to run
-sC or --script=default	run default scripts
-A	equivalent to -sV -O -sC --traceroute
-oN	save output in normal format
-oG	save output in grepable format
-oX	save output in XML format

Option	Meaning
-oA	save output in normal, XML and Grepable formats

Other Usefull ToDo about Nmap

Option	Purpose
--reason	explains how Nmap made its conclusion
-v	verbose
-vv	very verbose
-d	debugging

Option	Purpo-se
-dd	more details for debugg-ing

Some Important Points about

In other words, stealth SYN scan is not possible when option is chosen

Note that is often used during CTFs and when learning to scan on practice targets.

whereas is often used during real engagements where stealth is more important.

Other Nmap Cheat-sheet

https://github.com/marsam/cheatsheets/blob/master/nmap/nmap.rst

Directory Busting with GoBuster

Directory Busting with GoBuster

```
gobuster dir --url http://10.10.231.79:3333/ --wordlist /usr/share/wordlists/dirb/
common.txt
===============================================================
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.231.79:3333/
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Timeout:                 10s
===============================================================
2022/03/22 15:25:18 Starting gobuster in directory enumeration mode
===============================================================
/.hta                 (Status: 403) [Size: 293]
/.htaccess            (Status: 403) [Size: 298]
/.htpasswd            (Status: 403) [Size: 298]
/css                  (Status: 301) [Size: 317] [--> http://
10.10.231.79:3333/css/]
/fonts                (Status: 301) [Size: 319] [--> http://
10.10.231.79:3333/fonts/]
/images               (Status: 301) [Size: 320] [--> http://
10.10.231.79:3333/images/]
/index.html           (Status: 200) [Size: 33014]

/internal             (Status: 301) [Size: 322] [--> http://10.10.231.79:3333/
internal/]
/js                   (Status: 301) [Size: 316] [--> http://
10.10.231.79:3333/js/]
/server-status        (Status: 403) [Size: 302]
```

Useful about Gobuster

Useful about Gobuster

GoBuster is a tool used to brute-force URIs (directories and files), DNS subdomains and virtual host names. For this machine, we will focus on using it to brute-force directories.

GoBuster flag	**Description**
-e	Print the full URLs in your console
-u	The target URL
-w	Path to your wordlist
-U and -P	Username and Password for Basic Auth
-p <x>	Proxy to use for requests

GoBuster flag	Description
-c <http cookies>	Specify a cookie for simulating your auth

Exploitation

Exploitation

We will not only try to exploit as What THM has told us to do (**web server compromising**) but we also try to exploit this machine in many different ways based on our previous experiences

Steps from my experience rather than THM current guide

Steps from my experience rather than THM current guide

I used Ms17 Eternal Blue , Didn't Work for me

I also try to exploit with several different but I got tired may be this machine was design to be exploit only the way THM told us to do but It is worth to Try different approaches to **root** the machine.

Steps From THM

Steps from THM

• As we Found upload functionality we will try to exploit it.

• While trying we found that extensions are blocked so we will use world list **/usr/share/wordlists/dirb/extensions_common.txt** using **burp intruder**

• we found that .phtml is not blocked

• se we renamed our php-rev-shell.php as php-rev-shell.phtml and upload and got the **shell**

• we enumerate different PrivEsc Vectors and found **Systemctl** to be using **SUID** bit

• used https://gtfobins.github.io/gtfobins/systemctl/ and https://medium.com/@klockw3rk/privilege-escalation-leveraging-misconfigured-systemctl-permissions-bc62b0b28d49 to exploit this

• Successfully followed the step and got **rootshell**

Problems while exploiting

while using Burp Intruder I didn't notice any difference in response size, status code, even the grepable text was same for all the payloads including .**phtml** I will figure out why.