Meow Write-up

Prepared by: 0ne-nine9

Setting Up

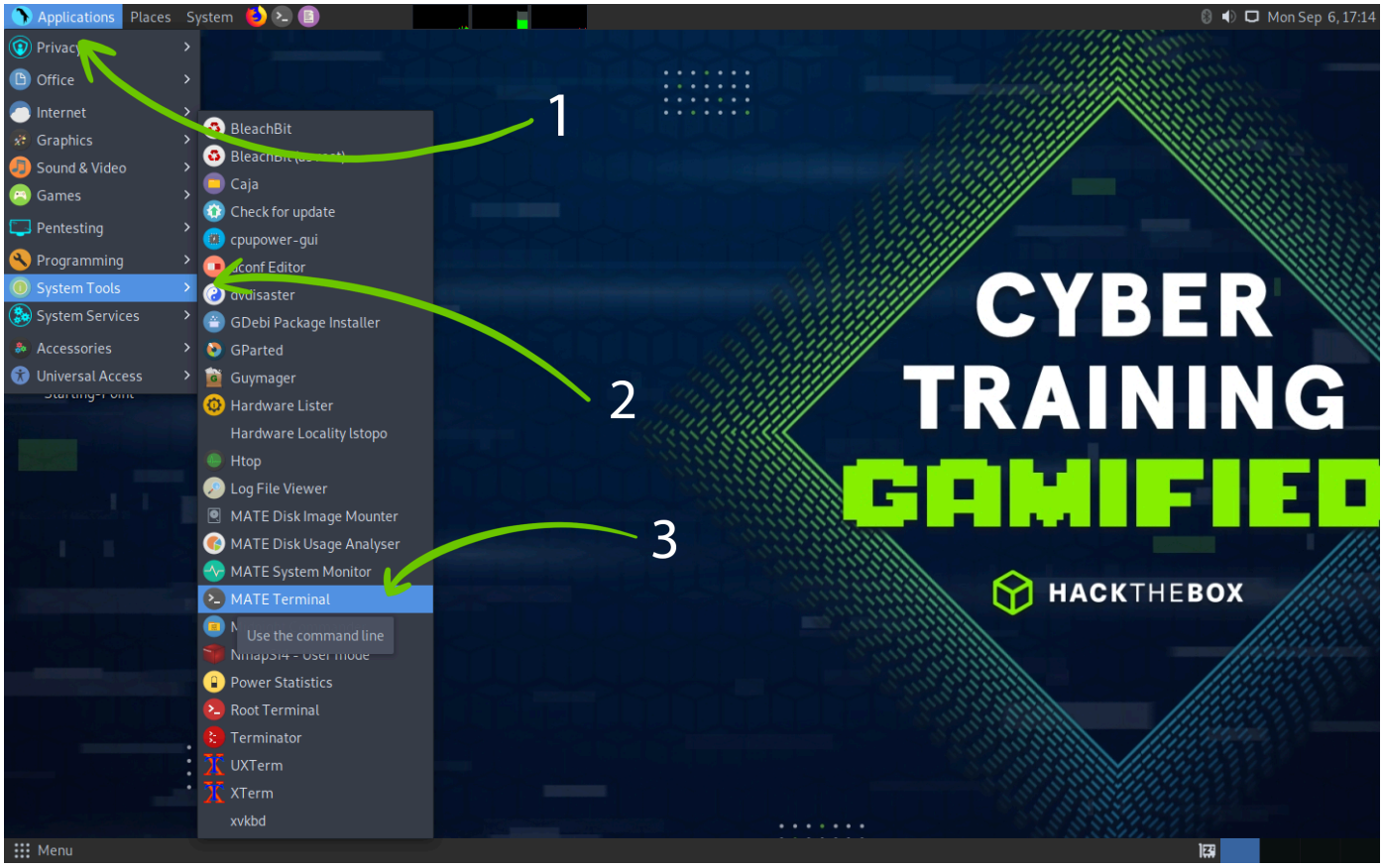
Welcome to Hack The Box!

Before we start with your very first vulnerable machine, let us make sure you are connected to the target's network and know your way around a terminal. When visiting the Starting Point lab's page, you might have been prompted to pick between a Pwnbox connection or a VPN configuration file that you can download and run on your Virtual Machine. If you have not learned how to set up a Virtual Machine yet, check out the Setting Up module on HTB Academy.

Running Pwnbox is straightforward, and you do not require any additional steps to connect to the target machine. If you boot up a new instance of Pwnbox under the Starting Point option, you will be automatically placed in the same network as the target. You can read more about Pwnbox in this article.

If you choose to download the VPN (.ovpn) configuration file on your Virtual Machine, then here is how to use it to connect to the target's network. In order to open a terminal window, you can click on the terminal icon on your Desktop.

Alternatively, you can navigate to the System Tools menu and select the terminal from there. In this case, we are using a MATE terminal. Ultimately, it does not matter what terminal you use as long as you do not get lost. Hovering over the terminal option, you can see the description of the tool: `Use the command line`, which is precisely what we will be doing next.



After selecting our terminal window, we will need to navigate where the .ovpn file was downloaded. In most cases, this is in the Downloads folder. In order to get there properly, let's start with a `cd` command, which stands for `change directory`, to make sure you're in your home folder for your current logged in user. Running this command without a specified location to navigate to will simply place you in your `home` directory. This way, we can make sure everyone reading this is at the right starting point (no pun intended). The next command we can type is `ls`, which will show us the home directory folders, some of them being `Desktop`, `Documents`, `Downloads` and more. `Downloads` is what we're interested in, and we use the `cd Downloads` command to navigate inside of it.

After navigating to the Downloads directory, type in `ls` to make sure the .ovpn file is present on the system, followed by the command to launch your OpenVPN client and connect to the Hack The Box internal network: `sudo openvpn {filename}.ovpn`, where {filename} should be replaced with the name of your .ovpn file for the Starting Point lab. The text marked in green and curly brackets `{}` is a replacement for your own version of input. This will be a recurring sight in the Starting Point write-ups, so keep that in mind!

After running the command, you will be prompted to input your super-user password, the same as the current password for your Operating System account. Don't worry if you can't see anything being typed into the terminal once you are typing your password. It's a security measure of Linux to stop other from shoulder-surfing you. Finish inputting your password and hit the Enter key once done to initialize the

openVPN connection.

Let the configuration script run until you see the `Initialization Sequence Completed` message at the very end of the output. Once that is present, make sure that there is no mention of multiple tunnel interfaces, such as `tun1`, `tun2`, and so forth. Having multiple tunnel interfaces can ruin the stability of your connection to the target and create routing conflicts on your Operating System, which would only bring frustration. There should only be `tun0` mentioned in the output as marked in the image below.

```
$ cd

$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos

$ cd Downloads

$ ls
{filename}.ovpn

$ sudo openvpn {filename.ovpn}
[sudo] password for {username}: {your_VM_password}

2021-09-24 20:20:41 WARNING: Compression for receiving enabled. Compression has been used in the past to break encryption. Sent
packets are not compressed unless "allow-compression yes" is also set.
2021-09-24 20:20:41 DEPRECATED OPTION: --cipher set to 'AES-128-CBC' but missing in --data-ciphers (AES-256-GCM:AES-128-GCM).
Future OpenVPN version will ignore --cipher for cipher negotiations. Add 'AES-128-CBC' to --data-ciphers or change --cipher
'AES-128-CBC' to --data-ciphers-fallback 'AES-128-CBC' to silence this warning.
2021-09-24 20:20:41 OpenVPN 2.5.1 x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO] [LZ4] [EPOLL] [PKCS11] [MH/PKTINFO] [AEAD] built on
May 14 2021


[..] Output omitted [..]

2021-09-24 20:20:41 net_addr_v6_add: {dead:beef:IPV6} dev tun0
2021-09-24 20:20:41 net_route_v4_add: 10.10.10.0/23 via 10.10.14.1 dev [NULL] table 0 metric -1
2021-09-24 20:20:41 net_route_v4_add: {IPV4} via 10.10.14.1 dev [NULL] table 0 metric -1
2021-09-24 20:20:41 add_route_ipv6(dead:beef::/64 -> {IPV6} metric -1) dev tun0
2021-09-24 20:20:41 net_route_v6_add: dead:beef::/64 via :: dev tun0 table 0 metric -1
2021-09-24 20:20:41 WARNING: this configuration may cache passwords in memory -- use the auth-nocache option to prevent this
2021-09-24 20:20:41 Initialization Sequence Completed
```

If you feel lost, you'll probably need to brush up on your Linux skills. In order to learn more about navigating and using Linux as a new user, you should check our our Linux Fundamentals module on HTB Academy. You'll be on the right track after completing it and you'll be able to return to Starting Point with a fresh set of skills that will tone down the frustration many new users feel when picking up pentesting for the first time with no prior contact with Linux, command line interfaces and big features.

After making sure everything in the output is in order, you can open a new terminal tab or window. Leave the current one running; otherwise, you will lose the connection to the target. You are now ready to start.

Introduction

When first starting a penetration test or any security evaluation on a target, a primary step is known as `Enumeration`. This step consists of documenting the current state of the target to learn as much as possible about it.

Since you are now on the same Virtual Private Network (VPN) as the target, you can directly access it as any user would. If the target is a web server, running a public web page, you can navigate to its IP address to see what the page contains. If the target is a storage server, you can connect to it using the same IP address to explore the files and folders stored on it, provided that you have the necessary credentials. The question is, how do you find these services? You cannot manually search for them because it would take a long time.

Every server uses `ports` in order to serve data to other clients. The first steps in the Enumeration phase involve scanning these open ports to see the purpose of the target on the network and what potential vulnerabilities might appear from the services running on it. In order to quickly scan for ports, we can use a tool called `nmap`, which we will detail more in the Enumeration chapter of this write-up.

After finding the open ports on the target, we can manually access each of them using different tools to find out if we have access to their contents or not. Different services will use different tools or scripts to be accessed. These can be discovered and learned by a beginner penetration tester only with time and practice (and some diligent Googling). 90% of penetration testing consists of research done on the internet about the product you are testing. Since the technological ecosystem is continuously evolving, it is impossible to know everything about everything. The key is to know how to look for the information you need. The ability to

research effectively is the skill you need to continuously adapt and evolve into your top quality.

The objective here is not speed but meticulousness. If a resource on the target is missed during the Enumeration phase of your test, you might lose a vital attack vector which would have potentially cut your worktime on the target in half or even less.

Enumeration

After our VPN connection is successfully established, we can ping the target's IP address to see if our packets reach their destination. You can take the IP address of your current target from the Starting Point lab's page and paste it into your terminal after typing in the `ping` command as illustrated below.

```
$ ping {target_IP}

PING {target_IP} ({target_IP}) 56(84) bytes of data.
64 bytes from {target_IP}: icmp_seq=1 ttl=63 time=20.4 ms
64 bytes from {target_IP}: icmp_seq=2 ttl=63 time=22.0 ms
64 bytes from {target_IP}: icmp_seq=3 ttl=63 time=20.2 ms
64 bytes from {target_IP}: icmp_seq=4 ttl=63 time=19.8 ms
^C
--- {target_IP} ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 19.788/20.603/22.026/0.849 ms
```

After four successful replies from the target, we can determine that our connection is formed and stable. We can cancel the ping command by pressing the `CTRL+C` combination on our keyboard, which will be displayed in the terminal as `^C` marked above in green. This will return control of the terminal tab to us, from where we can proceed with the next step - scanning all of the target's open ports to determine the services running on it. In order to start the scanning process, we can use the following command with the `nmap` script. `nmap` stands for Network Mapper, and it will send requests to the target's ports in hopes of receiving a reply, thus determining if the said port is open or not. Some ports are used by default by certain services. Others might be non-standard, which is why we will be using the service detection flag `-sV` to determine the name and description of the identified services. The text marked in green and curly brackets `{}` is a replacement for your own version of input. In this case, you will need to replace the `{target_IP}` part with the IP address of your own target.

```
$ sudo nmap -sV {target_IP}

Starting Nmap 7.92 ( https://nmap.org ) at 2021-09-24 20:36 BST
Nmap scan report for {target_IP}
Host is up (0.050s latency).
Not shown: 999 closed tcp ports (reset)

PORT   STATE SERVICE VERSION
23/tcp open  telnet  Linux telnetd
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 1.82 seconds
```

Following the completion of the scan, we have identified `port 23/tcp` in an `open` state, running the `telnet` service. Following a quick Google search of this protocol, we find out that telnet is an old service used for remote management of other hosts on the network. Since the target is running this service, it can receive telnet connection requests from other hosts in the network (such as ourselves). Usually, connection requests through telnet are configured with username/password combinations for increased security. We can see this is the case for our target, as we are met with a Hack The Box banner and a request from the target to authenticate ourselves before being allowed to proceed with remote management of the target host.

```
$ telnet {target_IP}

Trying {target_IP}...
Connected to {target_IP}.
Escape character is '^]'.

 Hack The Box


Meow login:
```

We will need to find some credentials that work to continue since there are no other ports open on the target that we could explore.

Foothold

Sometimes, due to configuration mistakes, some important accounts can be left with blank passwords for the sake of accessibility. This is a significant issue with some network devices or hosts, leaving them open to simple brute-forcing attacks, where the attacker can try logging in sequentially, using a list of usernames with no password input.

Some typical important accounts have self-explanatory names, such as:

- admin
- administrator
- root

A direct way to attempt logging in with these credentials in hopes that one of them exists and has a blank password is to input them manually in the terminal when the hosts request them. If the list were longer, we could use a script to automate this process, feeding it a wordlist for usernames and one for passwords. Typically, the wordlists used for this task consist of typical people names, abbreviations, or data from previous database leaks. For now, we can resort to manually trying these three main usernames above.

The first two were not so lucky for us. When things look down, it is essential to keep going, be persistent. We can't succeed unless we attempt all possibilities. Let us try the last one.

```
Meow login: root
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.4.0-77-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Mon 06 Sep 2021 03:15:22 PM UTC

  System load:  0.0               Processes:             195
  Usage of /:   41.7% of 7.75GB   Users logged in:       0
  Memory usage: 4%                IPv4 address for eth0: {target_IP}
  Swap usage:   0%

 * Super-optimized for small spaces - read how we shrank the memory
   footprint of MicroK8s to make it the smallest full K8s around.

   https://ubuntu.com/blog/microk8s-memory-optimisation

72 updates can be applied immediately.
29 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable


Last login: Wed Jul  7 10:55:01 UTC 2021 on tty1
```

Success! We have logged into the target system. We can now go ahead and take a look around the directory we landed in using the `ls` command. There is a possibility we might find what we are looking for.

```
# ls
flag.txt  snap

# cat flag.txt
b40abdfe23665f766f9c61ecba8a4c19
```

The `flag.txt` file is our target in this case. Most of Hack The Box's targets will have one of these files, which will contain a hash value called `a flag`. The naming convention for these targeted files varies from lab to lab. For example, weekly and retired machines will have two flags, namely `user.txt` and `root.txt`. CTF targets and other labs will have `flag.txt`. Challenges will, most of the time, not contain an actual file, but rather offer you snippets of the flag as you solve it, the respective parts being embedded into the challenge more homogeneously (text hidden in an image, or other examples).

You can read the file to have the hash value displayed in the terminal using the `cat` command. Copying the flag and pasting it into the Starting Point lab's page will grant you ownership of this machine, completing your very first task.

Congratulations!