UNIVERSIDAD NACIONAL DE INGENIERÍA

Facultad de Ingeniería Industrial y de Sistemas

Informes de exploración de vulnerabilidades en HTB

"De las máquinas: OpenAdmin, Fuse Magic, Remote "

ELABORADO POR:

- Alfonso Suárez, Luis
- Mottoccanche Tantaruna, Joseph
- Chi Jon, Lau

Índice

1. OpenAdmin **2**
1.1. Reconocimiento . . . 2
1.2. Escaneo de Vulnerabilidades . . . 2
1.3. Enumeración . . . 2
1.4. Explotación . . . 2
1.5. Post Explotación . . . 2

2. Remote **3**
2.1. Reconocimiento . . . 3
2.2. Escaneo de Vulnerabilidades . . . 3
2.3. Enumeración . . . 4
2.4. Explotación . . . 8
2.5. Post Explotación . . . 8

3. Fuse **9**
3.1. Reconocimiento . . . 9
3.2. Escaneo de Vulnerabilidades . . . 9
3.3. Enumeración . . . 9
3.4. Explotación . . . 9
3.5. Post Explotación . . . 9

1. OpenAdmin

1.1. Reconocimiento

1.2. Escaneo de Vulnerabilidades

1.3. Enumeración

1.4. Explotación

1.5. Post Explotación

2. Remote

2.1. Reconocimiento

Lo primero a hacer en este caso es un escaneo de nmap, para encontrar algunos puertos abiertos y servicios corriendo, en este caso se encontraron los puertos 21, 80 y 445 abiertos principalmente.

```
# Nmap 7.80 scan initiated Thu Oct  7 10:12:12 2021 as: nmap -Pn -p-
--min-rate=5000 -v -oN puertos 10.10.10.180
Nmap scan report for 10.10.10.180
Host is up (0.11s latency).
Not shown: 65528 filtered ports
PORT      STATE SERVICE
21/tcp    open  ftp
80/tcp    open  http
111/tcp   open  rpcbind
135/tcp   open  msrpc
445/tcp   open  microsoft-ds
2049/tcp  open  nfs
49666/tcp open  unknown

Read data files from: /usr/bin/../share/nmap
# Nmap done at Thu Oct  7 10:12:38 2021 -- 1 IP address (1 host up) s
canned in 26.44 seconds
```

Figura 1: nmap remote

2.2. Escaneo de Vulnerabilidades

Como primer escaneo de vulnerabilidades se intenta con el mismo nmap, con la opción -script vuln", esto probará as vulnerabilidades más comunes en el server.

```
# Nmap 7.80 scan initiated Sat Oct  9 15:26:38 2021 as: nmap -Pn -p 2
1,80,111,135,445,2049 --script vuln -v -oN vuln 10.10.10.180
Nmap scan report for 10.10.10.180
Host is up (0.12s latency).

PORT     STATE SERVICE
21/tcp   open  ftp
|_clamav-exec: ERROR: Script execution failed (use -d to debug)
|_sslv2-drown:
80/tcp   open  http
|_clamav-exec: ERROR: Script execution failed (use -d to debug)
|_http-csrf: Couldn't find any CSRF vulnerabilities.
|_http-dombased-xss: Couldn't find any DOM based XSS.
| http-enum:
|   /blog/: Blog
|   /home.aspx: Possible admin folder
|   /contact/: Potentially interesting folder
|   /home/: Potentially interesting folder
|_  /intranet/: Potentially interesting folder
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
111/tcp  open  rpcbind
|_clamav-exec: ERROR: Script execution failed (use -d to debug)
135/tcp  open  msrpc
|_clamav-exec: ERROR: Script execution failed (use -d to debug)
445/tcp  open  microsoft-ds
|_clamav-exec: ERROR: Script execution failed (use -d to debug)
2049/tcp open  nfs
|_clamav-exec: ERROR: Script execution failed (use -d to debug)
```

Figura 2: vulnerabilidades por nmap

2.3. Enumeración

Luego de ver los puertos, nmap no nos bota una vulnerabilidad por FTP, pero de todos modos nunca está de más probar si encontramos algo, sin embargo en esta ocasión no encontramos nada relevante.

```
❯ ftp 10.10.10.180
Connected to 10.10.10.180.
220 Microsoft FTP Service
Name (10.10.10.180:jmt): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230 User logged in.
Remote system type is Windows_NT.
ftp> ls
200 PORT command successful.
125 Data connection already open; Transfer starting.
226 Transfer complete.
ftp> _
```

Figura 3: logueo anónimo por FTP

Intentamos luego con la página ubicada en el puerto 80, a ver si encontramos algo, y efectivamente encontramos una página que tiene diferentes apartados para revisar, buscamos info en los cuadros y en toda la página pero es solo texto generado de relleno, así que no hay información relevante en estas páginas para diccionarios.

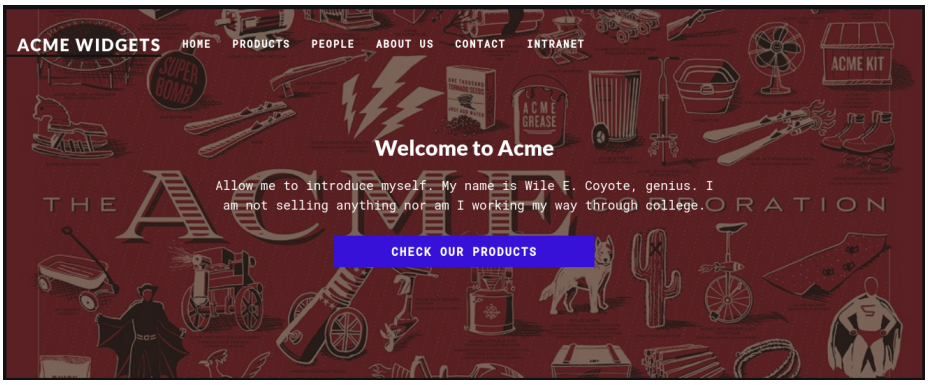

Figura 4: logueo anónimo por FTP

Entre todas las páginas encontramos un apartado de login, está en el mismo servidor así que se ve bastante interesante junto a que el framework es de umbraco según el wappalizer.

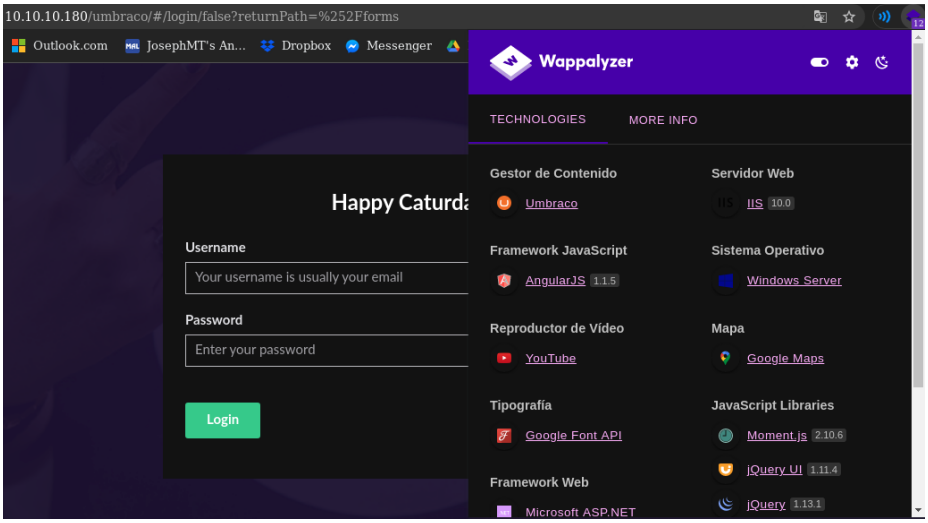

Figura 5: Resultados de wappalizer

Intentamos un escaneo con dirb para escanear los posibles directorios ocultos, donde se encontraron muchos directorios que de forma normal hubieran sido localizados y otros que hacen referencia a redirecciones, algunos que mostraron un error de configuración pero no grave.

```
000000038:   200        187 L    490 W      6703 Ch     "home"
000000032:   200        137 L    338 W      5011 Ch     "blog"
000000025:   200        124 L    331 W      7890 Ch     "contact"
000000042:   200        129 L    302 W      5330 Ch     "products"
000000155:   200        167 L    330 W      6739 Ch     "people"
000000157:   500        80 L     276 W      3420 Ch     "product"
000000286:   200        187 L    490 W      6703 Ch     "Home"
000000496:   200        129 L    302 W      5330 Ch     "Products"
000000592:   200        124 L    331 W      7890 Ch     "Contact"
000000715:   302        3 L      8 W        126 Ch      "install"
000001035:   200        137 L    338 W      5011 Ch     "Blog"
000001352:   200        167 L    330 W      6749 Ch     "People"
000001794:   500        80 L     276 W      3420 Ch     "Product"
000002430:   302        3 L      8 W        126 Ch      "INSTALL"
000002574:   500        80 L     276 W      3420 Ch     "master"
000002624:   200        123 L    283 W      4049 Ch     "1112"
000001119:   200        161 L    428 W      5441 Ch     "about-us"
000002959:   200        116 L    222 W      3313 Ch     "intranet"
000003012:   200        123 L    310 W      4234 Ch     "1114"
000002997:   200        81 L     201 W      2750 Ch     "1117"
```

Figura 6: Escaneo con la herramienta dirb

Luego para tratar de buscar por los archivos compartidos se usa el comando llamado showmonts, que viene en la herramienta nfs-common.

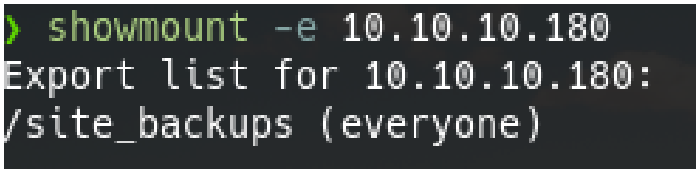

Figura 7: Obtención del backup

Luego creando una carpeta para guardar el contenido extraído con el comando "mount -t nfs 10.10.10.180:/site_backups".
una vez copiado esto tenemos carpetas interesantes, nuestro objetivo parecen ser credenciales de la base de datos para por medio de esas acceder al servidor original, entonces primero buscamos un poco. Entonces encontramos una password en hash dentro de .ªpp_Data/Umbraco.sdf"La obtuvimos

Figura 8: Revisado del backup

mediante el comando strings probando en diferentes archivos de configuración grepeando pass, luego de encontrarla en esta ruta vimos que grepeando pass no nos daba mucha información adicional al correo de login, así que probamos otro filtro.

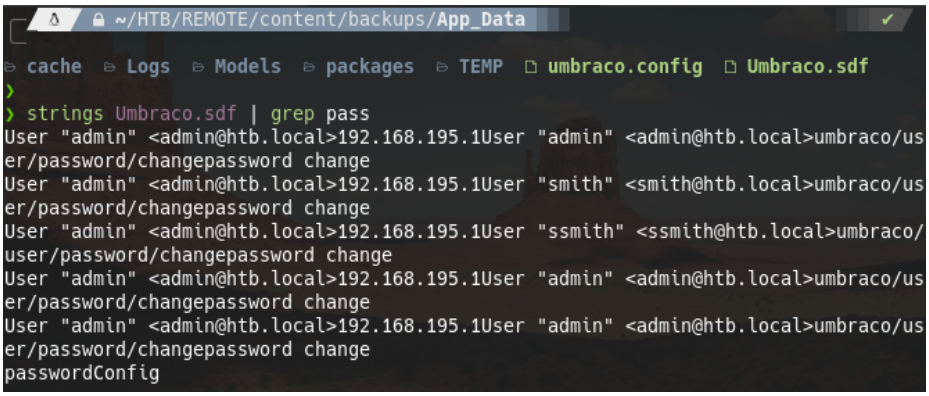

Figura 9: Encontrando el fichero con la contraseña

Entonces probando el filtro .ªdmin.en base a los resultados anteriors, y encontramos un hash, el cual mediante hash-identifier pudimos comprobar su naturaleza SHA1.

```
❯ strings Umbraco.sdf | grep admin
Administratoradmindefaulten-US
Administratoradmindefaulten-USb22924d5-57de-468e-9df4-0961cf6aa30d
Administratoradminb8be16afba8c314ad33d812f22a04991b90e2aaa{"hashAlgorithm":"SHA1"}e
n-USf8512f97-cab1-4a4b-a49f-0a2054c47a1d
adminadmin@htb.localb8be16afba8c314ad33d812f22a04991b90e2aaa{"hashAlgorithm":"SHA1"
}admin@htb.localen-USfeb1a998-d3bf-406a-b30b-e269d7abdf50
adminadmin@htb.localb8be16afba8c314ad33d812f22a04991b90e2aaa{"hashAlgorithm":"SHA1"
}admin@htb.localen-US82756c26-4321-4d27-b429-1b5c7c4f882f
User "admin" <admin@htb.local>192.168.195.1User "admin" <admin@htb.local>umbraco/us
er/password/changepassword change
User "admin" <admin@htb.local>192.168.195.1User "admin" <admin@htb.local>umbraco/us
er/sign-in/logoutlogout success
User "SYSTEM" 192.168.195.1User "admin" <admin@htb.local>umbraco/user/saveupdating
LastLoginDate, LastPasswordChangeDate, UpdateDate
User "SYSTEM" 192.168.195.1User "admin" <admin@htb.local>umbraco/user/sign-in/login
login success
User "admin" <admin@htb.local>192.168.195.1User "admin" <admin@htb.local>umbraco/us
er/sign-in/logoutlogout success
```

Figura 10: Encontrando la contraseña cifrada

Ahora posteriormente lo que sigue es intentar el crackeo de esta contraseña cifrada en SHA1, para nuestra suerte este tipo de cifrado es completamente obsoleto al poseer posiblidad de colisiones en su algoritmo. Por lo cual en diferentes sitios online se pueden encontrar formas de crackear la contraseña, y el resultado es la obtención de la contraseña "baconandcheese".

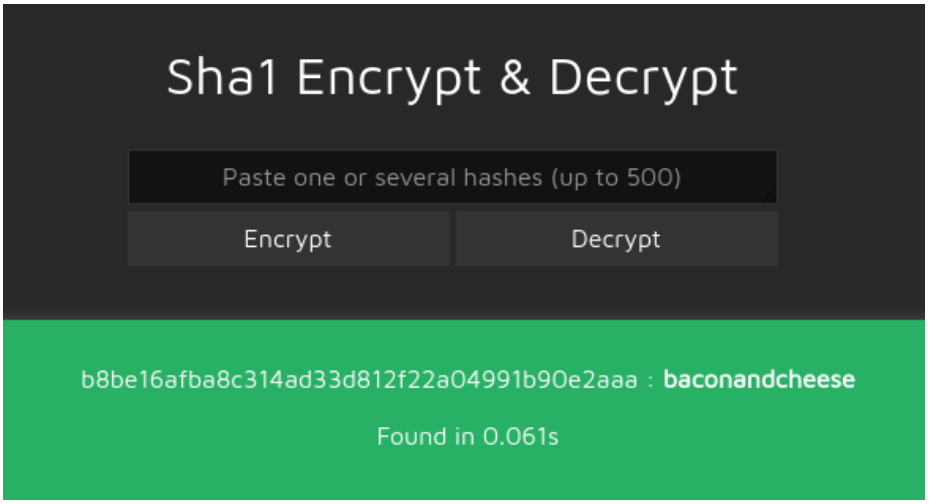

Figura 11: Encontrando la contraseña en texto claro

2.4. Explotación

2.5. Post Explotación

3. Fuse

3.1. Reconocimiento

3.2. Escaneo de Vulnerabilidades

3.3. Enumeración

3.4. Explotación

3.5. Post Explotación