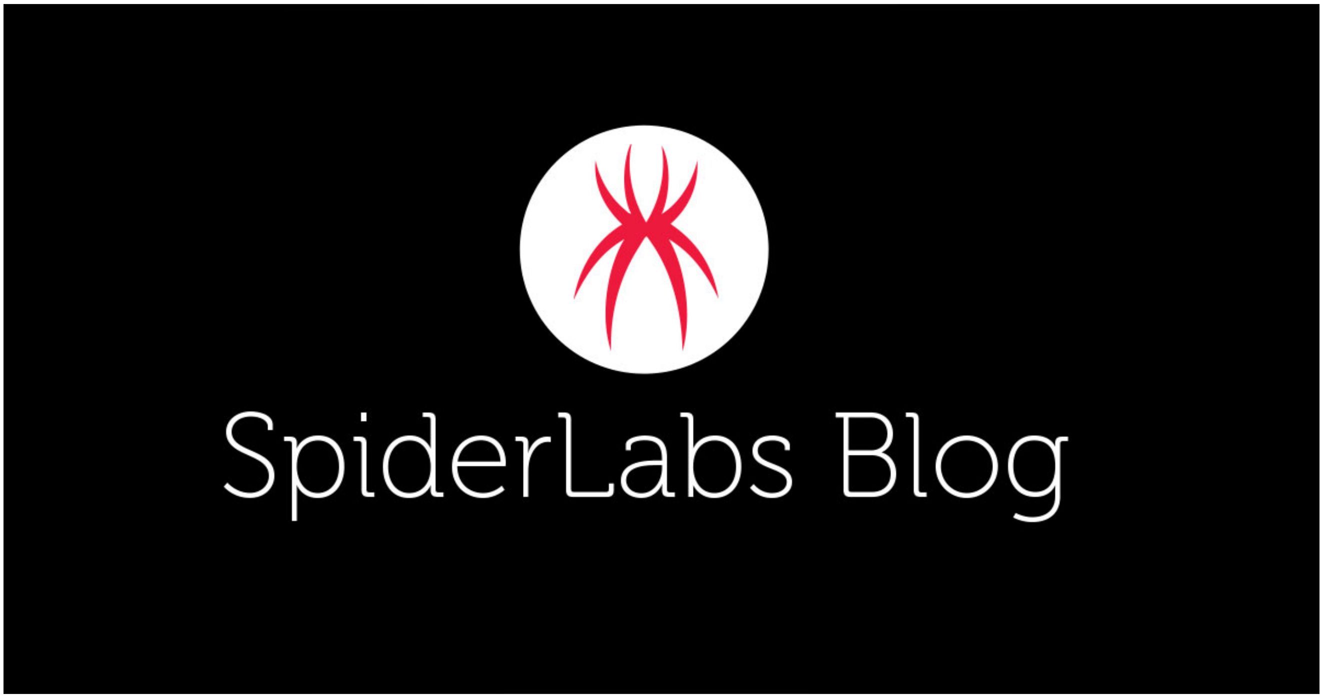
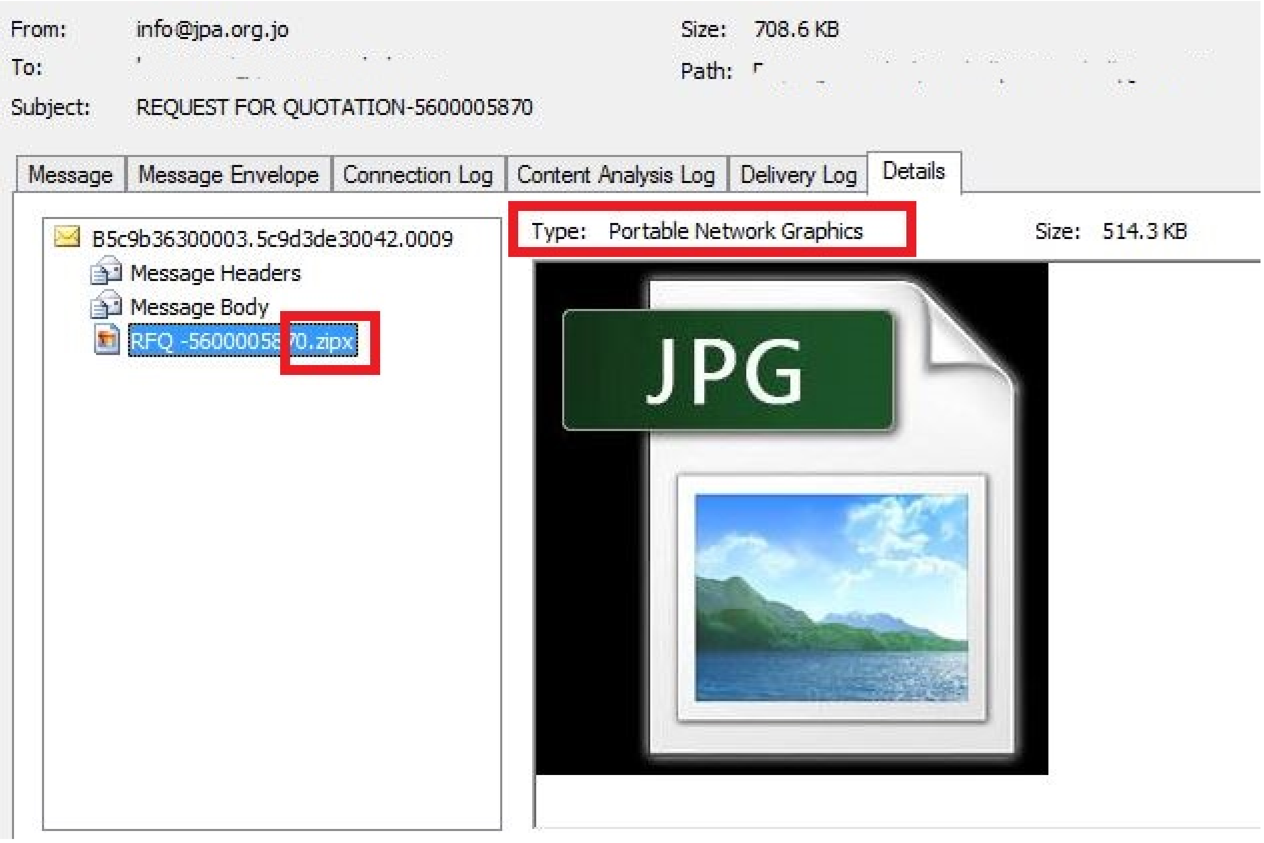
Spammed PNG file hides LokiBot

trustwave.com/en-us/resources/blogs/spiderlabs-blog/spammed-png-file-hides-lokibot/



Contributing authors: Phil Hay, Rodel Mendrez

Recently we came across a spam message from our traps that looked truly odd when viewed from our Secure Email Gateway console, as below:



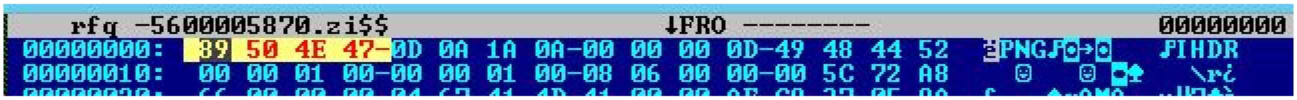
From: info@jpa.org.jo
To:
Subject: REQUEST FOR QUOTATION-5600005870
Size: 708.6 KB
Path:

Needless to say, there are several suspicious elements here:

- The attachment has a .zipx extension
- The gateway identified the message as a PNG image
- The image itself resembles a ‘JPG’ icon.
- The subject line is typical of those we see associated with malware.

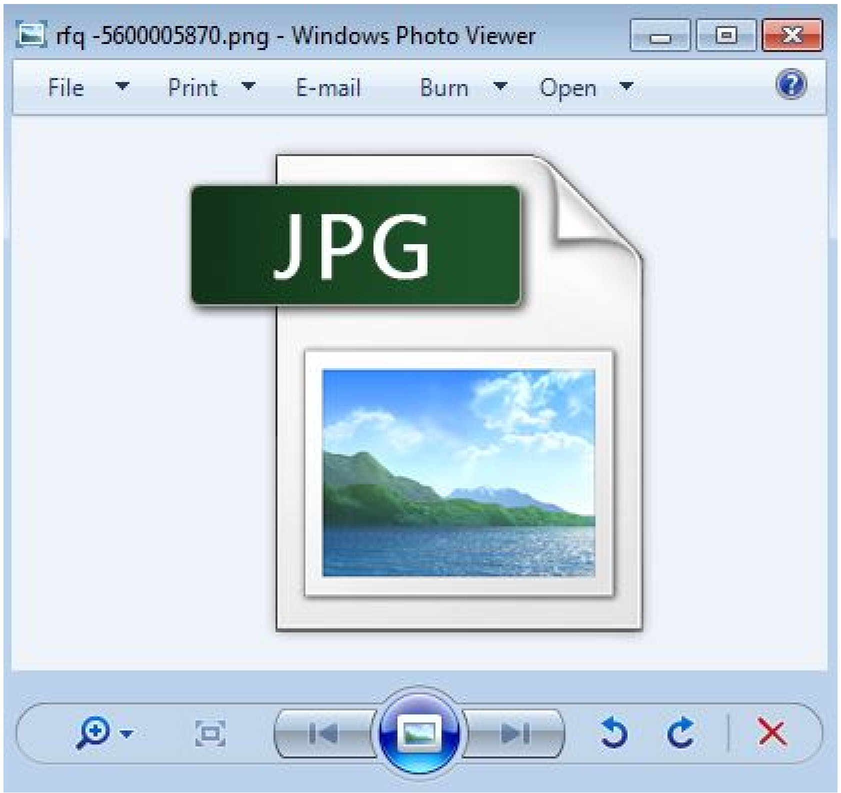
So what’s going on here? Let’s examine the file.

The header of the file indicates that it truly is a PNG.



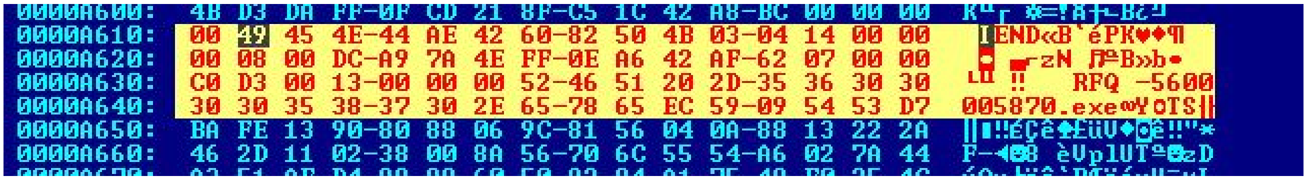
```
   rfq -5600005870.zi$$                  ↓FRO ---------                 00000000
 00000000:  89 50 4E 47-0D 0A 1A 0A-00 00 00 0D-49 48 44 52  èPNG♪◙→◙   ♪IHDR
 00000010:  00 00 01 00-00 00 01 00-08 06 00 00-00 5C 72 A8    ☺   ☺ ◘♠   \r¿
```

The image file can be opened with an image viewer:



However, the .zipx extension suggests something else is going on here. Let's look further into the file.

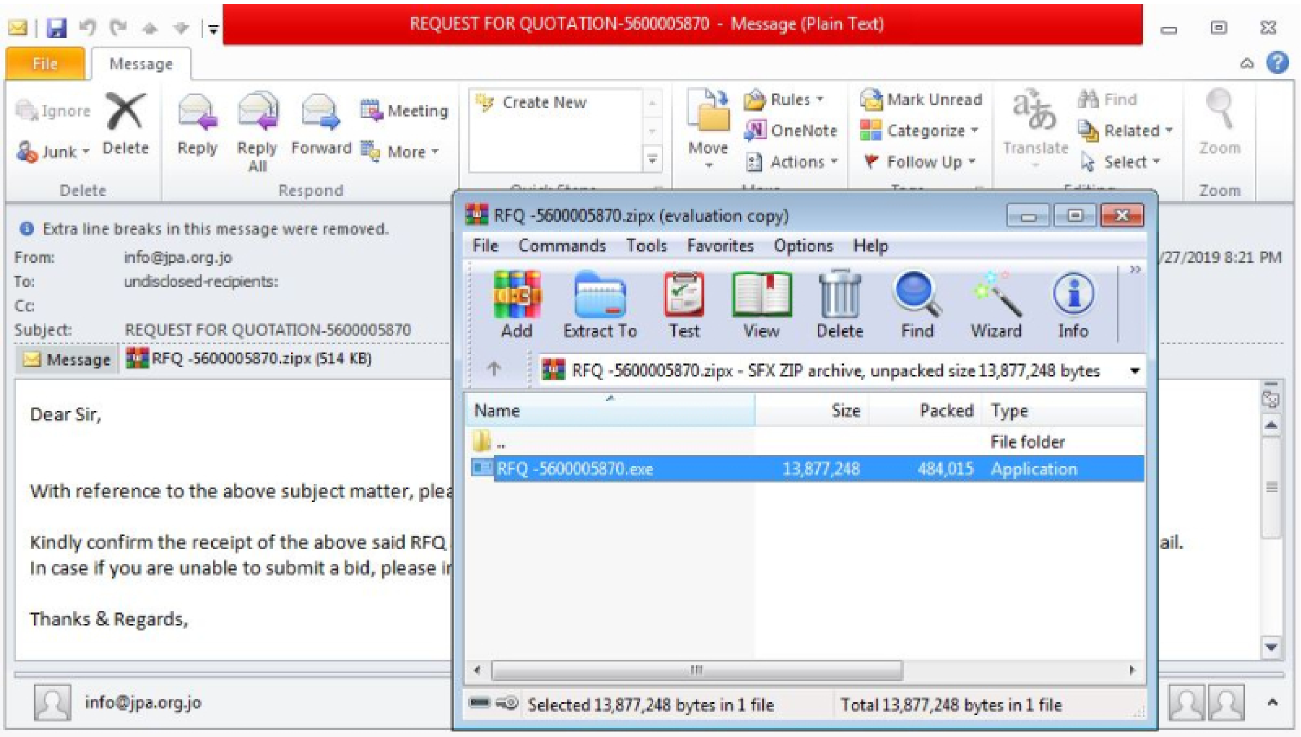
In a PNG file, IEND is supposed to mark the end of the image and is supposed to appear last. But in this file there is a bunch of data after IEND. If you look at the snippet below, you can see a PK header marker indicating some kind of zip archive, and a filename called RFQ -5600005870.exe. The PNG format specification appears to allow for such extraneous data, it is up to the application to decide to try and interpret or ignore such data.



```
0000A600:  4B D3 DA FF-0F CD 21 8F-C5 1C 42 A8-BC 00 00 00
0000A610:  00 49 45 4E-44 AE 42 60-82 50 4B 03-04 14 00 00   IEND«B`éPK♥♦¶
0000A620:  00 08 00 DC-A9 7A 4E FF-0E A6 42 AF-62 07 00 00      zN  B»b•
0000A630:  C0 D3 00 13-00 00 00 52-46 51 20 2D-35 36 30 30   └╙ ‼   RFQ -5600
0000A640:  30 30 35 38-37 30 2E 65-78 65 EC 59-09 54 53 D7  005870.exe∞Y○TS╫
0000A650:  BA FE 13 90-80 88 06 9C-81 56 04 0A-88 13 22 2A  ║■‼ÉÇê♠£üV♦◙ê‼"*
0000A660:  46 2D 11 02-38 00 8A 56-70 6C 55 54-A6 02 7A 44  F-◄☻8 èVplUTª☻zD
```

So, let's try unzipping it. WinZip and 7-Zip gave errors upon trying to unzip the file, but WinRAR had no such problem, happily extracting the file RFQ -5600005870.exe. Interestingly, if you alter the extension to anything other than zip or zipx, 7-Zip will also happily extract the .exe file. It seems that some decompression utilities will traverse the whole file looking for suitable stuff to unpack.

Here's how the message could have appeared to a user who had WinRAR - clicking the attachment would open up WinRAR for extracting the payload.



REQUEST FOR QUOTATION-5600005870 - Message (Plain Text)

Extra line breaks in this message were removed.

From: info@jpa.org.jo

To: undisclosed-recipients:

Cc:

Subject: REQUEST FOR QUOTATION-5600005870

/27/2019 8:21 PM

Message | RFQ -5600005870.zipx (514 KB)

Dear Sir,

With reference to the above subject matter, plea

Kindly confirm the receipt of the above said RFQ

In case if you are unable to submit a bid, please ir

ail.

Thanks & Regards,

info@jpa.org.jo

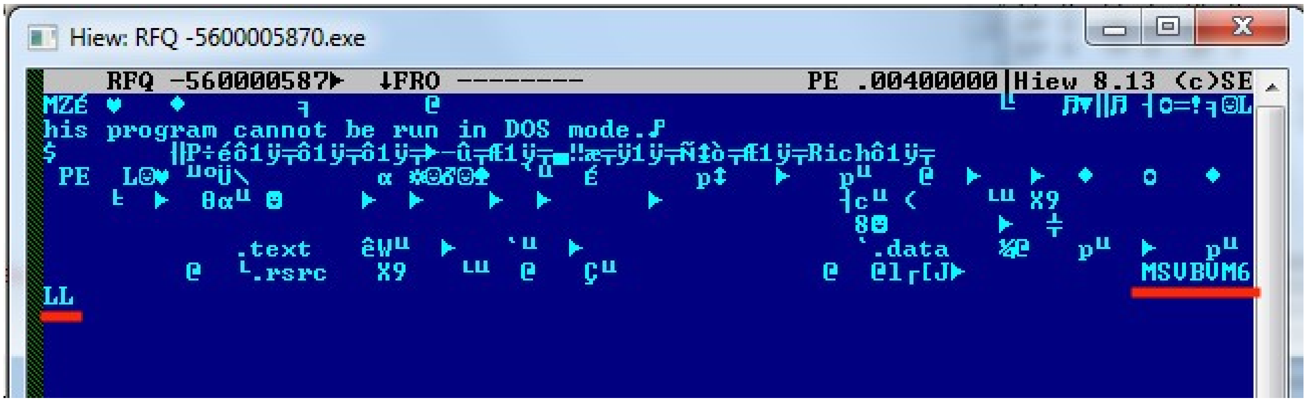
RFQ -5600005870.zipx (evaluation copy)

RFQ -5600005870.zipx - SFX ZIP archive, unpacked size 13,877,248 bytes

Name	Size	Packed	Type
..			File folder
RFQ -5600005870.exe	13,877,248	484,015	Application

Selected 13,877,248 bytes in 1 file | Total 13,877,248 bytes in 1 file

Analyzing the extracted .exe file indicates that it consists of multiple stages, with the first stage executable compiled with Visual Basic.

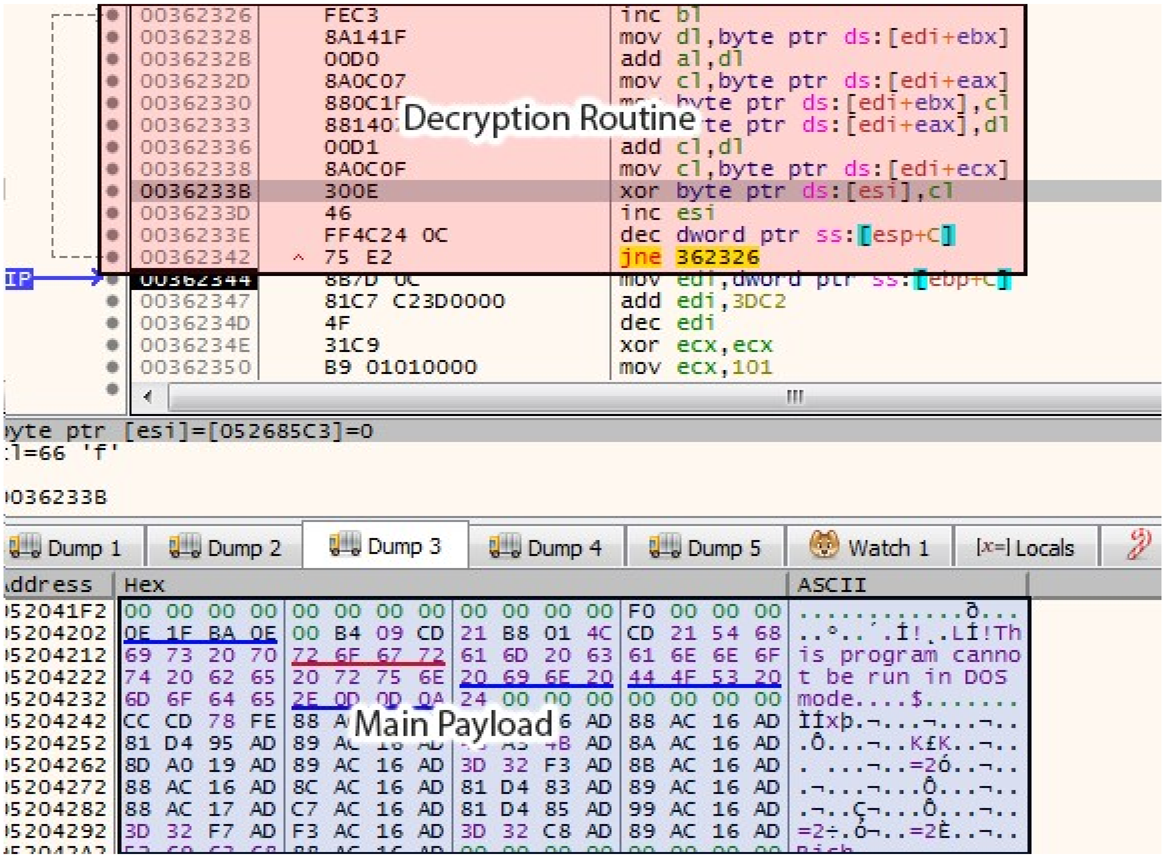


Hiew: RFQ -5600005870.exe

```
    RFQ -560000587▶  ↓FRO ---------             PE .00400000|Hiew 8.13 (c)SE
MZÉ ♥     ◆         ╕         @                                    ╚    ╓▾║╓ ┤ʘ=!╕☺L
his program cannot be run in DOS mode.♪
$        ║P÷éô1ÿ╤ô1ÿ╤ô1ÿ╤▶─û╤Æ1ÿ╤▄‼æ╤ÿ1ÿ╤Ñ‡ò╤Æ1ÿ╤Richô1ÿ╤
 PE   L☺♥ ᵘºÜ\          α ☼☺♂☺♠    `ᵘ    É        p‡    ▶     pᵘ   @   ▶    ▶    ◆    ○    ◆
    ŧ  ▶   θαᵘ ☻      ▶  ▶     ▶  ▶        ▶          ┤cᵘ (     ᴸᵘ X9
                                                        8☻      ▶  ┼
             .text   êWᵘ  ▶    `ᵘ   ▶                   `.data  ¾@   pᵘ  ▶   pᵘ
          @  ᴸ.rsrc   X9   ᴸᵘ  @    çᵘ                 @  @l┌[J▶             MSVBVM6
LL
```

This first stage function is to decrypt the main payload into the memory and execute it using a common technique called Process Hollowing, where a new process is created in a suspended state, its memory is unmapped and the malicious code replaces it.

Dumping the decrypted new process from memory, we end up with the main payload, as below.



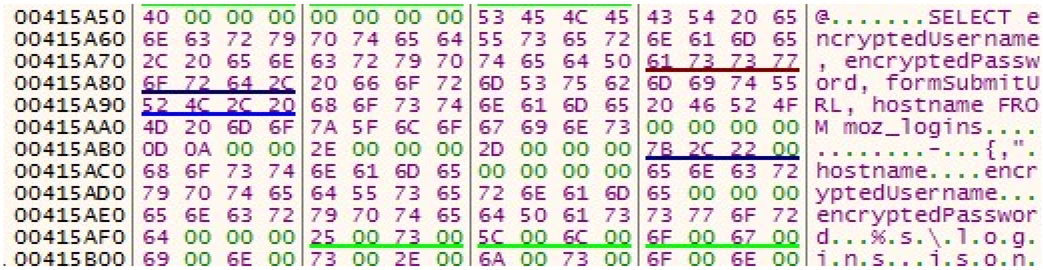
```
00362326    FEC3            inc bl
00362328    8A141F          mov dl,byte ptr ds:[edi+ebx]
0036232B    00D0            add al,dl
0036232D    8A0C07          mov cl,byte ptr ds:[edi+eax]
00362330    880C1F          mov byte ptr ds:[edi+ebx],cl
00362333    88140           mov byte ptr ds:[edi+eax],dl
00362336    00D1            add cl,dl
00362338    8A0C0F          mov cl,byte ptr ds:[edi+ecx]
0036233B    300E            xor byte ptr ds:[esi],cl
0036233D    46              inc esi
0036233E    FF4C24 0C       dec dword ptr ss:[esp+C]
00362342  ^ 75 E2           jne 362326
00362344    8B7D 0C         mov edi,dword ptr ss:[ebp+C]
00362347    81C7 C23D0000   add edi,3DC2
0036234D    4F              dec edi
0036234E    31C9            xor ecx,ecx
00362350    B9 01010000     mov ecx,101

yte ptr [esi]=[052685C3]=0
:l=66 'f'

)036233B
```

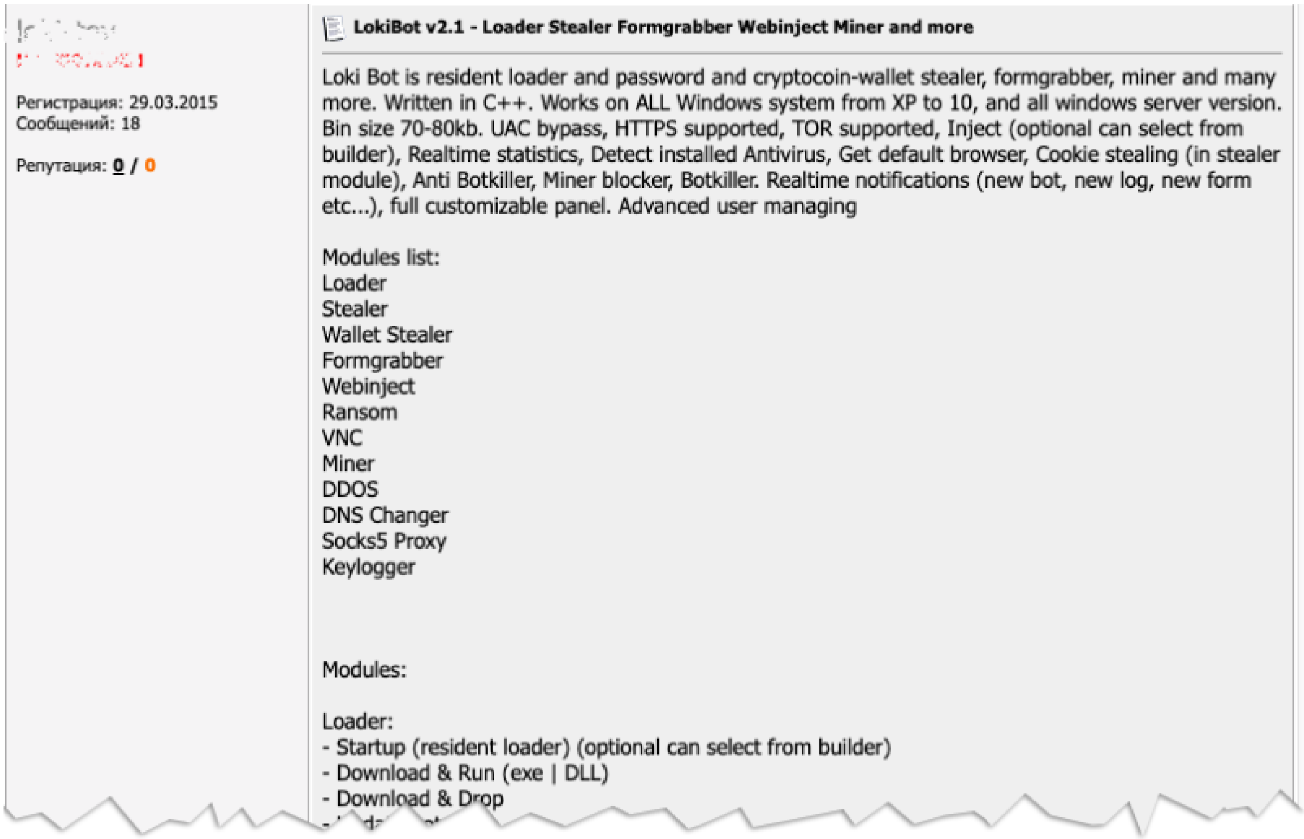
```
ddress   Hex                                              ASCII
52041F2  00 00 00 00 00 00 00 00 00 00 00 00 F0 00 00 00  ............ð...
5204202  0E 1F BA 0E 00 B4 09 CD 21 B8 01 4C CD 21 54 68  ..º..´.Í!¸.LÍ!Th
5204212  69 73 20 70 72 6F 67 72 61 6D 20 63 61 6E 6E 6F  is program canno
5204222  74 20 62 65 20 72 75 6E 20 69 6E 20 44 4F 53 20  t be run in DOS 
5204232  6D 6F 64 65 2E 0D 0D 0A 24 00 00 00 00 00 00 00  mode....$.......
5204242  CC CD 78 FE 88 AC 16 AD [illegible] AC 16 AD 88 AC 16 AD  ÌÍxþ.¬...¬...¬..
5204252  81 D4 95 AD 89 AC 16 AD 4B A3 4B AD 8A AC 16 AD  .Ô...¬..K£K..¬..
5204262  8D A0 19 AD 89 AC 16 AD 3D 32 F3 AD 8B AC 16 AD  . ...¬..=2ó..¬..
5204272  88 AC 16 AD 8C AC 16 AD 81 D4 83 AD 89 AC 16 AD  .¬...¬...Ô...¬..
5204282  88 AC 17 AD C7 AC 16 AD 81 D4 85 AD 99 AC 16 AD  .¬..Ç¬...Ô...¬..
5204292  3D 32 F7 AD F3 AC 16 AD 3D 32 C8 AD 89 AC 16 AD  =2÷.ó¬..=2È..¬..
```

Some interesting strings in the malware body start to give a hint of what this malware does.



```
00415A50  40 00 00 00 00 00 00 00 53 45 4C 45 43 54 20 65  @.......SELECT e
00415A60  6E 63 72 79 70 74 65 64 55 73 65 72 6E 61 6D 65  ncryptedUsername
00415A70  2C 20 65 6E 63 72 79 70 74 65 64 50 61 73 73 77  , encryptedPassw
00415A80  6F 72 64 2C 20 66 6F 72 6D 53 75 62 6D 69 74 55  ord, formSubmitU
00415A90  52 4C 2C 20 68 6F 73 74 6E 61 6D 65 20 46 52 4F  RL, hostname FRO
00415AA0  4D 20 6D 6F 7A 5F 6C 6F 67 69 6E 73 00 00 00 00  M moz_logins....
00415AB0  0D 0A 00 00 2E 00 00 00 2D 00 00 00 7B 2C 22 00  ........-...{,".
00415AC0  68 6F 73 74 6E 61 6D 65 00 00 00 00 65 6E 63 72  hostname....encr
00415AD0  79 70 74 65 64 55 73 65 72 6E 61 6D 65 00 00 00  yptedUsername...
00415AE0  65 6E 63 72 79 70 74 65 64 50 61 73 73 77 6F 72  encryptedPasswor
00415AF0  64 00 00 00 25 00 73 00 5C 00 6C 00 6F 00 67 00  d...%.s.\.l.o.g.
00415B00  69 00 6E 00 73 00 2E 00 6A 00 73 00 6F 00 6E 00  i.n.s...j.s.o.n.
```

Further analysis of the sample indicates that it is the well-known LokiBot information stealing Trojan. LokiBot is a multi-purpose modular trojan that attempts to steal passwords and other information from browsers, mail, FTP clients and other applications, as well as a raft of other functions. LokiBot is freely available in the underground markets where it can be bought quite cheaply - $300 can get you some password stealing capability.



[redacted]

Регистрация: 29.03.2015
Сообщений: 18

Репутация: 0 / 0

LokiBot v2.1 - Loader Stealer Formgrabber Webinject Miner and more

Loki Bot is resident loader and password and cryptocoin-wallet stealer, formgrabber, miner and many more. Written in C++. Works on ALL Windows system from XP to 10, and all windows server version. Bin size 70-80kb. UAC bypass, HTTPS supported, TOR supported, Inject (optional can select from builder), Realtime statistics, Detect installed Antivirus, Get default browser, Cookie stealing (in stealer module), Anti Botkiller, Miner blocker, Botkiller. Realtime notifications (new bot, new log, new form etc...), full customizable panel. Advanced user managing

Modules list:
Loader
Stealer
Wallet Stealer
Formgrabber
Webinject
Ransom
VNC
Miner
DDOS
DNS Changer
Socks5 Proxy
Keylogger

Modules:

Loader:
- Startup (resident loader) (optional can select from builder)
- Download & Run (exe | DLL)
- Download & Drop

LokiBot’s availability means it is widely used, and over the past year we have been seeing many spam messages with attached LokiBots, but never before one where the payload is hidden inside a PNG file.

The attacker likely used the PNG format to hide the executable from inspection by the email scanning gateway. The giveaway is the .zipx extension. If a user happened to have WinRAR installed, and received such a message, then clicking on the attachment would fire up WinRAR for the payload exe to be extracted by the user. The upshot is we may all want to inspect those PNG files a little closer.

IOCs:

.zipx attachment

SHA256: 3654011653e7289620041cb831bf93e6c480815feede72320cde94f20ce7f185

Extracted executable
SHA256: c9ddcf7d0cd026cdeac9586515b4d591c1ca63ee9c009cd00b198178e5e84f03

CnC of LokiBot
hxxp://slimcase[.]ml/evans/fre.php