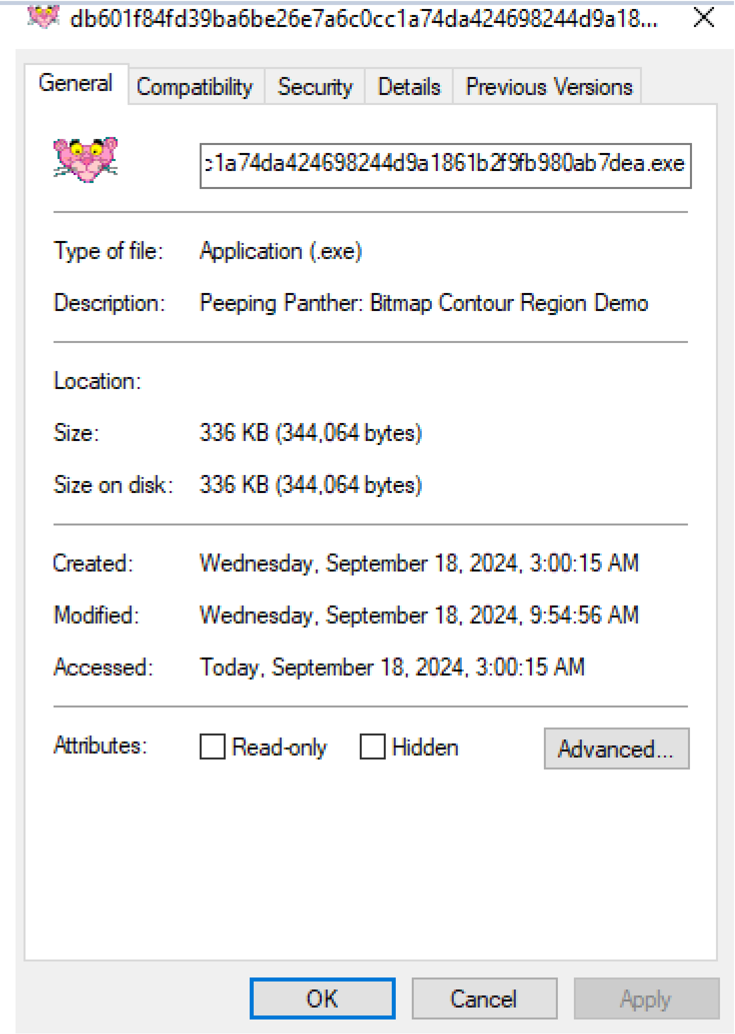
SHA256: db601f84fd39ba6be26e7a6c0cc1a74da424698244d9a1861b2f9fb980ab7dea

This was an attempt to try out in unpacking BuerLoader, but in this case it does not look like I was successful in unpacking the file and I probably dumped the wrong binary after attaching to the child process.

Figures 1 and 2 below show a simple look at the file details such as the name and size as well as the alleged author and legal trademarks.



db601f84fd39ba6be26e7a6c0cc1a74da424698244d9a18...

:1a74da424698244d9a1861b2f9fb980ab7dea.exe

Type of file: Application (.exe)

Description: Peeping Panther: Bitmap Contour Region Demo

Location:

Size: 336 KB (344,064 bytes)

Size on disk: 336 KB (344,064 bytes)

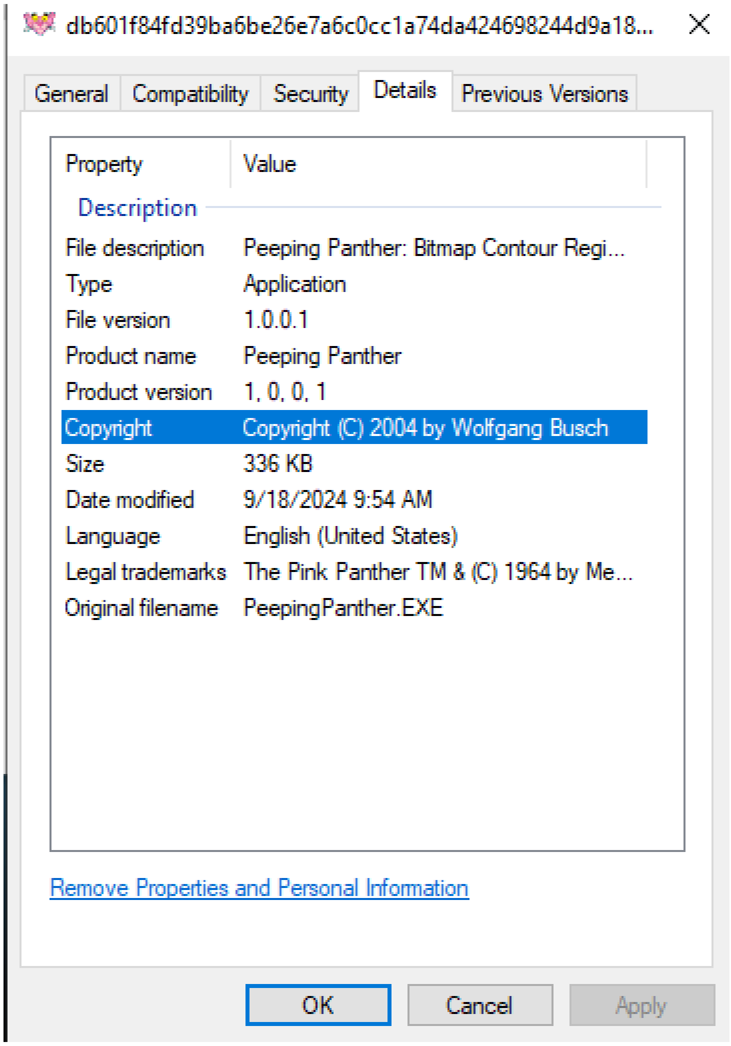
Created: Wednesday, September 18, 2024, 3:00:15 AM

Modified: Wednesday, September 18, 2024, 9:54:56 AM

Accessed: Today, September 18, 2024, 3:00:15 AM

Attributes: □ Read-only □ Hidden

Figure 1. File Details

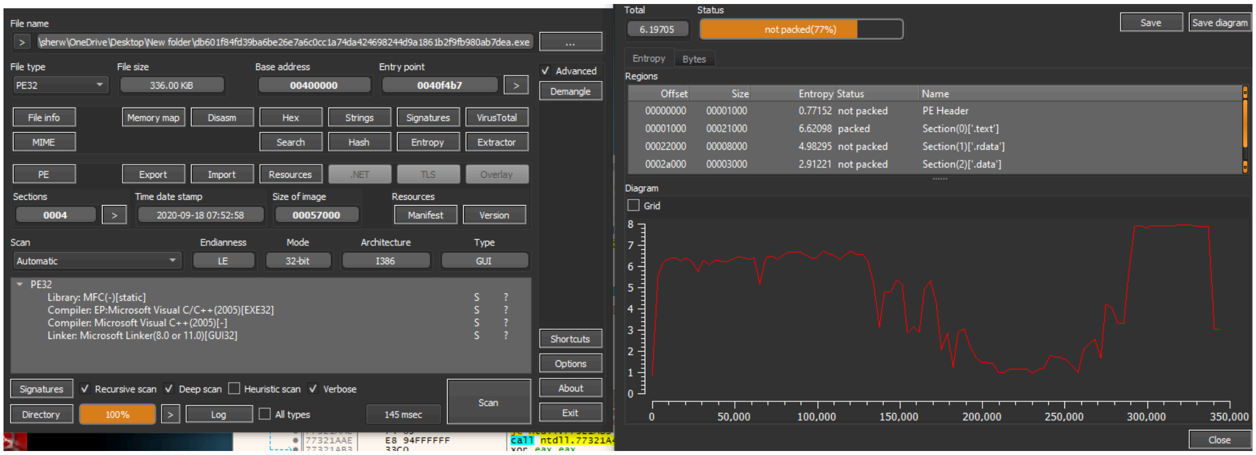


db601f84fd39ba6be26e7a6c0cc1a74da424698244d9a18...

Property	Value
Description	
File description	Peeping Panther: Bitmap Contour Regi...
Type	Application
File version	1.0.0.1
Product name	Peeping Panther
Product version	1, 0, 0, 1
Copyright	Copyright (C) 2004 by Wolfgang Busch
Size	336 KB
Date modified	9/18/2024 9:54 AM
Language	English (United States)
Legal trademarks	The Pink Panther TM & (C) 1964 by Me...
Original filename	PeepingPanther.EXE

Figure 2. File Details

Opening the file in Detect-it-Easy to gain information on the the file type and compiler as well as the entropy to see if the payload is packer or not. The file is also opened on PE-Studio to look for additional details and flags for the imports or strings as well.



File name: \sherw\OneDrive\Desktop\New folder\db601f84fd39ba6be26e7a6c0cc1a74da424698244d9a1861b2f9fb980ab7dea.exe

File type: PE32
File size: 336.00 KiB
Base address: 00400000
Entry point: 0040f4b7
✓ Advanced

Sections: 0004
Time date stamp: 2020-09-18 07:52:58
Size of image: 00057000

Scan: Automatic
Endianness: LE
Mode: 32-bit
Architecture: I386
Type: GUI

```
PE32
    Library: MFC(-)[static]                        S    ?
    Compiler: EP:Microsoft Visual C/C++(2005)[EXE32]  S    ?
    Compiler: Microsoft Visual C++(2005)[-]        S    ?
    Linker: Microsoft Linker(8.0 or 11.0)[GUI32]   S    ?
```

✓ Recursive scan ✓ Deep scan □ Heuristic scan ✓ Verbose
□ All types
100%
145 msec

Total: 6.19705
Status: not packed(77%)

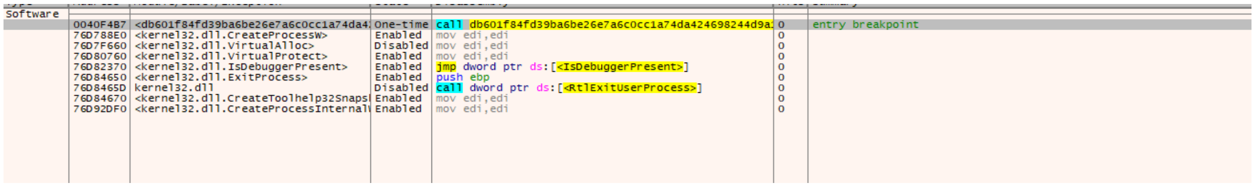
Regions

Offset	Size	Entropy	Status	Name
00000000	00001000	0.77152	not packed	PE Header
00001000	00021000	6.62098	packed	Section(0)['.text']
00022000	00008000	4.98295	not packed	Section(1)['.rdata']
0002a000	00003000	2.91221	not packed	Section(2)['.data']

Diagram
□ Grid

Figure 3. Detect it Easy

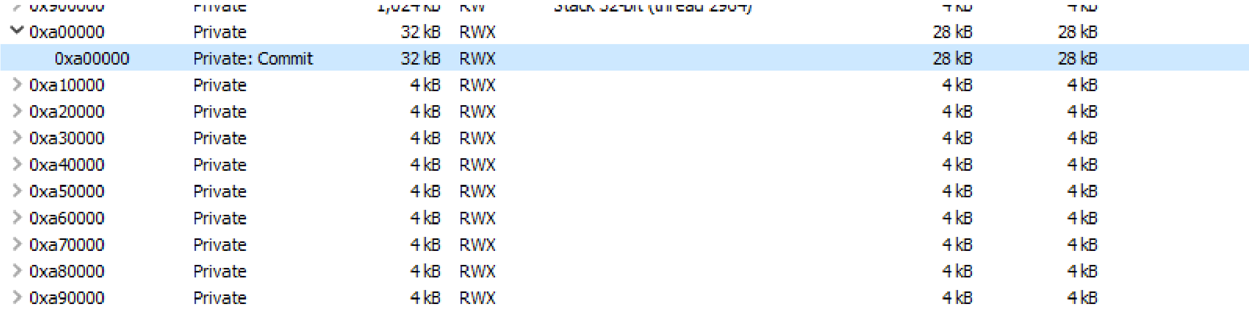
Initial set breakpoints on x32dbg.



```
Software
         0040F4B7 <db601f84fd39ba6be26e7a6c0cc1a74da4 One-time call db601f84fd39ba6be26e7a6c0cc1a74da424698244d9a 0  entry breakpoint
         76D788E0 <kernel32.dll.CreateProcessW>       Enabled  mov edi,edi                                    0
         76D7F660 <kernel32.dll.VirtualAlloc>         Disabled mov edi,edi                                    0
         76D80760 <kernel32.dll.VirtualProtect>       Enabled  mov edi,edi                                    0
         76D82370 <kernel32.dll.IsDebuggerPresent>    Enabled  jmp dword ptr ds:[<IsDebuggerPresent>]         0
         76D84650 <kernel32.dll.ExitProcess>          Enabled  push ebp                                       0
         76D8465D kernel32.dll                        Disabled call dword ptr ds:[<RtlExitUserProcess>]       0
         76D84670 <kernel32.dll.CreateToolhelp32Snaps Enabled  mov edi,edi                                    0
         76D92DF0 <kernel32.dll.CreateProcessInternal Enabled  mov edi,edi                                    0
```

Figure 4. Breakpoints

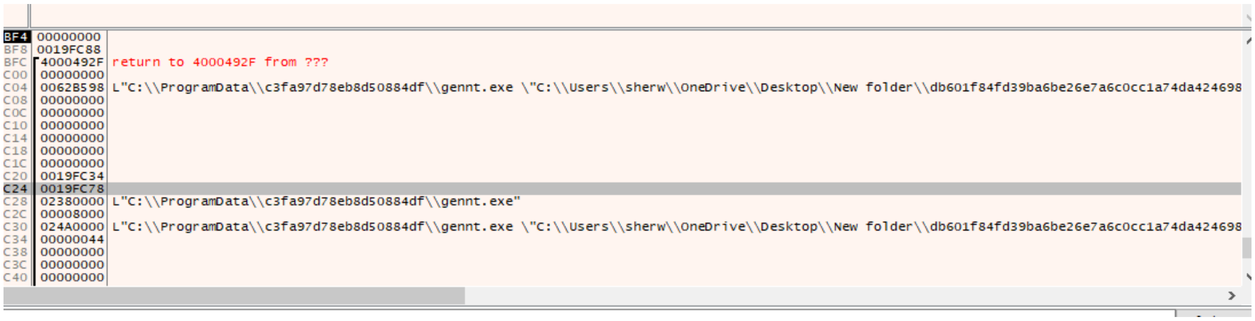
Looking at the memory regions for readable and writable memory to see if I have already missed something while following VirtualAlloc calls.



```
  0x900000    Private          1,024 kB  RW    Stack 32-bit (thread 2904)   4 kB    4 kB
v 0xa00000    Private             32 kB  RWX                               28 kB   28 kB
    0xa00000  Private: Commit     32 kB  RWX                               28 kB   28 kB
> 0xa10000    Private              4 kB  RWX                                4 kB    4 kB
> 0xa20000    Private              4 kB  RWX                                4 kB    4 kB
> 0xa30000    Private              4 kB  RWX                                4 kB    4 kB
> 0xa40000    Private              4 kB  RWX                                4 kB    4 kB
> 0xa50000    Private              4 kB  RWX                                4 kB    4 kB
> 0xa60000    Private              4 kB  RWX                                4 kB    4 kB
> 0xa70000    Private              4 kB  RWX                                4 kB    4 kB
> 0xa80000    Private              4 kB  RWX                                4 kB    4 kB
> 0xa90000    Private              4 kB  RWX                                4 kB    4 kB
```

Figure 5. Looking at the Memory Regions

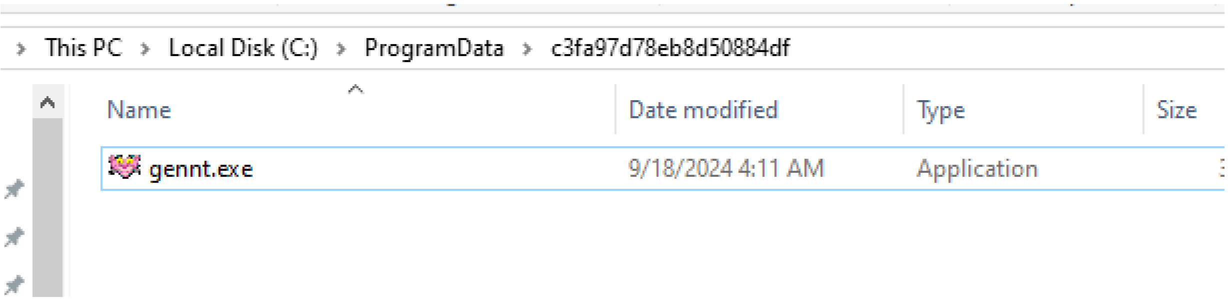
Following VirtualAlloc multiple times allocates data a multitude of times but it does not look like VirtualProtect was ever called. Instead it hits a CreateProcessW bp and shows that a file is going to be created on a folder in C:\Programdata.



```
BF4 00000000
BF8 0019FC88
BFC 4000492F return to 4000492F from ???
C00 00000000
C04 0062B598 L"C:\\ProgramData\\c3fa97d78eb8d50884df\\gennt.exe \"C:\\Users\\sherw\\OneDrive\\Desktop\\New folder\\db601f84fd39ba6be26e7a6c0cc1a74da424698
C08 00000000
C0C 00000000
C10 00000000
C14 00000000
C18 00000000
C1C 00000000
C20 0019FC34
C24 0019FC78
C28 02380000 L"C:\\ProgramData\\c3fa97d78eb8d50884df\\gennt.exe"
C2C 00008000
C30 024A0000 L"C:\\ProgramData\\c3fa97d78eb8d50884df\\gennt.exe \"C:\\Users\\sherw\\OneDrive\\Desktop\\New folder\\db601f84fd39ba6be26e7a6c0cc1a74da424698
C34 00000044
C38 00000000
C3C 00000000
C40 00000000
```

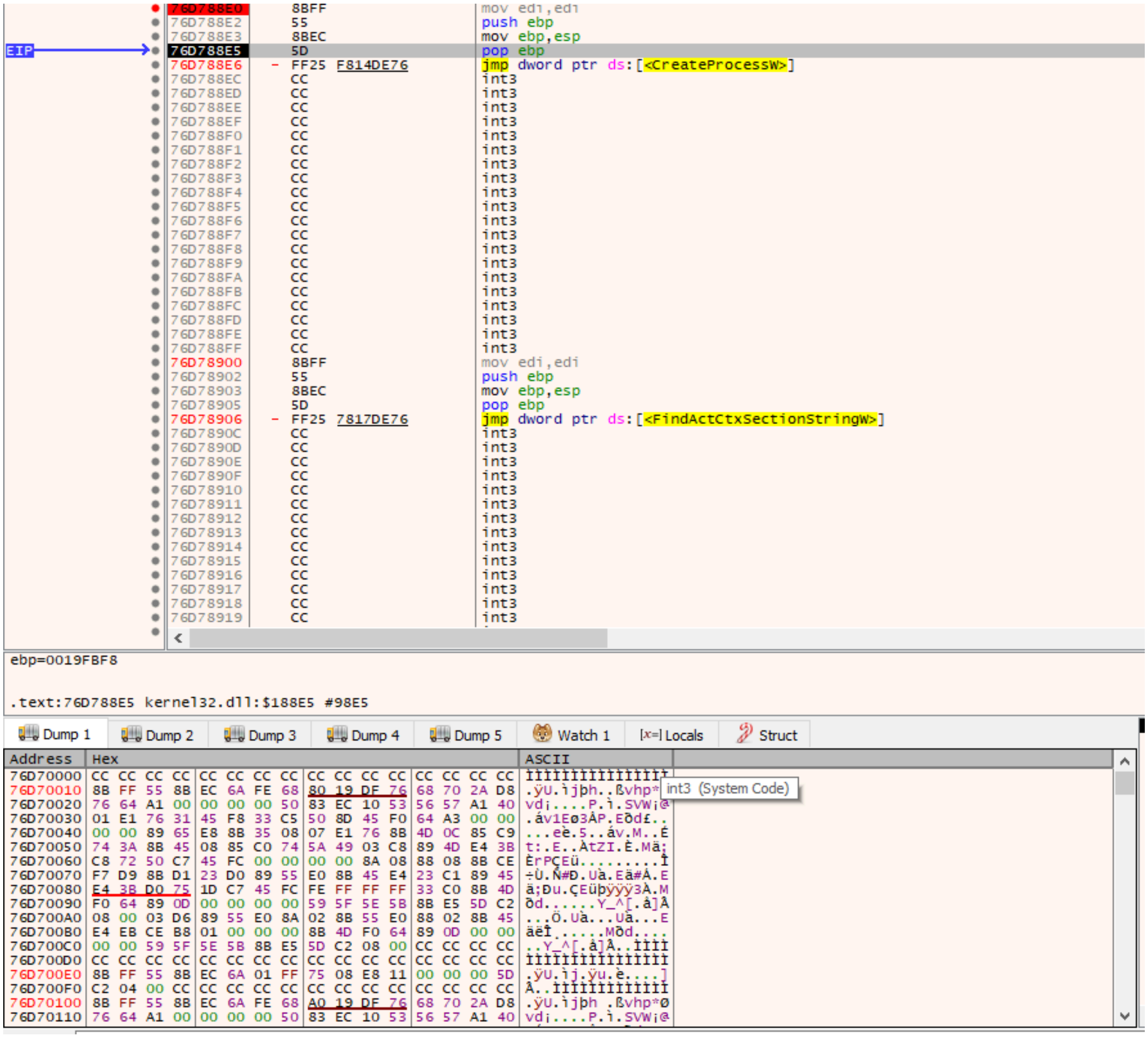
Figure 6. VirtualAlloc Calls

It looks like the process is going to create a new file called gennt.exe



```
> This PC > Local Disk (C:) > ProgramData > c3fa97d78eb8d50884df

Name                          Date modified         Type           Size
gennt.exe                     9/18/2024 4:11 AM     Application
```

Abnormal pop followed by a jmp. Following the data in this address shows suspicious content likely going to be injected into the new process.

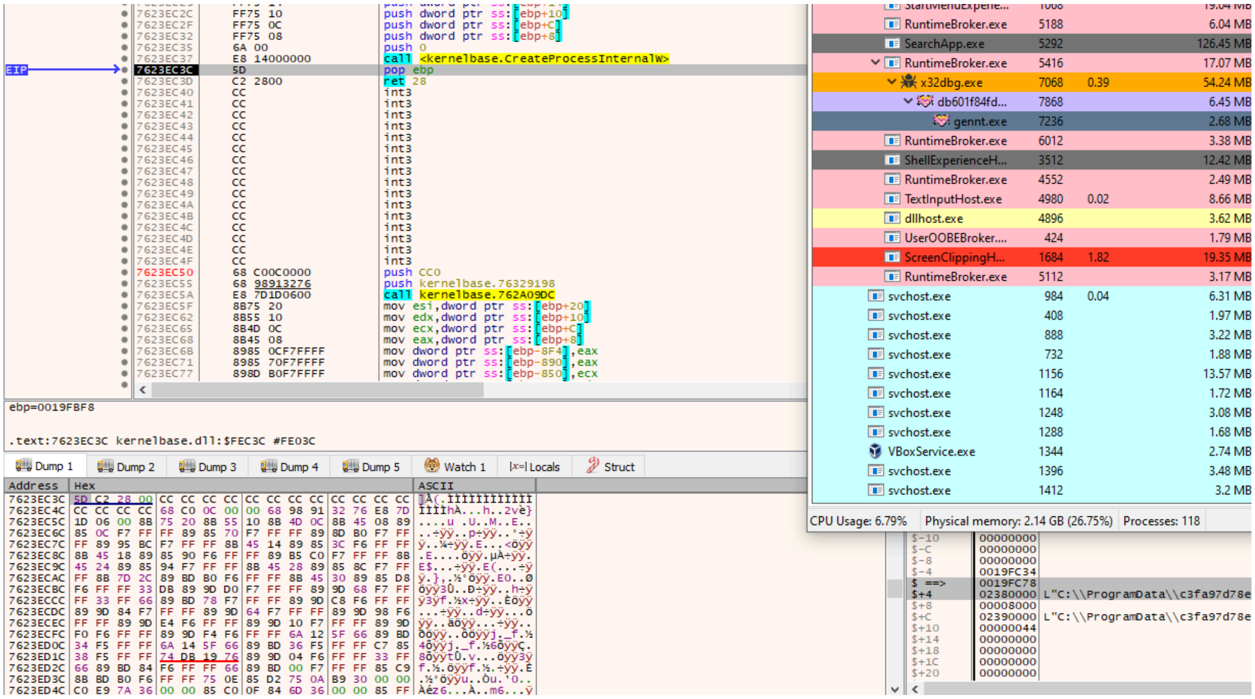


```
      76D788E0        8BFF              mov edi,edi
      76D788E2        55                push ebp
      76D788E3        8BEC              mov ebp,esp
EIP → 76D788E5        5D                pop ebp
      76D788E6      - FF25 F814DE76     jmp dword ptr ds:[<CreateProcessW>]
      76D788EC        CC                int3
      76D788ED        CC                int3
      76D788EE        CC                int3
      76D788EF        CC                int3
      76D788F0        CC                int3
      76D788F1        CC                int3
      76D788F2        CC                int3
      76D788F3        CC                int3
      76D788F4        CC                int3
      76D788F5        CC                int3
      76D788F6        CC                int3
      76D788F7        CC                int3
      76D788F8        CC                int3
      76D788F9        CC                int3
      76D788FA        CC                int3
      76D788FB        CC                int3
      76D788FC        CC                int3
      76D788FD        CC                int3
      76D788FE        CC                int3
      76D788FF        CC                int3
      76D78900        8BFF              mov edi,edi
      76D78902        55                push ebp
      76D78903        8BEC              mov ebp,esp
      76D78905        5D                pop ebp
      76D78906      - FF25 7817DE76     jmp dword ptr ds:[<FindActCtxSectionStringW>]
      76D7890C        CC                int3
      76D7890D        CC                int3
      76D7890E        CC                int3
      76D7890F        CC                int3
      76D78910        CC                int3
      76D78911        CC                int3
      76D78912        CC                int3
      76D78913        CC                int3
      76D78914        CC                int3
      76D78915        CC                int3
      76D78916        CC                int3
      76D78917        CC                int3
      76D78918        CC                int3
      76D78919        CC                int3

ebp=0019FBF8

.text:76D788E5 kernel32.dll:$188E5 #98E5
```

```
Address   Hex                                              ASCII
76D70000  CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC  ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
76D70010  8B FF 55 8B EC 6A FE 68 80 19 DF 76 68 70 2A D8  .ÿU.ìjþh..ßvhp*
76D70020  76 64 A1 00 00 00 00 50 83 EC 10 53 56 57 A1 40  vd¡....P.ì.SVW¡@
76D70030  01 E1 76 31 45 F8 33 C5 50 8D 45 F0 64 A3 00 00  .áv1Eø3ÅP.Eðd£..
76D70040  00 00 89 65 E8 8B 35 08 07 E1 76 8B 4D 0C 85 C9  ...eè.5..áv.M..É
76D70050  74 3A 8B 45 08 85 C0 74 5A 49 03 C8 89 4D E4 3B  t:.E..ÀtZI.È.Mä;
76D70060  C8 72 50 C7 45 FC 00 00 00 00 8A 08 88 08 8B CE  ÈrPÇEü.........Î
76D70070  F7 D9 8B D1 23 D0 89 55 E0 8B 45 E4 23 C1 89 45  ÷Ù.Ñ#Ð.Uà.Eä#Á.E
76D70080  E4 3B D0 75 1D C7 45 FC FE FF FF FF 33 C0 8B 4D  ä;Ðu.ÇEüþÿÿÿ3À.M
76D70090  F0 64 89 0D 00 00 00 00 59 5F 5E 5B 8B E5 5D C2  ðd......Y_^[.å]Â
76D700A0  08 00 03 D6 89 55 E0 8A 02 8B 55 E0 88 02 8B 45  ...Ö.Uà...Uà...E
76D700B0  E4 EB CE B8 01 00 00 00 8B 4D F0 64 89 0D 00 00  äëÎ¸.....Mðd....
76D700C0  00 00 59 5F 5E 5B 8B E5 5D C2 08 00 CC CC CC CC  ..Y_^[.å]Â..ÌÌÌÌ
76D700D0  CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC  ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
76D700E0  8B FF 55 8B EC 6A 01 FF 75 08 E8 11 00 00 00 5D  .ÿU.ìj.ÿu.è....]
76D700F0  C2 04 00 CC CC CC CC CC CC CC CC CC CC CC CC CC  Â..ÌÌÌÌÌÌÌÌÌÌÌÌÌ
76D70100  8B FF 55 8B EC 6A FE 68 A0 19 DF 76 68 70 2A D8  .ÿU.ìjþh .ßvhp*Ø
76D70110  76 64 A1 00 00 00 00 50 83 EC 10 53 56 57 A1 40  vd¡....P.ì.SVW¡@
```

Figure 7. Data on the Abnormal Epilogue

As seen on Process Hacker the child process gentt.exe has been created. X32dbg is then attached to this process for further analysis of the behavior of this child process.



```
7623EC2C  FF75 10        push dword ptr ss:[ebp+10]
7623EC2F  FF75 0C        push dword ptr ss:[ebp+C]
7623EC32  FF75 08        push dword ptr ss:[ebp+8]
7623EC35  6A 00          push 0
7623EC37  E8 14000000    call <kernelbase.CreateProcessInternalW>
7623EC3C  5D             pop ebp
7623EC3D  C2 2800        ret 28
7623EC40  CC             int3
7623EC41  CC             int3
7623EC42  CC             int3
7623EC43  CC             int3
7623EC44  CC             int3
7623EC45  CC             int3
7623EC46  CC             int3
7623EC47  CC             int3
7623EC48  CC             int3
7623EC49  CC             int3
7623EC4A  CC             int3
7623EC4B  CC             int3
7623EC4C  CC             int3
7623EC4D  CC             int3
7623EC4E  CC             int3
7623EC4F  CC             int3
7623EC50  68 C00C0000    push CC0
7623EC55  68 98913276    push kernelbase.76329198
7623EC5A  E8 7D1D0600    call kernelbase.762A09DC
7623EC5F  8B75 20        mov esi,dword ptr ss:[ebp+20]
7623EC62  8B55 10        mov edx,dword ptr ss:[ebp+10]
7623EC65  8B4D 0C        mov ecx,dword ptr ss:[ebp+C]
7623EC68  8B45 08        mov eax,dword ptr ss:[ebp+8]
7623EC6B  8985 0CF7FFFF  mov dword ptr ss:[ebp-8F4],eax
7623EC71  8985 70F7FFFF  mov dword ptr ss:[ebp-890],eax
7623EC77  898D B0F7FFFF  mov dword ptr ss:[ebp-850],ecx

ebp=0019FBF8

.text:7623EC3C kernelbase.dll:$FEC3C #FE03C
```

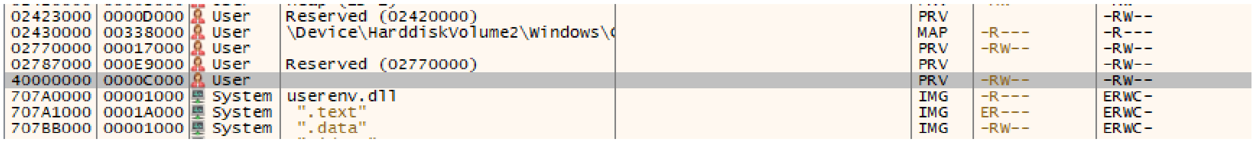
```
Address   Hex                                              ASCII
7623EC3C  5D C2 28 00 CC CC CC CC CC CC CC CC CC CC CC CC  ]Â(.ÌÌÌÌÌÌÌÌÌÌÌÌ
7623EC4C  CC CC CC CC 68 C0 0C 00 00 68 98 91 32 76 E8 7D  ÌÌÌÌhÀ...h..2vè}
7623EC5C  1D 06 00 8B 75 20 8B 55 10 8B 4D 0C 8B 45 08 89  ....u .U..M..E..
7623EC6C  85 0C F7 FF FF 89 85 70 F7 FF FF 89 8D B0 F7 FF  ..÷ÿÿ..p÷ÿÿ..°÷ÿ
7623EC7C  FF 89 95 BC F7 FF FF 8B 45 14 89 85 3C F6 FF FF  ÿ..¼÷ÿÿ.E...<öÿÿ
7623EC8C  8B 45 18 89 85 90 F6 FF FF 89 B5 C0 F7 FF FF 8B  .E....öÿÿ.µÀ÷ÿÿ.
7623EC9C  45 24 89 85 94 F7 FF FF 8B 45 28 89 85 8C F7 FF  E$...÷ÿÿ.E(...÷ÿ
7623ECAC  FF 8B 7D 2C 89 BD B0 F6 FF FF 8B 45 30 89 85 D8  ÿ.},.½°öÿÿ.E0..Ø
7623ECBC  F6 FF FF 33 DB 89 9D D0 F7 FF FF 89 9D 68 F7 FF  öÿÿ3Û..Ð÷ÿÿ..h÷ÿ
7623ECCC  FF 33 FF 66 89 BD 78 F7 FF FF 89 9D C8 F6 FF FF  ÿ3ÿf.½x÷ÿÿ..Èöÿÿ
7623ECDC  89 9D 84 F7 FF FF 89 9D 64 F7 FF FF 89 9D 98 F6  ...÷ÿÿ..d÷ÿÿ...ö
7623ECEC  FF FF 89 9D E4 F6 FF FF 89 9D 10 F7 FF FF 89 9D  ÿÿ..äöÿÿ...÷ÿÿ..
7623ECFC  F0 F6 FF FF 89 9D F4 F6 FF FF 6A 12 5F 66 89 BD  ðöÿÿ..ôöÿÿj._f.½
7623ED0C  34 F5 FF FF 6A 14 5F 66 89 BD 36 F5 FF FF C7 85  4õÿÿj._f.½6õÿÿÇ.
7623ED1C  38 F5 FF FF 74 DB 19 76 89 9D 04 F6 FF FF 33 FF  8õÿÿtÛ.v...öÿÿ3ÿ
7623ED2C  66 89 BD 84 F6 FF FF 66 89 BD 00 F7 FF FF 85 C9  f.½.öÿÿf.½.÷ÿÿ.É
7623ED3C  8B BD B0 F6 FF FF 75 0E 85 D2 75 0A B9 30 00 00  .½°öÿÿu..Òu.¹0..
7623ED4C  C0 E9 7A 36 00 00 85 C0 0F 84 6D 36 00 00 85 FF  Àéz6...À..m6...ÿ
```

```
RuntimeBroker.exe        5188                6.04 MB
SearchApp.exe            5292              126.45 MB
RuntimeBroker.exe        5416               17.07 MB
  x32dbg.exe             7068   0.39        54.24 MB
    db601f84fd...        7868                6.45 MB
      gennt.exe          7236                2.68 MB
RuntimeBroker.exe        6012                3.38 MB
ShellExperienceH...      3512               12.42 MB
RuntimeBroker.exe        4552                2.49 MB
TextInputHost.exe        4980   0.02         8.66 MB
dllhost.exe              4896                3.62 MB
UserOOBEBroker....        424                1.79 MB
ScreenClippingH...       1684   1.82        19.35 MB
RuntimeBroker.exe        5112                3.17 MB
svchost.exe               984   0.04         6.31 MB
svchost.exe               408                1.97 MB
svchost.exe               888                3.22 MB
svchost.exe               732                1.88 MB
svchost.exe              1156               13.57 MB
svchost.exe              1164                1.72 MB
svchost.exe              1248                3.08 MB
svchost.exe              1288                1.68 MB
VBoxService.exe          1344                2.74 MB
svchost.exe              1396                3.48 MB
svchost.exe              1412                 3.2 MB

CPU Usage: 6.79%   Physical memory: 2.14 GB (26.75%)   Processes: 118
```

```
$-10    00000000
$-C     00000000
$-8     00000000
$-4     0019FC34
$ ==>   0019FC78
$+4     02380000  L"C:\\ProgramData\\c3fa97d78e
$+8     00008000
$+C     02390000  L"C:\\ProgramData\\c3fa97d78e
$+10    00000044
$+14    00000000
$+18    00000000
$+1C    00000000
$+20    00000000
```

Figure 8. Child Process Created.

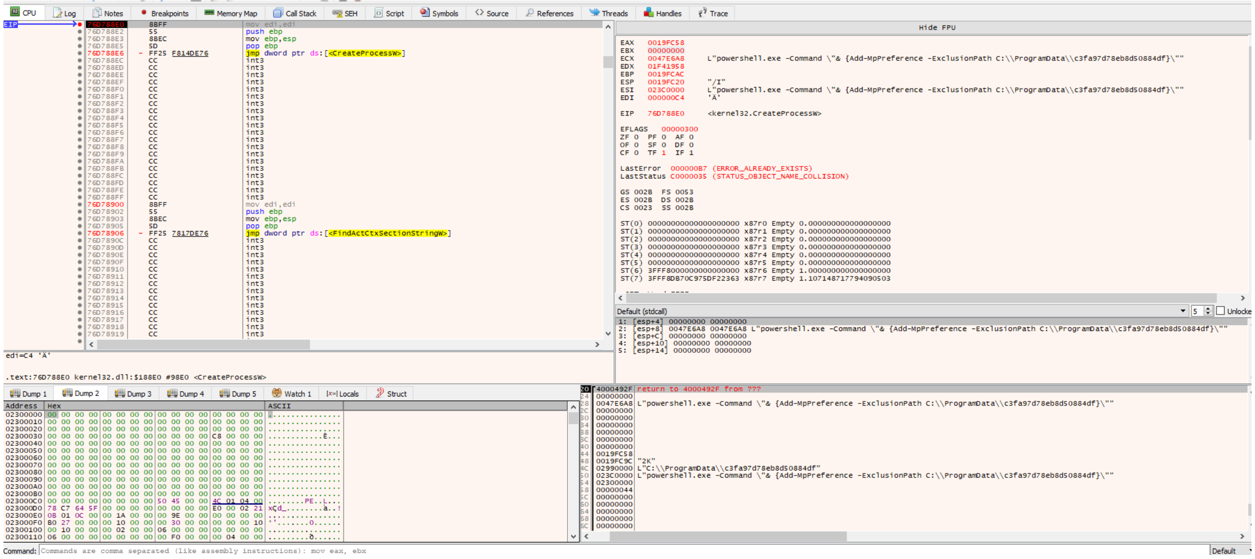
Following VirtualAlloc and VirtualProtect breakpoints on the newly created process shows it allocating memory on a RW memory address as of now .



```
02423000  0000D000  User    Reserved (02420000)                PRV           -RW--
02430000  00338000  User    \Device\HarddiskVolume2\Windows\C  MAP  -R---    -R---
02770000  00017000  User                                       PRV  -RW--    -RW--
02787000  000E9000  User    Reserved (02770000)                PRV           -RW--
40000000  0000C000  User                                       PRV  -RW--    -RW--
707A0000  00001000  System  userenv.dll                        IMG  -R---    ERWC-
707A1000  0001A000  System   ".text"                           IMG  ER---    ERWC-
707BB000  00001000  System   ".data"                           IMG  -RW--    ERWC-
```

Figure 9. Memory Regions for the Child Process.

The child process then hits a CreateProcessW breakpoint but this time the target process is now Powershell and puts in a command to exclude the folder from Defender's scanning.



```
EIP → 76D788E0   8BFF          mov edi,edi
      76D788E2   55            push ebp
      76D788E3   8BEC          mov ebp,esp
      76D788E5   5D            pop ebp
      76D788E6 - FF25 F814DE76 jmp dword ptr ds:[<CreateProcessW>]
      76D788EC   CC            int3
      76D788ED   CC            int3
      76D788EE   CC            int3
      76D788EF   CC            int3
      76D788F0   CC            int3
      76D788F1   CC            int3
      76D788F2   CC            int3
      76D788F3   CC            int3
      76D788F4   CC            int3
      76D788F5   CC            int3
      76D788F6   CC            int3
      76D788F7   CC            int3
      76D788F8   CC            int3
      76D788F9   CC            int3
      76D788FA   CC            int3
      76D788FB   CC            int3
      76D788FC   CC            int3
      76D788FD   CC            int3
      76D788FE   CC            int3
      76D788FF   CC            int3
      76D78900   8BFF          mov edi,edi
      76D78902   55            push ebp
      76D78903   8BEC          mov ebp,esp
      76D78905   5D            pop ebp
      76D78906 - FF25 7817DE76 jmp dword ptr ds:[<FindActCtxSectionStringW>]
      76D7890C   CC            int3
      76D7890D   CC            int3
      76D7890E   CC            int3
      76D7890F   CC            int3
      76D78910   CC            int3
      76D78911   CC            int3
      76D78912   CC            int3
      76D78913   CC            int3
      76D78914   CC            int3
      76D78915   CC            int3
      76D78916   CC            int3
      76D78917   CC            int3
      76D78918   CC            int3
      76D78919   CC            int3

edi=C4 'Ä'

.text:76D788E0 kernel32.dll:$188E0 #98E0 <CreateProcessW>
```

```
EAX   0019FC58
EBX   00000000
ECX   0047E6A8     L"powershell.exe -Command \"& {Add-MpPreference -ExclusionPath C:\\ProgramData\\c3fa97d78eb8d50884df}\""
EDX   01F41958
EBP   0019FCAC
ESP   0019FC20     "/I"
ESI   023C0000     L"powershell.exe -Command \"& {Add-MpPreference -ExclusionPath C:\\ProgramData\\c3fa97d78eb8d50884df}\""
EDI   000000C4     'Ä'

EIP   76D788E0     <kernel32.CreateProcessW>

EFLAGS   00000300
ZF 0  PF 0  AF 0
OF 0  SF 0  DF 0
CF 0  TF 1  IF 1

LastError  000000B7 (ERROR_ALREADY_EXISTS)
LastStatus C0000035 (STATUS_OBJECT_NAME_COLLISION)

GS 002B  FS 0053
ES 002B  DS 002B
CS 0023  SS 002B

ST(0) 0000000000000000000 x87r0 Empty 0.000000000000000000
ST(1) 0000000000000000000 x87r1 Empty 0.000000000000000000
ST(2) 0000000000000000000 x87r2 Empty 0.000000000000000000
ST(3) 0000000000000000000 x87r3 Empty 0.000000000000000000
ST(4) 0000000000000000000 x87r4 Empty 0.000000000000000000
ST(5) 0000000000000000000 x87r5 Empty 0.000000000000000000
ST(6) 3FFF8000000000000000 x87r6 Empty 1.000000000000000000
ST(7) 3FFF8DB70C975DF22363 x87r7 Empty 1.107148717794090503
```

```
1: [esp+4] 00000000 00000000
2: [esp+8] 0047E6A8 0047E6A8 L"powershell.exe -Command \"& {Add-MpPreference -ExclusionPath C:\\ProgramData\\c3fa97d78eb8d50884df}\""
3: [esp+C] 00000000 00000000
4: [esp+10] 00000000 00000000
5: [esp+14] 00000000 00000000
```

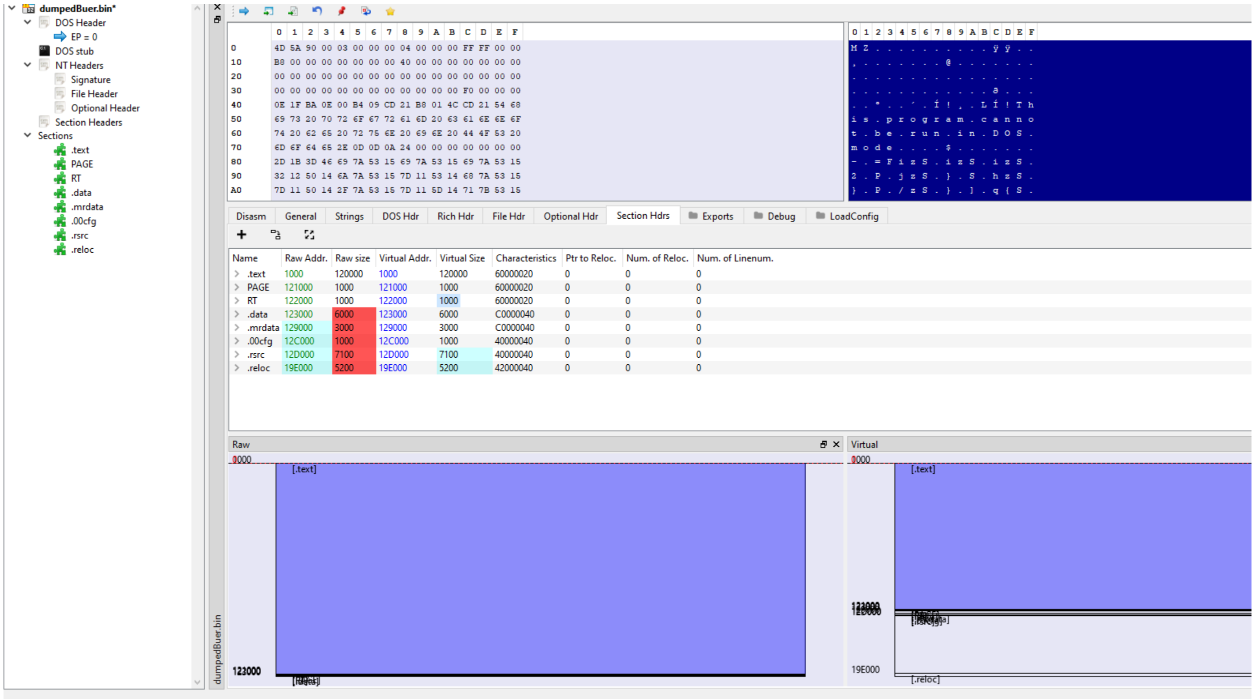
```
Address  Hex                                              ASCII
02300000 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
02300010 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
02300020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
02300030 00 00 00 00 00 00 00 00 00 00 00 00 C8 00 00 00 ............È...
02300040 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
02300050 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
02300060 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
02300070 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
02300080 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
02300090 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
023000A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
023000B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
023000C0 00 00 00 00 00 00 00 00 50 45 00 00 4C 01 04 00 ........PE..L...
023000D0 78 C7 64 5F 00 00 00 00 00 00 00 00 E0 00 02 21 xÇd_........à..!
023000E0 0B 01 0C 00 00 1A 00 00 00 9E 00 00 00 00 00 00 ................
023000F0 B0 27 00 00 00 10 00 00 00 30 00 00 00 00 00 10 °'.......0......
02300100 00 10 00 00 00 02 00 00 06 00 00 00 00 00 00 00 ................
02300110 06 00 00 00 00 00 00 00 00 F0 00 00 00 04 00 00 .........ð......
```

```
4000492F return to 4000492F from ???
00000000
0047E6A8 L"powershell.exe -Command \"& {Add-MpPreference -ExclusionPath C:\\ProgramData\\c3fa97d78eb8d50884df}\""
00000000
00000000
00000000
00000000
00000000
00000000
0019FC58
0019FC9C "2K"
02990000 L"C:\\ProgramData\\c3fa97d78eb8d50884df"
023C0000 L"powershell.exe -Command \"& {Add-MpPreference -ExclusionPath C:\\ProgramData\\c3fa97d78eb8d50884df}\""
02300000
00000044
00000000
00000000
00000000
00000000
00000000
```

Command: Commands are comma separated (like assembly instructions): mov eax, ebx

Figure 10. Powershell

Dumping Gennt.exe and fixing section headers with PE-Bear. This binary has a few different exports from the original file but it does not yet look like that I was successful in this attempt and that I did not dump from the right place.



```
    0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F     0 1 2 3 4 5 6 7 8 9 A B C D E F
0   4D 5A 90 00 03 00 00 00 04 00 00 00 FF FF 00 00    M Z . . . . . . . . . . ÿ ÿ . .
10  B8 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00    , . . . . . . . @ . . . . . . .
20  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00    . . . . . . . . . . . . . . . .
30  00 00 00 00 00 00 00 00 00 00 00 00 F0 00 00 00    . . . . . . . . . . . . ð . . .
40  0E 1F BA 0E 00 B4 09 CD 21 B8 01 4C CD 21 54 68    . . º . . ´ . Í ! , . L Í ! T h
50  69 73 20 70 72 6F 67 72 61 6D 20 63 61 6E 6E 6F    i s . p r o g r a m . c a n n o
60  74 20 62 65 20 72 75 6E 20 69 6E 20 44 4F 53 20    t . b e . r u n . i n . D O S .
70  6D 6F 64 65 2E 0D 0D 0A 24 00 00 00 00 00 00 00    m o d e . . . . $ . . . . . . .
80  2D 1B 3D 46 69 7A 53 15 69 7A 53 15 69 7A 53 15    - . = F i z S . i z S . i z S .
90  32 12 50 14 6A 7A 53 15 7D 11 53 14 68 7A 53 15    2 . P . j z S . } . S . h z S .
A0  7D 11 50 14 2F 7A 53 15 7D 11 5D 14 71 7B 53 15    } . P . / z S . } . ] . q { S .
```

Name	Raw Addr.	Raw size	Virtual Addr.	Virtual Size	Characteristics	Ptr to Reloc.	Num. of Reloc.	Num. of Linenum.
.text	1000	120000	1000	120000	60000020	0	0	0
PAGE	121000	1000	121000	1000	60000020	0	0	0
RT	122000	1000	122000	1000	60000020	0	0	0
.data	123000	6000	123000	6000	C0000040	0	0	0
.mrdata	129000	3000	129000	3000	C0000040	0	0	0
.00cfg	12C000	1000	12C000	1000	40000040	0	0	0
.rsrc	12D000	7100	12D000	7100	40000040	0	0	0
.reloc	19E000	5200	19E000	5200	42000040	0	0	0

Figure 11. Fixing the IAT