Unpacking Dridex malware

muha2xmad.github.io/unpacking/dridex/

January 11, 2022

Muhammad Hasan Ali

Malware Analysis learner

2 minute read

As-salamu Alaykum

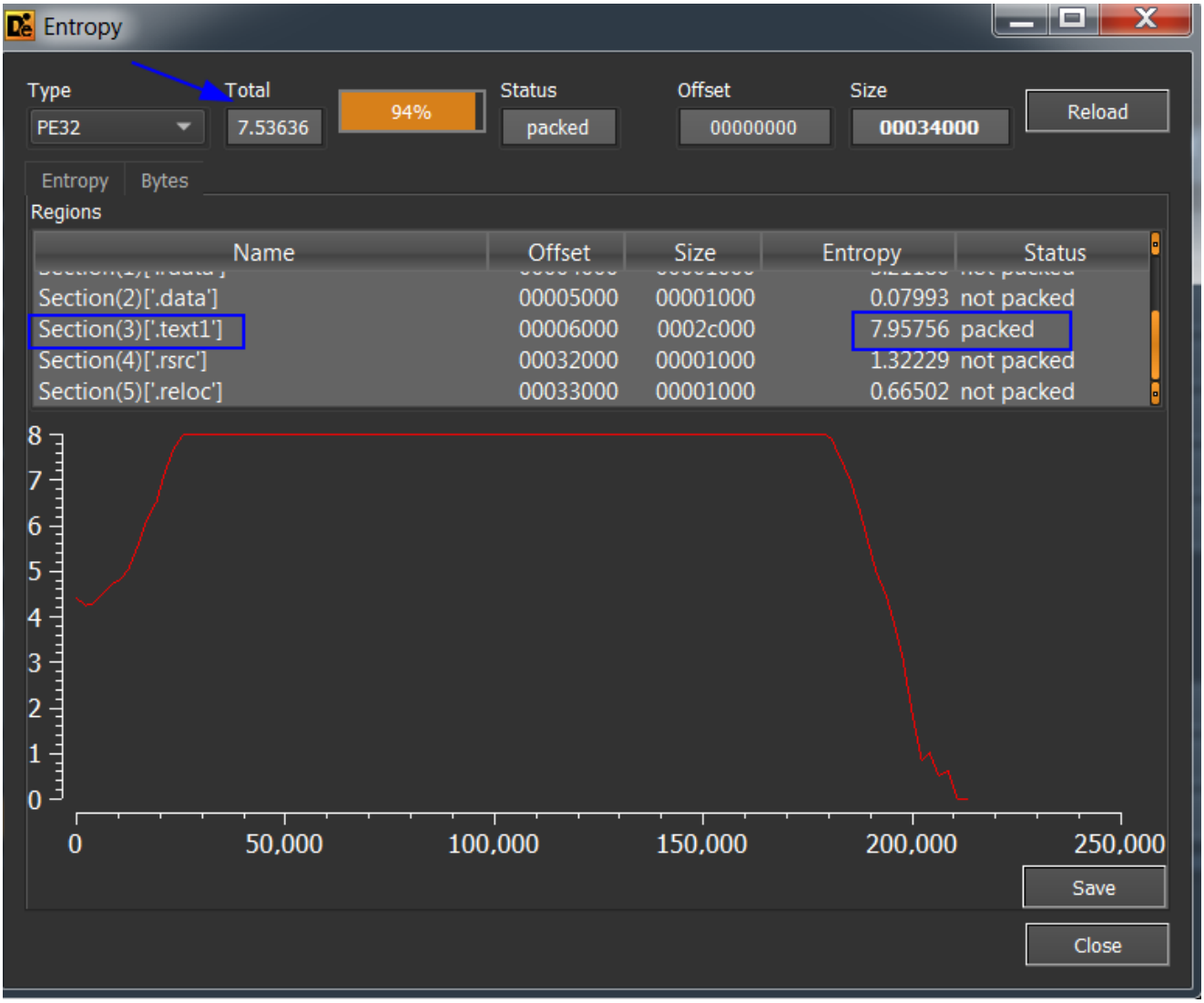
Introducton

Dridex is a famous banking Trojan which appeared around 2011 and is still very active today. This is because of its evolution and its complex architecture, which is based on proxy layers to hide the main command and control servers (C&C). The APT known as TA505 is associated to Dridex. 1

MD5: 6A8401448A5BD2B540850F811B20A66D

Static

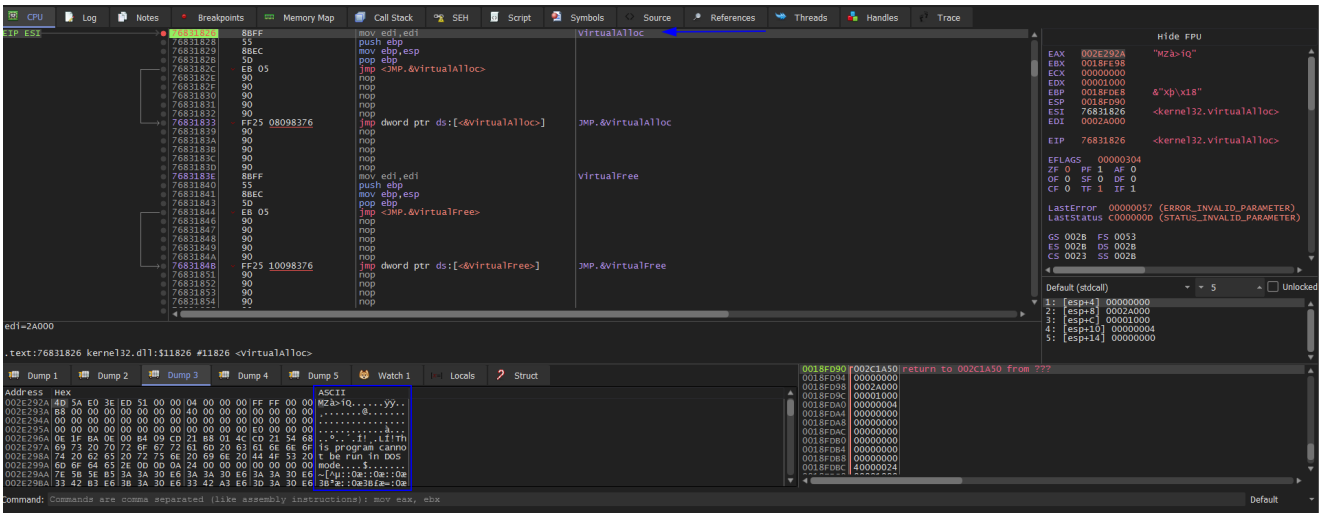
Open it in `DiE` to see `Entropy`. It's too high and if we open the sample in `IDA`. `IDA` will pop up a warning which means that the sample is packed. After it's opened in `IDA` we see that it's less number of functions and no imports.



Figure(1):

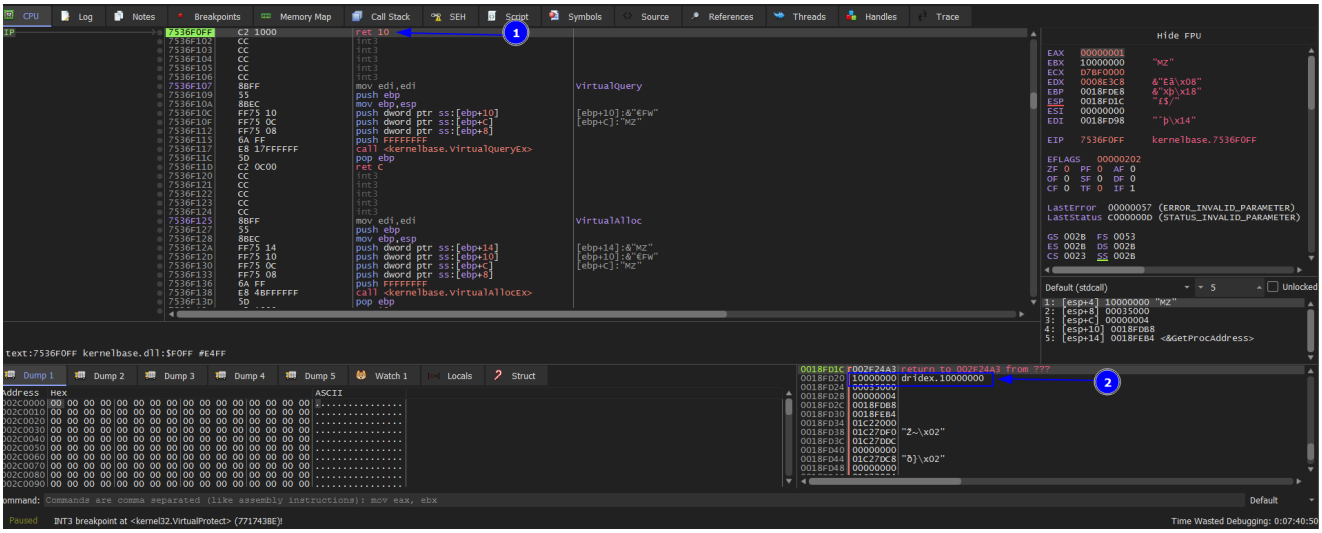
Unpacking Process

Then open the sample with `x32dbg` and set 3 breakpoints `VirtualAlloc` and `VirtualProtect` and `CreateProcessInternalW`. Then press `F9` to hit the first BP. We hit `VirtualAlloc` then `Execute till return` and dump `EAX`. We see that's empty. Then run again and `Execute till return` and dump `EAX`. Then run again for the third time and `Execute till return` and dump `EAX`. We notice that there's a PE header starts with `MZ` magic byte which we can assume that's our unpacked file. **Don't close `x32dbg` yet** `Follow in memory map` and save it then open it in `pestudio`. It shows that we didn't finish the unpacking process.



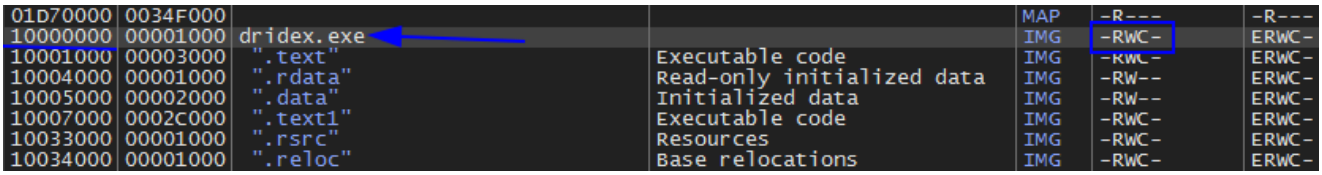
Figure(2):

We keep doing the same steps till we hit `VirtualProtect` BP. then `Execute till retrun` then we look at the second parameter which will change the permission at this memory location which is `10000000` .



Figure(3):

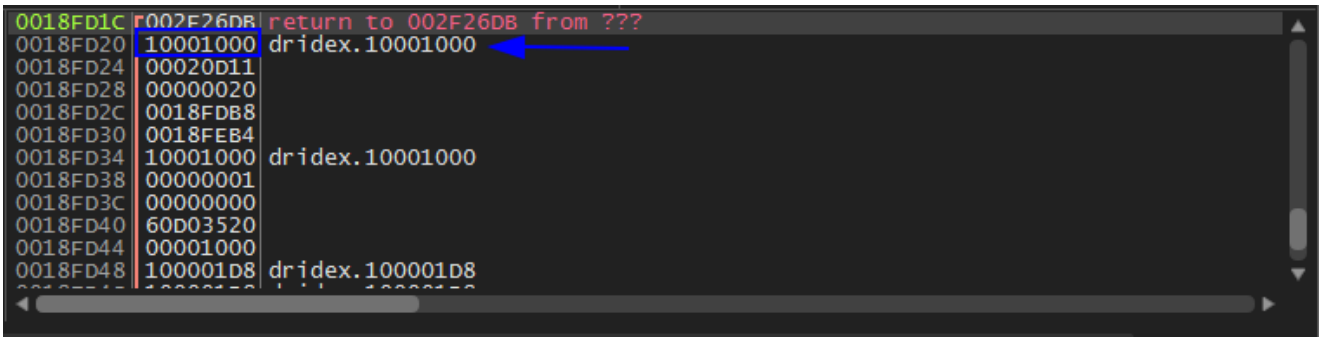
If you `follow in Memory Map` you will see `RWC` which means that it still writing which still unpacking.



01D70000	0034F000			MAP	-R---	-R---
10000000	00001000	dridex.exe		IMG	-RWC-	ERWC-
10001000	00003000	".text"	Executable code	IMG	-RWC-	ERWC-
10004000	00001000	".rdata"	Read-only initialized data	IMG	-RW--	ERWC-
10005000	00002000	".data"	Initialized data	IMG	-RW--	ERWC-
10007000	0002C000	".text1"	Executable code	IMG	-RWC-	ERWC-
10033000	00001000	".rsrc"	Resources	IMG	-RWC-	ERWC-
10034000	00001000	".reloc"	Base relocations	IMG	-RWC-	ERWC-

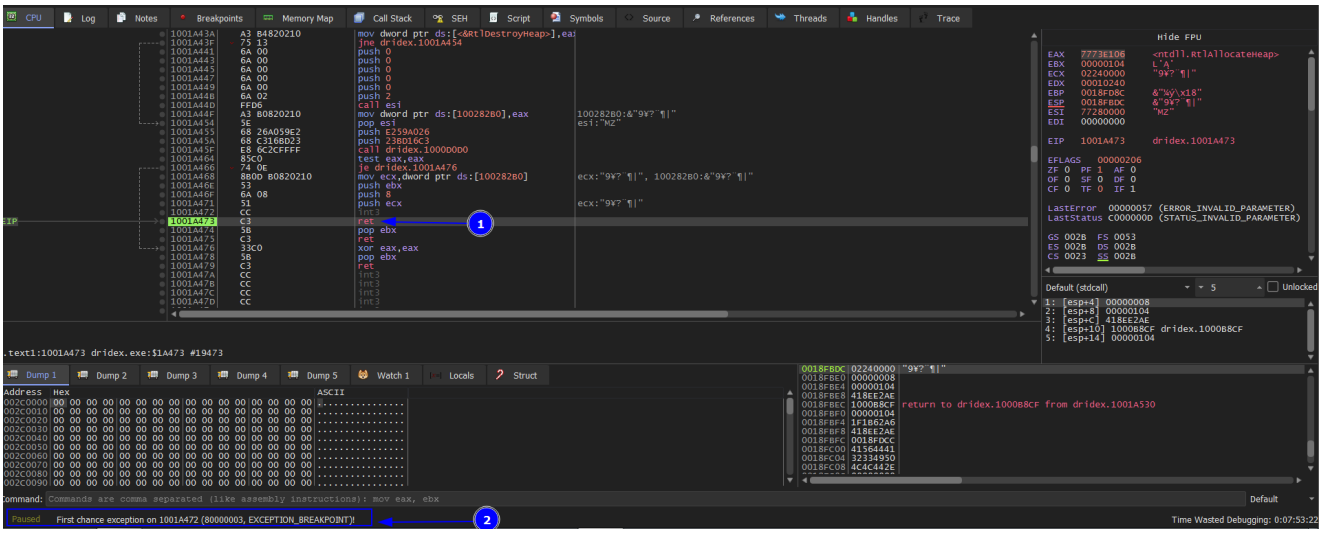
Figure(4):

Then run again which will hit `VirtualProtect` again. Then `Execute till return` and it points at the same location `10000000`, but this time shows us it's only `R` in memory map. Now we don't know that if it's done or not. So `run` again and `Execute till return` we see it points at another location but in the same region of memory of `10000000` which is `10001000`.



Figure(5):

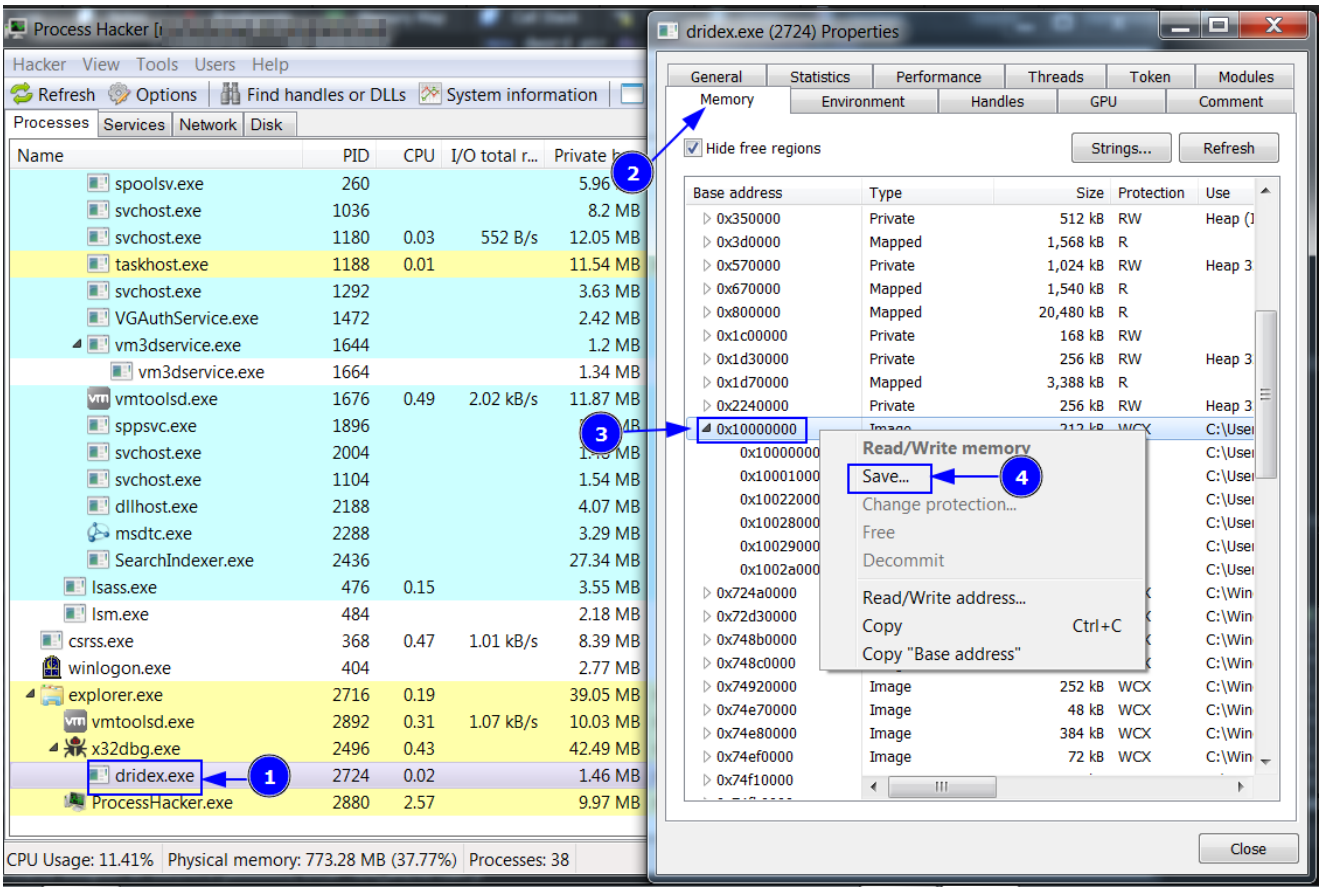
This the 3rd time to hit `VirtualProtect` BP we keep doing the same steps which will point at another locations but in the same region of memory of `10000000`. and in the 6th one we **stop**. We press `F9` to run. It shows us `EXCEPTION_BREAKPOINT` it's done unpacking. **Don't close the debugger**.



Figure(6):

Now we hit `VirtualAlloc` 3 times and `VirtualProtect` 6 times at the 6th one we finished unpacking. We will restore the snapshot and do it again till hit `VirtualProtect` at the 6th time and `Execute till return`.

Then open `Process Hacker` tool to to save the unpacked file.



Figure(7):

Unmapping

Then we repair the unapcked file to restore the imports table. As we did Here and save it. Then open it in `pestudio` if you unmapped it right, will show the imports table and you could analyze it easy.

Article quote

> النعيمُ لا يُدرك بالنعيم

REF

1- https://cyber-anubis.github.io/malware%20analysis/dridex/