

Forum:[Hack Forums](#)

Thread: [Process Injection Methods](#)

TheBrokenWasp - June 2, 2020, 7:06 pm

I know that there are a lot of different process injection methods, such as RunPE, and I would like your opinions on the different types. I know RunPE is very common and looking into it, it seems like AV programs detect it very easily, so I want to know if anybody here uses alternatives. If so, which ones? Why?

Basically I just want people to talk about process injection as it's a very interesting topic to me. Thanks.

VisualSoft - June 2, 2020, 9:19 pm

its not that all runpe are detected, if you mode a bit the same runpe then it will be fud on scantime + runtime.

TheBrokenWasp - June 2, 2020, 9:28 pm

(Jun 2, 2020 09:19 PM)VisualSoft Wrote: [\[CITE\]](#) its not that all runpe are detected, if you mode a bit the same runpe then it will be fud on scantime + runtime.

Alright, well I just based my knowledge off of [this](#) article

VisualSoft - June 2, 2020, 9:31 pm

(Jun 2, 2020 09:28 PM)TheBrokenWasp Wrote: [\[CITE\]](#) Alright, well I just based my knowledge off of [this](#) article

it looks to be an old article.

TheBrokenWasp - June 2, 2020, 9:37 pm

(Jun 2, 2020 09:31 PM)VisualSoft Wrote: [\[CITE\]](#) it looks to be an old article. probably :/

VisualSoft - June 3, 2020, 3:41 am

(Jun 2, 2020 09:37 PM)TheBrokenWasp Wrote: [\[CITE\]](#) probably :/
There are some other Private methods but they are not yet public.

Jean-Jacques Rousseau - June 9, 2020, 4:32 am

There are loads of things you can do, overwrite callbacks, use APC, map a transacted section, etc etc, to map a bootstrap + shellcode + malware. It is just that crypter developers use RunPE because they don't know any better. Either way, a crypter should not use code injection to begin with, that is the work of bot.

TheBrokenWasp - June 9, 2020, 7:09 am

(Jun 9, 2020 04:32 AM)Supercut Wrote: [\[CITE\]](#) There are loads of things you can do, overwrite callbacks, use APC, map a transacted section, etc etc, to map a bootstrap + shellcode + malware. It is just that crypter developers use RunPE because they don't know any better. Either way, a crypter should not use code injection to begin with, that is the work of bot.

Why? What's the difference?

Jean-Jacques Rousseau - June 9, 2020, 8:35 am

(Jun 9, 2020 07:09 AM)TheBrokenWasp Wrote: [\[CITE\]](#) Why? What's the difference?

Any decent bot will use its own process injection into a trusted process. Better let the bot do the work by itself, you should just map the executable and execute it.

3DES - June 10, 2020, 5:22 am

(Jun 9, 2020 08:35 AM)Supercut Wrote: [\[CITE\]](#) Any decent bot will use its own process injection into a trusted process. Better let the bot do the work by itself, you should just map the executable and execute it. More commonly known as pe loader / LoadPE.

Jean-Jacques Rousseau - June 10, 2020, 7:17 am

(Jun 10, 2020 05:22 AM)3DES Wrote: [\[CITE\]](#) More commonly known as pe loader / LoadPE. Exactly, map the PE according to its standard, and execute. No API.

TheBrokenWasp - June 10, 2020, 7:22 am

(Jun 10, 2020 07:17 AM)Supercut Wrote: [\[CITE\]](#) Exactly, map the PE according to its standard, and execute. No API.

So a quick lookup leads me to another HF post where it's explained that it executes it within its own process (I might be misunderstanding that) but if that leaves the malware running within the same process, would it not be a bit ineffective against AV as they would scan the process as an unknown process anyways, unlike where RunPE hides the malware in a remote process which is trusted and won't be scanned?

Jean-Jacques Rousseau - June 10, 2020, 7:39 am

(Jun 10, 2020 07:22 AM)TheBrokenWasp Wrote: [\[CITE\]](#) So a quick lookup leads me to another HF post where it's explained that it executes it within its own process (I might be misunderstanding that) but if that leaves the malware running within the same process, would it not be a bit ineffective against AV as they would scan the process as an unknown process anyways, unlike where RunPE hides the malware in a remote process which is trusted and won't be scanned?

Depends on what type of malware. Malware with bad footprint will be better off injected, more sophisticated malware should be mapped in current process. When AVs conduct memory scanning, they scan commonly injected processes as well. Also remember that some RunPEs inject to itself, which is completely useless.

TheBrokenWasp - June 10, 2020, 7:40 am

(Jun 10, 2020 07:39 AM)Supercut Wrote: [\[CITE\]](#) Depends on what type of malware. Malware with bad footprint will be better off injected, more sophisticated malware should be mapped in current process. When AVs conduct memory scanning, they scan commonly injected processes as well. Also remember that some RunPEs inject to itself, which is completely useless.

What if you look for a bunch of user-downloaded software, like chrome, discord, firefox, steam, etc, which are likely to be running most of the time while the user is on their PC and inject into those?

Jean-Jacques Rousseau - June 10, 2020, 7:43 am

(Jun 10, 2020 07:40 AM)TheBrokenWasp Wrote: [\[CITE\]](#) What if you look for a bunch of user-downloaded software, like chrome, discord, firefox, steam, etc, which are likely to be running most of the time while the user is on their PC and inject into those?

I don't have that knowledge on top of my head, might do tests later. But that defeats the purpose, to hide the code in a trusted process. But i know that AVs will scan other processes.

TheBrokenWasp - June 10, 2020, 7:48 am

(Jun 10, 2020 07:43 AM)Supercut Wrote: [\[CITE\]](#) I don't have that knowledge on top of my head, might do tests later. But that defeats the purpose, to hide the code in a trusted process. But i know that AVs will scan other processes.

How so? These are signed processes that are trusted all around the world by everyone, right?

Jean-Jacques Rousseau - *June 10, 2020, 8:23 am*

(Jun 10, 2020 07:48 AM)TheBrokenWasp Wrote: [\[CITE\]](#) How so? These are signed processes that are trusted all around the world by everyone, right?

Yes, however, if I am not wrong (which I might be), some processes are treated differently in terms of ring3 hooks and kernel callbacks. I haven't done tests to support this though, it is just what I believe.