

Forum:[Hack Forums](#)

Thread: [Prevent DLL injection into my process?](#)

Profesorel - November 3, 2018, 11:20 am

Hey i would like to know what are the best methods to prevent dll from getting injected into my process?

debug - November 3, 2018, 12:38 pm

Probably the easiest would be to loop through the process modules and and detect any foreign ones. However, this would be pretty easy to defeat. The other method would be to debug your own process in which you could get notifications of any new loaded dlls.

xHeaven - November 3, 2018, 12:57 pm

This might help a little. <https://stackoverflow.com/questions/869320/how-do-i-prevent-dll-injection>

You won't be able to fully stop DLL injection.
May I ask what are you trying to make?

Profesorel - November 3, 2018, 1:24 pm

(Nov 3, 2018 12:57 PM)xHeaven Wrote: [\[CITE\]](#) This might help a little.

You won't be able to fully stop DLL injection.

May I ask what are you trying to make?

Well i am coding cheats for PUBG and it seems that my app got "leaked".

I have protected it with themida but it's not enough.

xHeaven - November 3, 2018, 2:28 pm

(Nov 3, 2018 01:24 PM)Profesorel Wrote: [\[CITE\]](#) Well i am coding cheats for PUBG and it seems that my app got "leaked".

I have protected it with themida but it's not enough.

Themida won't do shit with .NET applications.

You need a properly coded launcher, as serversided as possible, and a proper obfuscation. After that, maybe crypting everything, obfuscating the crypted output, because why not, then protecting the whole thing with VMProtect.

What do you even want to achieve with blocking DLL injection? I mean, if you are using a loader and some internal cheat, just make everything serversided, even in the DLL, and they can't do anything with it, even if they dump the DLL itself (which should be protected with VMProtect or something like that of course). If it's an external, then there is no point of doing this.

Profesorel - November 4, 2018, 3:10 am

(Nov 3, 2018 02:28 PM)xHeaven Wrote: [\[CITE\]](#) Themida won't do shit with .NET applications.

You need a properly coded launcher, as serversided as possible, and a proper obfuscation. After that, maybe crypting everything, obfuscating the crypted output, because why not, then protecting the whole thing with VMProtect.

What do you even want to achieve with blocking DLL injection? I mean, if you are using a loader and some internal cheat, just make everything serversided, even in the DLL, and they can't do anything with it, even if they dump the DLL itself (which should be protected with VMProtect or something like that of course). If it's an external, then there is no point of doing this.

it is an external cheat. I edit the files of the game, then it install cheats.

Profesorel - November 5, 2018, 12:34 pm

I just bought VMProtect, which other software do you recommend to me for obfuscating before VMProtect ?

The problem is that my themida protected software is getting leaked everyday. The dude who do that can literally change everything in the software LOL

Nobelium Uranium - November 5, 2018, 1:27 pm

(Nov 5, 2018 01:34 PM)Profesorel Wrote: [\[CITE\]](#) I just bought VMProtect, which other software do you recommend to me for obfuscating before VMProtect ?

The problem is that my themida protected software is getting leaked everyday. The dude who do that can literally change everything in the software LOL

It's .NET, anyone can do that. .NET languages (C# and VB.NET) use JIT compilers meaning the code technically isn't compiled until it is ran, most other languages on the other hand are compiled when they are built, like C++.

0xBB - November 6, 2018, 2:06 am

(Nov 5, 2018 02:27 PM)\$supreme Wrote: [\[CITE\]](#) It's .NET, anyone can do that. .NET languages (C# and VB.NET) use JIT compilers meaning the code technically isn't compiled until it is ran, most other languages on the other hand are compiled when they are built, like C++.

Id argue .net languages are compiled. JIT != interpreted, also @op you may slow a good reverse engineer, but you will never stop one. You may change your business modal into users depending on your update and implementing server/client system to load your sensitive data in memory thus making the task of cracking every update requiring a valid "access" to your system.

3DES - November 6, 2018, 3:40 am

Hook LdrLoadDll in your process and NtOpenProcess in other process if you want to be sure.

Nobelium Uranium - November 6, 2018, 5:55 am

(Nov 6, 2018 03:06 AM)ibennz Wrote: [\[CITE\]](#) Id argue .net languages are compiled. JIT != interpreted Well yes, but I was saying that way so it was easier to understand. It's true they are somewhat "compiled" but again, due to the nature of JIT and the way .NET is compiled in general is what makes it fairly easy to reverse even with the best efforts to stop it from being reversed.

At the end of day using a language like C/C++ would give you the best chance of not getting reversed if you used it right. Another thing worth noting is that unless you're dealing with some sort of hardware, there's nothing 100% uncrackable, it really all comes down to how much time and effort is the person willing to put into your software. An example of this would be companies like Microsoft and Adobe who pay millions to protect their products from piracy but yet we can find cracked versions of Microsoft Office and Windows online as well as Adobe Photoshop and many other things in the Adobe Suite.