

Forum:[Hack Forums](#)

Thread: [process memory protection](#)

xAmka1488 - March 2, 2019, 8:22 pm

Hello. Please tell me how you can protect the memory of your process from writing and reading? I would be very grateful, thanks in advance!

StarZ - March 2, 2019, 11:50 pm

<https://www.pinvoke.net/default.aspx/kernel32.virtualprotectex>

I believe this could help you.

0xDEAD10CC - March 3, 2019, 12:22 am

(Mar 3, 2019 12:50 AM)StarZ Wrote: [\[CITE\]](#)

<https://www.pinvoke.net/default.aspx/kernel32.virtualprotectex>

I believe this could help you.

That's not really going to help here; that function is the reason why.

The question is also not descriptive enough - protect memory from read/writes from the perspective of what? The process itself, or from an observer process?

You can't really protect yourself from external read/writes. You'd need to write a complicated driver to do something remotely like that.

xAmka1488 - March 3, 2019, 1:11 am

(Mar 3, 2019 01:22 AM)0xDEAD10CC Wrote: [\[CITE\]](#) That's not really going to help here; that function is the reason why.

The question is also not descriptive enough - protect memory from read/writes from the perspective of what? The process itself, or from an observer process?

You can't really protect yourself from external read/writes. You'd need to write a complicated driver to do something remotely like that.

the process itself

(Mar 3, 2019 01:22 AM)0xDEAD10CC Wrote: [\[CITE\]](#) That's not really going to help here; that function is the reason why.

The question is also not descriptive enough - protect memory from read/writes from the perspective of what? The process itself, or from an observer process?

You can't really protect yourself from external read/writes. You'd need to write a complicated driver to do something remotely like that.

I want to create a secure program in which will be anti dll inject, anti dump, anti Write Write ProcessMemory / ReadProcessMemory, anti tamper. And so on

Vexallus - March 3, 2019, 9:51 am

you can't just do this with .NET, you need top level accessibility in order to do such thing , aka a Driver, just like those games running xigncode3 , apex shield ,Battle-eye etc...

my advise for you is to have a look for a decent obfuscator and include your own protection, you can pretty much make everything protected if you know what you are doing.

Also even if you use drivers to protect your memory privileges, there will be always a way, C.E itself is using a kernel driver that has a full access over memory even if its protected, so yeah...better change your ideas about that.

Antlion - April 1, 2019, 12:44 am

This task is generally unsolvable.

Private solution: write a driver that sniffs the opening functions process (XP-Srv2003) or will use a special kernel API for its protection (Vista SP1 and above) and thus will not allow to write to the process memory. This saves from ~ 90% of typical threats.

Now if the attacker is not able to get on computer administrator rights, then everything would be much easier. In this case, it would be enough to run your process under another account because they don't have access by default each other's processes.