

Forum:[Hack Forums](#)

Thread: [Polymorphic program](#)

indetectable - January 29, 2022, 2:35 pm

Hi, I just want to know how to make my program polymorphic and if just write :
Console.WriteLine(randomthing); would be enough (I "code" in c#) ?

And if not how can I change the signature of my program at every launch ?

Zettabit - January 30, 2022, 1:10 pm

(Jan 29, 2022 04:38 PM)Incog Wrote: [\[CITE\]](#)
<https://github.com/bytecode77/self-morphing-csharp-binary>

Downside being source code needs to be read somehow but it is an example of self mutation

If this is in terms of malware, why touch disk at all? Signature based detection is worthless on what it can't read

Even that source isn't true polymorphism. All that's doing is changing variable names / obfuscation. If you want true polymorphism then the entire flow of the application must change each time as well. A signature could still be made for that app that would apply to all future variances as well.

@OP, achieving something like true polymorphism is a tricky task. Not saying it can't be done, but it's likely not worth the time like Incog mentioned.

Hackers - January 31, 2022, 9:07 am

A very simple polymorphic example of what you are saying would be to have a namespace (or in other langs, a class) that declares a function - for example, something to print out text. It's generic and only takes an argument, prints it.

It becomes polymorphic when you have others like 'English', 'Spanish', etc extend the namespace with little sub functions. So like, you could have a func 'helloworld()' in your main namespace, which accepts a string argument/parameter. The English extension would have the same helloworld(), and it would send the text "Hello World" to your main namespace. The Spanish version's helloworld function would send "Hola Mundo" to it.

This is polymorphic because the main namespace / class has a general purpose that is altered by those extensions.

yvnkxx - February 4, 2022, 6:52 am

(Jan 31, 2022 10:07 AM)nodes Wrote: [\[CITE\]](#) A very simple polymorphic example of what you are saying would be to have a namespace (or in other langs, a class) that declares a function - for example, something to print out text. It's generic and only takes an argument, prints it.

It becomes polymorphic when you have others like 'English', 'Spanish', etc extend the namespace with little sub functions. So like, you could have a func 'helloworld()' in your main namespace, which accepts a string argument/parameter. The English extension would have the same helloworld(), and it would send the text "Hello World" to your main namespace. The Spanish version's helloworld function would send "Hola Mundo" to it.

This is polymorphic because the main namespace / class has a general purpose that is altered by those extensions.

I m think its just randomize,but not polymorh.