

Re: Am I the only one with doubts about .NET for commercial apps?

Source: <http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.general/2004-05/1608.html>

From: David Sworder (dsworder_at_cts.com)

Date: 05/15/04

Date: Sat, 15 May 2004 11:15:20 -0700

> *What kind of applications do YOU write???*

Financial apps... essentially every piece of analytical logic I use has been around for decades or can be copied out of an article or book.

> *I spent a few years developing signal processing/image processing applications, and I can tell you: finding out how an algorithm works isn't half as easy as you seem to think.*

Probably true... but then again, I know nothing about signal processing/image processing. Do you think that if you gathered a bunch of signal processing experts together in a room and showed them your app, they'd be SO impressed with your algorithm that they'd say to themselves "wow, I've never seen anything like that! Instead of creating our own similar algorithm, let's reverse engineer her code, recompile it into our own app, and SELL it!"

Please don't misinterpret my tone here. I'm not being sarcastic. If this actually applies to you, then hey, congratulations! ... and you're right, you'd better look for a sophisticated way to protect your algorithm.

> *I guess the same would apply to highly optimized graphics engines as they are found in computer games (if they were written in managed code)*

I'll buy that argument. For example, I remember when ID Software released "Doom." Other developers were genuinely amazed. I remember them asking "how did ID Software do that?"... but that was written in C/C++, *not* managed code.

> *proprietary communication standards*

I don't know... It seems like the hacks against com standards have been done by using a sniffer to examine network traffic, not by reverse engineering code. I suspect that the spammer that figured out how the "proprietary" Windows messenger service worked did so by examining network traffic, not by trying to decompile the messenger service EXE.

microsoft.public.dotnet.general: Re: Am I the only one with doubts about .NET for commercial apps?

> *or high-speed-databases...*

...but these aren't written in .NET managed code (but they presumably will be at some point). I assume you're talking about databases like SQL Server and Oracle? If I were writing my own high speed database, I'd turn to books, articles, and classes on database theory. I educate myself the same way the designers of SQL Server did. This would be vastly more effective than trying to decompile the .EXEs of a competitor's product.

> *virtually*

> *every piece of code that required thought when it was written.*

This is a gross exaggeration! Yesterday I wrote a helper app that waits for a file to arrive, reads it, does some analysis, and uploads the results to a database. This app "required thought," but should I worry about someone reverse engineering my code and discovering:

- a) I used a FileSystemWatcher to wait for the file
- b) I used an SqlConnection object to connect to the database and called some methods against an IDbCommand object to upload the data
- c) I used some well-known algorithms (well known in financial analyst circles) to perform the analysis that can be looked up on Google and read in **plain English** (as opposed to reading them in unobfuscated decompiled cryptic C# code.)

My app required thought, but any competitor who wanted to duplicate my efforts would be better off reading some financial analysis articles and the documentation for ADO.NET and FileSystemWatcher than he would trying to decompile my app, unobfuscate it, etc.

I'm not claiming that my situation applies to everyone. If your signal/image processing logic is truly awesome and warrants protection, so be it. If you're designing encryption logic for the dept of defense, don't use .NET. Same would be true if you're designing a revolutionary 3D game engine that is so frigging' unique that it isn't just a derivative of the techniques taught in any of the "Teach Yourself 3D Game Programming" books at Barnes and Noble... what I **am** saying is that a very large fraction of the apps out there don't fall into this category and the authors of some of these apps have decided to use C++ instead of C# simply because they're worried about someone reverse engineering their precious code -- and I think that's a shame.

David