

Forum:[Hack Forums](#)

Thread: [C++ RunPE & .NET support](#)

snowman13 - February 25, 2013, 1:03 pm

Hi guys! Im making a c++ crypter from scratch. Currently it does the basics with no prob at all, but guess what, the runPE i made doesn't support .NET apps. I guess the structure of a .NET PE is not the same with the native languages. If somebody could help me with some advices or some documantation it would be much appreciated!

tweety - February 25, 2013, 1:05 pm

The structure is not the issue, it's loading the Framework modules in which is the problem. You may want to invest some money into buying Dragon Hunters Virtual Net or you can make your own managed .NET RunPE and load the library for the RunPE if selected. If you want to purchase one instead, HMU.

snowman13 - February 25, 2013, 2:23 pm

Thanks for ur response, i guess I'll make a C# RunPE...

tweety - February 25, 2013, 2:24 pm

(Feb 25, 2013 03:23 PM)snowman13 Wrote: [\[CITE\]](#) Thanks for ur response, i guess I'll make a C# RunPE...

Make sure your not working with Win API's. You need to look into the Reflection method to load an assembly in memory. Assembly.Load() is the typical way to do this but depending on the wrapped code around it, it probably will be detected. Be creative. Good luck.

snowman13 - February 25, 2013, 2:54 pm

(Feb 25, 2013 03:24 PM)Tweety. Wrote: [\[CITE\]](#) Make sure your not working with Win API's. You need to look into the Reflection method to load an assembly in memory. Assembly.Load() is the typical way to do this but depending on the wrapped code around it, it probably will be detected. Be creative. Good luck.

Ok thanks for the tip! You helped me a lot! ?

notsteve - February 25, 2013, 3:27 pm

Take a look at a .NET stub on a standard exe and see what happens before it transfers to the .net framework? All .NET is is a small stub that sets up the .NET vm/framework and then points it towards the .net assembly entry.

mose3c - March 11, 2021, 4:23 pm

Take a look at this its should solve your problem .

<https://github.com/Mose3c/netstub>

Joseph Ducreux - March 12, 2021, 2:28 am

im sorry if this sounds vauge but i have to so some noob doesnt ruin this for everybody, but you can use live off land techniques/LOLBINS) to directly inject base64 natively in OS, in a way that works naturally with how windows was designed to work using defaultly installed and defaultly whitelisted programs instead of doing shady shit like runpe. Net dependencies will work, in fact it is interconvertable with the underlying CLR that .net is based off of. the beauty of this is you can in essence create a runtime crypter with nothing but cmd, vbscript, jscript, delphi basically any language with the ability to pipe and parse arguments and pass them over to another file now has instant

runtime crypter capability, even C++

runpe is something that microsoft developers really didnt want to be able to happen so easily. even if your program is completely authentic and non malicious just the very act triggers alot of AV, hell all next generation AI based Endpoint / NIDS systems will immediately hate it

try to work more with the system in a more natural way to accomplish your goals, ive done both (ive previously made runpe in delphi7 it was not easy and way to complicated for such a simple actual result intended)

why not do someting that is much easier, takes only 5 minutes (i can do it in 2 line batch file lol, or 1 single line in another framework windows language) instead of couple hours of coding, and will remain fud for awhile, if done correctly your .net exe is never dropped to disk and runs completely in memory, it is in essence completely scan time fud, and does not evoke any suspicious WINAPI calls that runpe definetly will.

let the whitelisted pre-installed lolbin programs windows comes with do those calls for you so your cryper doesnt have to.

second rant:

rerember to that .NET is a JIC language, if you browse to your .net framework folder in C:\windows you will eventually find several very useful exe's, naturally .net is only partially compiled and only when exe is ran does it pull more shit, if you mess with some of the compiler EXE's in that net folder, things like CSC.exe and shit you can create precompiled exes that may come with the right libraries, but thats all experimental, if i know what you are asking and having trouble go with my first rant.