Lab: Decompiling

Environment Needed:

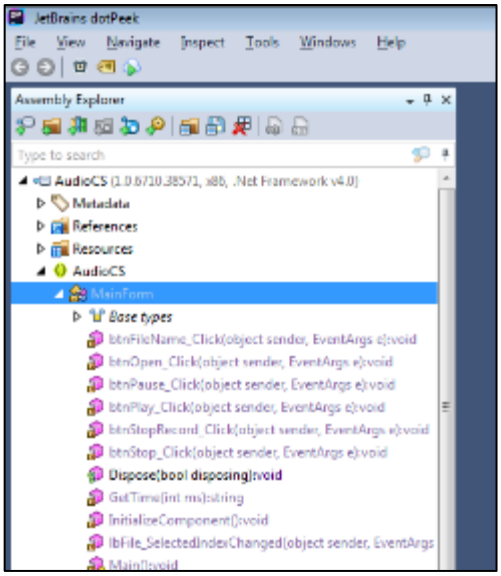
- Windows Virtual Machine
- .Net

Tool to install: JetBrains dotPeek

Links for any tools are available in the github under 'Tools'

1. Alvas Audio is a .NET audio editor and recorder tool.
2. After downloading the lab source folder, navigate to: Alvas.Audio\Alvas.Audio\bin .
3. Launch AudioCS
    a. Poke around, just see its basic capabilities.
4. Launch JetBrains dotPeek
5. We want to load AudioCS for examination.
    a. File > Open >
6. Navigate to <your lab folder>\Alvas.Audio\Alvas.Audio\bin\AudioCS.exe .
7. Once imported, you can navigate through the project; all of the code is located under AudioCS>AudioCS>Mainform.

X86-SOFTWARE-REVERSE-ENGINEERING-CRACKING-AND-COUNTER-MEASURES
ISBN: **1394199880**



8. Spend a few minutes exploring the source code. Note how all of the code – GUI handlers, application logic, everything – has been recovered by the decompiler.
9. Since our ultimate goal is *reverse engineering*, we won't explore Visual Studio in this class; however, at this point in the process, the recovered source code can be directly dropped into Visual Studio and recompiled back to the original program, or chopped up, modified, extracted, or reused in any way we see fit. What sorts of interesting pieces of code could an attacker modify, reuse, or repurpose in this program?

X86-SOFTWARE-REVERSE-ENGINEERING-CRACKING-AND-COUNTER-MEASURES
ISBN: **1394199880**