Reverse Engineering April, 2023

Lab session 0x07

In this lab session, we will analyze .NET and Android binaries.

1 Lab files

The files for this lab session are available at https://pwnthybytes.ro/unibuc_re/09-lab-files.zip and this time there is no password for the zip file (why is there no password this time?).

2 Tools we use (Windows)

In the Windows environment, you need to install the CFF Explorer[1] and dnSpy[2]. For the Android tasks you need the Android emulator[3] and Bytecode Viewer[4].

3 Lab tasks: debugging .NET and Android applications

3.1 Task: reversing .NET binaries (6p)

Perform the following tasks using the task1.exe binary:

- Investigate task1.exe however you see fit. Spend no more than 10-15 minutes trying to approach the binary as usual, with IDA Pro.
- Next, open the binary in the CFF Explorer and look for the "FileDescription" field. What value does the binary have and what does it mean?
- Open the binary in 7z or Winrar, extract the underlying executable and open it in IDA Pro. Explain why you think the extraction works. (2p)
- What type of file is recognized in this executable by IDA Pro? Notice that decompilation does not work and reading the assembly is pretty hard.
- Now use dnSpy (64 bit) to open the binary and poke around in btnDecode_Click. Find the correct output (either through static analysis or dynamic analysis, both using dnSpy). (4p)

3.2 Task: reversing Android binaries (9p)

Perform the following tasks using the task2 binary:

- What type of file is it? How can you unpack it? Run the application in an emulator. (2p)
- Use *Bytecode Viewer* to open the same file. Look under "com" through the *Activity classes*. Where is the password checked? What does *Loadlibrary* do and where is the library file?
- Open it in IDA, look in the "Java_com_flareon_flare_ValidateActivity_validate" function.
- Load *jni.h* using File/Load File/Parse C header file and retype the function as "int __fastcall Java_com_flareon_flare_ValidateActivity_validate(JNIEnv *jnienv, int a2, jstring input)"
- Reverse the function and find the correct input. (7p)

[1] https://ntcore.com/files/CFF_Explorer.zip
[2] https://github.com/dnSpy/dnSpy/releases
[3] https://developer.android.com/studio/run/emulator
[4] https://github.com/Konloch/bytecode-viewer