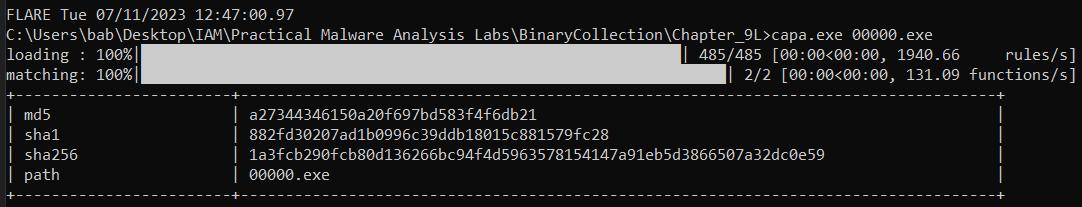
**Lab 15: Patching EXEs with Ollydbg**

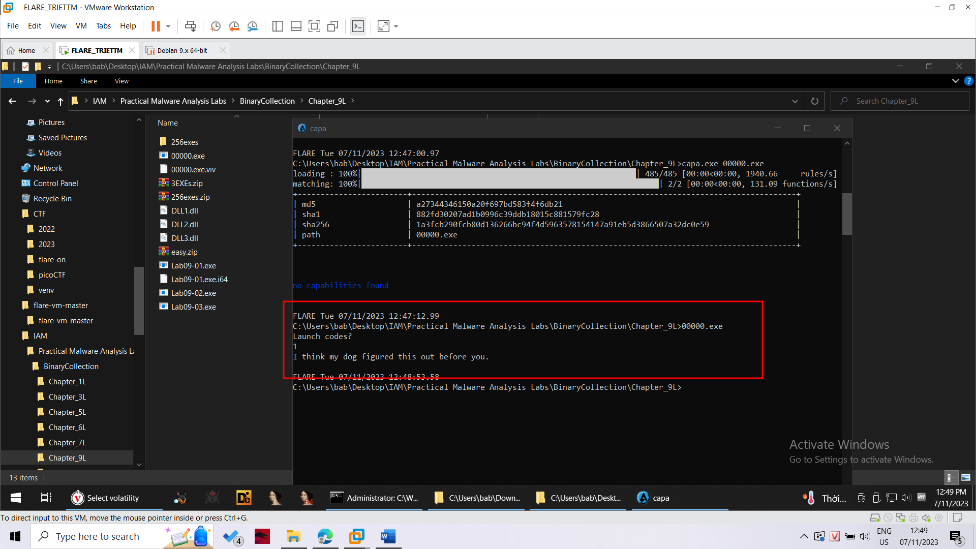
**Patching an EXE**

Check hash file exe

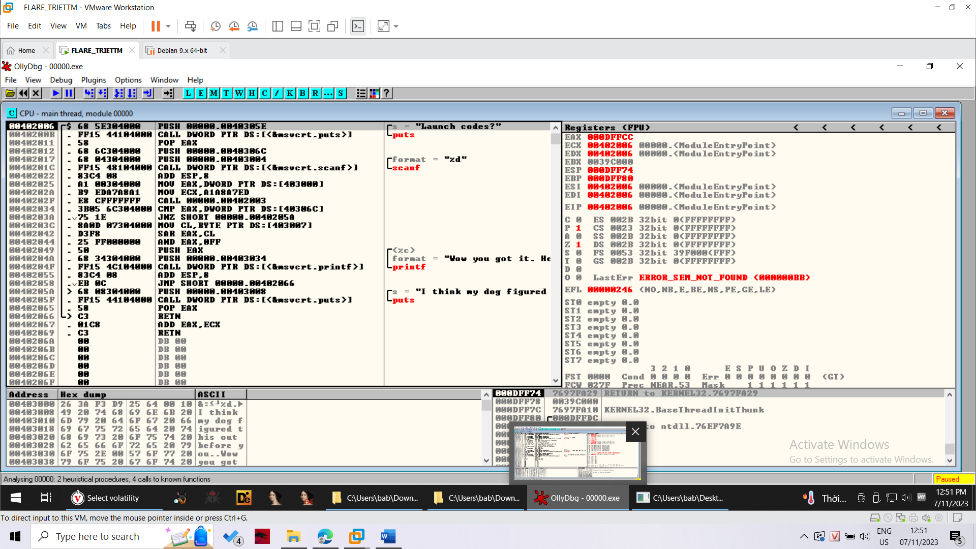


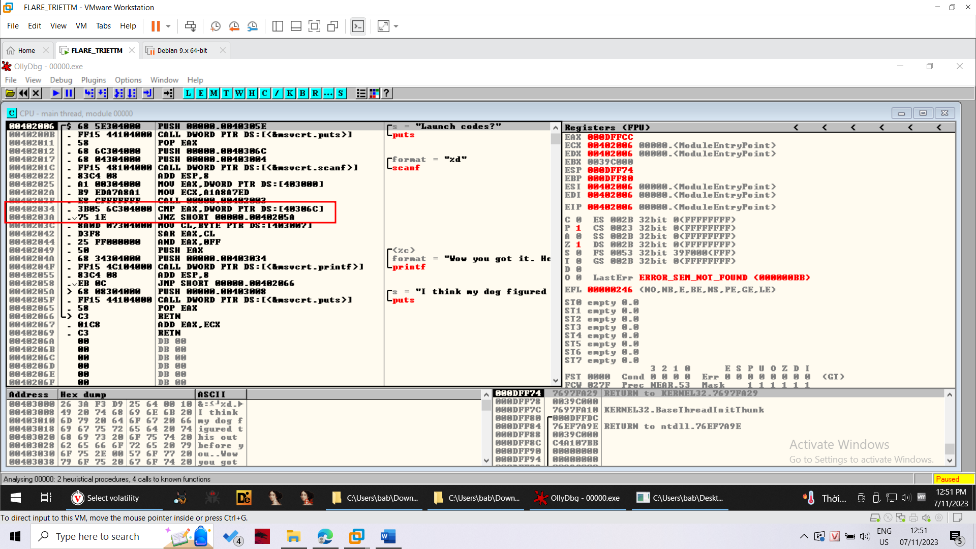
**Running the EXE**

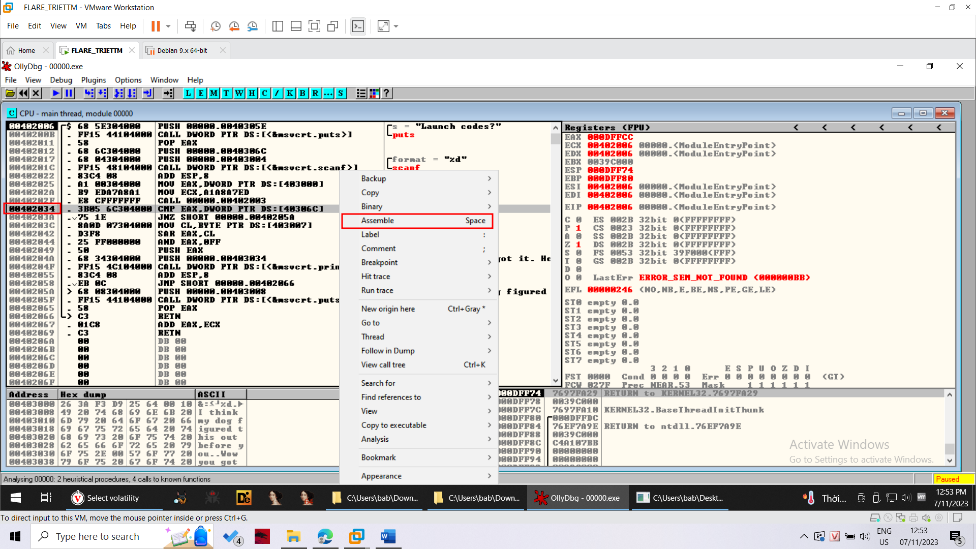
Chạy thử file exe này.

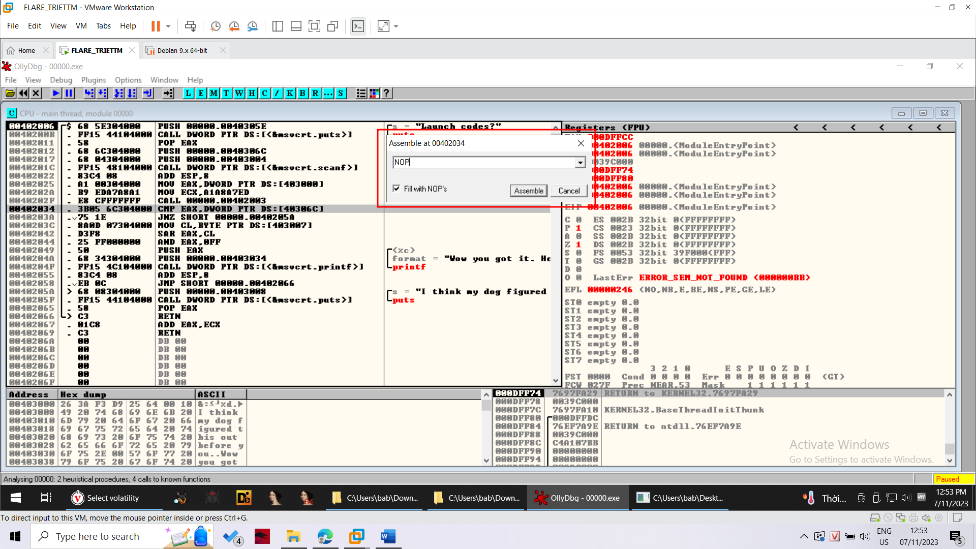


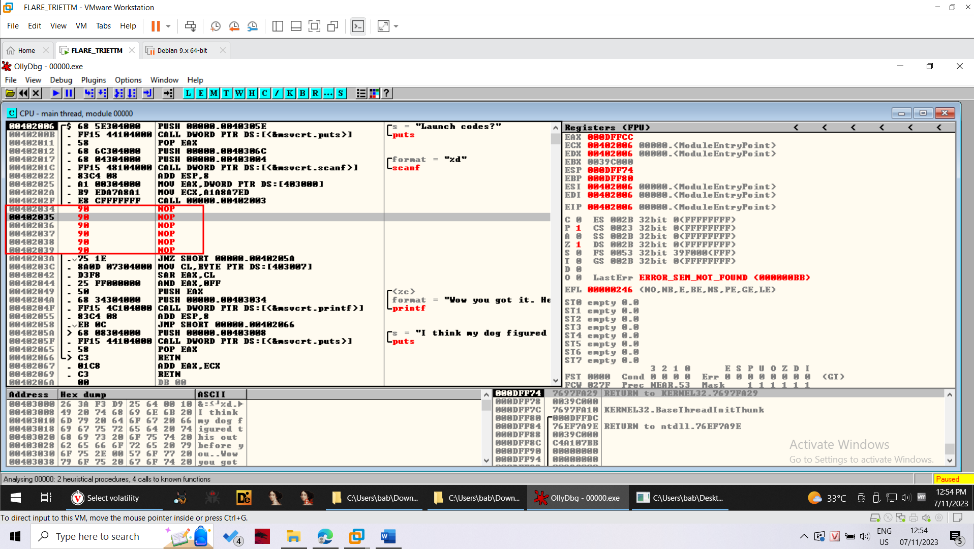
**Examining the EXE with Ollydbg**

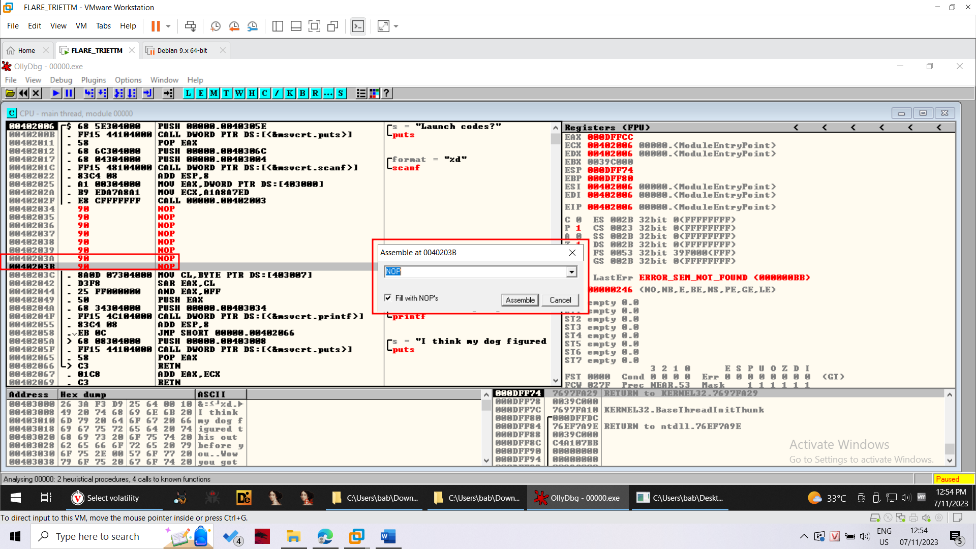


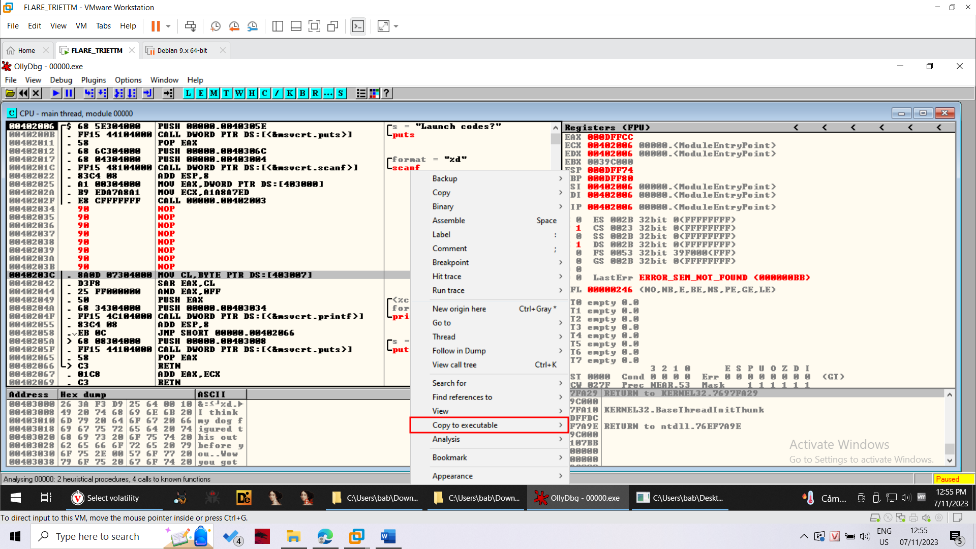


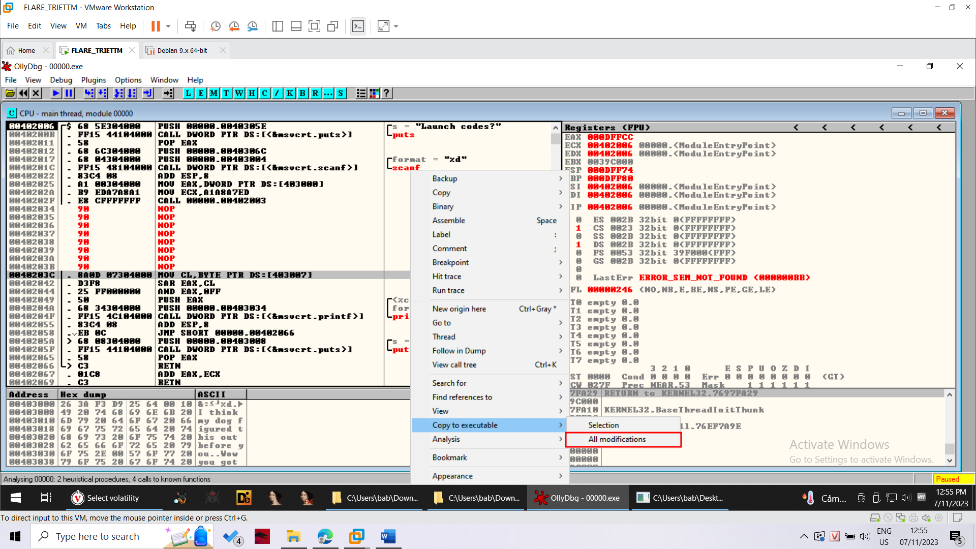


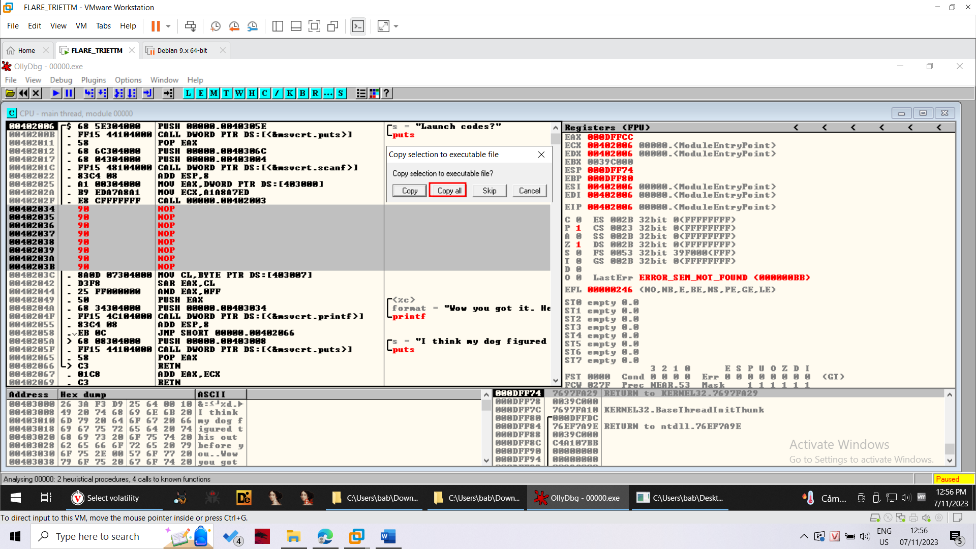


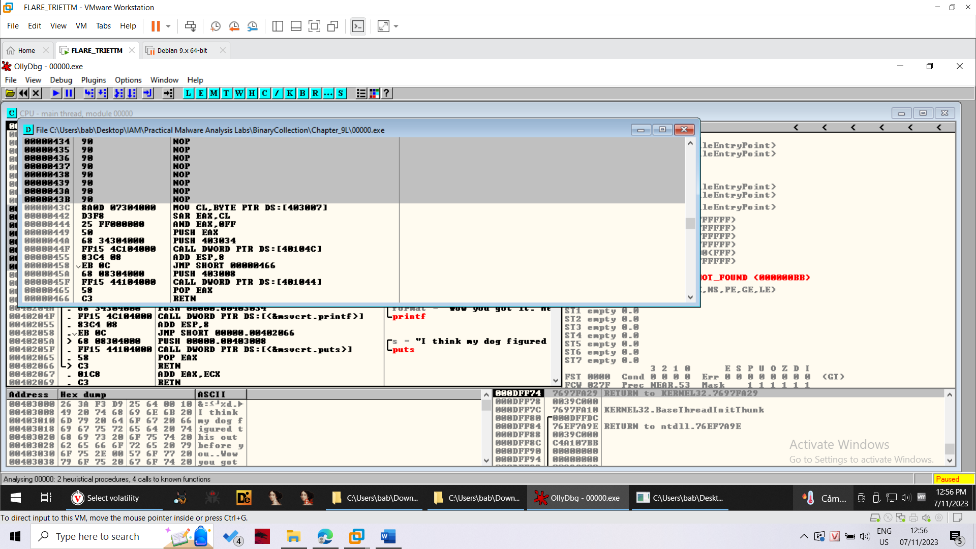


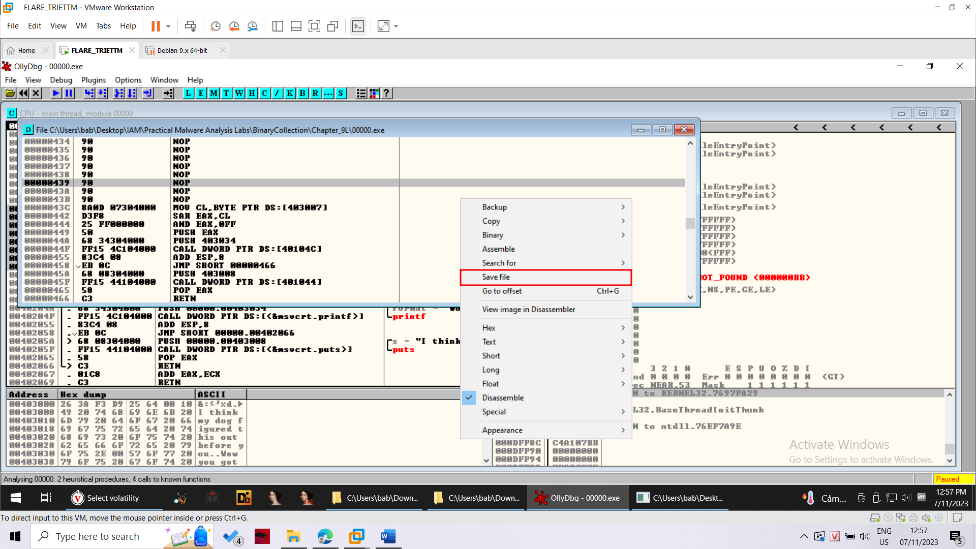


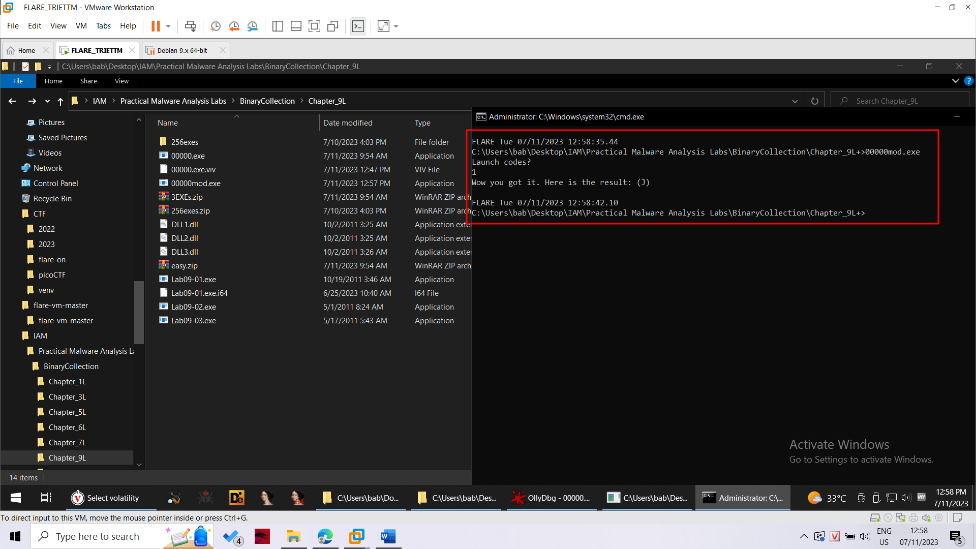




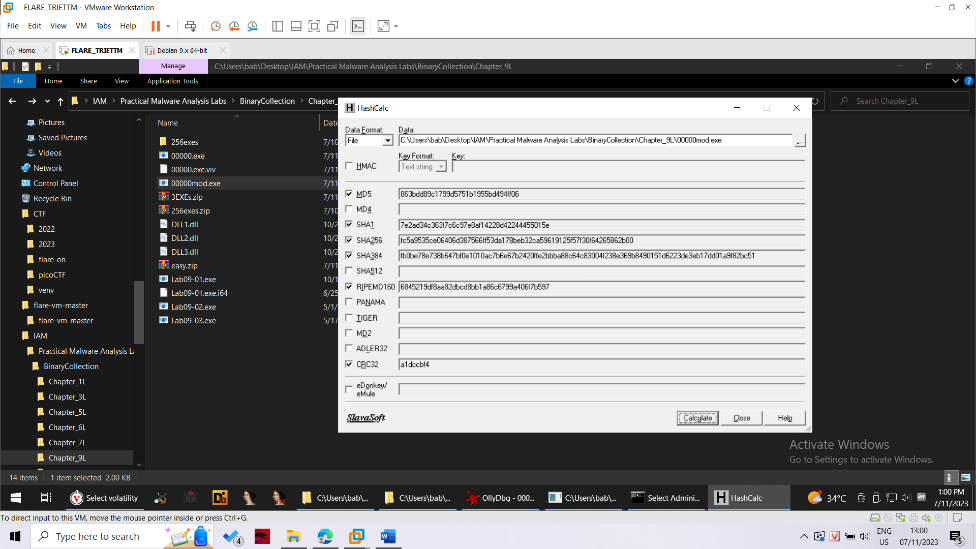




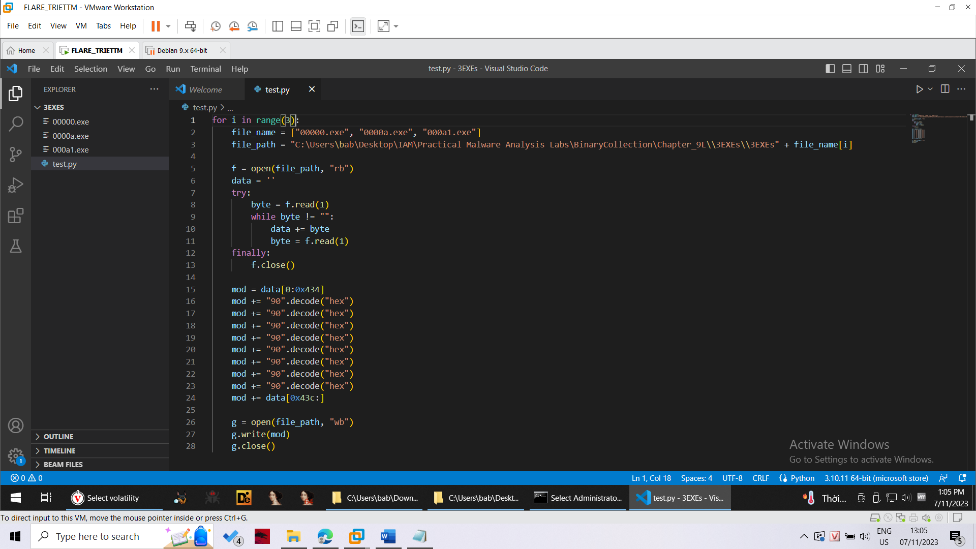


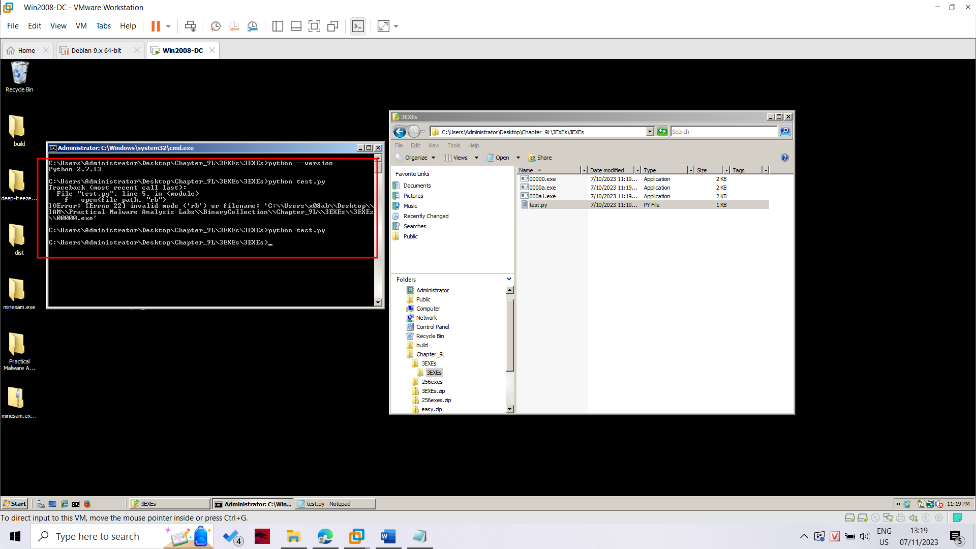


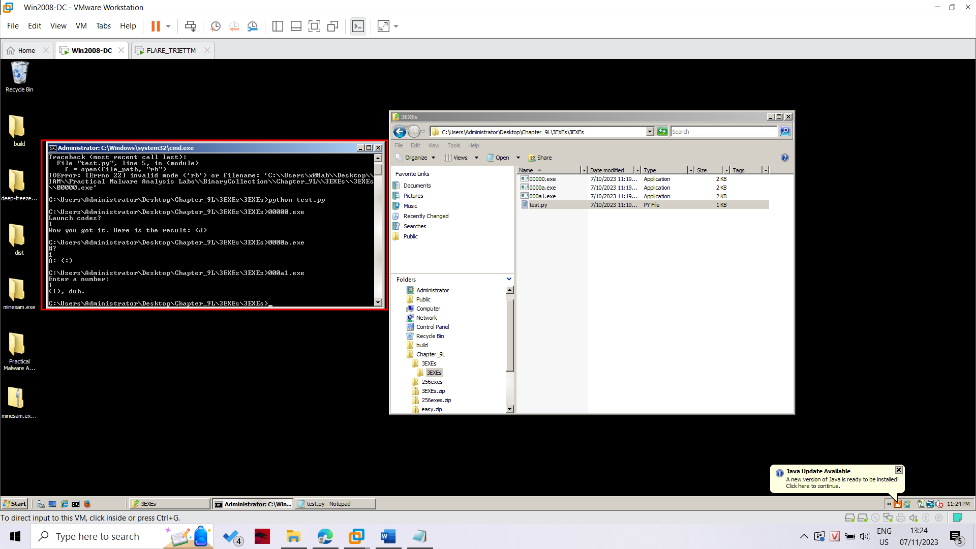
**Checking the Hash**



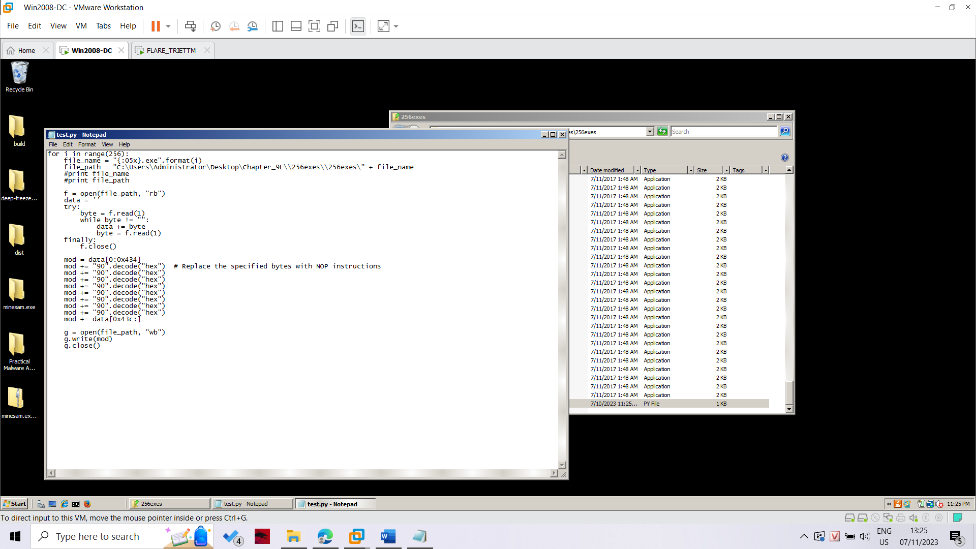
**Patching Three EXEs**

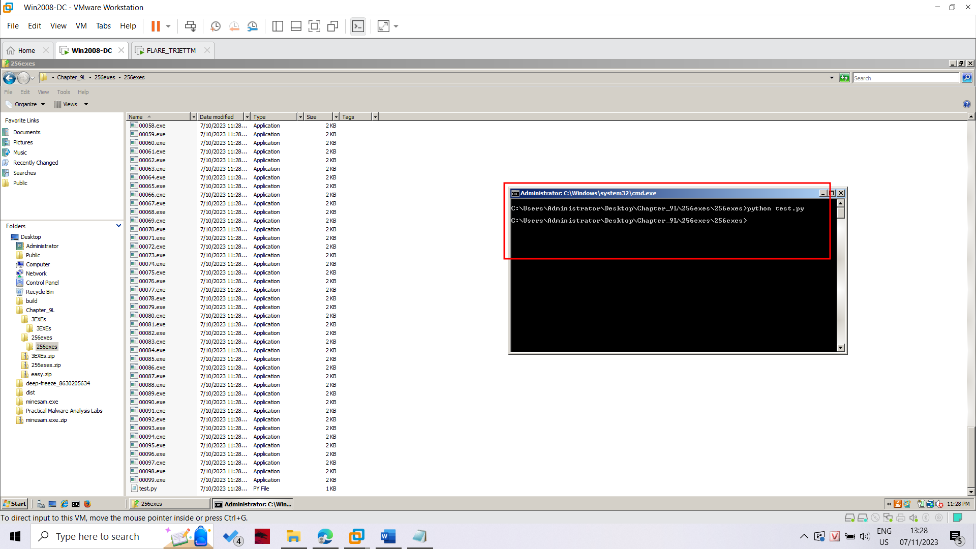






**Patching 256 EXEs**





Because 256 file have the different offset, so we have to scripting a bit difficult to patch. I use following script

import os

import pefile

from capstone import \*

from capstone.x86 import \*

dir = "C:\\Users\\bab\\Desktop\\256exes\\"

for filename in os.listdir(dir):

    print(filename)

    file = pefile.PE(filename)

    code\_section = None

    for section in file.sections:

        if section.Characteristics & pefile.SECTION\_CHARACTERISTICS["IMAGE\_SCN\_MEM\_EXECUTE"]:

            code\_section = section

            break

    CODE\_BASE = code\_section.VirtualAddress

    CODE\_SIZE = code\_section.SizeOfRawData

    code\_data = file.get\_memory\_mapped\_image()[CODE\_BASE : CODE\_BASE + CODE\_SIZE]

    md = Cs(CS\_ARCH\_X86, CS\_MODE\_32)

    for insn in md.disasm(code\_data, CODE\_BASE):

        if( insn.mnemonic == "cmp" ):

            patched\_code = b"\x90\x90\x90\x90\x90\x90\x90\x90"

            file.set\_bytes\_at\_rva(insn.address, patched\_code)

            break

    file.write("patched\_" + filename)

Using above script then all file will corectly patched.