Final Project

Prof. Alberto LaCava

CY -640

MALWARE ANALYSIS & DEFENSE

Nikhil Patel

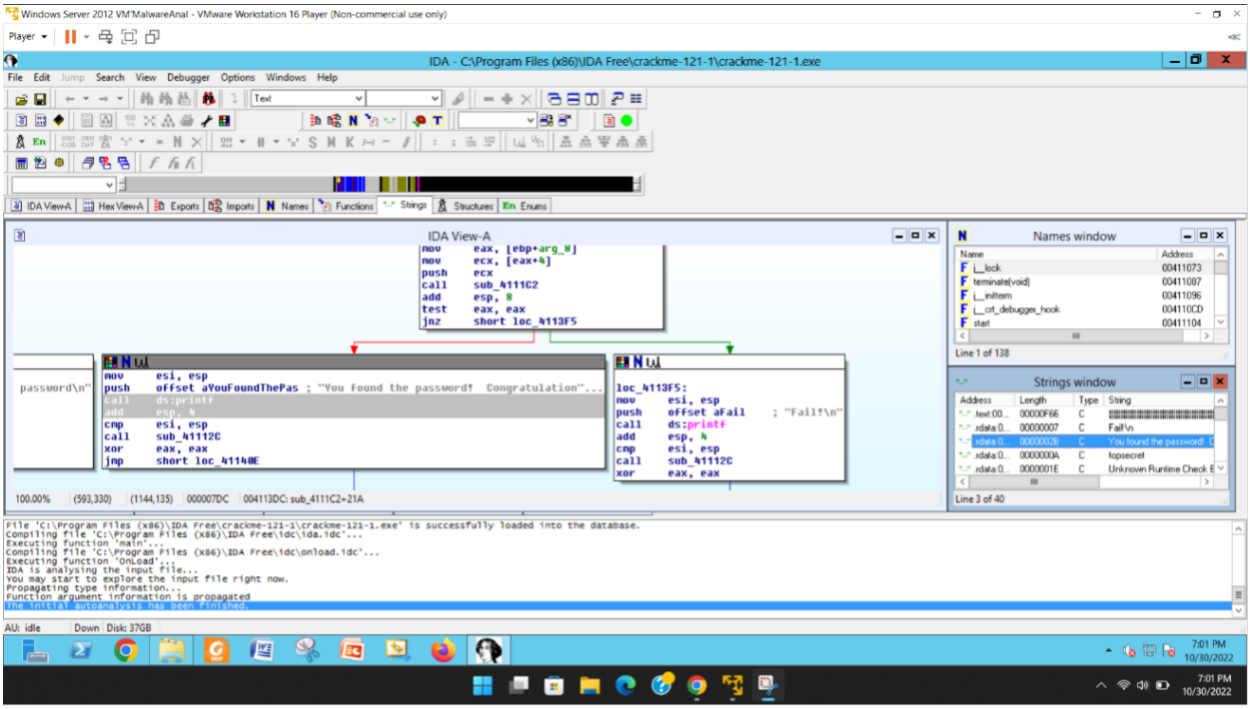
11/22/2022

Project 2: Reverse Engineering with IDA Pro Freeware

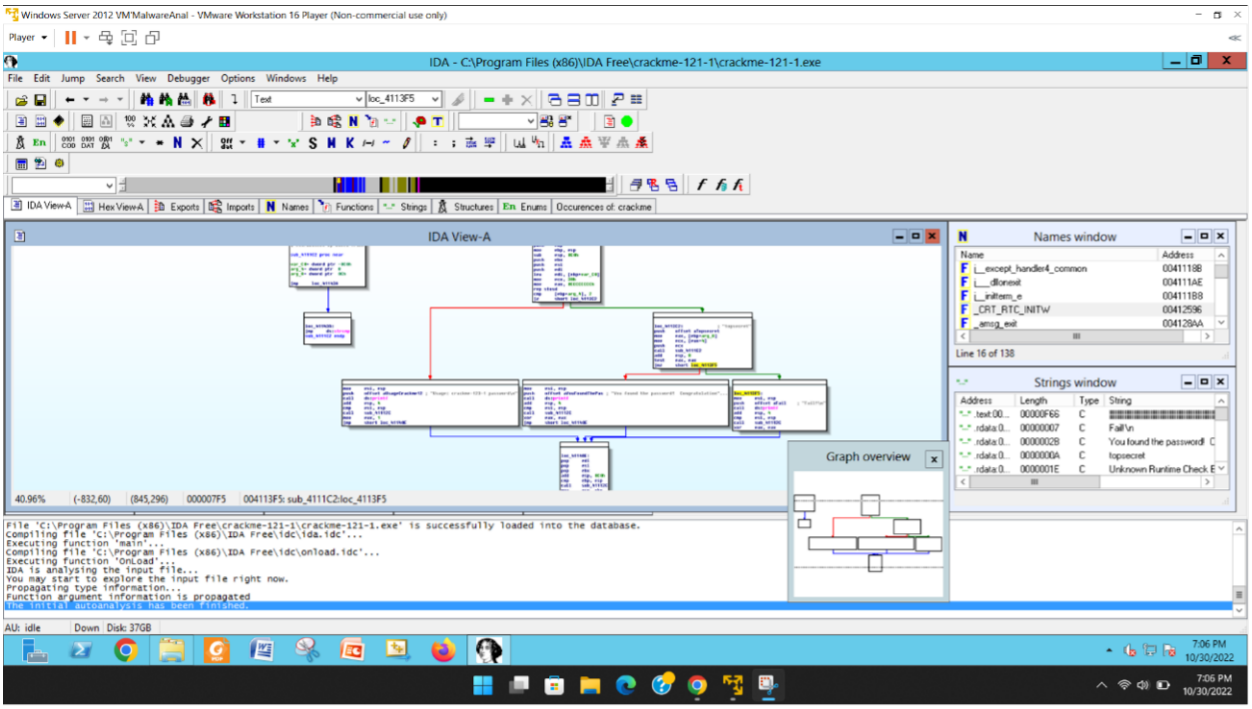
Overview: In this lab, I examined four different types of executable files and attempted to decrypt them using IDAPro.

Procedure:

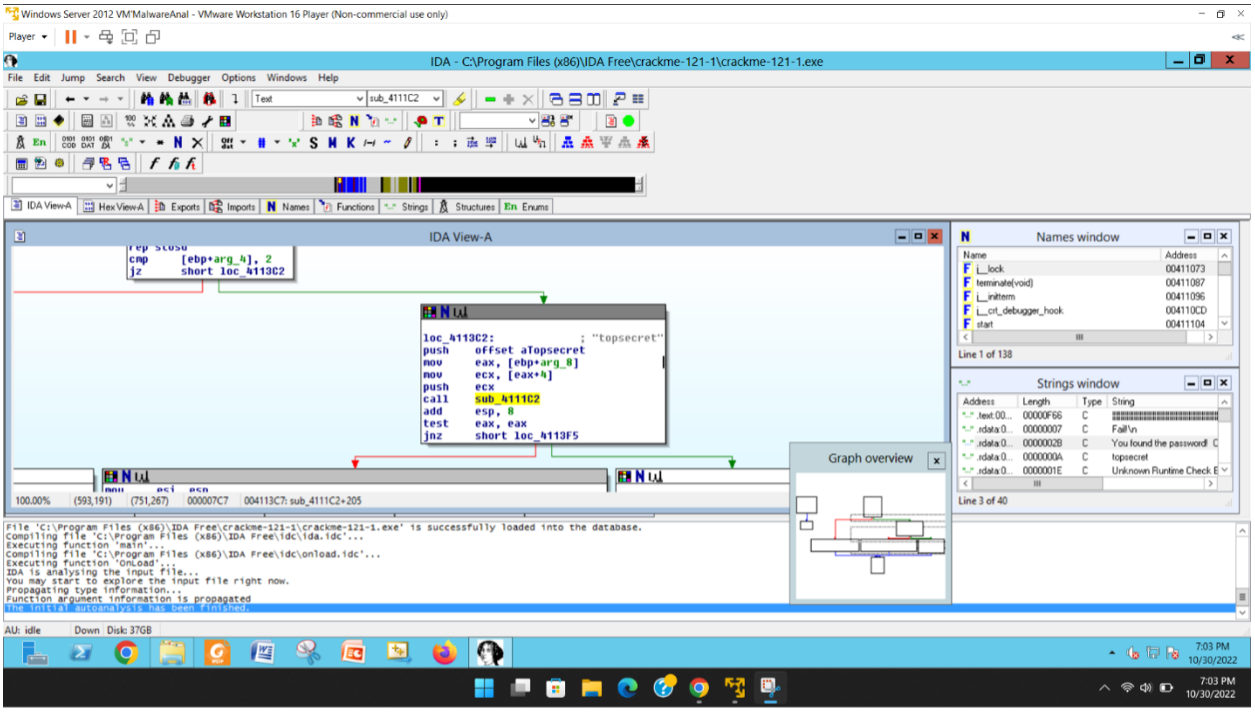
1. Go to Blackboard and look for a file called crackme-121-1.exe. Open the file with IDA Pro after downloading it.
2. Now, after it's opened, look in the string module for the string "You discovered password."



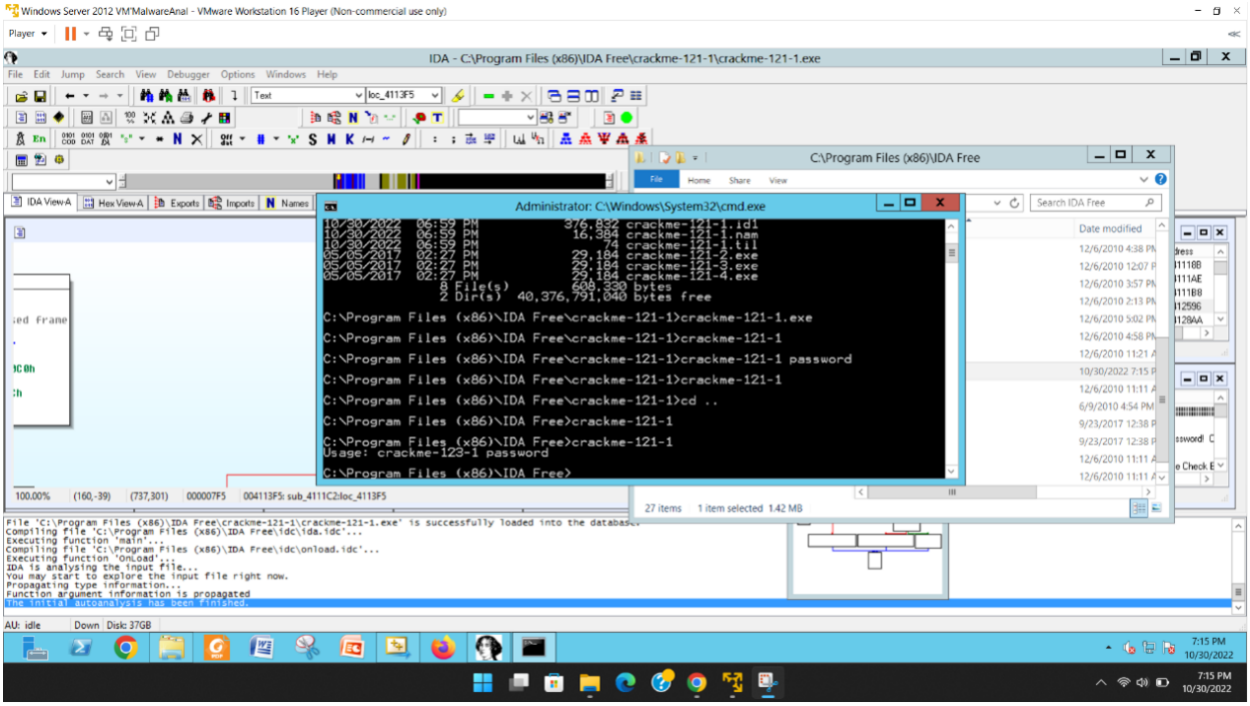
3. Zoom out to examine the whole flow of the exe file.

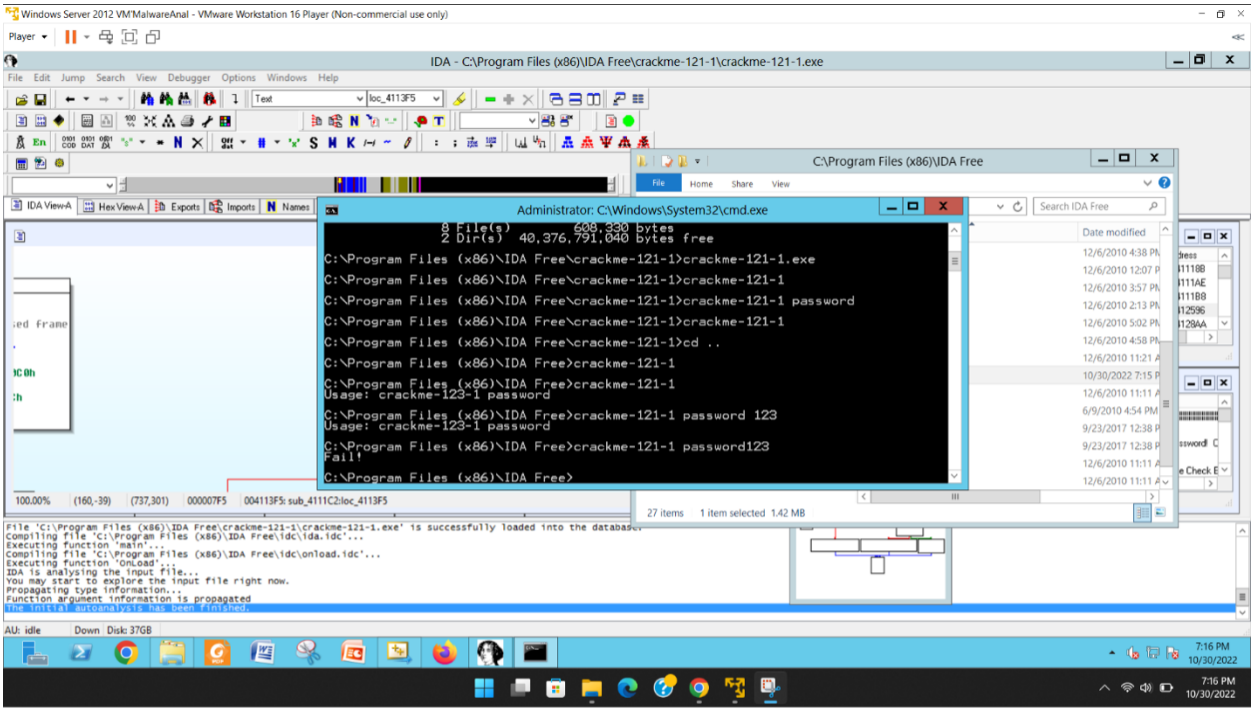


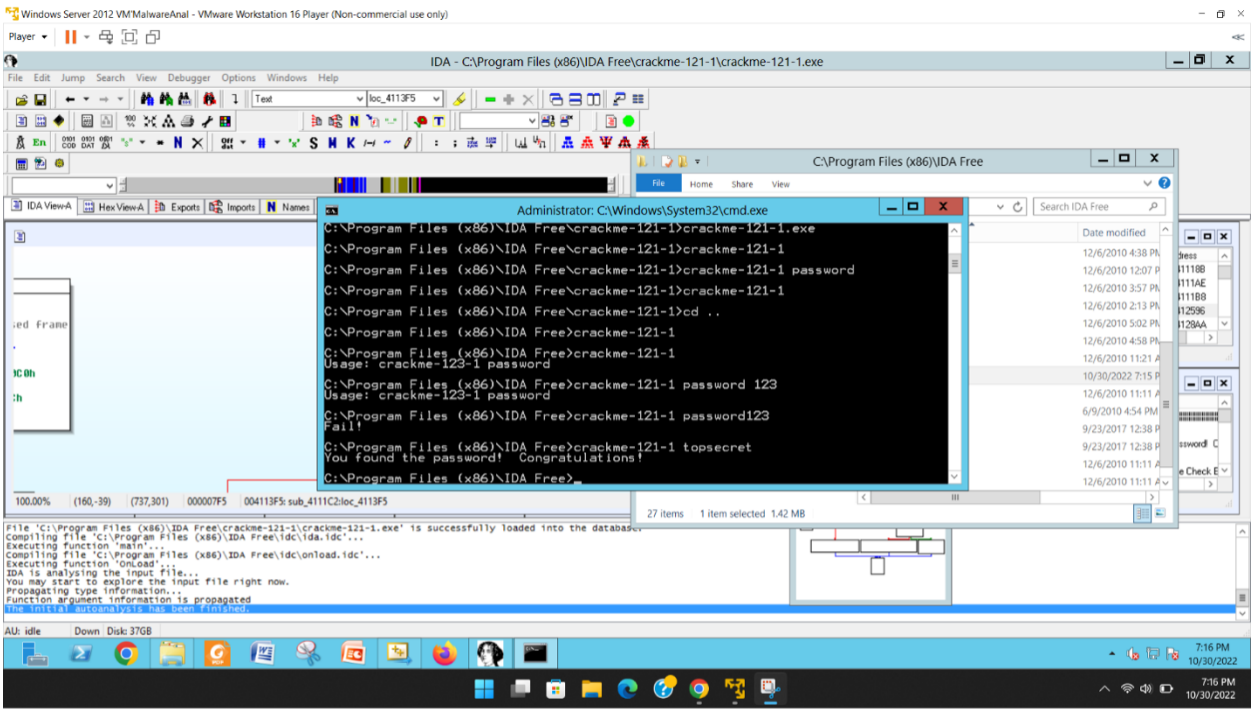
4. Following the flow will lead you to a password (the password is "top-secret").



5. Open cmd and go to the crackme directory, then execute the crackme-121-1.exe file, which will display instructions (it will ask you to enter a password). So enter the password you discovered in IDAPro.

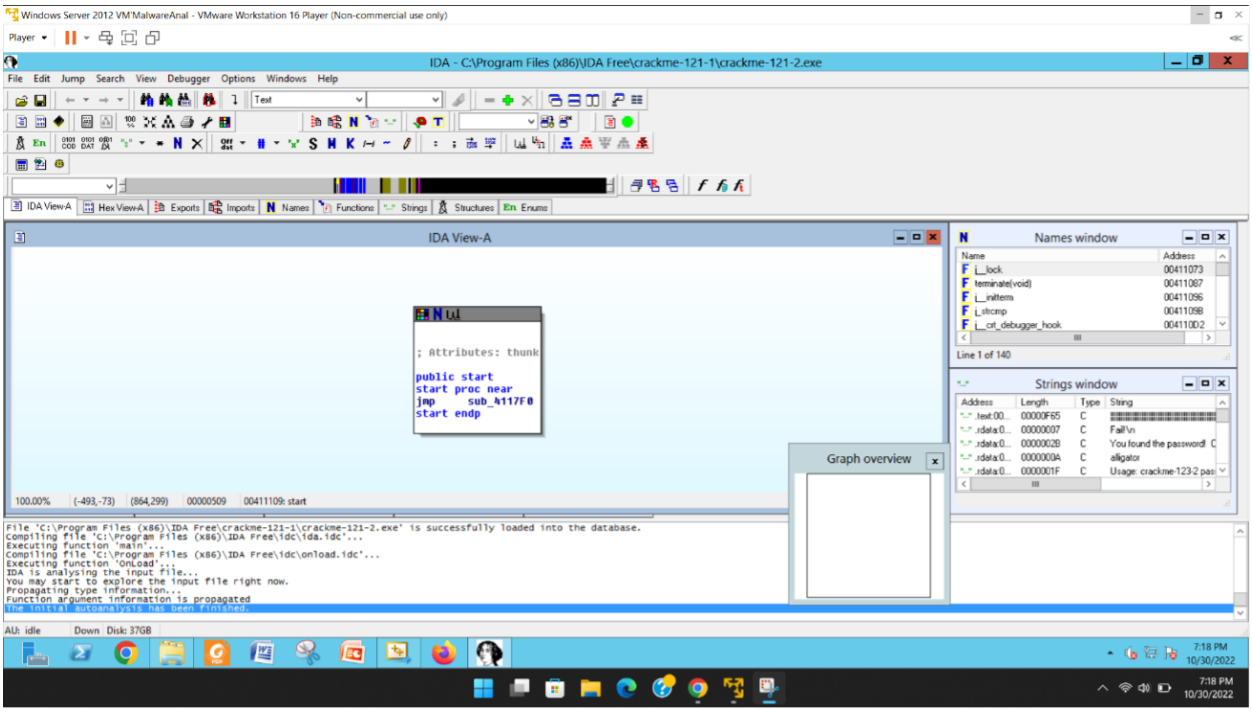

Windows Server 2012 VM'MalwareAnal - VMware Workstation 16 Player (Non-commercial use only)
IDA - C:\Program Files (x86)\IDA Free\crackme-121-1\crackme-121-1.exe
Administrator: C:\Windows\System32\cmd.exe
C:\Program Files (x86)\IDA Free\crackme-121-1>crackme-121-1.exe
C:\Program Files (x86)\IDA Free\crackme-121-1>crackme-121-1
C:\Program Files (x86)\IDA Free\crackme-121-1>crackme-121-1 password
C:\Program Files (x86)\IDA Free\crackme-121-1>crackme-121-1
C:\Program Files (x86)\IDA Free\crackme-121-1>cd ..
C:\Program Files (x86)\IDA Free>crackme-121-1
C:\Program Files (x86)\IDA Free>crackme-121-1
Usage: crackme-123-1 password
C:\Program Files (x86)\IDA Free>
C:\Program Files (x86)\IDA Free
27 items 1 item selected 1.42 MB
7:15 PM
10/30/2022

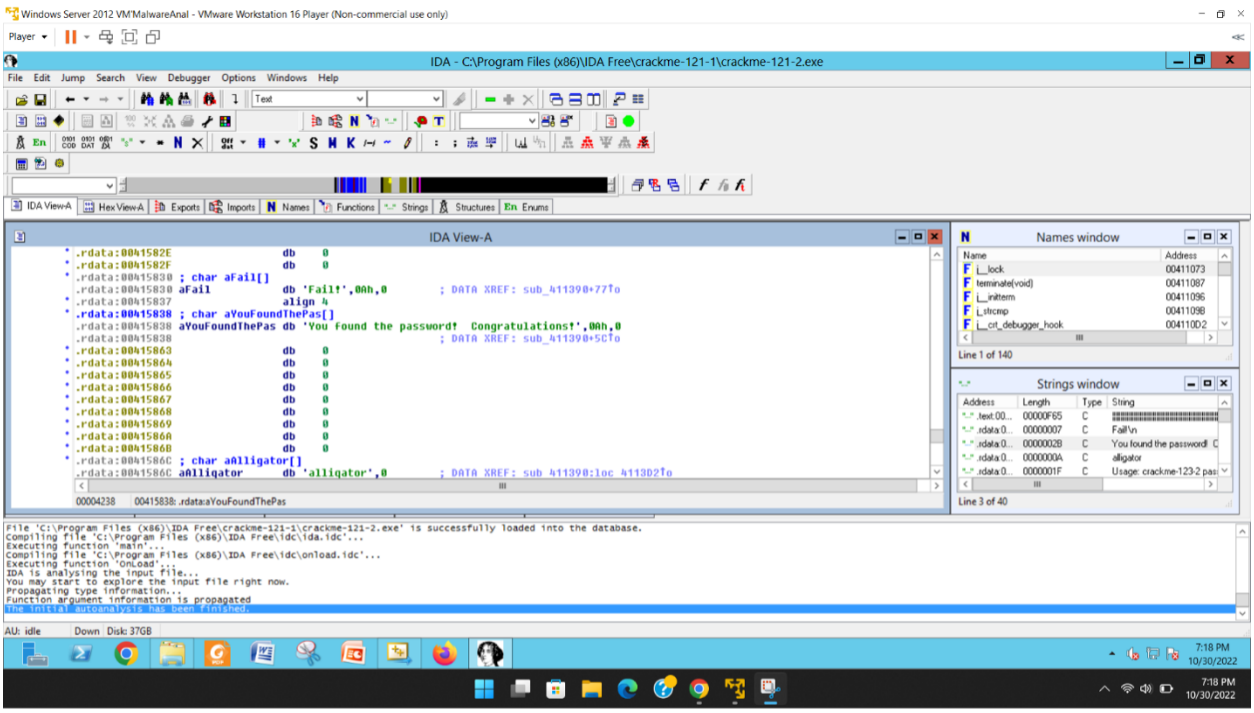

Windows Server 2012 VM'MalwareAnal - VMware Workstation 16 Player (Non-commercial use only)
IDA - C:\Program Files (x86)\IDA Free\crackme-121-1\crackme-121-1.exe
Administrator: C:\Windows\System32\cmd.exe
C:\Program Files (x86)\IDA Free\crackme-121-1>crackme-121-1.exe
C:\Program Files (x86)\IDA Free\crackme-121-1>crackme-121-1
C:\Program Files (x86)\IDA Free\crackme-121-1>crackme-121-1 password
C:\Program Files (x86)\IDA Free\crackme-121-1>crackme-121-1
C:\Program Files (x86)\IDA Free\crackme-121-1>cd ..
C:\Program Files (x86)\IDA Free>crackme-121-1
C:\Program Files (x86)\IDA Free>crackme-121-1
Usage: crackme-123-1 password
C:\Program Files (x86)\IDA Free>crackme-121-1 password 123
Usage: crackme-123-1 password
C:\Program Files (x86)\IDA Free>crackme-121-1 password123
Fail!
C:\Program Files (x86)\IDA Free>
C:\Program Files (x86)\IDA Free
27 items 1 item selected 1.42 MB
7:16 PM
10/30/2022



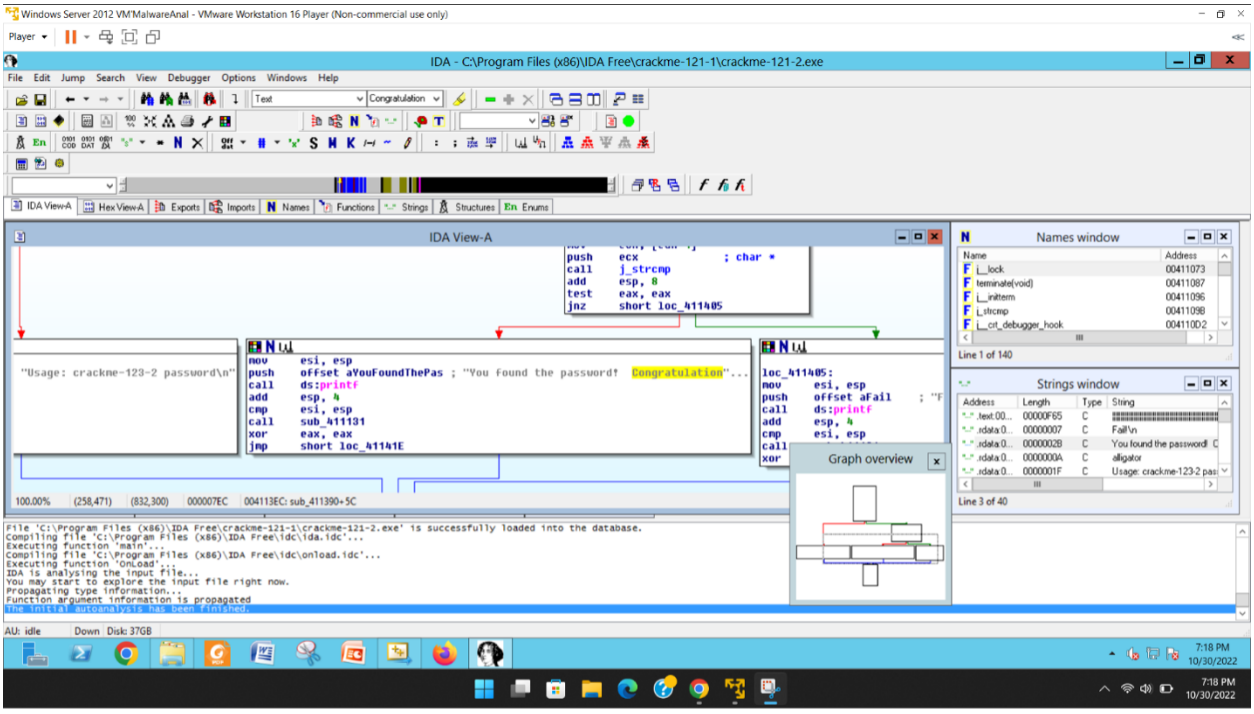
Crackme-121- 2.exe

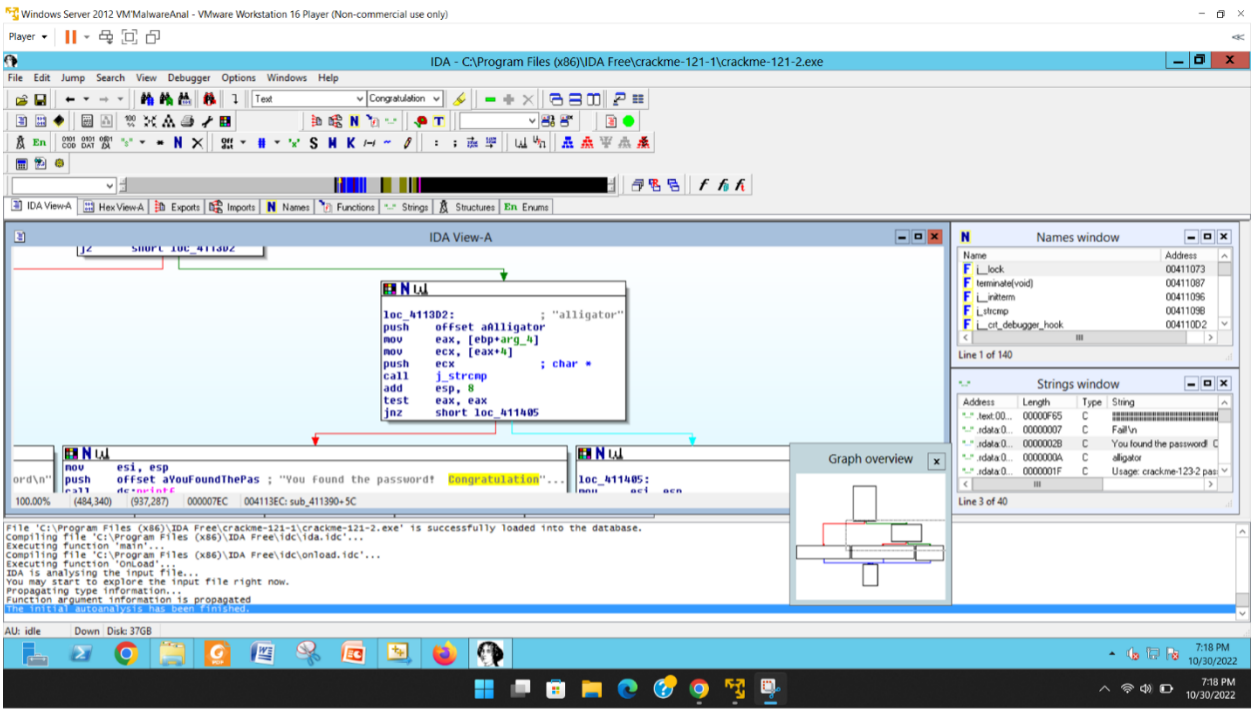
1. Load crackme-121-2.exe into IDAPro and look in the strings module for the string "discovered password."
2. It will take you to the DATA XREF: sub 411390+5c location, which you should double-click on.



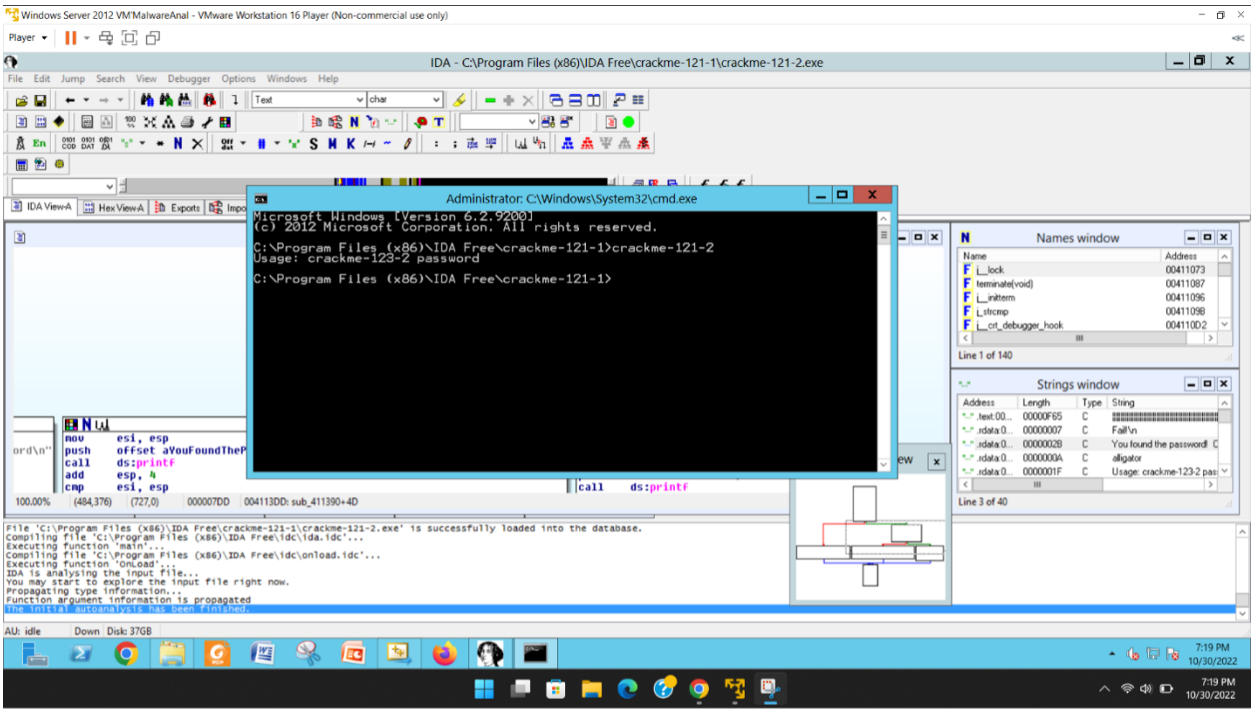


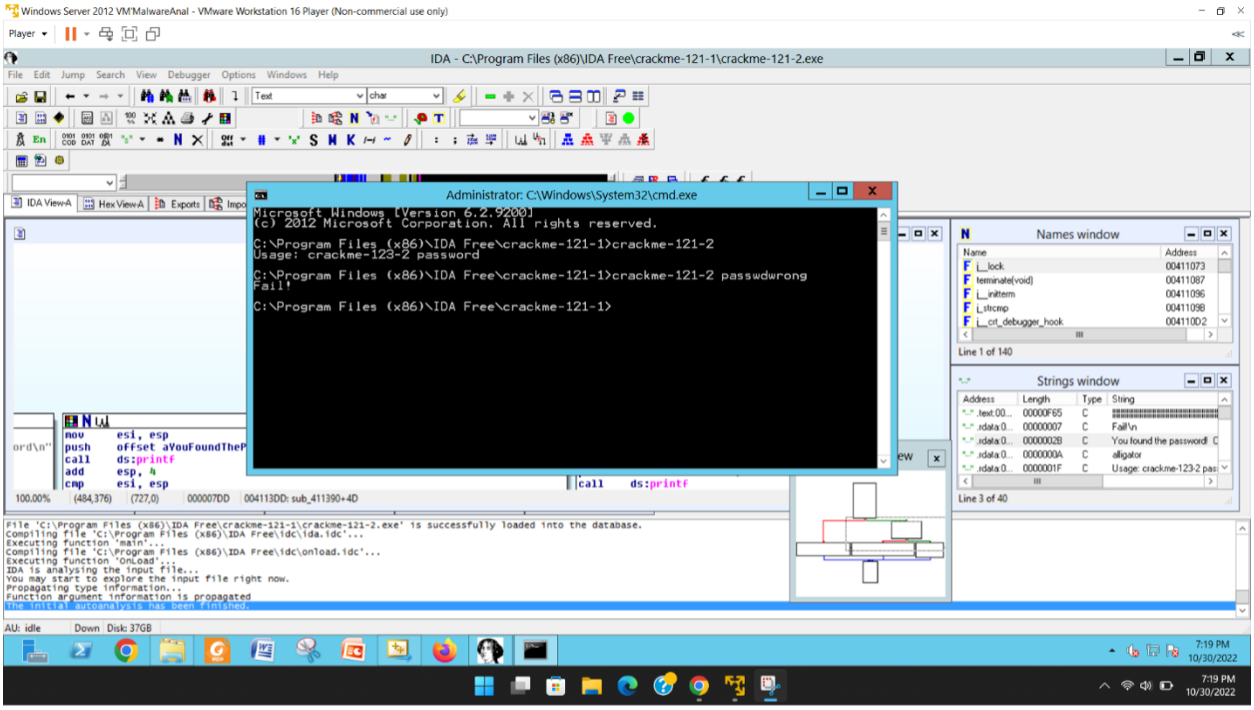
3. It will show you how it flows. So, zoom out the graph and start looking for the password.
4. You have discovered the password.

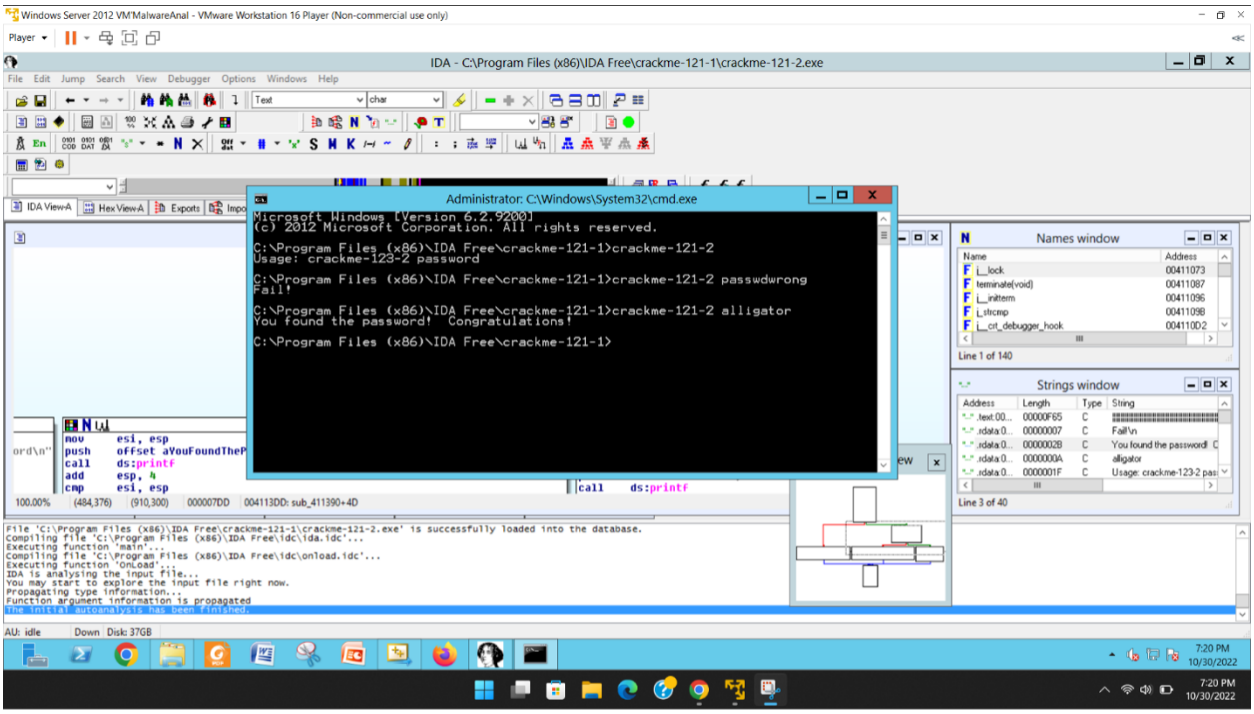




5. Open cmd and run crackme-121-2.exe, which will prompt you for a password.

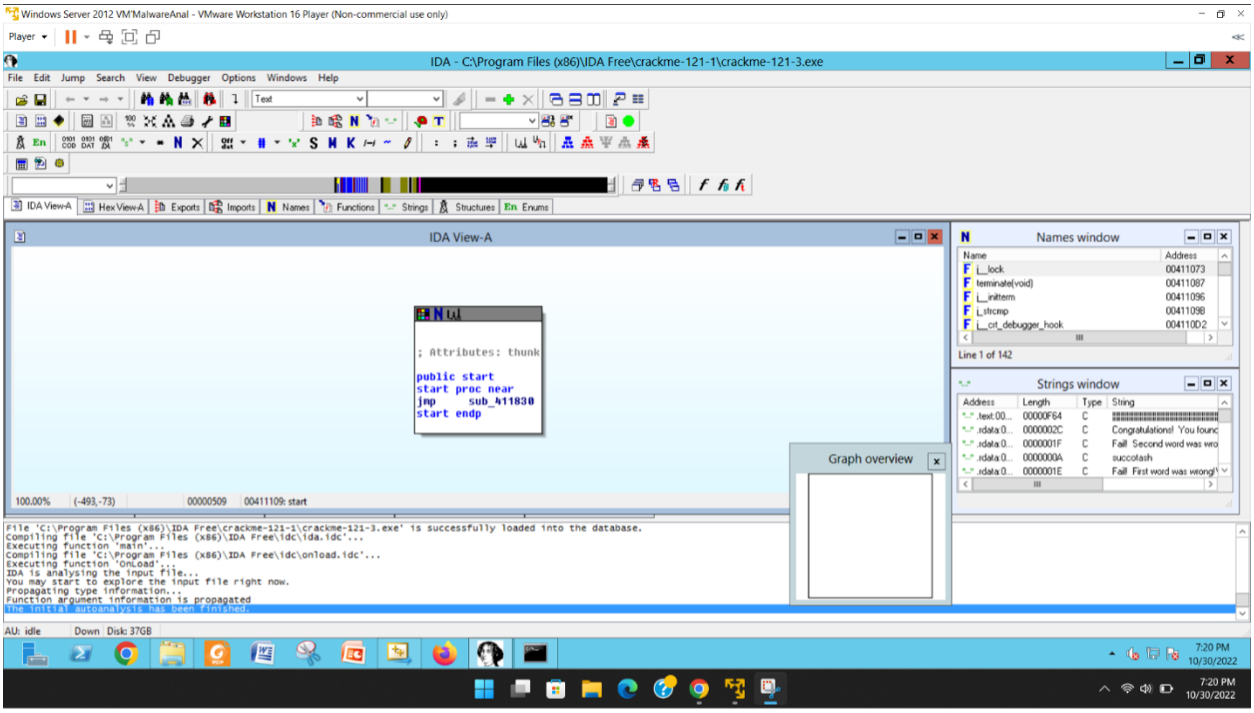



Windows Server 2012 VM'MalwareAnal - VMware Workstation 16 Player (Non-commercial use only)
IDA - C:\Program Files (x86)\IDA Free\crackme-121-1\crackme-121-2.exe
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.2.9200]
(c) 2012 Microsoft Corporation. All rights reserved.
C:\Program Files (x86)\IDA Free\crackme-121-1>crackme-121-2
Usage: crackme-123-2 password
C:\Program Files (x86)\IDA Free\crackme-121-1>crackme-121-2 passwdwrong
Fail!
C:\Program Files (x86)\IDA Free\crackme-121-1>
Names window
Strings window
7:19 PM
10/30/2022

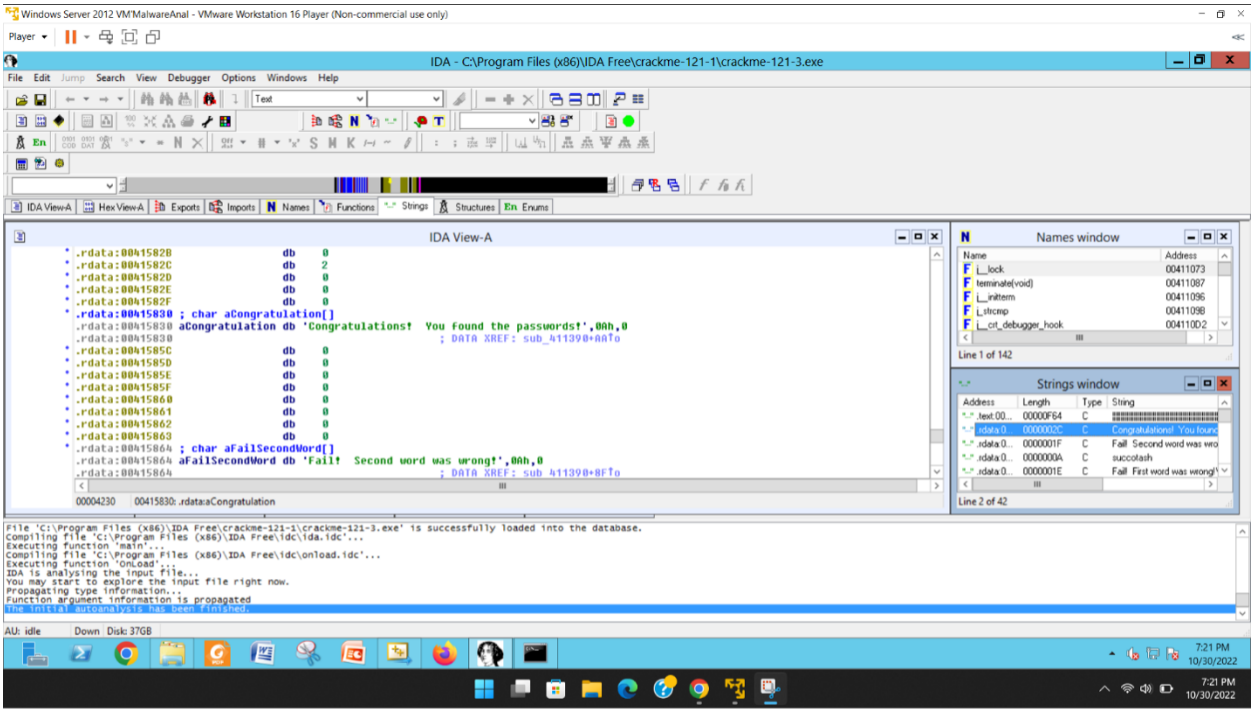

Windows Server 2012 VM'MalwareAnal - VMware Workstation 16 Player (Non-commercial use only)
IDA - C:\Program Files (x86)\IDA Free\crackme-121-1\crackme-121-2.exe
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.2.9200]
(c) 2012 Microsoft Corporation. All rights reserved.
C:\Program Files (x86)\IDA Free\crackme-121-1>crackme-121-2
Usage: crackme-123-2 password
C:\Program Files (x86)\IDA Free\crackme-121-1>crackme-121-2 passwdwrong
Fail!
C:\Program Files (x86)\IDA Free\crackme-121-1>crackme-121-2 alligator
You found the password! Congratulations!
C:\Program Files (x86)\IDA Free\crackme-121-1>
Names window
Strings window
7:20 PM
10/30/2022

Crackme-121- 3.exe

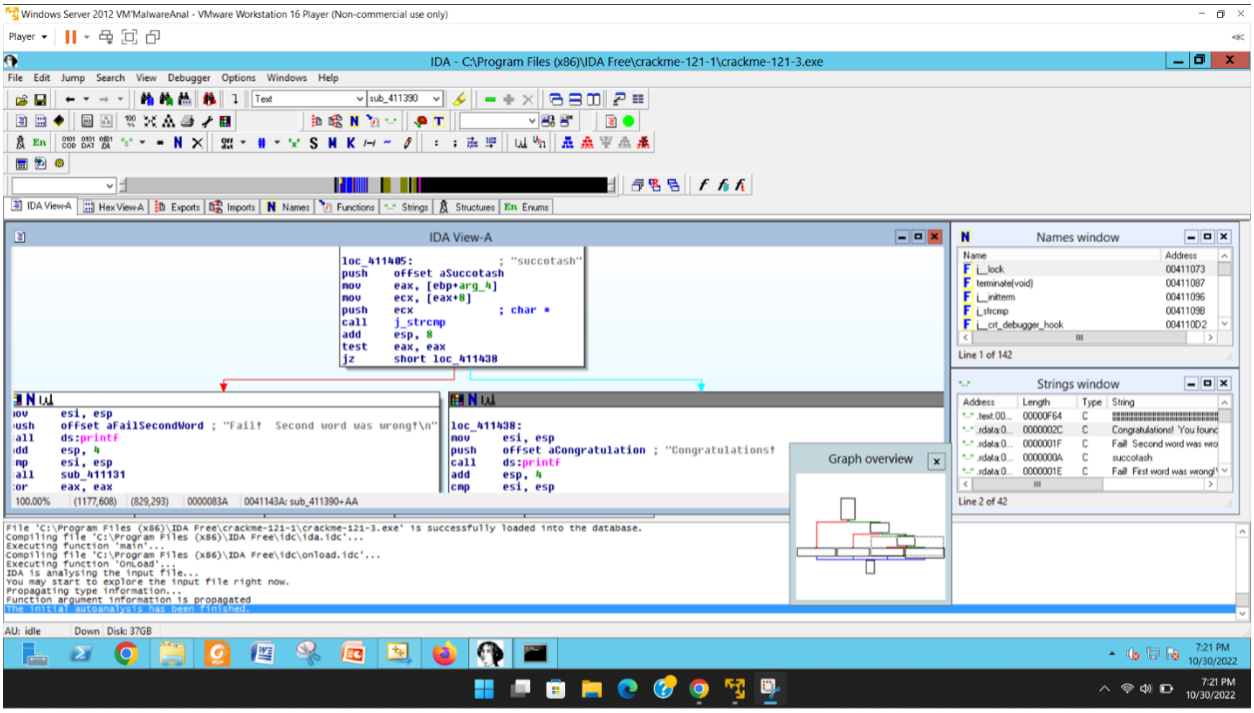
1. In IDAPro, open crackme-121-3.exe.



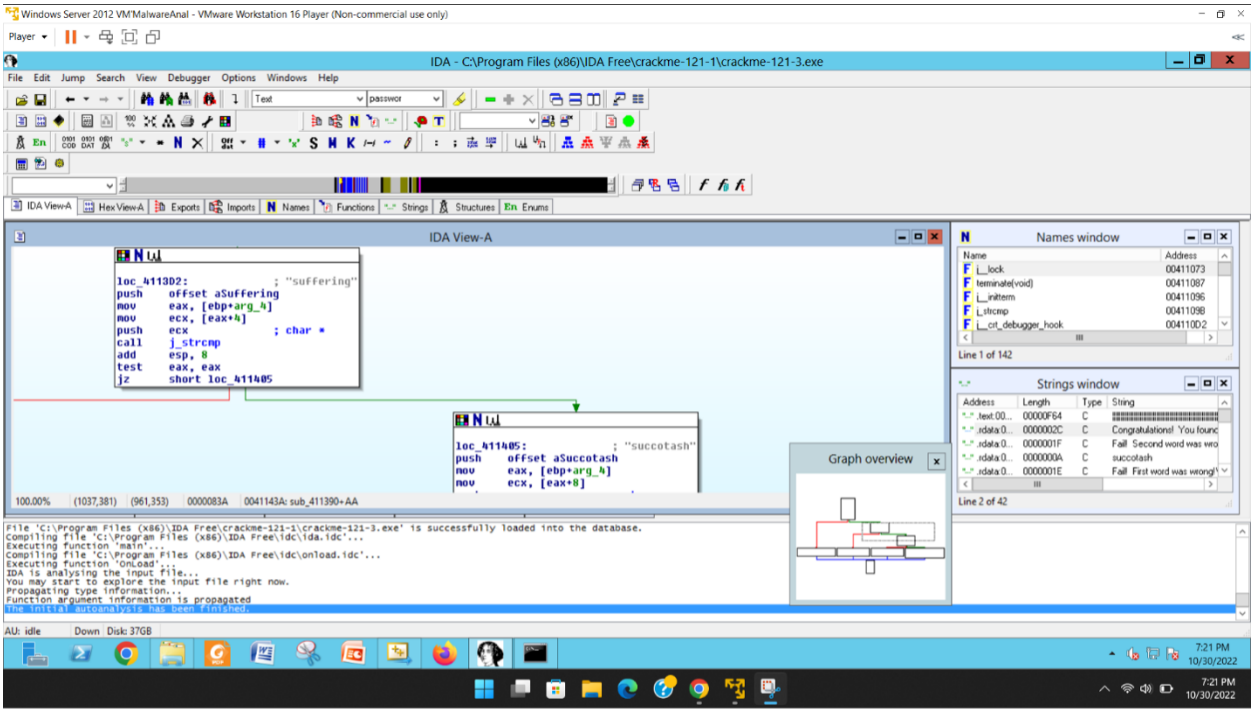
2. Double-click on the string "Congratulations."



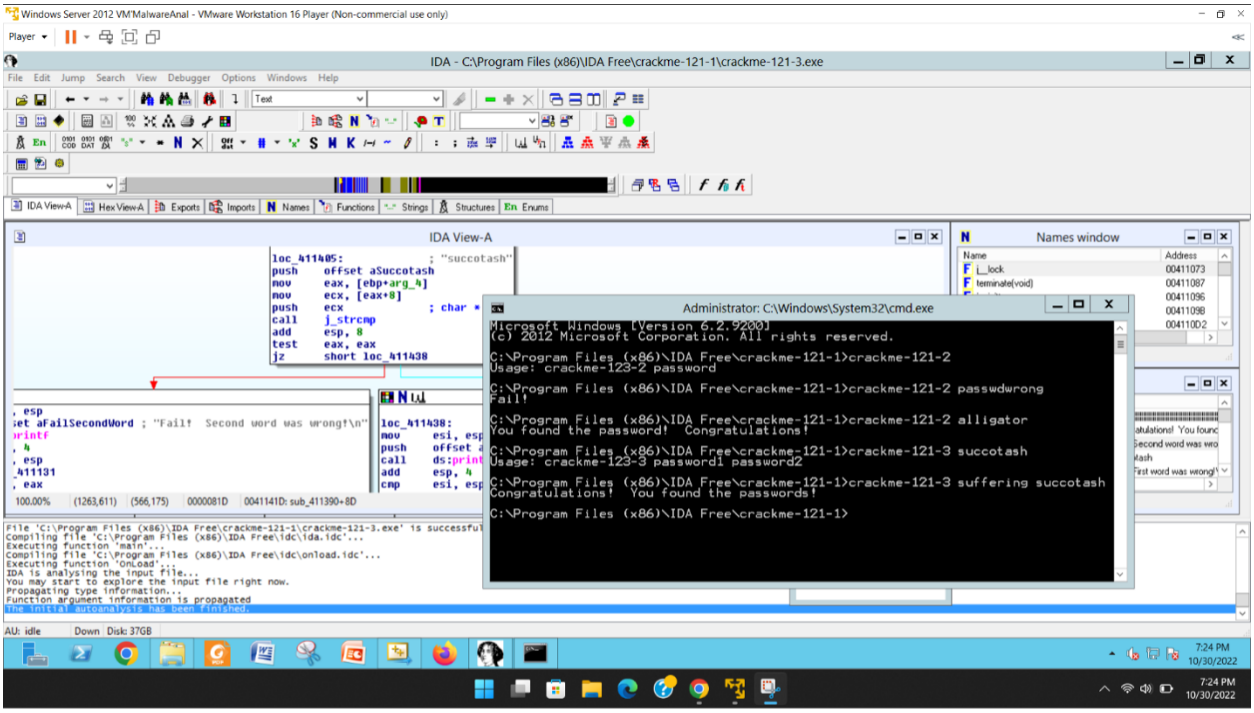
3. Now that you know where it is. When you double-click it, it will take you to a graphical depiction.



4. Find the password; you will be given two passwords that differ from the last exe file. We comprehend it by referring to the graph flow chart (password1=suffering, password2=succotash).

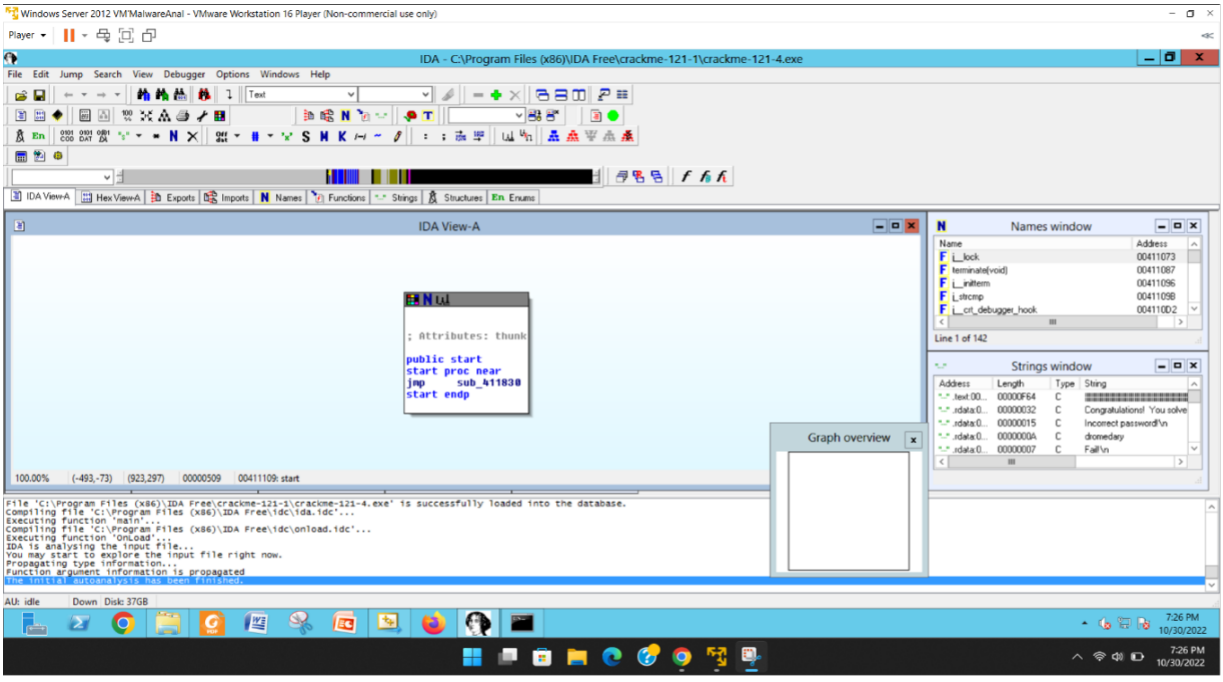


5. Open cmd and run the crackme-121-3.exe file (when we run it, it will prompt us to enter both passwords).

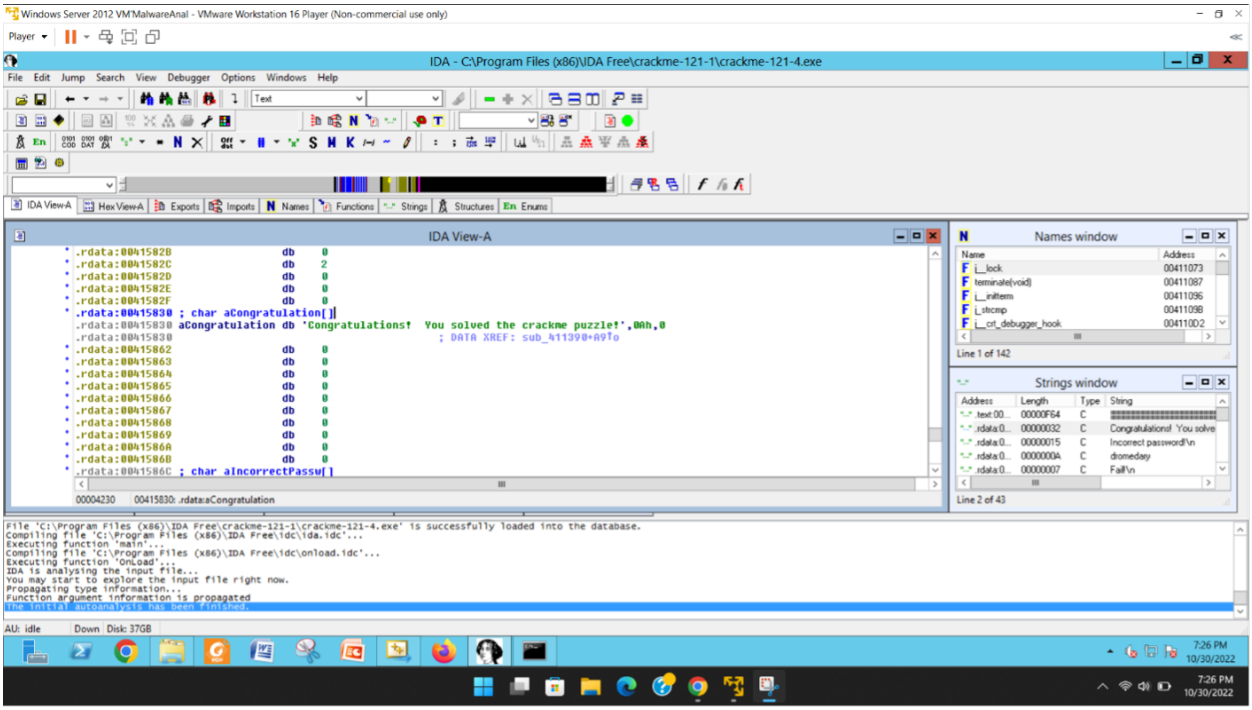


Crackme-121- 4.exe

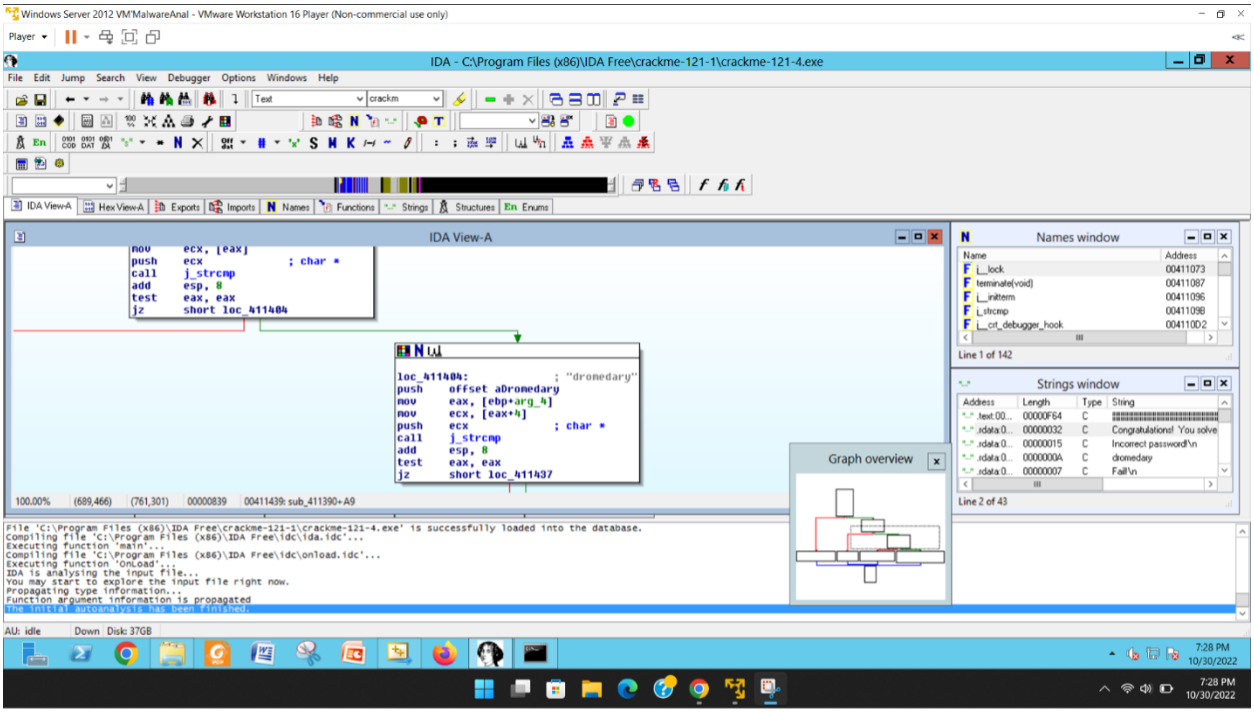
1. In IDAPro, open crackme-121-4.exe.



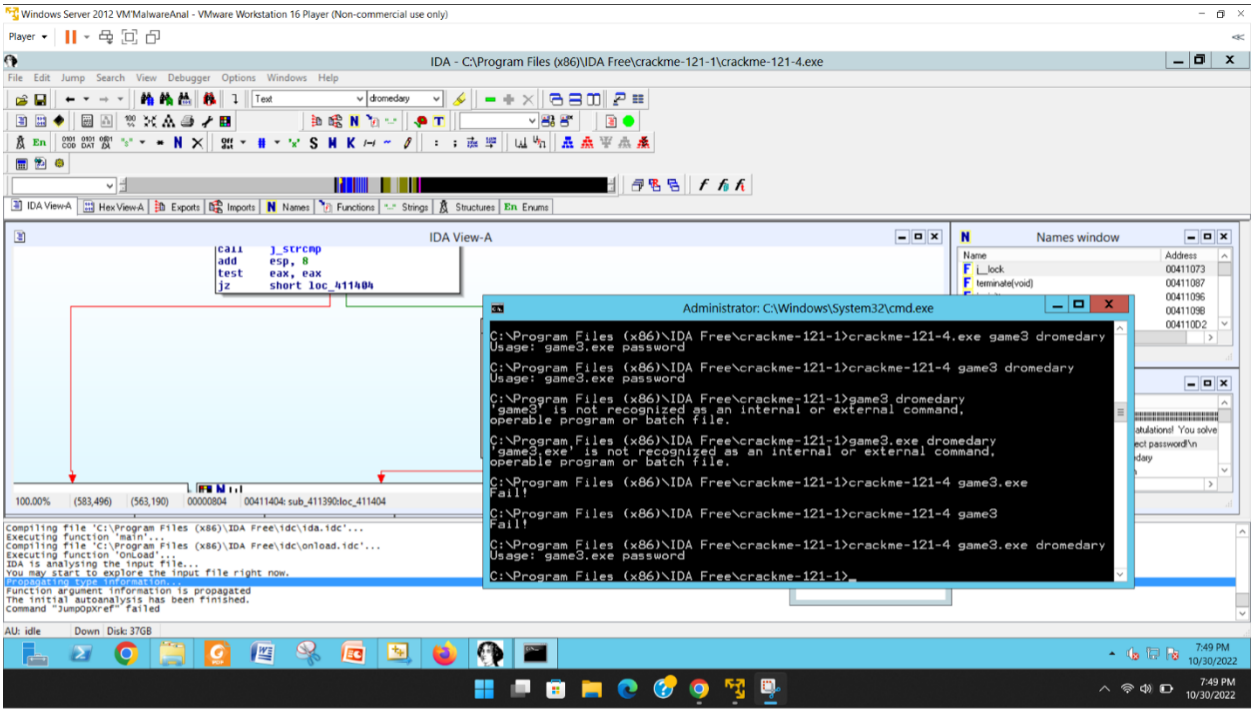
2. In the strings module, look for the string "Password discovered."



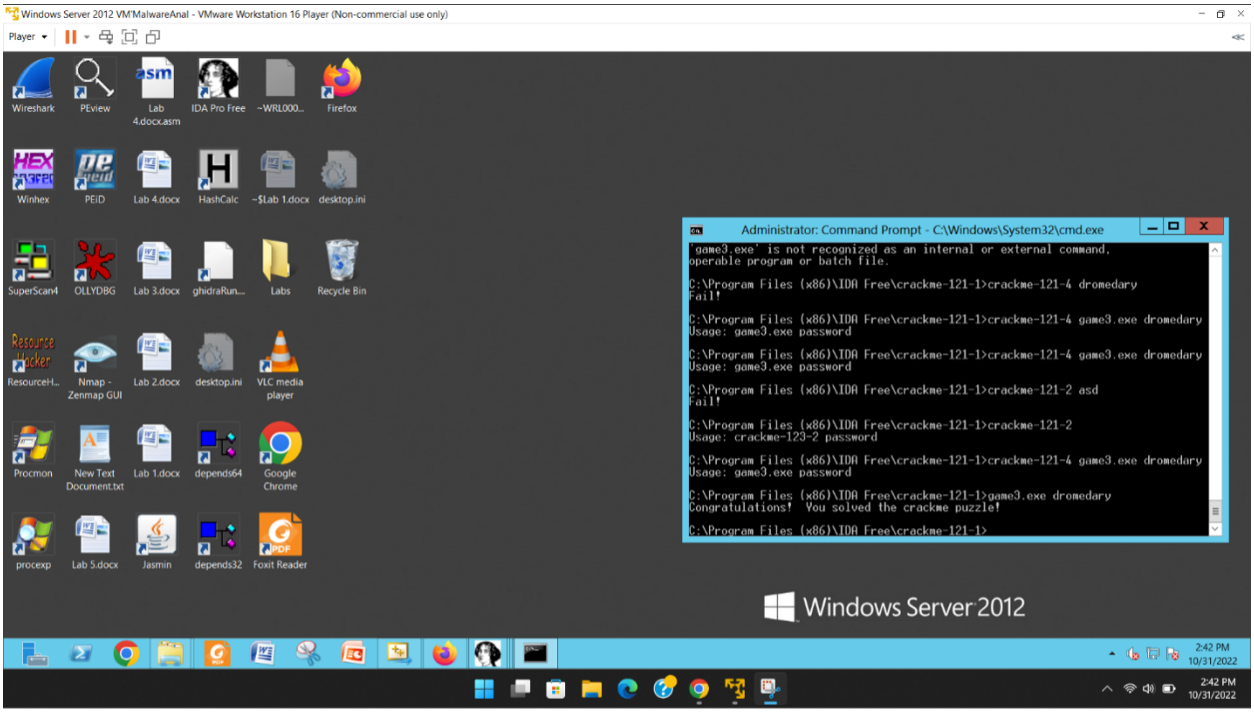
3. Double-click on the spot where we found it.



4. We obtained a password dromedary by following the graph flow, however, it would not execute in cmd. Launch cmd and run crackme-121-4.exe.



5. So we need to comprehend what it suggests to do here. As we can see, it uses the game3.exe password, thus we need to execute the game3.exe file and then supply the password, but we don't have game3.exe, so what do we do now? So we go to the crackme folder and rename crackme-121-4.exe to game3.exe so that we can run game3.exe and then enter the password.

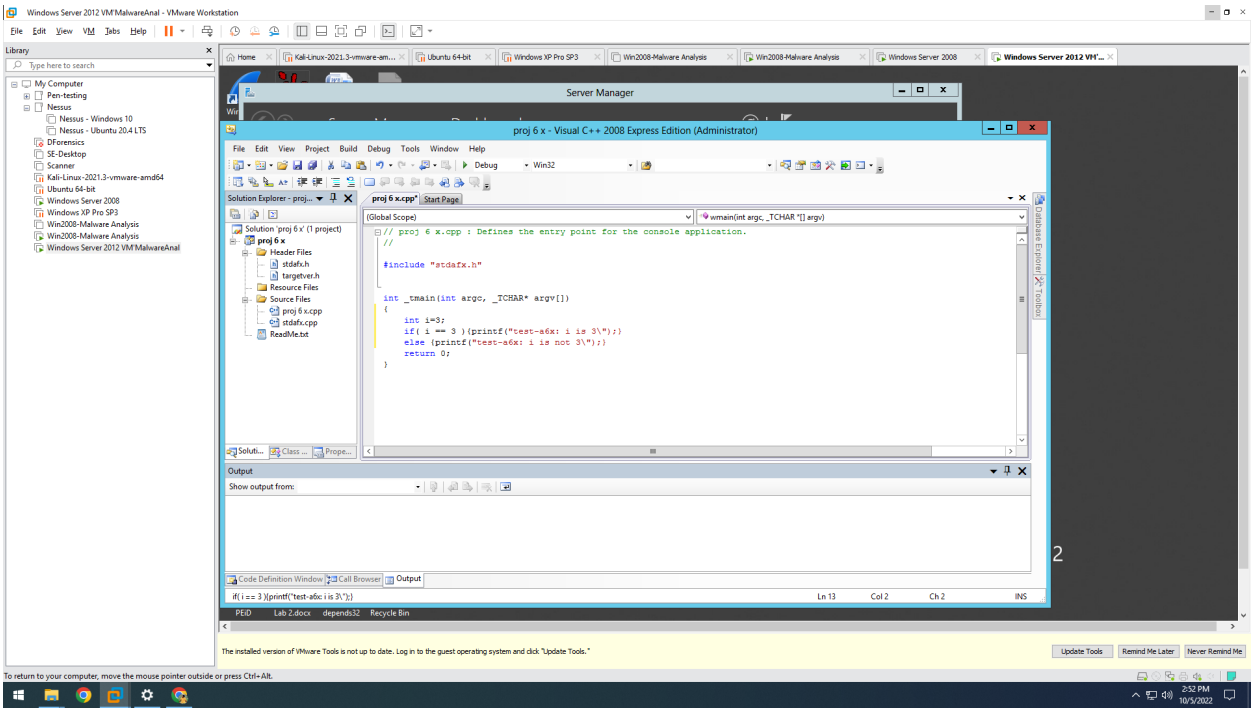


Proj 6: Disassembling C on windows part 3

Overview: In this lab, we will create a program utilizing if statements. And analyze it with an IDAPro disassembler to see how assembly language appears.

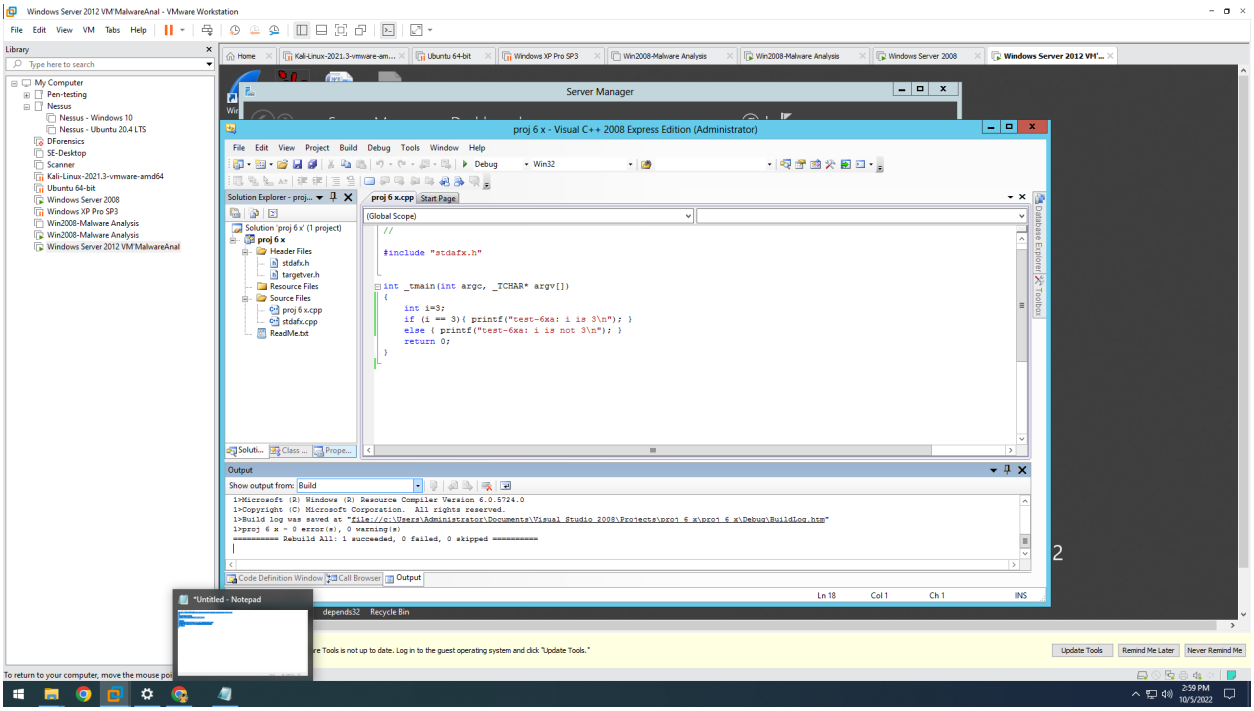
Open Visual C++ and create a c program. Select and accept the win32 console application after clicking on win32.

```
#include "stdafx.h"
int _tmain(int argc, _TCHAR* argv[])
{
        int i=3;
        if (i == 3)
          { printf("YOURNAME-6xa: i is 3\n"); }
                else { printf("YOURNAME-6xa: i is not 3\n"); }
           return 0;
}
```

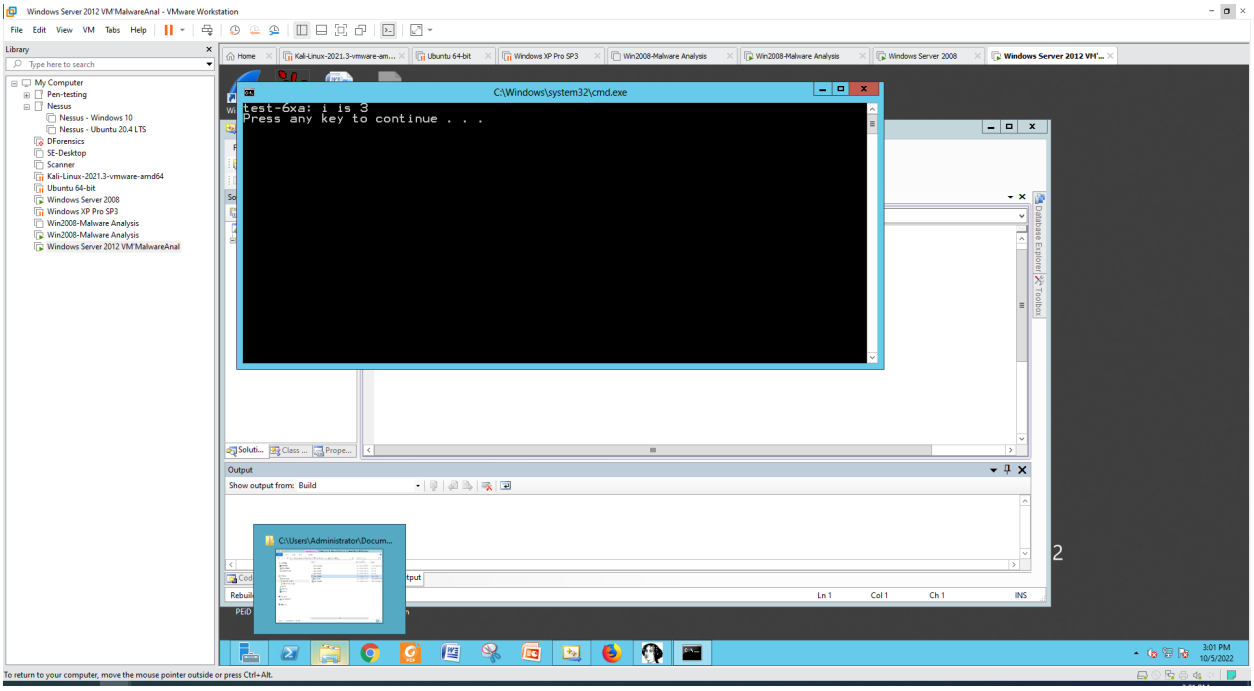


Procedure:

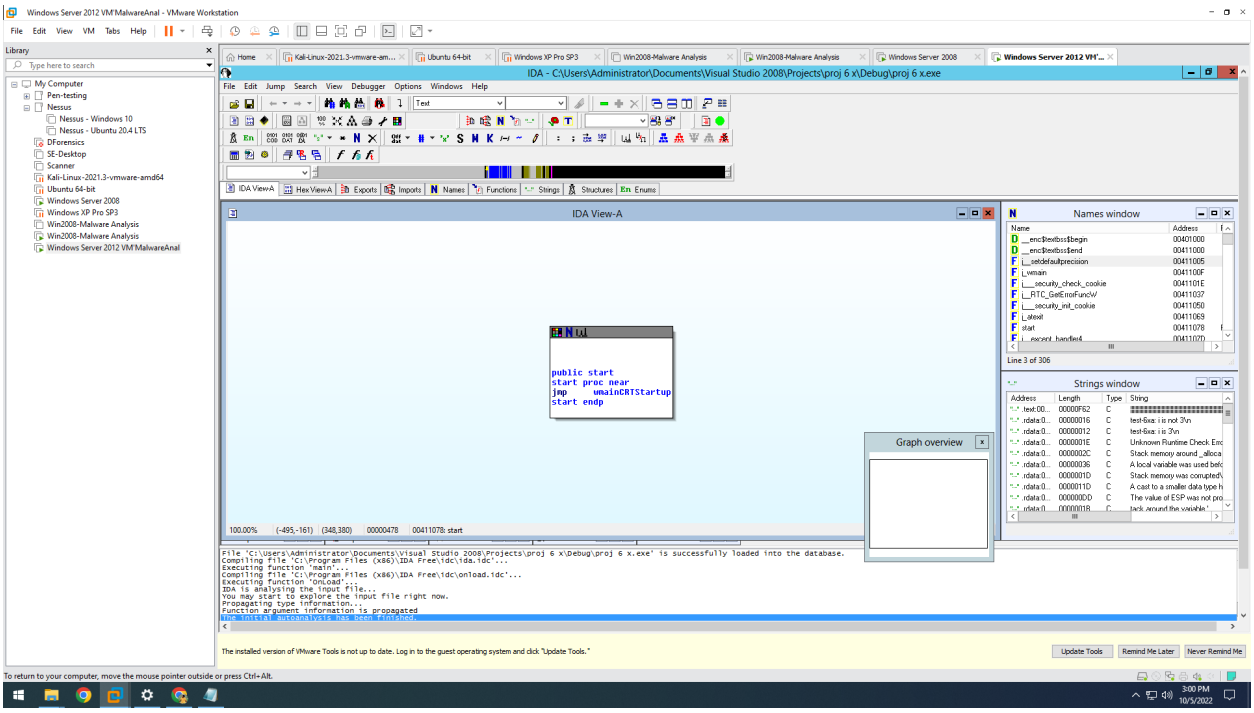
1. Build and execute the solution without debugging.

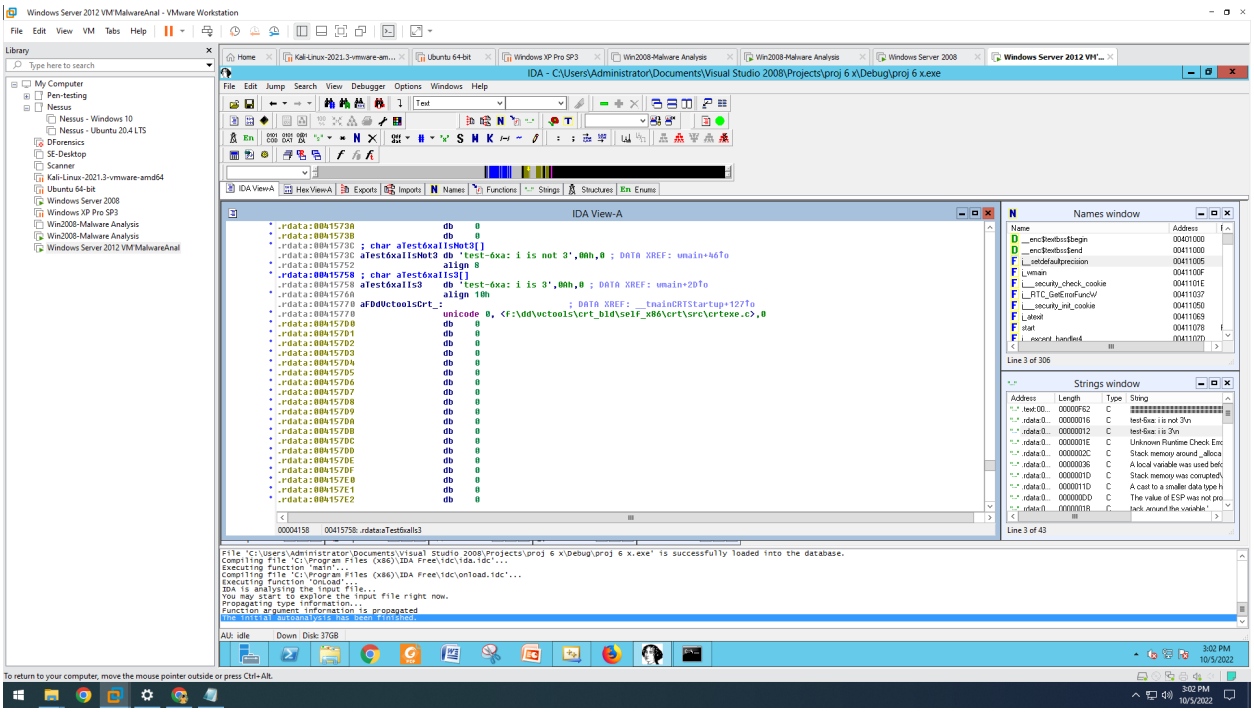


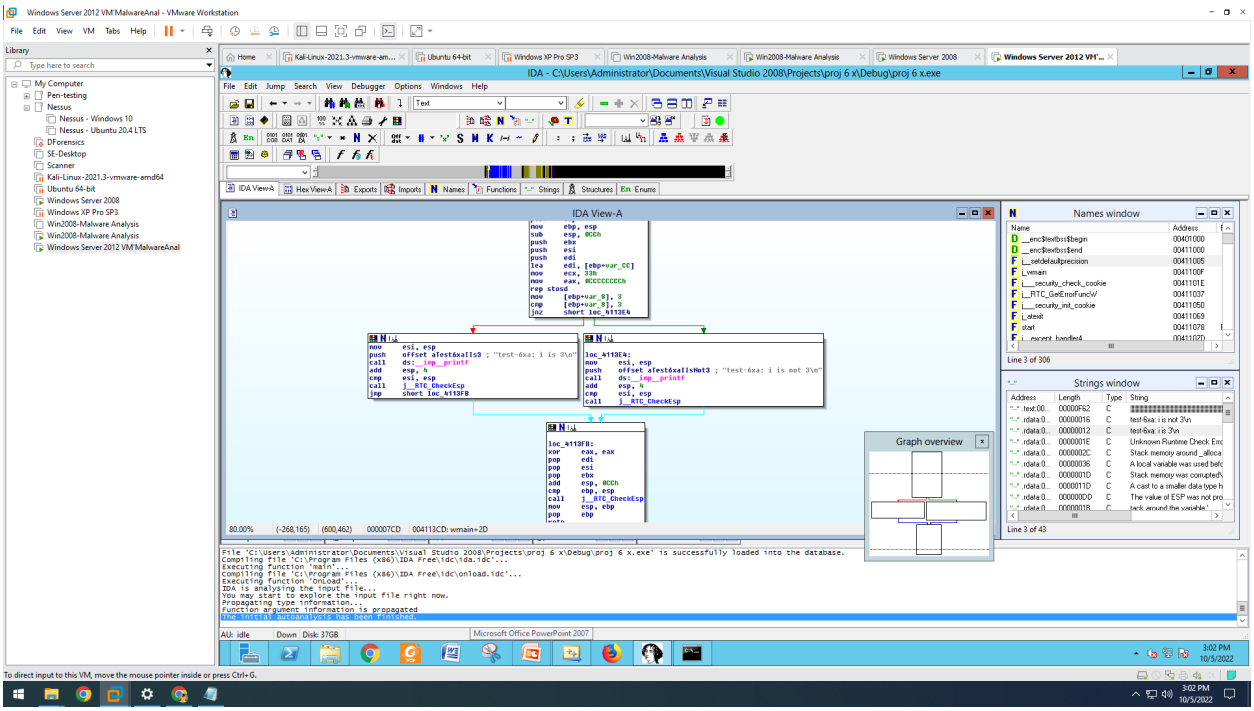
2. In IDAPro, open the c code exe file.



3. Find and select test-6ax in the string module. Then it will redirect you to the XREF address, where you should double-click.





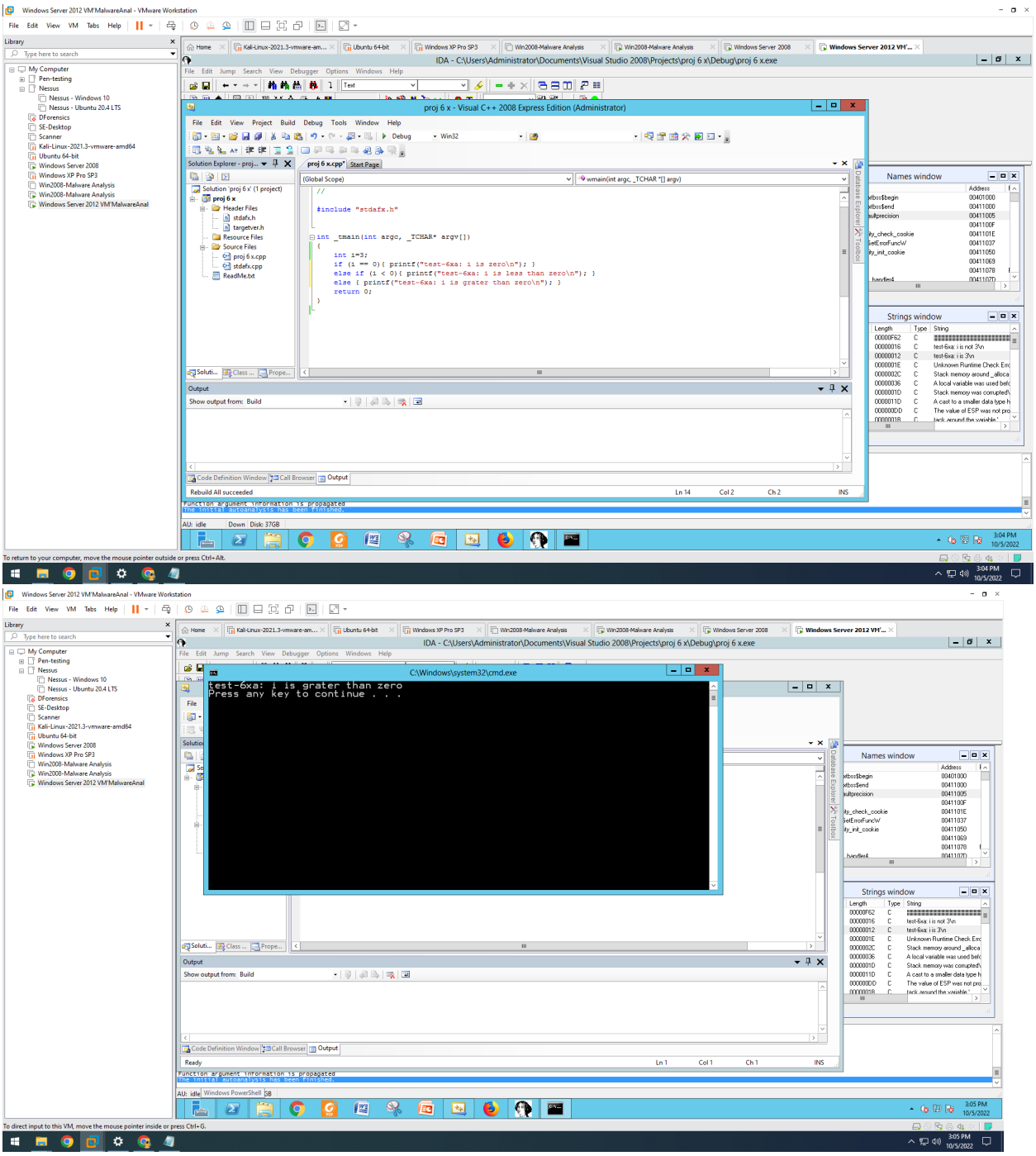


CHALLENGE

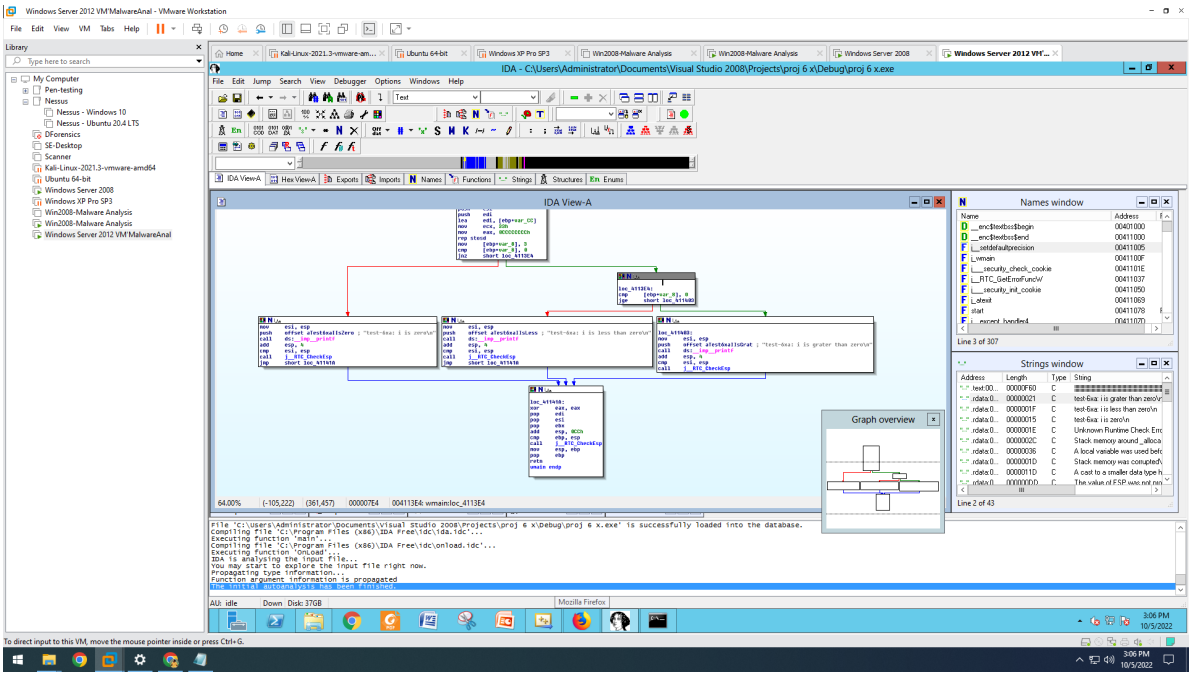
- Modify the c program to run a three-way test to determine whether a variable is less than, equal to, or higher than zero.
- In all three circumstances, print suitable messages mentioning your name.
- Compile and disassemble it to generate an assembly code identical to the one shown below.
- Create a c++ program with Visual Studio.

Procedure:

1. Build the solution and run it without debugging, like in the previous procedure, to determine the flow of code in IDAPro.



2. In IDAPro, open the c code exe file.
3. To do so, open IDAPro and search for a string in the strings module before clicking it. You will then be taken to the XERF location, so double-click it.

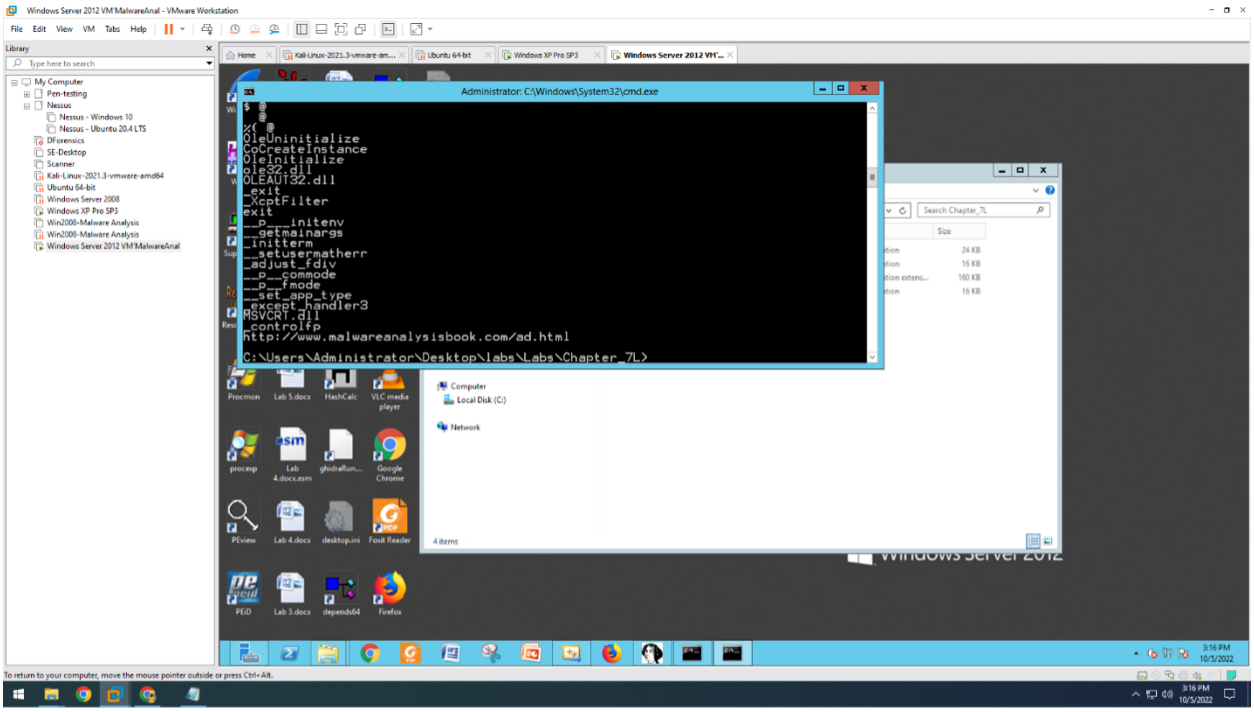

Windows Server 2012 VM'MalwareAnal - VMware Workstation
IDA - C:\Users\Administrator\Documents\Visual Studio 2008\Projects\proj 6 x\Debug\proj 6 x.exe
IDA View-A
Names window
Strings window
Graph overview
File 'C:\Users\Administrator\Documents\Visual Studio 2008\Projects\proj 6 x\Debug\proj 6 x.exe' is successfully loaded into the database.
Compiling file 'C:\Program Files (x86)\IDA Free\idc\ida.idc'...
Executing function 'main'...
Compiling file 'C:\Program Files (x86)\IDA Free\idc\onload.idc'...
Executing function 'OnLoad'...
IDA is analysing the input file...
You may start to explore the input file right now.
Propagating type information...
Function argument information is propagated
The initial autoanalysis has been finished.

Proj 7: Analyzing malicious windows programs

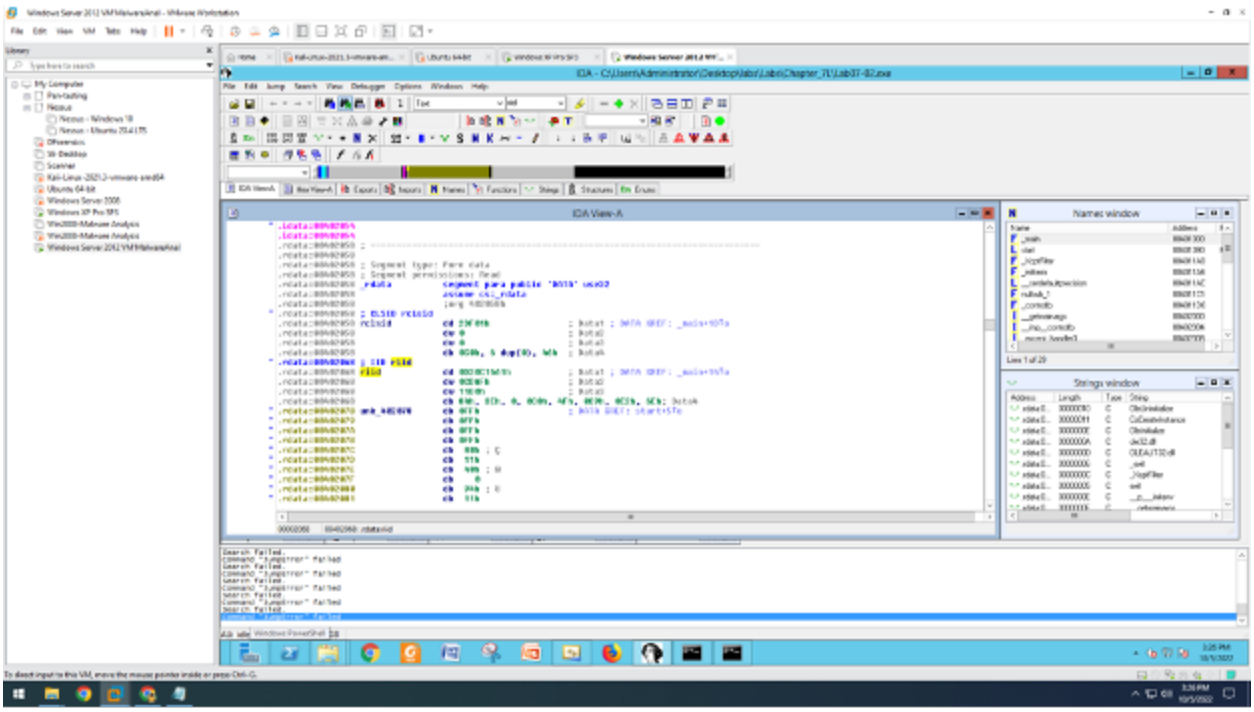
Overview: In this lab, I examined a malicious file that contained a virus and examined how it changed the registry after infecting the system.

Procedure:

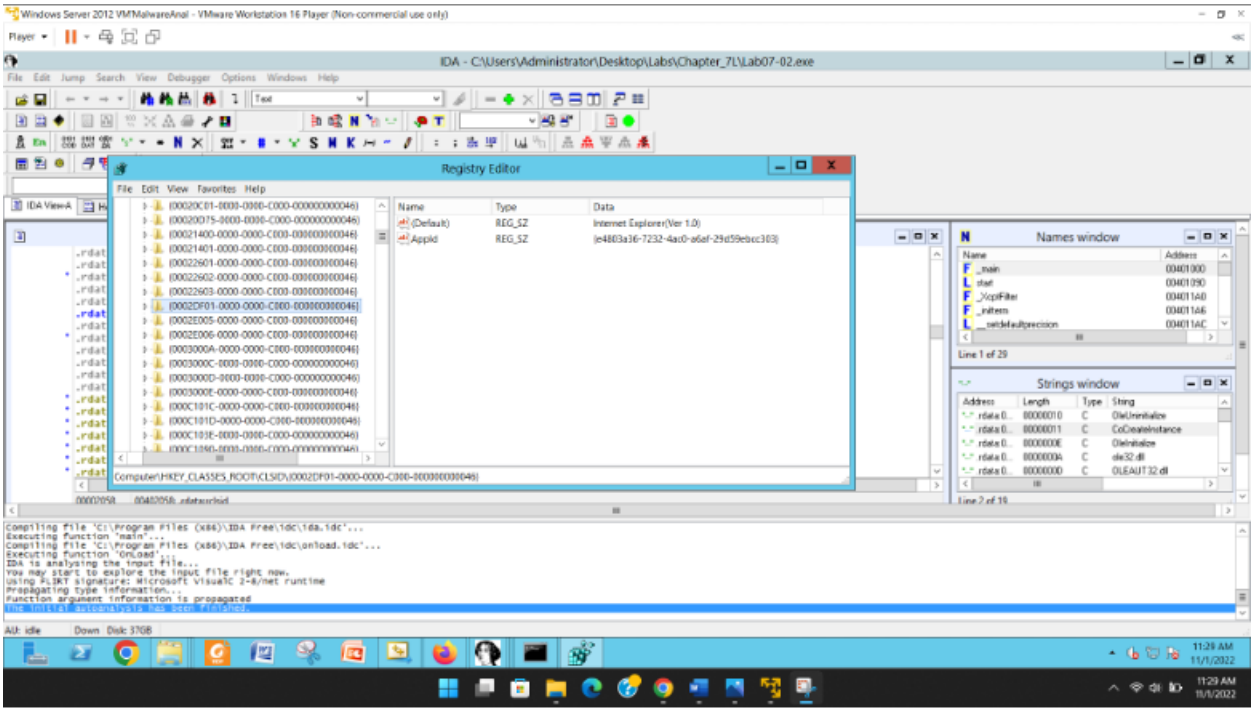
1. Use command to run strings. So, first, go to the lab Chapter 7 folder and launch strings Lab07-02.exe, as it displays a malicious website called malwareanaysisbooks.com.



2. Enter Lab07-02.exe into IDAPro and look for the rclsid. It's already coded across numerous lines, as seen below, but you can reconstruct it from a single 128-bit value string beginning with 2DF0 and ending with 0046.



3. To open the registry, go to Windows Run (crtl+r), type Regedit, then option, locate, and type 0002D.

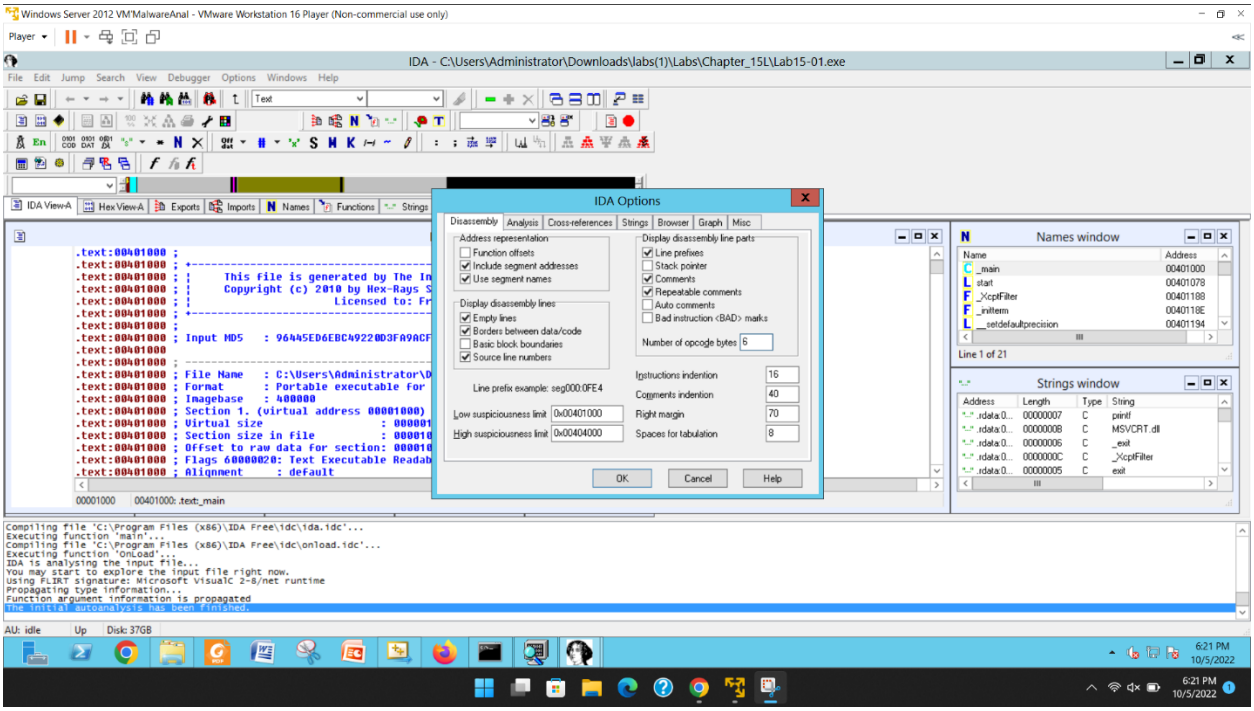


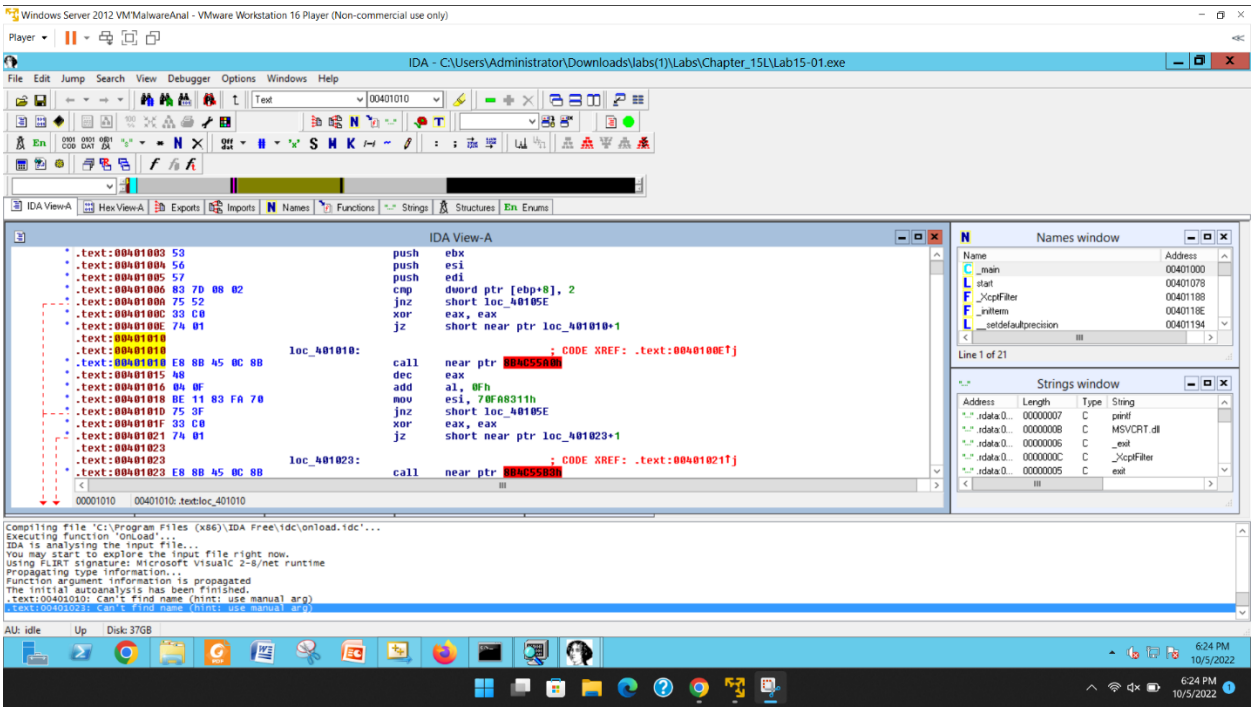
Proj 12: Anti-Disassembly

Overview: In this lab, I will utilize Lab15-01.exe in IDAPro for disassembly to overcome anti-disassembly.

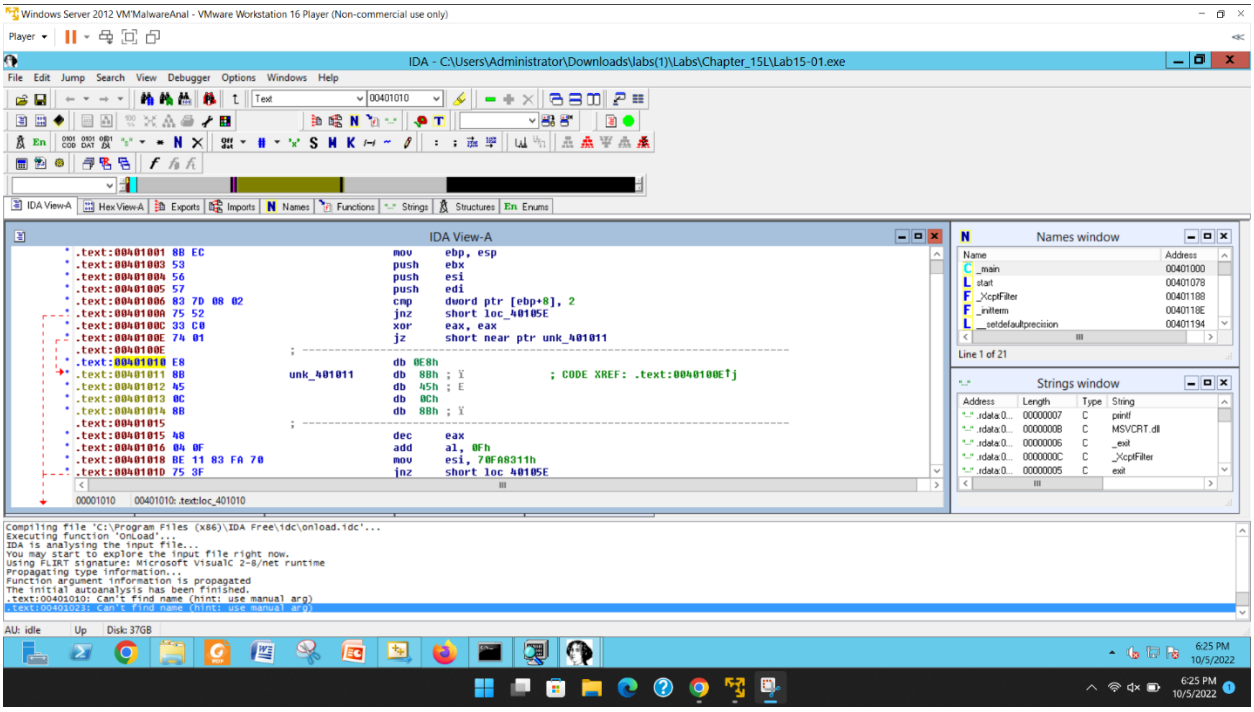
Procedure:

1. Load Lab15-01.exe into IDAPro, then choose Options, General, and then "Line preference." Enter 6 copy opcode bytes and then click OK.

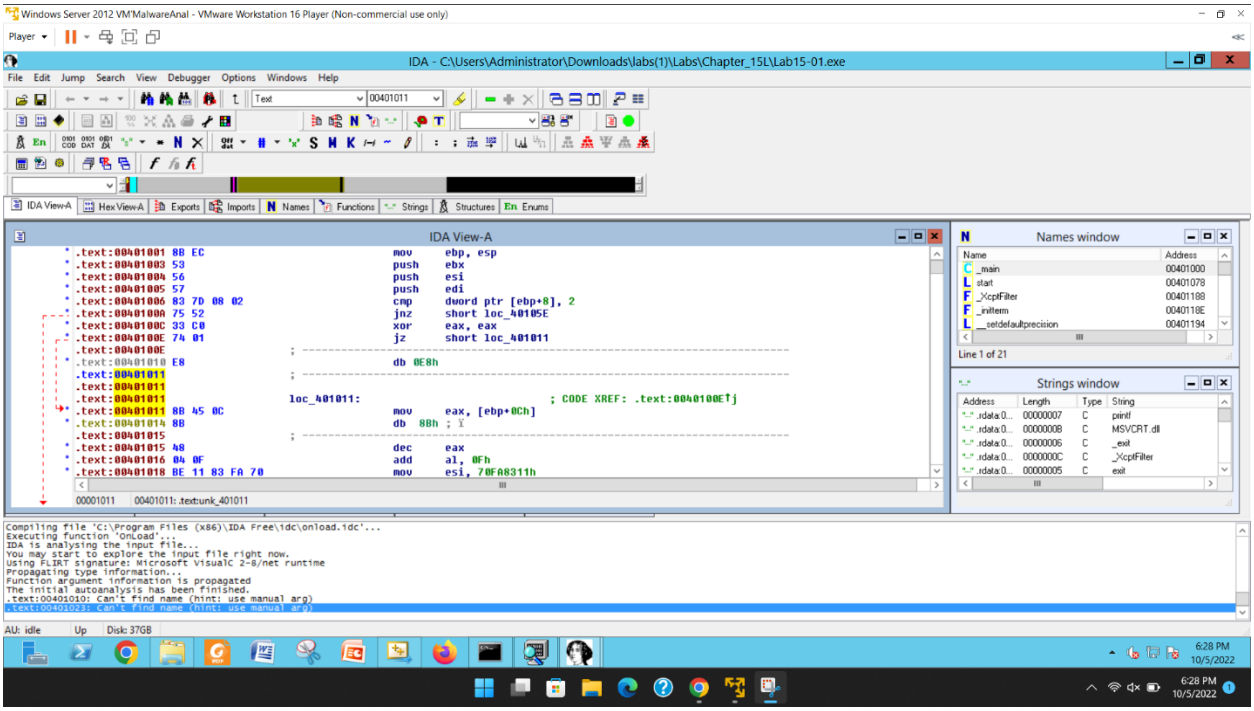




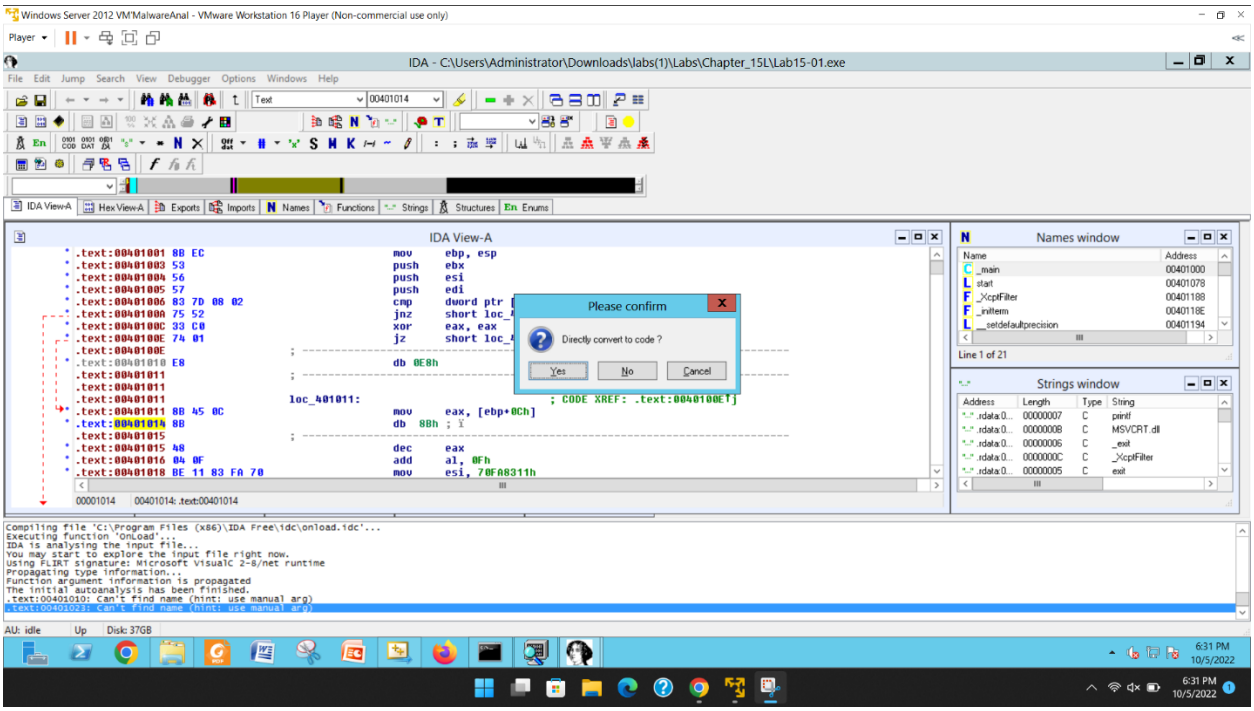
2. 401011 code is being fixed. The first issue is obvious: the instruction at location 40100E is a jz to 401010+1, or 401011. The condition is always true since the preceding instruction was xor eax, eax, hence the code will always skip over the byte at location 40100E. Click any of the 401010 addresses in the left column to adjust the disassembly. They all turn yellow at the same time. To convert these five bytes to data, hit d on the keyboard. A box with the words "Please confirm" appears. Select Yes. Following the jz instruction, the code now displays five bytes of data, as shown below.

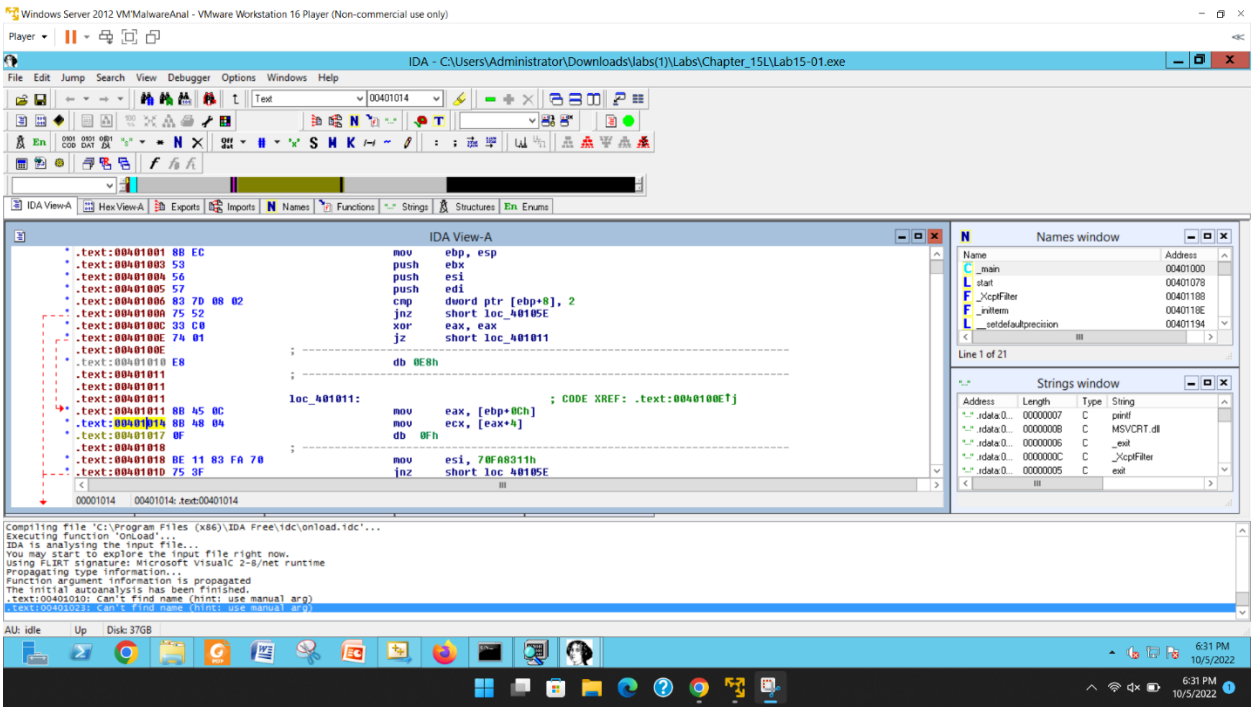


3. Because the jz restarts execution at 401011, we must instruct IDA to interpret that address as code.

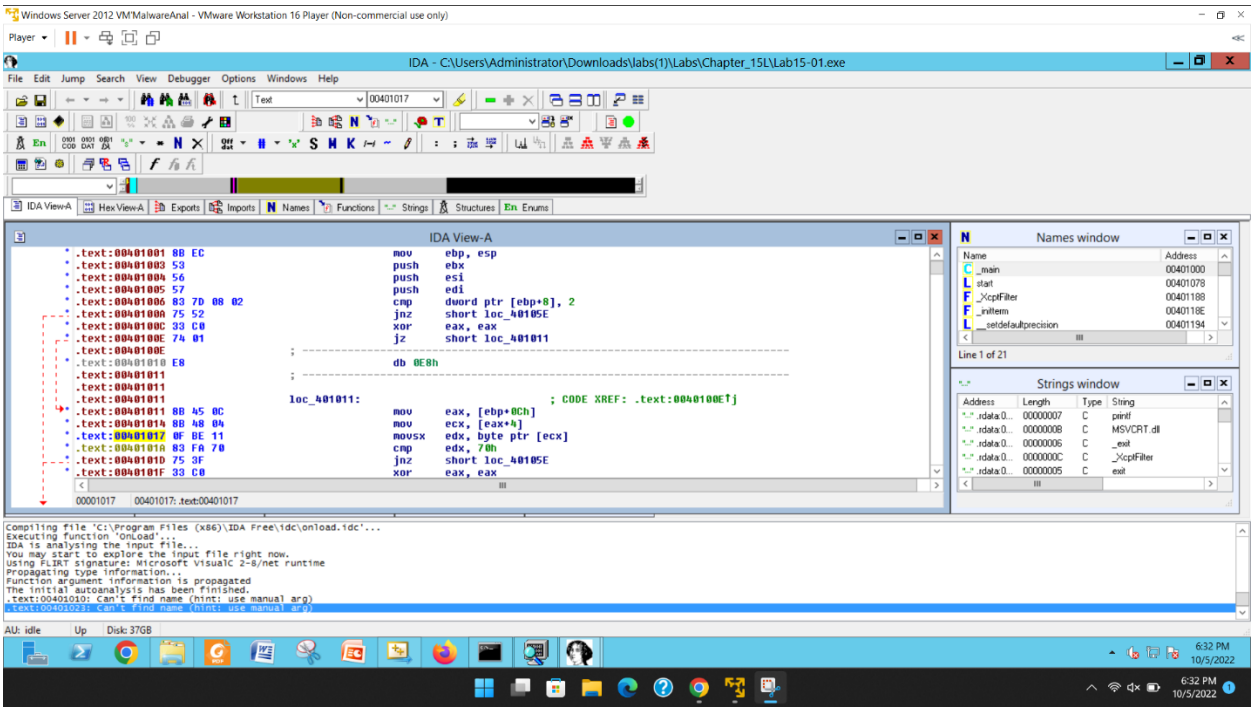


4. Click the 401011 addresses in the left column. To convert these five bytes to code, hit c on the keyboard.
5. As shown below, three bytes beginning at 401011 are now interpreted as a MOV instruction.
6. However, the byte at 401014 is still interpreted as data, rendering the disassembly below it incorrect.
7. Click the 401014 addresses in the left column. To convert these five bytes to code, hit C on the keyboard. A box with the words "Please confirm" appears. Select Yes.
8. This directs IDA to successfully disassemble another instruction while leaving another byte abandoned and interpreted as data at 401017, as illustrated below.





9. Click the 401017 addresses in the left column. To convert these five bytes to code, hit c on the keyboard. A box with the words "Please confirm" appears. Select Yes.
10. Finally, as shown below, IDA displays the right code, with no stray "db" bytes in the center.



11. Use the same method for 401023, 401024, 401027, 40102A, 40102E, 401031, 401033, 401037, 401038, 40104C, and 401063.