Project Reversus

Prerequisites:

- Basic programming skills in any high-level language (C++, Python, etc.)
- A little bit of Assembly language skills
- Basic Operating System knowledge

Introduction:

Welcome to Project Reversus, where you will learn how to reverse engineer small programs. The process of reverse engineering is about learning how software works without having the source code on hand. There are two types of analysis done on programs for this process which are static and dynamic. Static analysis is analyzing software without actually executing the program. On the other hand, dynamic analysis is about analyzing software while the program is being executed. Most of the challenges in this module can be solved with just static analysis since running the program isn't a requirement. When doing dynamic analysis, other tools are also employed to capture network traffic or process activity as a program may modify something or call out to the Internet.

During this workout, you will log onto a Windows server with the programs to reverse engineer named *crackthis.* Starting with the first one, the difficulty will increase up to the sixth *crackthis.* You will use tools like IDA or Ollydbg to disassemble each program to find the password that will validate it.

Logging onto your computer:

- Use the generated username and password to login through Guacamole

Your mission:

- Once you are logged in, you should see a folder that says Reverse Engineering.
- After clicking on Reverse Engineering, click on another folder that says Crackthis.
- Here, you will be presented with six different binary files.
- Some will not open up by just double-clicking and may require the command line to open it. There is a provided CMD shortcut in the folder.
- **With reversing tools and techniques, find the hidden codes that will validate each program.**