Reverse Engineering

Reverse Engineering

[Task 1] Debugging and File Permission

In this task, we'll be learning the basics of reverse engineering and assembly. Here are some important things to do before starting the task:
• These files have been compiled with the lowest level of optimisation on Unix based machines and are intended to be run on Linux/Mac.
• Make sure you set up a debugger - it would be good to get comfortable with radare2 which can be downloaded from here. You can also use other debuggers like gdb, which come installed in most Unix based operating systems.
• When these files have been downloaded, change the permissions of these files using the command **chmod +x filename**
These tasks will make use of crackme files. The objective of these files is to understand the assembly code to uncover the right password for the file.
Here are some of the important things you will learn in this course:
• If statements in assembly
• Loops in assembly
• standard function calls in assembly
• Calling Convention in assembly

click me	click me
#1	Set up debugger(if you haven't already)

No answer needed

[Task 2] crackme1

This first crackme file will give you an introduction to if statements and basic function calling in assembly.

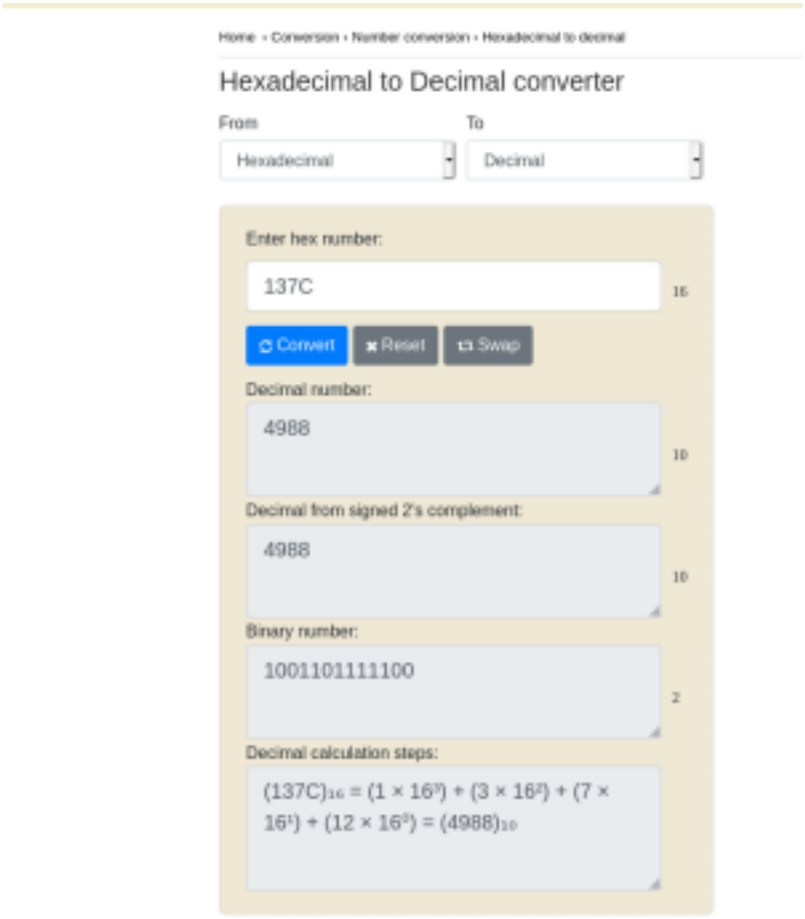
click me	click me
#1	what is the correct password

hax0r

[Task 3] crackme2

This is the second crackme file - Unlike the second file, this will involve examining registers, how and where values are compared

click me	click me
#1	What is the correct password?



137C hex == 4988 dec

4988

[Task 4] crackme3

This crackme will be significantly more challenging - it involves learning how loops work, and how they are represented in assembly

click me	click me
#1	What are the first 3 letters of the correct pass

azt