Faculty of Science and Engineering

Coursework – 2020/21 Academic Year

Module Code:

Module Title: hitecture and Low Level Programming

Module Leader: Dr Vasilios Kelefouras

School: Computing, Electronics and Mathematics

SEC204 Coursework Assignment 2020/2021

Assignment Brief

Assignment Title:	omputer Architecture and Low Level rogramming - Coursework
Submission Deac	lease check the DLE submission point
Submission:	nline (DLE)
Contribution to M	0%
Individual/Group Assignment:	Individual
Module:	SEC204
Module Leader:	Dr Vasilios Kelefouras

Overview

This is individual coursework consisting of two tasks:

1. Hack/Unlock the '*Tower of Hanoi*' game provided. To this end, you must reverse engineer the binary code provided, analyse what it does and extract the appropriate information [70%].
2. Write a small report describing methods and techniques to mitigate against buffer overflow security exploit [30%].

Learning Outcomes

The above parts cover the following **module learning outcomes**, as described into the module record on the DLE:

1. Interpret and manipulate assembly code via hardware debugging techniques
2. Apply reverse engineering techniques to identify main software flaws
3. Identify relevant countermeasures for main software flaws.

Part 1: Hack the *Tower of Hanoi* game

You are provided with a x86-64 binary file. This binary file runs only on Linux and contains a well-known mathematical game / puzzle called '*The Tower of Hanoi'.* The game is developed in the C programming language. To play the game you need to type a valid username and password which are unknown. Your task is to extract the username and password and thus unlock the game. To this end, you must reverse engineer the binary file provided, analyse what it does and extract the information needed. You are expected to use '*gdb*' debugger, as you did in the '*bomb*' lab session.

This is an individual coursework. ***Collaboration with other students will be considered as plagiarism and*** you may be required to attend a verbal examination on request of the module leader.

The marking criteri[illegible] (please see the rubric table below):

1. Extract the [illegible]fy the procedure followed. Provide the gdb commands [illegible] username. [20 marks]
2. Extract the [illegible]fy the procedure followed. Provide the gdb commands [illegible] username. [25 marks]
3. Unlock the [illegible]ame. Justify the procedure followed. Provide the gdb commands used to extract the username. [25 marks]
   Hint #1*: the input is stored as an array of characters (1 byte each).*
   Hint #2*: In the beginning of encrypt_phase2(), the values $0x236b6f23 and $0x236b23 refer to ASCII characters* [30 marks]

Marking criteria				
Question.1 marks	**0 marks**	**0-7**	**8-14**	**15-20**
Question.1 marking criteria	Nothing or **just the username** is provided.	**One of the following occurs:** **A.** The student has not solved the problem but has reached to a point close to the solution. The username is not provided, but the procedure followed to this point has been appropriately justified and explained; all the gdb commands needed to reach this point are provided. The student has justified why each step/action has been followed. **B.** Username is provided but the procedure followed is poorly justified and explained **C.** Username is provided but the student has not provided all the gdb commands needed.	The username and all the gdb commands needed, are provided. The student has appropriately justified the procedure followed. The student has appropriately explained why each step/action has been followed. The student has **not** accurately explained the functionality of the critical functions.	The username and all the gdb commands needed, are provided. The student has appropriately justified the procedure followed. The student has appropriately explained why each step/action has been followed. The student has appropriately reversed engineer the critical functions and has accurately explained their functionality.
Question.2 marks	**0 marks**	**0-8**	**9-15**	**16-25**
Question.2 marking criteria	Same as above	Same as above	Same as above	Same as above
Question.3 marks	**0 marks**	**0-8**	**9-15**	**16-25**
Question.3 marking criteria	Same as above	Same as above	Same as above	Same as above

Part 2: Write a small report describing methods and techniques to mitigate against buffer overflow security exploit.

Write a small report (less than 700 words) describing methods and techniques to mitigate against bu[illegible]rity exploit. [30 marks]

- Desc[illegible]s to defend against the attack. [15 marks]
- Disc[illegible]ness of existing detection and prevention mech[illegible]s the extent to which relevant secure software deve[illegible]ogies could help alleviate the problem [15 marks].

Submission Details and Deadlines

You must submit a *‘.decx’* file for part1 and part2.

Feedback and marks will be returned within 20 working days. Note that you may be required to attend a verbal examination on request of the module leader. This can be face to face or using Skype for Business. The purpose of this might be to clarify any issues to facilitate assessment, as well as to help verify this is your own work. Failure to participate may have a detrimental impact on your final grade.