Student name: Adam Logan Student Number: 40293585

CSC3063 – Secure Software Development

Lab 1 feedback (30%)

First some generic feedback:

Most submissions were of really good quality. However, there is room for improvement all around. One major problem was the language used in the report. Many wrote this in first person or with some informal language being used. You need to treat these as formal documents, so you need to try to write in 3rd person form (Instead of 'I ran some tests' you can use 'some tests were run', etc) and try to keep the language neutral, stating facts rather than opinions. Also remember your audience is technically proficient, so avoid vagueness, ambiguity and statements that provide no clear information. You can assume the reader understands how the stack works, what is a buffer overflow, etc.

Arguments given to demonstrate knowledge and understanding of the main issues were overall appropriate and well delivered in the mitigation section, but you should be able to provide more references to back up your arguments in your "bibliography". The average number of references used is very low. You need to step it up.

One common failure is not recognizing that the most effective mitigation strategy against a buffer overflow is input sanitation. Several students recommended instead the use of an interpreted language as the best solution, quoting their inherent immunity to buffer overflows, but failed to recognize that these may not always be the most appropriate platform to write a solution for the project. When reporting these mitigation strategies, you should always address the most robust and cost-effective strategy first, which in this case is really input sanitation, following then with the more generic options such as secure functions, PIE, ASLR and Canaries. You should also emphasise the benefits of an in-depth security approach, including adding checks during the development life cycle and embedding security from the early stages of the project.

When choosing a secure function to use, you need to make sure you do not introduce another vulnerability in the code. For example, using strcpy_s is an option, but you need to guarantee that the resulting string complies to the standards that C uses. In C, strings must be terminated in Null character. Using strcpy_s without care, may result in a string that is not Null terminated, which can lead to a Buffer Overread vulnerability.

Finally, a better effort needs to be made when citing and referencing material (including that coming from the notes). Choose a style and use it appropriately.

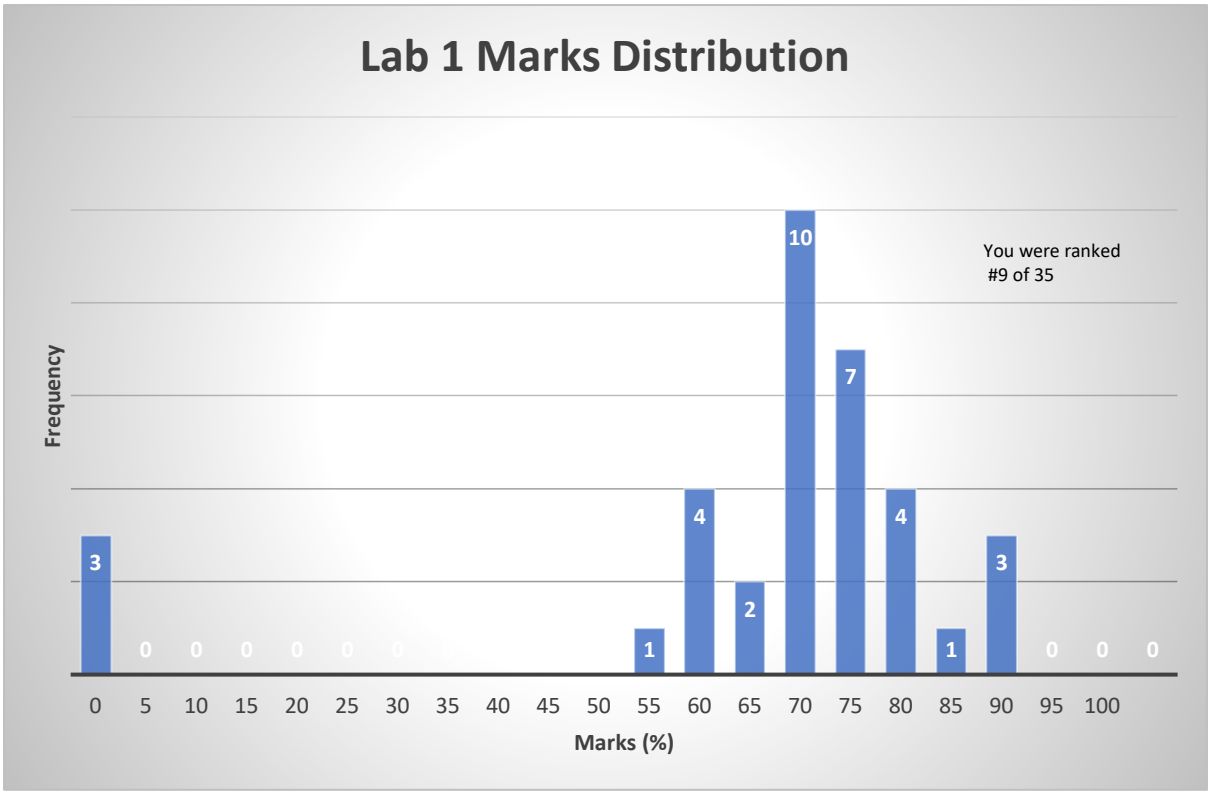
In the next page you will find the feedback specific for your work:

Student name: Adam Logan Student Number: 40293585

Adam Logan

Feedback	*Assessment Mark*
Followed format requested. Good Executive Summary. Very Good Introduction. Very clear explanation of the methodology used, confidence in the approach supported by relevant evidence and resources. Very good actions taken, showing well-developed arguments and evidencing wide use of learning resources.. Very good approach to mitigation of the vulnerability showing knowledge and understanding of the main issues involved and their relevance. A good level of independence of thought and critical judgement and a level of critical analysis. Excellent referencing using a consistent strategy for reporting and citing sources. Excellent work. You typically want to know what security (if any) is enabled on the file before selecting your approach to pentesting, so your checksec should have been the very first thing to run on the executable. Your recomendations are sound, but you should recognise that it is not the function itself that is insecure, it is the usage of the function in conjunction with a failure to sanitise user input which is also partially responsible for a buffer overflow attack being successful. You do recognise in your recommendations that it is not always possible to change the chosen development language. You could emphasise more the "defence in depth" mentality, and recommend implementation of as many deterrants as possible. The overall level of work in this task is very good - very well done.	**75%**

For reference, here's the mark distribution for this assignment:



Student name: Adam Logan Student Number: 40293585