Faculty of Computing Engineering and Science Assessment Brief

Module Title: Secure Software Development
Module Code: CS2S562
Module Leader/Tutor: Dr Janusz Kulon / Alun King
Assessment Type: Report
Assessment Title: Report (CW) 1
Weighting: 40%
Submission Date: 02/05/2025 (by 23:59)
Return Date: 20 Working days from the submission date.

Assessment Description

Your task

Download three zipped Visual Studio program projects from the 'Learning Materials' page of the CS2S562 module on Blackboard. Validate all three programs.

According to IEEE-STD-610 validation is defined as: 'The process of evaluating software during or at the end of the development process to determine whether it satisfies specified requirements.'

In this coursework you validate at the end of the development process. The specified requirements are those listed under non-functional requirements for the application, namely:

- it must be implemented by applying best practice coding procedures
- it must be designed using Secure Design Principles and Patterns
- In real life there would of course be validations against all other requirements as well but in this coursework we focus on secure coding aspect only.

For validation use the 'Validation Report Guide' that is also provided on the 'Learning Materials' page on Blackboard.

Submission instructions

1. Check the marking grid. Does your validation 'tick all the boxes' (ideally in the 1st Class / Distinction column)?
2. Name your Validation Report file like this: `CS2S562_CW1_EnrolmentNumber_FirstName_LastName.doc` (or docx, odt, or pdf) (Replace the placeholders above with your enrolment number, first and last name respectively)
3. Go to the Blackboard pages of the module, select the 'Assessment' link and use the upload facility there to submit the file to Blackboard.
4. After uploading the system should show you a receipt screen. If that is not the case or in case of problems e-mail alun.king1@southwales.ac.uk or j.kulon@southwales.ac.uk immediately.

Guidance on Format of Assessment

Note: Students are reminded **not** to include this assignment brief with the assignment submission

Learning Outcomes Assessed

(as specified in the validated module descriptor https://icis.southwales.ac.uk/):
1) To be able to describe the integration of security into the software development life-cycle and reflect on best practice in minimising code vulnerabilities.
2) To be able to apply principles of protection mechanisms, software security and secure design.
3) To be able to conduct static and dynamic security verification and assessment of a software application.

Marking Criteria/Rubric
Note: All grades are provisional until they are ratified by the exam board
The rubric is provided at the end of this assessment.

What happens next?

Your marked assessment should be available 20 working days after submission. However, please be advised that this may be subject to change in the event of Bank Holidays, University Closure or staff sickness. If there is something about the feedback you have been given that you are unclear about, please see your module tutor.

Feedback Method

Feedback will be provided through Blackboard.

Late Submission

Refer University Policies and Procedures about Late Submissions:
https://advice.southwales.ac.uk/a2z/assessment-submission/

Retrieval in the Event of Failure

Standard university policy will be applied.
IYR is available for this assessment.

Extenuating Circumstances

https://advice.southwales.ac.uk/a2z/extenuating-circumstances

Referencing, Plagiarism and Good Academic Practice

https://advice.southwales.ac.uk/a2z/referencing-plagiarism-and-good-academic-practice

Learning Support Resources

https://studyskills.southwales.ac.uk

Student Checklist

1. Check the marking grid. Does your validation 'tick all the boxes' (ideally in the 1st Class / Distinction column)?
2. Name your Validation Report file like this: `CS2S562_CW1_EnrolmentNumber_FirstName_LastName.doc` (or docx, odt, or pdf) (Replace the placeholders above with your enrolment number, first and last name respectively)
3. Go to the Blackboard pages of the module, select the 'Assessment' link and use the upload facility there to submit the file to Blackboard.
4. After uploading the system should show you a receipt screen. If that is not the case or in case of problems e-mail alun.king1@southwales.ac.uk or j.kulon@southwales.ac.uk immediately.

Marking Scheme:

	Fail (0%-29%)	Narrow Fail (30%-39%)	3rd Class / Pass (40%-49%)	Lower 2nd Class / Pass (50%-59%)	Upper 2nd Class / Merit (60%-69%)	1st Class / Distinction (70%-100%)
Integer vulnerability validation 10%	☐ Very poor Integer vulnerability validation	☐ Poor Integer vulnerability validation	☐ Satisfactory Integer vulnerability validation	☐ Good Integer vulnerability validation	☐ Very good Integer vulnerability validation	☐ Excellent Integer vulnerability validation
String vulnerability validation 10%	☐ Very poor String vulnerability validation	☐ Poor String vulnerability validation	☐ Satisfactory String vulnerability validation	☐ Good String vulnerability validation	☐ Very good String vulnerability validation	☐ Excellent String vulnerability validation
Memory vulnerability validation 10%	☐ Very poor Memory vulnerability validation	☐ Poor Memory vulnerability validation	☐ Satisfactory Memory vulnerability validation	☐ Good Memory vulnerability validation	☐ Very good Memory vulnerability validation	☐ Excellent Memory vulnerability validation
Formatted IO vulnerability validation 10%	☐ Very poor Formatted IO vulnerability validation	☐ Poor Formatted IO vulnerability validation	☐ Satisfactory Formatted IO vulnerability validation	☐ Good Formatted IO vulnerability validation	☐ Very good Formatted IO vulnerability validation	☐ Excellent Formatted IO vulnerability validation
File IO vulnerability validation 10%	☐ Very poor File IO vulnerability validation	☐ Poor File IO vulnerability validation	☐ Satisfactory File IO vulnerability validation	☐ Good File IO vulnerability validation	☐ Very good File IO vulnerability validation	☐ Excellent File IO vulnerability validation
Pointer vulnerability validation 10%	☐ Very poor Pointer vulnerability validation	☐ Poor Pointer vulnerability validation	☐ Satisfactory Pointer vulnerability validation	☐ Good Pointer vulnerability validation	☐ Very good Pointer vulnerability validation	☐ Excellent Pointer vulnerability validation
Automated Tool usage validation 10%	☐ Very poor Automated Tool usage validation	☐ Poor Automated Tool usage validation	☐ Satisfactory Automated Tool usage validation	☐ Good Automated Tool usage validation	☐ Very good Automated Tool usage validation	☐ Excellent Automated Tool usage validation
Secure Pattern 1 15%	☐ Very poor Secure Pattern 1	☐ Poor Secure Pattern 1	☐ Satisfactory Secure Pattern 1	☐ Good Secure Pattern 1	☐ Very good Secure Pattern 1	☐ Excellent Secure Pattern 1
Secure Pattern 2 15%	☐ Very poor Secure Pattern 2	☐ Poor Secure Pattern 2	☐ Satisfactory Secure Pattern 2	☐ Good Secure Pattern 2	☐ Very good Secure Pattern 2	☐ Excellent Secure Pattern 2