The University of Melbourne

SWEN90010: High Integrity Software Engineering

Assignment 3

Due Date: 11:59pm, Sunday 17 May, 2015

1 Introduction

The assignment is worth 20% of your total mark and is done in pairs (the same pairs as assignment 2).

The aim of this assignment is to specify a formal model in Alloy of (part of) the ICD system, and to produce a fault-tolerant design for this. The assignment evaluates your ability to apply specification and design techniques to engineer a security- and safety-critical system.

2 Your tasks

The tasks for the assignment are listed below:

1. The file `ICD.als` (available on the LMS) is an Alloy model describing the roles and ICD state from Assignment 1. This is intended to allow you to describe the access control permissions for ICDs. In this assignment you will work through the intended access control rules, using Alloy to model your rules and detect whether they have the implications you want them to have. The model includes the state of a person's ICD, along with some roles for people who may change or read the state of that ICD.

   In addition, there are declarations for four predicates corresponding the operations that can be performed in the *off mode*, and have security requirements related to them. However, the bodies of these predicates are empty.

   Your sub-tasks are:

   (a) **Modelling explicit permissions in Alloy (6 marks)**. Reread Assignment 1 and fill in the empty predicates with appropriate bodies.

   (b) **Thinking about implicit security facts (4 marks)** Think at a high level about what this sort of system should be trying to achieve. In other words, try to think of some security requirements that are NOT explicitly stated in Assignment 1 but which, by commonsense, you think should be true. (Hint: What happens when a cardiologist is admitted to hospital?) Write and check at least one assertion relating to this, and one more assertion (e.g. one from assignment 1). Did they check out the way you expected?

   (c) **Updating the model for implicit constraints (2 marks)** Update your model so that your assertions hold in the way that you intended. You can either modify your predicates or add facts to constrain the model, but you can NOT modify the existing signatures or the inputs/outputs to the predicates.

   Write a brief report of no more than a page describing why you chose the assertions you did, what instances you found that weren't supposed to be there (or what you

didn't find that was supposed to be there), and how you updated the model. Any text that is more than a page will not be marked.

2. In assignment 2, you derived a hazard log. This task asks you to consider how to mitigate *some* of those hazards using fault-tolerant design principles. Your sub-tasks are:

   (a) **Designing a fault-tolerant architecture and algorithms (5 marks)**. Applying fault-tolerant techniques from lectures, modify the ICD architecture from assignment 1 (outlined in Figure 2 in the assignment 1 handout) to mitigate those hazards that you believe can be mitigated using fault-tolerant techniques.

   Show your design as a high-level architecture (at a similar level of detail to that in assignment 1) and describe the (new) algorithms used for achieving fault tolerance in the design.

   (b) **Justifying your design (2 marks)**. Justify the decisions that you have made in your design. In particular, link each decision back to a hazard or set of hazards from your hazard log, and to the relevant theory in the notes.

   (c) **Updating the requirements (1 mark)**. Briefly list any new system requirements (added to those from assignments 1 and 2) that must be added to handle the new fault tolerant design.

3 Criteria

Criterion	Description	Marks
Alloy model [12 marks]		
Predicate models	The predicates are correct and complete. The appropriate level of abstraction has been used. The solution is clear and succinct.	6 marks
Security implications	At least two valid and sensible implications have been designed and modelled.	4 marks
Updated model	The updated model correctly and completely addresses the identified security implications.	2 marks
Fault-tolerant design [8 marks]		
Architecture	A correct architecture has been chosen and designed.	3 marks
Algorithms	The correct algorithms have been chosen for the design.	2 mark
Justification	The justification is sound and clear. All fault-tolerant aspects have been justified.	2 mark
Requirements	The fault-tolerant aspects of the design are sensibly addressed by the new requirements.	1 mark
Total		20 marks

4 Submission

Submit the assignment using the submission link on the subject LMS. Go to the SWEN90010 LMS page, select *Assignments* from the subject menu, and then select *View/Complete* from the *Assignment 3 submission* item. Following the instructions, upload:

1. An alloy file, named `FirstAttempt.als` containing your solutions to the modelling and verification sub-tasks in Task 1.
2. An alloy file, named `SecondAttempt.als` containing your *updated* solution, for which the assertions now pass.
3. A PDF file containing your one page description for Task 1, and your solution for Task 2.

Only *one* student from the pair should submit the solution, and the submission should clearly identify both authors.

Late submissions Late submissions will attract a penalty of 2 marks for every day that they are late. If you have a reason that you require an extension, email Tim *well before the due date* to discuss this.

Please note that having assignments due around the same date for other subjects is not sufficient grounds to grant an extension. It is the responsibility of individual students to ensure that, if they have a cluster of assignments due at the same time, they start some of them early to avoid a bottleneck around the due date. Starting early on this is highly encouraged.

5 Academic Misconduct

The University misconduct policy applies to this assignment. Students are encouraged to discuss the assignment topic, but all submitted work must represent the individual's understanding of the topic.

The subject staff take plagiarism very seriously. In the past, we have successfully prosecuted several students that have breached the university policy. Often this results in receiving 0 marks for the assessment, and in some cases, has resulted in failure of the subject.