Computer Science and Creative Technologies

Coursework Specification

Module Details

Module Code	UFCFYN-15-M
Module Title	Analysis and Verification of Concurrent Systems
Module Leader	Rakib Abdur
Module Tutors	Rakib Abdur
Year	2020-21
Component/Element Number	B 1
Total number of assessments for this module	One
Weighting	This coursework is worth 100 marks representing 40% of your total course grade.
Element Description	This assignment is to be completed on your own as an individual assignment.

Dates

Date issued to students	March 9, 2021
Date to be returned to students (marks/feedback)	June 4, 2021
Submission Date	May 6, 2021
Submission Place	Blackboard
Submission Time	14:00
Submission Notes	Please submit a portfolio via Blackboard before 14:00 May 6, 2021 as a ZIP file with your NuSMV code (.smv) along with a PDF design/results file, as well as you need to make a short video to run and show your encoding and results. (See Deliverable section).

Feedback

Feedback provision will be	Written feedback uploaded to Blackboard as appropriate.

Contents

Section 1: Overview of Assessment

This assignment assesses the following module learning outcomes:

- Demonstrate the application of formalisms to specify system properties using temporal logics like Linear-time Temporal Logic (LTL) and Computation Tree Logic (CTL)
- Use tools and analysis techniques to study and reason about critical properties of the concurrent systems, including security protocols

The assignment is worth 100 marks representing ***40%*** of the overall mark for the module.

The learning objective of this coursework is to design, analyse, and verify authentication protocol using the NuSMV model checker. This will give you hands-on experience with the NuSMV tool and understanding its input language for specifying systems and desired system properties.

The assignment is described in more detail in Section 2.

Assignment type: this is an **INDIVIDUAL** assignment. Do not copy and paste work from any other source or work with any other person. You are strictly forbidden from discussing your answers with another student. Text-matching software will be used on all submissions.

Working on this assignment will help you to achieve the learning outcomes mentioned above as well as you would be aware of when and how you might deploy formal verification techniques. If you have questions about this assignment, I will be happy to discuss those during our lecture/lab session.

Section 2: Task Specification

Consider the following four-step communication protocol, which is known as Kerberos protocol and its aim is to guarantee authentication and key exchange between a client and a server [1, 2].

(1) A -> S : A, B

(2) S -> A : $\{T_s, L, K_{ab}, B\}K_{as}$, $\{T_s, L, K_{ab}, A\}K_{bs}$

(3) A -> B : $\{T_s, L, K_{ab}, A\}K_{bs}$, $\{A, T_a\}\ K_{ab}$

(4) B -> A : $\{T_a+1\}K_{ab}$

The protocol involves the principals A (client/initiator) and B (server/responder), and an authentication server S. The server S is a trusted party which shares a key K_{as} with A and a key K_{bs} with B, and also responsible for generating new session keys K_{ab}. The above protocol makes use of the time stamps T_a and T_b, and the lifetime L. In step (1) above, A contacts S in order to communicate its claimed identity and the name of the server B. In step (2), S sends to A two encrypted components. The first component contains the session key K_{ab} generated by S, a time stamp T_s specifying when the session key has been generated, the interval of validity of such key, and the name of the server B. The second component is called *ticket* having similar information, however, A will not be able to decrypt it. In step (3), A forwards the *ticket* to server B, with a authenticator component encrypted with the new session key. After receiving the above message, B can extract the session key from the *ticket*, and uses it to decrypt the authenticator. If the key used to encrypt authenticator matches with the key contained in the ticket, the server B can assume that the authenticator was generated by A. At this point, in order to authenticate the client A, the server B must also check the time stamp T*a* to make sure that the authenticator is recent. Thus B can recognise A if the result of verification is positive. In step (4), B demonstrate its identity to A sending a message with increased time stamp encrypted with the session key K_{ab}.

The model of the above protocol could be composed of several variables and processes (or agents). The protocol shall ensure authentication and secrecy. Such properties shall be verified against an intruder I with the following capabilities:

- I is a known agent, it can act either as initiator or as responder of a protocol session;
- I can eavesdrop and store any message sent by any agent;
- I can exploit its knowledge to generate new messages or use previously stored messages as they are.

Model as a concurrent system in NuSMV the protocol described above as the interaction of 4 agents, A, B, S and I. Multiple sessions may overlap, asynchronously. However, to ensure finiteness of the model state space, consider a maximum number n of sessions, and verify the satisfaction of the properties above under such a limitation. To model and verify the desired system properties you need to complete the following tasks:

Task 1: Design and draw a state transition diagram of the system considering four agents mentioned above. Please note that this diagram would be a high-level diagram and not considering NuSMV's low level diagram (as it would be very complicated); (**20 marks**)

Task 2: In your NuSMV model (code using the SMV language) all the agents should work concurrently, and in an asynchronous manner; (**30 marks**)

Task 3: Identify and express six authentication and secrecy properties using CTL/LTL; (**6*5=30 marks**)

Task 4: Demonstrate, verify all the properties identified above by running your code. (**20 marks**)

You may get some idea on how to write authentication and secrecy properties from the following paper.

M. Panti , L. Spalazzi , S. Tacconi. Using the NuSMV Model Checker to verify the Kerberos Protocol (2002) http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.21.8906

Section 3: Deliverables

Please submit a portfolio via Blackboard on or before May 6, 2021@14:00, as a ZIP file with your NuSMV code (.smv) along with a PDF file containing: (i) system design and state transition diagram, and (ii) video recording presentation of verification results showing the execution logs. The logs should clearly show verified properties as "true" or "false". In case of "false" it must show a counter example.

You can create your video in many ways:

simply using your smart phone

https://atomisystems.com/screencasting/record-screen-windows-10/

http://fetliu.net/blog/new-aug-19-kaltura-submissions-what-do-i-need-to-tell-my-students/

Or any other way you may find easier

Your video recording:

(i) a record of your presentation that displays the screen on which you run your code;

(ii) you should be running through all the properties to demonstrate the how your code works (assuming that your tutor is watching the presentation);

(iii) since presentation timing is an important factor, detailed code explanation would not be possible. However, you should go through the code lightly after presenting/showing/running all the properties. Later on, your tutors may look at your code if necessary;

(iv) your voice will be enough (if you don't want to show your face);

(v) your presentation should not be too short or too long, ideally 10/15(max) minutes;

(vi) do not leave submission to the very last minute. Always allow time in case of technical issues and uploading time.

Section 4: Marking Criteria

	0-29	30-39	40-49	50-59	60-69	70-84	85-100	Mark & Advice for Improvement
Design and state transition diagram	Little or no diagram provided	Partial diagram provided but arcs and/or nodes are only partially correct, and description of security threats are not considered when designing a (security) solution/system (overall less than 40% complete)	Partial diagram provided, arcs and/or nodes are correct, but description of security threats are not considered when designing a (security) solution/system (overall less than 50% complete)	Almost complete diagram provided, arcs and/or nodes are only partially correct, but description of security threats are considered when designing a (security) solution/system (overall less than 60% complete)	Almost complete diagram provided, arcs and/or nodes are correct, and description of security threats are considered when designing a (security) solution/system (overall less than 70% complete)	Complete diagram provided, arcs and/or nodes are partially correct, and description of security threats are considered when designing a (security) solution/system (overall less than 85% complete)	Complete diagram provided, arcs and nodes are correct, and description of security threats are considered when designing a (security) solution/system (100% complete)	
NuSMV model (code in SMV language)	Little or no implementation provided	Partial implementation, variable declaration	Partial implementation, variable declaration	Complete implementation, but variable declaration	Complete implementation, variable declaration	Complete implementation, variable declaration	Complete implementation, excellent model with no	

		and their use are not correct (overall less than 40% complete)	and their use are correct but model does not fulfil required features (overall less than 50% complete)	and their use are not fully correct, so model does not fulfil required features (overall less than 60% complete)	and their use are correct, but the interleaved execution of processes gives minor error that can be fixed easily	and their use are correct, and the interleaved execution of processes work perfectly.	redundant variable declaration and their use are correct, and the interleaved execution of processes work perfectly.	
Authentication and secrecy properties using CTL/LTL	For each property identified: CTL/LTL expression incorrect as well as the property is irrelevant 0 mark CTL/LTL expression is correct, but the property is irrelevant 50% marks CTL/LTL expression is incorrect, but the property is relevant 50% marks CTL/LTL expression is correct, and the property is relevant 100% marks							
Verification result	For each correct and relevant CTL/LTL property verified: 100% marks							
Video presentation	Absent 0 mark Inadequate (barely able to explain the analysis/design and/or work done) 6 marks Good (good explanation of analysis/design and/or work done) 12 marks Very Good (very good explanation of analysis/design and/or work done) 16 marks Excellent (excellent explanation of analysis/design and/or work done) 20 marks							

Section 5: Feedback mechanisms

Written feedback uploaded to Blackboard as appropriate.

This page is intentionally left blank