FIT3056
Secure and trusted software systems

Unit Guide

Semester 2, 2009

Last updated : 15 Jul 2009

Table of Contents

FIT3056 Secure and trusted software systems - Semester 2, 2009

Chief Examiner:

Dr Phu Le
Fax: +61 3 9903 1247

Contact hours: 11AM - !3PM - Friday

Lecturer(s) / Leader(s):

Caulfield

Dr Phu Le
Fax: +61 3 9903 1247

Additional communication information:

Dr. Phu Dung Le

Phone: +61 3 9903 23 99

Office: Faculty of IT - Monash university

900 Dandenong Rd, Caulfield East Vic 3145, Australia

H.706

Introduction

Welcome to Secure and Trusted Software (FIT3056)!

This unit will be a core unit in the Security major of BITS. It can be an elective unit for students who do not major in security (subject to the approval from the school). This unit will provide students with the knowledge and experience of identifying software vulnerabilities, principles for constructing secure and trusted software, basic software security testing and verification.

Unit synopsis

Students are introduced to some of the most common security issues involved in the development of software, including secure coding practices, secure database access, secure data communications, security of web applications, use of encryption techniques and security testing. Students are provided with a range of practical exercises to reinforce their skills, including authenticating and authorizing users programmatically, user input validation, developing secure web, mobile/wireless and database applications, encrypting and hashing data programmatically, generating digital signatures programmatically, security testing, designing logging and auditing mechanisms.

Learning outcomes

At the completion of this unit students should have knowledge of the main security concepts and issues involved in the development of software, including:

1. Software security versus other aspects of computer security;
2. Goals of secure and trusted software;
3. Vulnerabilities versus threats;
4. Best software development principles and practices;
5. Buffer overflows;
6. Security of programming platforms;
7. Authentication and authorisation;
8. Principle of least privilege;
9. Security features are not equal to secure features; 10. Secure use of encryption;
10. User input validation;
11. Reliable software components;
12. Data privacy;
13. Auditing and logging;
14. Security testing.

At the completion of this unit students will acquire an understanding and appreciation of:

1. the importance of developing secure software in today's electronic world;
2. They will also learn that security features are not equal to secure features.

In developing secure and trusted software, students will be able to:

1. Design applications with security in mind
2. Validate user input;
3. Implement secure authentication mechanisms;
4. Authorise user's access to various protected resources;

5. Encrypt files and hash passwords;
6. Store session data securely in web applications;
7. Perform secure database access;
8. Set up secure transfer of data;
9. Create security logs;
10. Test software for security vulnerabilities.

Contact hours

2 hour lecture/week, 2 hour tutorial/week

Workload

two-hour lecture and two-hour tutorial (or laboratory) (requiring preparation in advance) a minimum of 4 hours of personal study per one hour of contact time in order to satisfy the reading and assignment expectations. You will need to allocate up to 8 hours per week in several weeks, for use of a computer, including time for group and individual assignments.

Unit relationships

Prerequisites

FIT1019, FIT1002

Prohibitions

CSE3207 (Translation for CSE3207)

Relationships

FIT3105 is a core unit of the Security major of BITS.

Teaching and learning method

Timetable information

For information on timetabling for on-campus classes please refer to MUTTS, http://mutts.monash.edu.au/MUTTS/

Tutorial allocation

On-campus students should register for tutorials/laboratories using the Allocate+ system:
http://allocate.cc.monash.edu.au/

Unit Schedule

Week	Topic	Key dates
1	Introduction to software design and implementation	
2	Principles of secure software design and implementation	
3	Principles of secure software design and implementation (continued)	
4	Computer system software problems and solutions	
5	Cryptography and secure software applications	
6	Cryptography and computer software practice (continued)	
7	Large computer software and security	
8	Concurrent programming and software security	assignment 1 due 4PM - Friday
9	Building secure distributed applications	
10	Secure software testing and verification	
Mid semester break		
11	Secure software testing and verification (continued)	
12	Research in secure software design and implementation	assignment 2 due 4PM - Friday
13	Revision	

Unit Resources

Prescribed text(s) and readings

Howard, M. and LeBlank, D. (2002), "Writing secure code", 2nd edn, Microsoft Press, Redmond. Online references supplied on the unit website Text books are available from the Monash University Book Shops. Availability from other suppliers cannot be assured. The Bookshop orders texts in specifically for this unit. You are advised to purchase your text book early.

Recommended text(s) and readings

http://www.windowsecurity.com/articles/Analysis_of_Buffer_Overflow_Attacks.htmlhttp://www.isoc.org/isoc/conferences/n (Testing C programs for vulnerabilities)http://www.cgisecurity.com/lib/sips.html (Perl scripts and security issues)http://www.mirrors.wiretapped.net/security/info/reference/nist/special-publications/sp-800-8.txt (SQL and security issues)

Equipment and consumables required or provided

Students studying off-campus are required to have the minimum system configuration specified by the Faculty as a condition of accepting admission, and regular Internet access. On-campus students, and those studying at supported study locations may use the facilities available in the computing labs. Information about computer use for students is available from the ITS Student Resource Guide in the Monash University Handbook. You will need to allocate up to **n** hours per week for use of a computer, including time for newsgroups/discussion groups.

Study resources

Study resources we will provide for your study are:

Assessment

Overview

Examination (3 hours): 60%
Assignments: 40%

Faculty assessment policy

To pass a unit which includes an examination as part of the assessment a student must obtain:

- 40% or more in the unit's examination, and
- 40% or more in the unit's total non-examination assessment, and
- an overall unit mark of 50% or more.

If a student does not achieve 40% or more in the unit examination or the unit non-examination total assessment, and the total mark for the unit is greater than 44% then a mark of no greater than 44-N will be recorded for the unit.

Assignment tasks

Assignment coversheets

Assignment coversheets are available via "Student Forms" on the Faculty website:
http://www.infotech.monash.edu.au/resources/student/forms/
You MUST submit a completed coversheet with all assignments, ensuring that the plagiarism declaration section is signed.

Assignment submission and return procedures, and assessment criteria will be specified with each assignment.

- **Assignment task 1**

  Title:
  assignment 1
  Description:
  Identify software design and implementation vulnerabilities and propose solutions.
  Weighting:
  20%
  Due date:
  4PM - Friday - Week 8

- **Assignment task 2**

  Title:
  assignment 2
  Description:
  Design and implementation secure applications using cryptography.
  Weighting:
  20%
  Due date:
  4PM - Friday - Week 12

Examination

- **Weighting:** 60%
  Length: 3 hours
  Type (open/closed book): Closed book

See Appendix for End of semester special consideration / deferred exams process.

Due dates and extensions

Please make every effort to submit work by the due dates. It is your responsibility to structure your study program around assignment deadlines, family, work and other commitments. Factors such as normal work pressures, vacations, etc. are not regarded as appropriate reasons for granting extensions. Students are advised to NOT assume that granting of an extension is a matter of course.

Students requesting an extension for any assessment during semester (eg. Assignments, tests or presentations) are required to submit a Special Consideration application form (in-semester exam/assessment task), along with original copies of supporting documentation, directly to their lecturer within two working days before the assessment submission deadline. Lecturers will provide specific outcomes directly to students via email within 2 working days. The lecturer reserves the right to refuse late applications.

A copy of the email or other written communication of an extension must be attached to the assignment submission.

Refer to the Faculty Special consideration webpage or further details and to access application forms:
http://www.infotech.monash.edu.au/resources/student/equity/special-consideration.html

Late assignment

Assignments received after the due date will be subject to a penalty of 10% for one day late, 20% for two days late, 40% for three days late, 80% for four days late and 100% for five or more days late.

Return dates

Students can expect assignments to be returned within two weeks of the submission date or after receipt, whichever is later.

Appendix

Please visit the following URL: http://www.infotech.monash.edu.au/units/appendix.html for further information about:

- Continuous improvement
- Unit evaluations
- Communication, participation and feedback
- Library access
- Monash University Studies Online (MUSO)
- Plagiarism, cheating and collusion
- Register of counselling about plagiarism
- Non-discriminatory language
- Students with disability
- End of semester special consideration / deferred exams