# Table of Contents

**FIT5003 Software security - Semester 2, 2014**

- Mode of Delivery ................................................................. 1
- Workload Requirements ....................................................... 1
- Unit Relationships ............................................................... 1
  - Prerequisites ................................................................. 1
- Chief Examiner .................................................................... 1
- Campus Lecturer ............................................................... 1
  - Caulfield ......................................................................... 2
- Tutors ................................................................................ 2
  - Caulfield ......................................................................... 2
- Your feedback to Us ......................................................... 2
- Previous Student Evaluations of this Unit ......................... 2

## Academic Overview

- Learning Outcomes .......................................................... 3

## Unit Schedule

- Teaching Approach .......................................................... 4
- Assessment Summary ....................................................... 4

## Assessment Requirements

- Assessment Policy ............................................................ 5
- Assessment Tasks .............................................................. 5
  - Participation .................................................................... 5
- Examinations ..................................................................... 6
  - Examination 1 ............................................................... 6
- Learning resources ........................................................... 6
- Feedback to you ............................................................... 6
- Extensions and penalties .................................................. 7
- Returning assignments ..................................................... 7
- Assignment submission .................................................... 7
- Online submission ........................................................... 7

## Other Information

- Policies .............................................................................. 8
- Faculty resources and policies ......................................... 8
  - Graduate Attributes Policy ............................................. 8
- Student Charter .............................................................. 8
- Student services ............................................................. 8
- Monash University Library ............................................... 9
- Disability Liaison Unit ...................................................... 9
FIT5003 Software security - Semester 2, 2014

This unit aims to introduce the secure software development issues including secure software development life cycle, secure software design principles, secure coding practices, threat evaluation models, secure software testing, deployment and maintenance, software development and security policy integration. Students are provided with a range of practical exercises and tasks to reinforce their skills including: identification of security bugs in programs written in different programming languages, design, implementation, and testing of secure concurrent and networked applications, identification of vulnerabilities in networked and mobile/wireless applications. In addition, students will learn input validation techniques to minimise security risks, man-in-the-middle attack techniques to be able to build more secure networked applications, practical secure software testing techniques to be able to test applications for security bugs.

Mode of Delivery

Caulfield (Day)

Workload Requirements

Minimum total expected workload equals 12 hours per week comprising:

(a.) Contact hours for on-campus students:

• Two hours of lectures
• One 2-hour tutorial

(b.) Additional requirements (all students):

• A minimum of 8 hours independent study per week for completing lab and project work, private study and revision.

Unit Relationships

Prerequisites

((FIT5131 or FIT9017) and (FIT5134 or FIT9018) and (FIT5132 or FIT9003 or FIT9019) and (FIT5135 or FIT9020) and (FIT5136 or FIT4037) and (FIT5130 or FIT9030)) or equivalent

Prerequisite knowledge: Programming experience, preferably in C or C++

Chief Examiner

Professor Bala Srinivasan

Campus Lecturer
Your feedback to Us

Monash is committed to excellence in education and regularly seeks feedback from students, employers and staff. One of the key formal ways students have to provide feedback is through the Student Evaluation of Teaching and Units (SETU) survey. The University’s student evaluation policy requires that every unit is evaluated each year. Students are strongly encouraged to complete the surveys. The feedback is anonymous and provides the Faculty with evidence of aspects that students are satisfied and areas for improvement.

For more information on Monash’s educational strategy, see:

www.monash.edu.au/about/monash-directions/ and on student evaluations, see:
www.policy.monash.edu/policy-bank/academic/education/quality/student-evaluation-policy.html

Previous Student Evaluations of this Unit

This unit is being offered for the first time this year as part of the Master of Network and Security degree.

If you wish to view how previous students rated this unit, please go to
Academic Overview

Learning Outcomes

At the completion of this unit students will have an understanding of central issues and concepts involved in secure software development, including:

- importance of security and risk assessment;
- vulnerabilities versus threats;
- common attack techniques (malicious input injection, buffer overflows);
- malware;
- common web application vulnerabilities and design aspects (cross-side scripting, SQL injection, cookies, session management, user authentication);
- operating system security;
- secure development principles and techniques (including principle of least privilege, input validation, tools for writing secure code);
- software testing techniques (including bug discovery techniques, fuzzing);
- software deployment and maintenance issues.
Unit Schedule

<table>
<thead>
<tr>
<th>Week</th>
<th>Activities</th>
<th>Assessment</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No formal assessment or activities are undertaken in week 0</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Introduction to Software Security</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Threats, Vulnerabilities and Attacks</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Threat Modeling</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Secure (and Insecure) Coding Techniques I</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Secure (and Insecure) Coding Techniques II</td>
<td>Part I of Assessment task 1 due</td>
</tr>
<tr>
<td>6</td>
<td>Web Application Security</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Security Testing Techniques</td>
<td>Part II of Assessment task 1 due</td>
</tr>
<tr>
<td>8</td>
<td>Programming Languages and Security</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Programming Languages and Sandboxing</td>
<td>Assessment task 2 due</td>
</tr>
<tr>
<td>10</td>
<td>Malware</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Implementation Pitfalls</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Summary and Revision</td>
<td>Assessment task 3 due</td>
</tr>
<tr>
<td></td>
<td>SWOT VAC</td>
<td>No formal assessment is undertaken in SWOT VAC</td>
</tr>
<tr>
<td></td>
<td>Examination period</td>
<td>LINK to Assessment Policy:</td>
</tr>
</tbody>
</table>

*Unit Schedule details will be maintained and communicated to you via your learning system.

Teaching Approach

Lecture and tutorials or problem classes

This teaching and learning approach helps students to initially encounter information at lectures, discuss and explore the information during tutorials, and practice in a hands-on approach both in a lab environment as well as by accessing a server from outside the University.

Assessment Summary

Examination (2 hours): 50%; In-semester assessment: 50%

<table>
<thead>
<tr>
<th>Assessment Task</th>
<th>Value</th>
<th>Due Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use of a Software Security Tool for Code Review</td>
<td>20%</td>
<td>Week 5 and 7( Two part submission)</td>
</tr>
<tr>
<td>SQL injection Vulnerability</td>
<td>20%</td>
<td>Week 9</td>
</tr>
<tr>
<td>Penetration Testing of Software</td>
<td>10%</td>
<td>Week 12</td>
</tr>
<tr>
<td>Examination 1</td>
<td>50%</td>
<td>To be advised</td>
</tr>
</tbody>
</table>
Assessment Requirements

Assessment Policy

Faculty Policy - Unit Assessment Hurdles

Academic Integrity - Please see resources and tutorials at
http://www.monash.edu/library/skills/resources/tutorials/academic-integrity/

Assessment Tasks

Participation

• Assessment task 1

  Title: Use of a Software Security Tool for Code Review
  Description: Develop a concurrent program using threads for an application in C or C++ and identify its vulnerabilities using a source code review tool.

  More details will be provided on the assignment specification.
  Weighting: 20%
  Criteria for assessment:
  Code development (40%)
  Installation of code review tool and the analysis of supplied source code for vulnerabilities (40%)
  Defend or refute the identified vulnerabilities (20%)
  Due date: Week 5 and 7 (Two part submission)

• Assessment task 2

  Title: SQL injection Vulnerability
  Description: This assignment will develop code to demonstrate the SQL injection vulnerability. You will extend the program that is being discussed in the lab session so as to exploit the SQL injection while accessing a database, and then implement sound countermeasure.
  Weighting: 20%
  Criteria for assessment:
  Code development (30%)
  Successful exploitation and penetration of the code (40%)
  Countermeasures to fix the vulnerabilities (30%)
  Due date:
Assessment Requirements

Week 9

• Assessment task 3

  Title: Penetration Testing of Software
  Description: Students will be given access to a software component and challenged to breach its security using techniques covered in the unit.
  Weighting: 10%
  Criteria for assessment: Report contents will be marked on the following attributes:
    ♦ Identification of vulnerabilities in the supplied software (40%);
    ♦ Implementation of exploits for and countermeasures to vulnerabilities (40%); and
    ♦ Quality of the submitted report (20%).
  Due date: Week 12

Examinations

• Examination 1

  Weighting: 50%
  Length: 2 hours
  Type (open/closed book): Closed book
  Electronic devices allowed in the exam: None

Learning resources

Monash Library Unit Reading List (if applicable to the unit)
http://readinglists.lib.monash.edu/index.html

Faculty of Information Technology Style Guide

Feedback to you

Examination/other end-of-semester assessment feedback may take the form of feedback classes, provision of sample answers or other group feedback after official results have been published. Please check with your lecturer on the feedback provided and take advantage of this prior to requesting individual consultations with staff. If your unit has an examination, you may request to view your examination script booklet, see
http://intranet.monash.edu.au/infotech/resources/students/procedures/request-to-view-exam-scripts.html

Types of feedback you can expect to receive in this unit are:

• Informal feedback on progress in labs/tutes
Assessment Requirements

- Graded assignments without comments
- Interviews

Extensions and penalties

Submission must be made by the due date otherwise penalties will be enforced.


Returning assignments

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

Assignment submission

It is a University requirement ([http://www.policy.monash.edu/policy-bank/academic/education/conduct/student-academic-integrity-managing-plagiarism-collusion-procedures.html](http://www.policy.monash.edu/policy-bank/academic/education/conduct/student-academic-integrity-managing-plagiarism-collusion-procedures.html)) for students to submit an assignment coversheet for each assessment item. Faculty Assignment coversheets can be found at [http://www.infotech.monash.edu.au/resources/student/forms/](http://www.infotech.monash.edu.au/resources/student/forms/). Please check with your Lecturer on the submission method for your assignment coversheet (e.g. attach a file to the online assignment submission, hand-in a hard copy, or use an online quiz). Please note that it is your responsibility to retain copies of your assessments.

Online submission

If Electronic Submission has been approved for your unit, please submit your work via the learning system for this unit, which you can access via links in the my.monash portal.
Other Information

Policies

Monash has educational policies, procedures and guidelines, which are designed to ensure that staff and students are aware of the University’s academic standards, and to provide advice on how they might uphold them. You can find Monash’s Education Policies at: www.policy.monash.edu.au/policy-bank/academic/education/index.html

Key educational policies include:

- Student Academic Integrity Policy and Student Academic Integrity: Managing Plagiarism and Collusion Procedures; http://www.policy.monash.edu/policy-bank/academic/education/conduct/student-academic-integrity-policy.html
- Special Consideration; http://www.policy.monash.edu/policy-bank/academic/education/assessment/special-consideration-policy.html
- Grading Scale; http://www.policy.monash.edu/policy-bank/academic/education/assessment/grading-scale-policy.html
- Discipline: Student Policy; http://www.policy.monash.edu/policy-bank/academic/education/conduct/student-discipline-policy.html
- Academic Calendar and Semesters; http://www.monash.edu.au/students/dates/
- Orientation and Transition; http://intranet.monash.edu.au/infotech/resources/students/orientation/

Faculty resources and policies

Important student resources including Faculty policies are located at http://intranet.monash.edu.au/infotech/resources/students/

Graduate Attributes Policy

http://www.policy.monash.edu/policy-bank/academic/education/management/monash-graduate-attributes-policy.html

Student Charter


Student services

The University provides many different kinds of support services for you. Contact your tutor if you need advice and see the range of services available at http://www.monash.edu.au/students. For Malaysia see http://www.monash.edu.my/Student-services, and for South Africa see http://www.monash.ac.za/current/.
Monash University Library

The Monash University Library provides a range of services, resources and programs that enable you to save time and be more effective in your learning and research. Go to www.lib.monash.edu.au or the library tab in my.monash portal for more information. At Malaysia, visit the Library and Learning Commons at http://www.lib.monash.edu.my/. At South Africa visit http://www.lib.monash.ac.za/.

Disability Liaison Unit

Students who have a disability or medical condition are welcome to contact the Disability Liaison Unit to discuss academic support services. Disability Liaison Officers (DLOs) visit all Victorian campuses on a regular basis.

- Website: http://www.monash.edu/equity-diversity/disability/index.html
- Telephone: 03 9905 5704 to book an appointment with a DLO; or contact the Student Advisor, Student Community Services at 03 55146018 at Malaysia
- Email: dlu@monash.edu
- Drop In: Equity and Diversity Centre, Level 1, Building 55, Clayton Campus, or Student Community Services Department, Level 2, Building 2, Monash University, Malaysia Campus