COMP6443/6843
Web Application Security and Testing

Semester 1, 2018
6 UoC — for UG + PGC

Course Staff

Lecturer-in-Charge
Prof Richard Buckland (https://www.openlearning.com/u/richard.buckland)

Lecturers and Teaching Staff
Norman Yue (https://www.openlearning.com/u/ilikeplatypodes)
Abhijeth Dugginapeddi (https://www.openlearning.com/u/abhijeth)
Sean Yeoh (https://www.openlearning.com/u/seanyeoh)
John Cramb (https://www.openlearning.com/u/z3377807)

Course Convenor
Prof Richard Buckland (https://www.openlearning.com/u/richard.buckland)

Course Admin
Cameron Lonsdale (https://www.openlearning.com/u/cameronlonsdale/)
For confidential course enquiries, email: cs6443@cse.unsw.edu.au (mailto:cs6443@cse.unsw.edu.au)

Classes

Lectures:

- 6[48]43: Tues 18:00 - 20:00 CivEng 109
- 6843: Tues 20:00 - 21:00 CivEng 109

Tutorials:

- 6843: Wed 11:00 - 13:00 SecLab
- 6443 (one of):
  - Wed 15:00 - 17:00 Quad G040
  - Thurs 13:00 - 15:00 SecLab
  - Fri 18:00 - 20:00 SecLab

Course Summary

Web Applications are currently the predominant source of software vulnerabilities exploited in online attacks. There is a growing need and demand for web programmers to be security literate.

These courses covers the main types of web application vulnerabilities and current professional coding and testing best practices to be able to successfully develop and test secure web applications.

The core course content (6443 and 6843) includes:

- Reconnaissance
- Authentication
- Session management
- Access control
- Client-side attacks, such as cross-site scripting
- Server-side attacks, such as SQL injection
- Common web service vulnerabilities

For COMP6843 students, additional course content will be covered in the extended lecture. (Core students are welcome to attend the extended lecture too if they are interested). Extended students will learn, in overview:

- Advanced Asset Discovery
- Sandbox Escapes for Cross-site Scripting (XSS)
- Server-side Request Forgery (SSRF)
- XML External Entity (XXE)
- Same Origin Policy (SOP) bypass, Content Security Policies (CSP) and Cross-site Scripting header protections
- Single Sign On (SSO), OAuth and Security Assertion Markup Language (SAML)
- Mobile Security
- Advanced Injection
- Amazon Web Services (AWS) and Cloud

Course coverage will be updated over time to reflect emerging vulnerabilities and practices.

BEWARE

To get the most from these courses you will need to engage in independent study and act as a self directed learner. Attending lectures alone will not be sufficient to pass the course. You will need to devote considerable practice to all the techniques we cover and read further on topics which interest you or which you do not fully understand. For a credit level result we expect you will spend 11 hours per week in total on this course, including over the mid semester break.

Seek feedback from your friendly tutor and class peers constantly over the semester and closely monitor yourself to make sure you are not falling behind. We treat you like adults and do not force you to do the self directed work and practice - but experience has shown that students who do not work hard at the course do not do well, and often express disappointment later on at the missed opportunity. (Since we have awesome tutors and speakers here for you during the course - make sure you make full use of them and your time.)

Assumed Knowledge

You need to have taken and passed COMP6441 or COMP6841 or COMP3441 or COMP9321. We expect you to have or be prepred to teach yourself web programming skills such as are taught in Web Applications Engineering - COMP9321

Prior to commencing the course, students should have an understanding of how the web works, including the following:

1. Web technologies: HTML, CSS, JavaScript, Databases, Server-side Scripting (e.g. PHP)
2. Moderate familiarity with at least one web development language like Java, Python, PHP, etc
3. Basic knowledge of network and web related protocols (e.g. TCP/IP, UDP, HTTP, HTTPS)

Course Learning Outcomes

After completing this course, you will:

- Understand how modern web applications work
- Be able to use secure coding practices to develop web applications
- Be able to assess the security of a web application using white/grey/black-box approaches
- Know the many common pitfalls and security vulnerabilities that affect web applications and associated infrastructure
- Understand how to defend against these attacks and how companies can implement a Secure Development Life Cycle (SDLC) to mitigate potential vulnerabilities
- Have a basic understanding of mobile application security concerns
- Have practical knowledge of specialised security tools which assist in assessing the security of a web application

This course contributes to the following Graduate Capabilities:

- Scholars, capable of independent and collaborative enquiry, rigorous in their analysis, critique and reflection, and able to innovate by applying their knowledge and skills to the solution of novel as well as routine problems
- Entrepreneurial leaders, capable of initiating and embracing innovation and change, as well as engaging and enabling others to contribute to change
- Professionals, capable of ethical, self-directed practice and independent lifelong learning
- Global citizens, who are culturally adept and capable of respecting diversity and acting in a socially just and responsible way

Teaching Strategies

COMP6443/6843 are taught in parallel. The lectures are common for all courses. Students doing Extended Web Application Security and Testing (COMP6843) will have an extra 1 hour extended lecture. Tutorials are allocated for both COMP6443 and COMP6843 students.

Lectures will be used to introduce students to theoretical and practical concepts, and will include live and recorded demonstrations. There will be guest lecturers coming in from industry to share practical specialised experience with the students. COMP6443 students have an allocated 2 hour lecture, COMP6843 students have an extra 1 hour lecture covering extended topics. A detailed list of lecture topics and the slides used for the lectures will be posted on the course website as session progresses.

Tutorials: Their primary purpose is to allow students to further develop lecture concepts through collaborative learning and experienced instruction. Part of the allocated tutorial time will be devoted towards solving problems and discussing solutions related to lecture topics. The remaining time will be used to discuss topics relevant to assignments. Any work not completed during the 2 hour tutorial time slot becomes work students must complete in their own managed study time for these courses. There are no marks for tutorial attendance.

Assignments: Students are expected to apply the knowledge gained in practical environments known as **war games**. These will vary in difficulty and will cover various aspects of Web Application Security. Additionally, COMP6443 students will be asked to create their own Web Applications to develop their secure coding practice skills. These exercises will be assessed and make up a percentage of your final mark. Students will work in small groups to complete these assignments.

For COMP6843 it is assumed that students are already able to build simple web apps with ease, and therefore the assignments will focus heavily on advanced exploitation of more difficult web vulnerabilities.

Assignments will be subject to a late penalty. Due to their diversity, the late penalty will be tailored for each individual assignment.

- Assignments will be marked as a group, i.e. groups of one will be marked the same as groups of two or three
- All group members are expected to contribute equally to EACH assignment. In submitting an assignment, the group is agreeing that all members have contributed equally unless we receive an agreed-to statement from all group members identifying differing levels of contribution

The **Final Exam** will be a theoretical and practical exam where students apply their knowledge and skills learnt over the entire course. Note: Extended and standard Web Application Security have different exams with approximately 75% common content.

Supplementary exams will only be awarded in well justified cases, in accordance with School policy for *Special Consideration*, **not** as a second chance for poorly performing students. In particular, it is unlikely that a supplementary will be awarded to students who have actually sat the proper exam. **Make up your mind whether or not you are sick before attempting the exam!**

Assessment

6443

- **Build & Break Coursework**: Practical homework assignments done in groups. Weekly; worth 45%
- **Final Examination**: theoretical and practical examination. During the university's exam period; worth 50%
- **Something Awesome**: Portfolio demonstrating your individual effort into something awesome. Throughout the semester; worth 5%
- **Bonus**: Portfolio of bug bounties verified by third party or similar. Throughout the semester; worth up to 5%

6843

- **Break Coursework**: Practical homework assignments done in groups. Weekly; worth 45%
- **Final Examination**: A theoretical and practical examination. During the university's exam period; worth 50%
- **Bug Bounties**: Portfolio of bug bounties verified by third party or similar. Throughout the semester; worth 5%

A mark of 50% overall **AND** a mark of 50% in the final exam is required in order to pass the course. In the circumstance that a student does not pass the final exam, their mark for coursework will be capped by their exam mark. For example, a student scoring 40% in their final exam, but who previously scored 60% in the course work will have their course work mark modified down to 40%.

Student Conduct

UNSW has an ongoing commitment to fostering a culture of learning informed by academic integrity. All UNSW staff and students have a responsibility to adhere to this principle of academic integrity. Plagiarism undermines academic integrity and is not tolerated at UNSW. Plagiarism at UNSW is defined as using the words or ideas of others and passing them off as your own.

If you haven't done so yet, please take the time to read the full text of UNSW's policy regarding academic honesty and plagiarism (https://student.unsw.edu.au/plagiarism).

The pages below describe the policies and procedures in more detail:

- Student Code Policy (https://www.gs.unsw.edu.au/policy/documents/studentcodepolicy.pdf)
- Plagiarism Policy Statement (https://www.gs.unsw.edu.au/policy/documents/plagiarismpolicy.pdf)
- Plagiarism Procedure (https://www.gs.unsw.edu.au/policy/documents/plagiarismprocedure.pdf)
- Student Misconduct Procedure (https://www.gs.unsw.edu.au/policy/documents/studentmisconductprocedures.pdf)

Good Faith Policy

These courses expects a high standard of professionalism from its students with regard to how Web Application Security Testing is conducted. We expect all students to act in good faith at all times - including but not limited to:

- Respect the property of others and the university
- Always abide by the law and university regulations
- Be considerate of others to ensure everyone has an equal learning experience
- Always check that you have written permission before performing a security test on a system

Your actions speak volumes; It is our responsibility to uphold the reputation of the course, the course staff, fellow students, the school, the university and the ICT profession. **If you are unsure whether your actions may violate this policy, ask one of the course staff for guidance.**

Failure to adhere to this policy may result in an academic penalty.

Course Evaluation and Development

These courses are still relatively new, and we strongly encourage students to actively provide feedback about the course's progress.

These courses will be evaluated by UNSW's myExperience program. You'll receive an email to your student email address with instructions on completing this; we'll also (endeavour to) send out a notification.

These courses will also be evaluated by the CSE Student Representatives' mid-session Course Survey.

Text and Reference Books

Textbook
There is no official textbook for these courses.

Reference Books
- Stuttard, Dafydd, and Marcus Pinto. The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws. John Wiley & Sons, 2012
- Zalewski, Michal. The Tangled Web: a Guide to Securing Modern Web Applications. No Starch Press, 2012