Software Security - Spring 2018 (4 cr.)

This is a 4 credit course at Aalto University Department of Computer Science in spring 2018. The course code is CS-E4330. Lectures will be held by Antti Vähä-Sipilä from F-Secure Cyber Security Services.

Course staff

- Antti Vähä-Sipilä (lecturer, avs@iki.fi, Twitter)
- N. Asokan (professor responsible for the course, n.asokan_at_aalto.fi)
- Lachlan Gunn (local organization, lachlan.gunn@aalto.fi)
- Mustafa Khalid Masood (local organization, mustafa.khalidmasood@aalto.fi)
- **All of the above (preferred, ssec-staff@list.aalto.fi)**

Course staff can be reached by email. For anything related to the university (enrollment, grades, etc.), please contact the Aalto contacts. For questions regarding the content of the course, please contact Antti.

There is also a Slack channel, which you can use to freely discuss, and it could be the quickest way to get hold of Antti. Invite yourself using this link. It is suggested that lecture subject matter Q&A will be done on the Slack channel. However, the Slack channel is **not** the official information source; for that, there is the course email list and this page.

The course uses MyCourses for returning weekly exercises.

Thanks

I'd like to thank F-Secure for their support in making the course possible this year.

News

- No news at the moment. Please read the course email list.

Course Overview

This is a basic course on software security. The target group are software developers and generalists who are not necessarily security specialists. However, the course provides a lot of practical insight to those contemplating security related career options. The content of the course is geared towards practical and commercial software development.

The course will explain how software breaks in the security sense, and how to determine whether software is broken. We look at the variety of activities and strategies available for software developers and organizations to create more secure software. We learn how to perform security and privacy threat modelling (architectural risk analysis and privacy impact assessment), and how to apply language-theoretic approach to security engineering. We conclude with economics of software security, its regulation (mainly from a Finnish and EU viewpoint), and relationship to society.

Prerequisites

Especially for the two first weeks' topic, the students need to be familiar with operating system basics such as the concepts of processes and memory allocation, fluent in at least one programming language, understand the concept of a protocol stack, know how HTTP works, and be comfortable on the command line on GNU/Linux, Mac OS X, or Windows. Prior information security knowledge is not a requirement.

For some of the weekly exercises, you need access to a computer with a Java JRE. It is likely you will be better off using a UNIX-y system such as a GNU /Linux distro, macOS, or possibly the Windows Subsystem for Linux.

Grading and passing the course

Weekly exercises

The course has no exam, but has five weekly graded home exercises, whose average score will be the final grade. Students must complete at least three of the five exercises in order to get a final grade. A skipped exercise will be evaluated as zero. Grading guidelines are available from MyCourses.

Each exercise deadline is one week after the lecture. Exercises are returned on the MyCourses page.

Responses to weekly assignments can be written in either English or Finnish. Writings must be returned in plain text (including Moodle text fields), HTML or PDF formats. Images (diagrams drawn in some assignments) must be returned in PDF, JPEG or PNG formats. If using lossy compression, please use maximum quality settings.

Lectures

Although the six lectures do not, strictly speaking, have mandatory attendance, we would ask you not to enroll on the course unless you really also plan to attend the lectures. The course has limited space and there *will* be someone else who wants to attend. If you are looking for a completely virtual course, Coursera still has a Software Security course.

Concept inventory tests

This course is also an experiment in computer security education. We will measure the participants' knowledge of some key concepts immediately before and after the course using a 15-minute multiple choice test. This test is pseudonymous; it will not affect your grade, and we are not interested in the identity of any single responder. The only thing we require is that we can correlate the responses before and after the course.

Course Registration

Registration period will start on March 12th, 2018 at 9:00 and close on April 16th, 2018 at 23:59.

Important note on registrations: The course will have limited capacity. Based on past years' experience, it is likely that it will be 'sold out'. To ensure that everyone who starts the course will also finish it, we will kindly ask for ***your commitment to finish the course***. ~~Registrants will get an email prior to the course, and **it is necessary to reply to that email** in order to secure your place. Also, we'd suggest you register as soon as you know you can take the course.~~

Schedule

- **Lecture:** Lectures will be mostly (see MyCourses) held on **Tuesday afternoon (14:15 - 17:00)** in room TU5 of the Aalto TUAS building, Maarintie 8. There is no lecture on Tuesday 1 May. Instead, a lecture will be held on Friday 11 May (09:15-12.00pm). Lectures usually aim at two hours, with the third hour being reserved for extra material.

Date	Lecture topic	Lecture notes	Exercise and deadline (all deadlines 07:00 UTC+3)
Tuesday **10 Apr**	• Concept inventory test • Introduction • How software breaks (low level)	Lecture 1: How software breaks (1/2)	Assignment 1: Fuzzing *Grades posted on MyCourses.*
Tuesday **17 Apr**	• How software breaks (web apps) • Concepts of language theoretic security	Lecture 2: How software breaks (2/2)	Assignment 2: Intercepting proxies *Due 24 April, 07:00am UTC+3.*
Tuesday **24 Apr**	• Security in a software project	Lecture 3: Security in a Software Project	Assignment 3: The role of security tools in software projects *Due 8 May, 7:00am UTC+3.* Based on past and this year's assignment solutions, I maintain a document with difficult questions to ask from security tool salespersons: *SecurityTestToolConsiderations_20180523.pdf*
Tuesday **8 May**	• Threat modelling (architectural risk analysis) • **Location T6**	Lecture 4: Threat modelling	Assignment 4: Threat modelling
Friday **11 May**	• Privacy engineering • EU General Data Protection Regulation and software security • **Note** a different weekday. • **Location TU5**	Lecture 5: Privacy, the GDPR and Privacy Engineering	Assignment 5: Privacy considerations The result from Assignment 4 is required as a basis for Assignment 5. If you skip Assignment 4, you can obtain a reference answer from the course staff after the Assignment 4 deadline has passed. However, this only leaves you a couple of days, so working on Assignment 4 is much recommended.
Tuesday **15 May**	• Software security in the society • Economics of software security • EU and national regulation • ~~Visitor: A consultant's view - Henri Lindberg~~ ◦ Cancelled due to force majeure • Concept inventory test • **Location R037/1171-72**	Lecture 6: Software Security, Society and Economics	(None, have a nice summer)

Books

The course has no textbook as such. There are a couple of options that will help you, but getting the books with your own money up front is perhaps not the best alternative. Having said this, both of the following books have excellent material for those who are contemplating a career in security.

1. Adam Shostack: Threat Modeling: Designing for Security. Wiley, 2014. http://threatmodelingbook.com/
   Note: E-book copy is available for 14 days for Aalto students. Link - http://site.ebrary.com/lib/aalto/detail.action?docID=10837601
2. Craig Jackson, Scott Russell, Susan Sons: Security from First Principles. O'Reilly, 2017.

Other

This course was lectured for the first time in 2014 at the University of Helsinki. You can find the old course pages on their wiki. Please don't copy their answers.