Introduction

TDDC90 – Software Security

Ulf Kargén

Division for Database and Information Techniques (ADIT) at the
Department of Computer and Information Science (IDA)

LiU EXPANDING REALITY

Agenda

- Why study software security?
- Organization of the course
  - Course contents
  - Prerequisites
  - Lectures overview
  - Labs
  - Reading material
  - Course evaluation

Examiner
Ulf Kargén

Lecturer
Ahmed Rezine

Lecturer
Kristian Sandahl

Lab assistant
Alireza Mohammadinodooshan

Why study software security?

1. What kind of software is security critical?
2. Why do people try to hack software?

20 years ago

1) Mostly server software
2) Fun

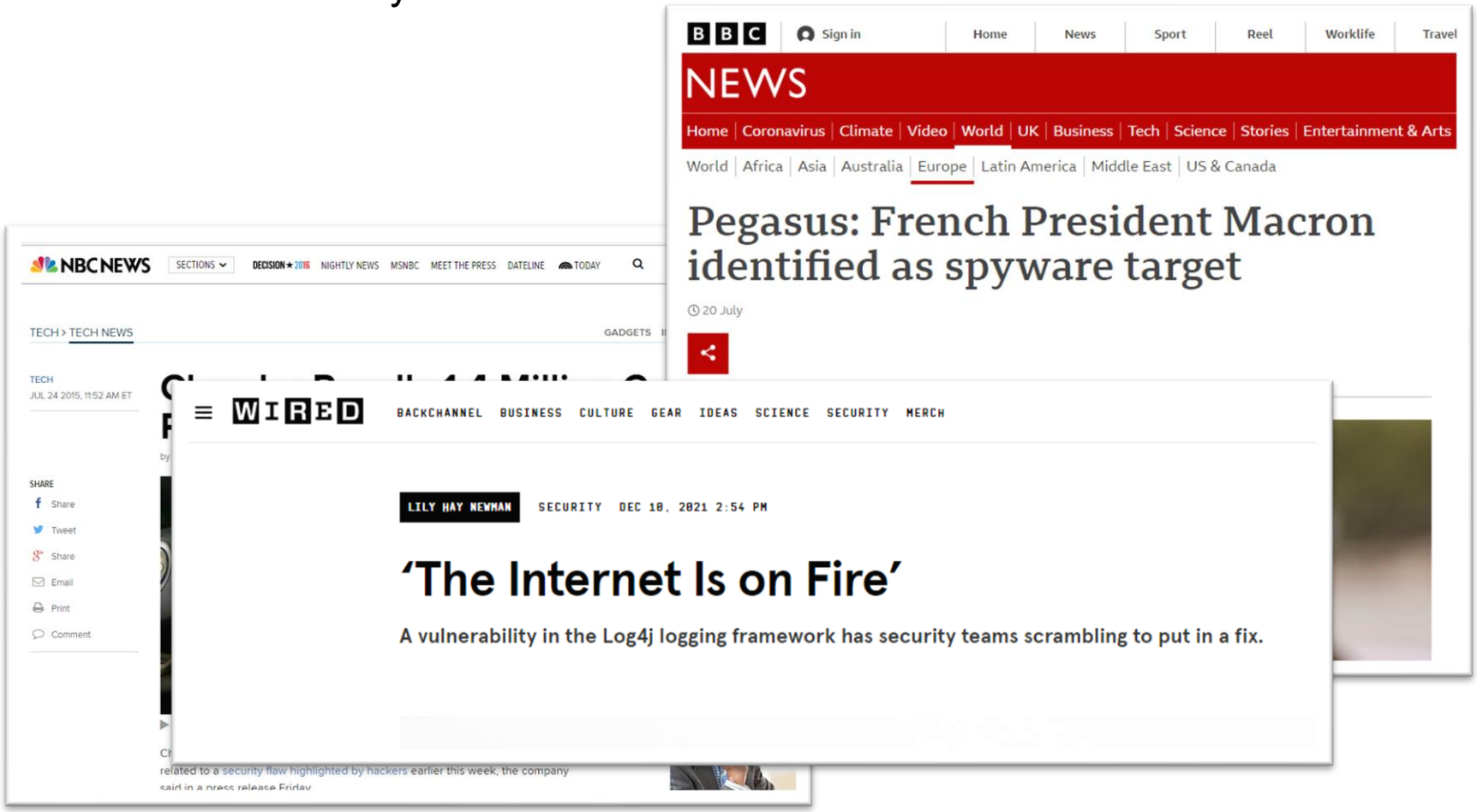
Today

1) Pretty much all software
2) Profit

- Hackers increasingly target end-user equipment
- Break-ins happen increasingly by exploiting client-side software (document viewers, media players, browsers), not by attacking central infrastructure
- “Everything” is connected to the internet – attacks against poorly secured IoT targets are very common

Why study software security?

Software is everywhere …



Developing secure software requires…

- Security-aware developers
  - Know about common vulnerability types
  - Know common attacks
  - “Think like a hacker”
  - The devil is in the details…

- Adequate software engineering processes
  - Methods for eliciting security requirements
  - Security in the specification, architecture and design
  - Secure coding guidelines and patterns
- Software security assurance methods and tools
  - Many methods:
    Code reviews, formal methods, static analysis, fuzzing, etc.

Organization of the course

Organization

- 9 lectures
- 3 mandatory labs
  - Pong – the insecure ping
  - Web security
  - Static analysis
- Examination:
  - Written exam (3 hp)
  - Labs (3 hp)

Detailed information on course organization, lecture slides, lab instructions, etc., is available on the course web site:

https://www.ida.liu.se/~TDDC90/index.en.shtml

Changes due to surgery

- My lectures will all be via Zoom
  - Recordings will also be published on course web site
  - Lectures given by Ahmed and Kristian on-campus as usual
  - Lab supervision also on-campus as usual
- Sick leave from November 12 (probably 3–4 weeks)
  - Direct questions of administrative nature to Alireza Mohammadinodooshan
  - Direct subject-specific questions on Ahmed or Kristian's parts to them
- If possible, I will try to book a Q&A session at the end of the course (mid-December)
  - Opportunity to ask questions, get help with old exams, etc.

Details on course start page and schedule page:

- https://www.ida.liu.se/~TDDC90/index.en.shtml
- https://www.ida.liu.se/~TDDC90/timetable/index.en.shtml

Prerequisites

- Required:
  - Basic computer security course
  - Programming experience
  - Course in software engineering

- Recommended:
  - Operating systems and assembly programming basics
  - Some prior experience with C-programming
  - Basic course in logic
  - Basic web programming
    (HTML, JavaScript, some server-side language)

For those unfamiliar with C

Google these things (in this order):

- ✓ C pointers
- ✓ Pointer arithmetic
- ✓ Pointers and arrays
- ✓ C dynamic memory allocation
- ✓ C sizeof operator
  - Pay special attention to the difference between sizeof on pointers and arrays!

Lectures

- Secure software development (1 lecture)
  Given by Ulf Kargén
  - Secure software development processes
  - Secure design patterns
  - Modeling and risk analysis

- Vulnerabilities and exploits (2 lectures)
  Given by Ulf Kargén
  - Common vulnerabilities in C/C++ programs
  - Known attack techniques
  - OS and compiler mitigations

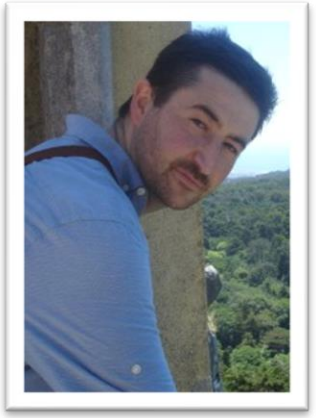
Lectures (continued)

- Web security (1 lecture)
  Given by Ulf Kargén
  - Common vulnerabilities in web applications
  - Attack techniques and protections

- Code reviews (1 lecture)
  Given by Kristian Sandahl
  - Software inspections and other techniques

Lectures (continued)

- Static analysis (2 lectures)
  Given by Ahmed Rezine
  - Introduction to static analysis
    - Abstract interpretation
    - Symbolic execution



- Security testing and course wrap-up (1 lecture)
  Given by Ulf Kargén
  - Fuzzing, concolic testing
  - Course wrap-up

Labs

- Pong – the insecure ping
  - Perform a code review to find vulnerabilities
  - Exploit a buffer overflow to gain root
  - Fix all vulnerabilities
  - **Requires considerable time and effort, especially if you don't posses all recommended prerequisite knowledge**
- Websec
  - Deliberately vulnerable web app
  - Study common weaknesses and understand attack techniques
  - Typical time needed: 1-2 lab sessions
- Static
  - Study common static analysis techniques described in the lectures
  - Typical time needed: 1-2 lab sessions
  - **Note 1:** Lab sessions on **Nov 29** and **Dec 1** are **half-class** only! (Webreg Groups A and B, respectively)
  - **Note 2:** Requires demoing for Ahmed. The other labs do **not** require demos.

Labs

- Different assistants for some labs – see lab page on course web
- Webreg signup deadline **November 8th**
  - Unregistered students not allowed to sign up!
- Labs are meant to be done in pairs
  - *Might* be possible to do labs alone if you have a good motivation, **however:**
  - If too many sign up alone, we may randomly group lone students.
- **Hard** deadline for handing in solutions is **December 15th**
  - Complete all labs **at least one week before this** to allow time for corrections and re-submission
  - Hand in solutions continuously during the study period – don't save everything for the last week!
  - Start with labs as early as possible, especially Pong!

Reading material

- No course book (no one book covers all topics in the course)
- Mandatory reading:
  - Papers/articles, web resources, and lecture slides
  - Lectures don’t cover all articles, and vice versa
- Also a list of extra reading for interested students
  - Not needed for exam

Previous year's course evaluation

- Overall score last year was 4.18 (of 5)
- Scores of all evaluation items available at: https://admin.evaliuate.liu.se/search?lang=en

Suggestions on improvements from students:

- "Want answers for old exams"
  - **Comment**: Old exams are provided primarily to give an idea about exam structure and what topics to focus on when studying for the exam. This year, I will provide at least one example exam with answers/hints.
- "Examiner was late for lectures"
  - **Comment**: This is of course unfortunate. I will try to have better time margins this year.

Previous year's course evaluation

Positive remarks:

- "The web and static analysis labs were excellent in teaching their respective methods […]"
- "I really liked the labs and the lab assistant Alireza did a great job explaining everything intuitively!"
- "The labs did a good job of covering the different subject areas, and help foster a deeper understanding of them."

Questions?