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

Why study software security?

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

20 years ago

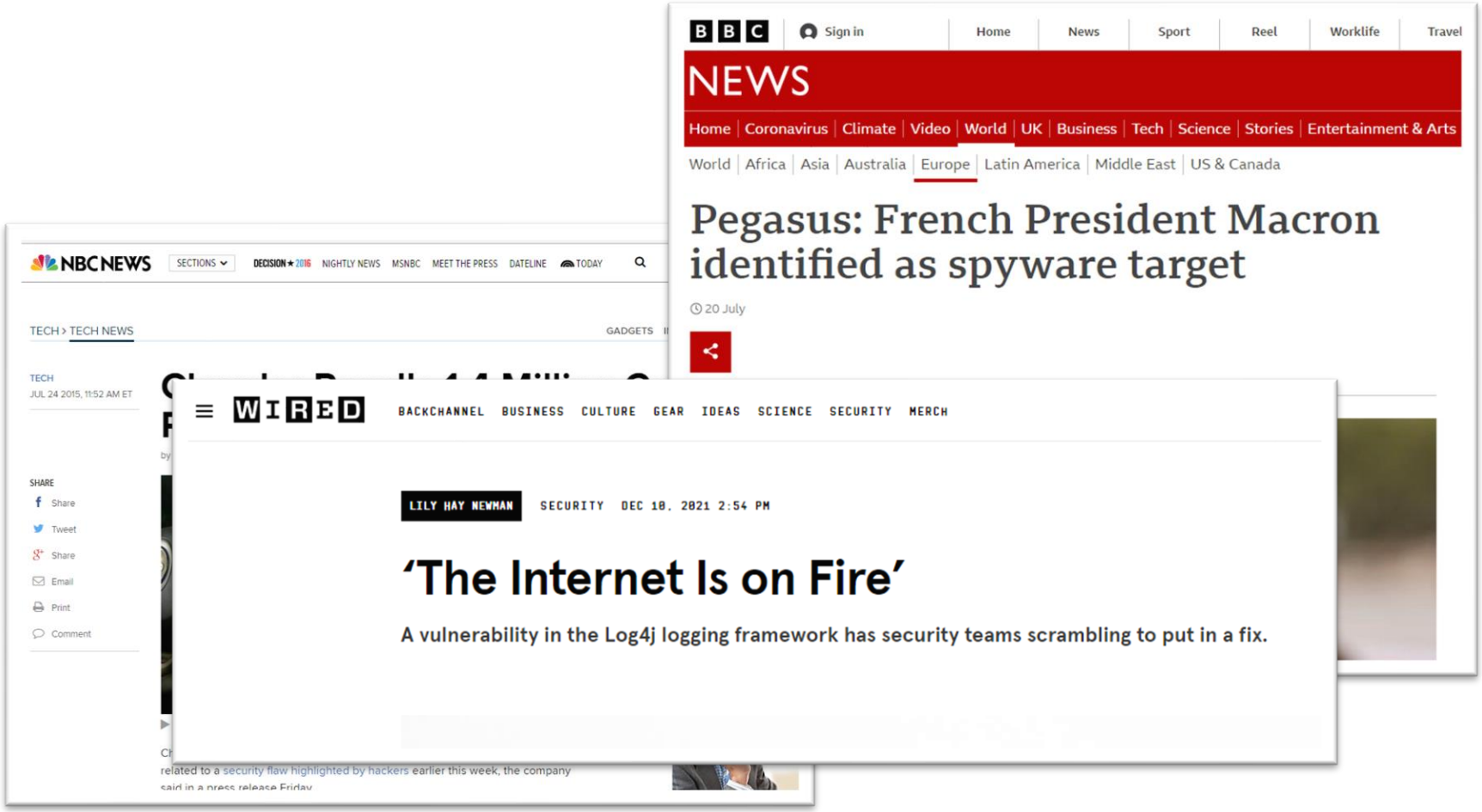
1) Mostly server software
2) Fun

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, messaging apps), not by attacking central infrastructure
- “Everything” is connected to the internet – attacks against poorly secured IoT targets are very common

Why study software security?



Why study software security?

Provide a list of actively exploited software vulnerabilities in major software products discovered during 2025.

CVE ID	Product/Component	Vulnerability Type & Impact	
CVE-2025-59287	Microsoft Windows Server Update Services (**WSUS**)	Critical Remote Code Execution (**RCE**)	Added to CISA's KEV catalog shortly after an emergency patch in October 2025. Exploitable **unauthenticated** by an attacker sending a crafted request.
CVE-2025-2783	Google **Chrome**	RCE and Sandbox Escape	Actively exploited in targeted espionage attacks (Operation ForumTroll) to deploy spyware (**LeetAgent/Dante**). Tied to a flaw in the Mojo IPC system.
CVE-2025-43300	Apple **iOS**	**Zero-click** Vulnerability	An emergency update was pushed for an exploit used to install spyware on targeted iPhones.
CVE-2025-55177	**WhatsApp** (on iOS)	Unauthorized Code Execution	A severe bug allowing specially crafted messages to trigger RCE, reportedly exploited against civil society targets.
CVE-2025-47827	**IGEL OS**	Secure Boot Bypass	Allowed a **local user** to escalate privileges by improperly verifying cryptographic signatures when mounting an image.
CVE-2025-	Microsoft Windows (**Agere Modem Driver**)	Elevation of Privilege (**EoP**)	Affected nearly all Windows versions. Allowed a **local attacker** with minimal

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

- 8 lectures
- 6 mandatory labs/assignments
  - Pong – the insecure ping
    - Split into 3 assignments: Review, Exploit, Mitigation
  - Web security
  - Static analysis
  - Inspection (performing a code review on small pieces of code)
- 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

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 + code reviews (1 lecture)
  Given by Ulf Kargén
  - Secure software development processes
  - Modeling and risk analysis
  - Introduction to manual code reviews

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

Lectures (continued)

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

Lectures (continued)

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

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

Labs/Assignments

- Pong – the insecure ping
  - Perform a code review to find vulnerabilities (**Review** part)
  - Exploit a buffer overflow to gain root (**Exploit** part)
  - Apply protections and fix all vulnerabilities (**Mitigation** part)
  - **Requires considerable time and effort, especially if you don't posses all recommended prerequisite knowledge**
- Static
  - Study common static analysis techniques described in the lectures
  - Typical time needed: 1-2 lab sessions
  - **Note 1:** Requires demoing for Ahmed or Ulf.
    The other labs do **not** require demos.
  - **Note 2:** Ahmed is only available for demoes during **two** specific lab sessions.
    Check the lab schedule on the web!
- Websec
  - Deliberately vulnerable web app
  - Study common weaknesses and understand attack techniques
  - Typical time needed: 1-2 lab sessions

Labs/Assignments

- Inspection
  - Perform a code review on several small pieces of vulnerable C/C++ code
  - Performed during **mandatory** seminar on **November 27**
  - You will be required to present your findings during the seminar
    - So make sure to have a solid grasp of memory safety vulnerabilities prior to seminar!
  - More detailed info on course web

Labs/Assignments

- Different assistants for some labs – see lab page on course web
- Webreg signup deadline **November 12th**
  - 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 17th**
  - 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!
    - OK (and recommended) to hand in individual parts of Pong separately, but please **use the same email thread for all parts**
  - Start with labs as early as possible, especially Pong!

Labs/Assignments

- Pong and Websec labs will run on Cybersecurity lab backend starting this year (same as Ethical hacking course)
- Separate logins for VMs will be sent out via email on **November 13**
- Instructions on how to access VMs will soon be published on course web site

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 4.25 of 5 (12 respondents)
- Scores of all evaluation items available at: https://admin.evaliuate.liu.se/search?lang=en

Suggestions on improvements from students:

- "I think that the exam covered a bit too much. I think some vulnerabilities, attacks, mitigations and terms could be skipped"
  - **Action**: Code inspection task moved from exam to seminar this year.
- "I have used many static analysis tools in Go previously but I did not find the static analysis lab to be very helpful. Something more modern and inspirational would be good. The websec lab seemed to be covered by TSIT02 already."
  - **Comment**: Static analysis labs are intended to introduce some of the theory of how tools work "under the hood". We must still cover basics from TSIT02 since all students have not necessarily taken it.

Previous year's course evaluation

Suggestions on improvements from students (continued):

- "The labs for Web Security and Static Analysis could have been more in depth more practical testing […]"
  - **Comment**: Something we will look into, but limited time for labs.
- "In my case, I was lost on some topics of the course due to a lack of seminars. […] add in the future seminars to help student understanding the outcomes of the courses on exercises and to help them carrying out typical exercises which can be found in the final examination. […]"
  - **Action/Comment**: Again, moved the final code inspection question (which many students find challenging to solve "on the spot" during exams) to a seminar instead.
- Lectures need to become better, they are pretty messy and there are multiple parts that are not covered by the lectures or mandatory reading that still ends up on the exam which is not cool
  - **Comment**: Not certain what is meant here. Slides + reading material should cover everything on exam.

Previous year's course evaluation

Positive remarks:

- "The pong lab was very interesting and fun!"
- "Great keeping the on distance lectures."
- "The static lab was very good. Never been on a lab where the lab assistant cared so much that we should understand. Instead of only check our answers we had an explaination of why things worked as they did."
- "Pong is challenging and interesting!"
- "The labs were fun!"
- "The PONG lab was great!"
- "The pong lab was both fun and aided in understanding in how buffer overflows can be exploited and how to indentify other bugs. The structure of the exam is good in my opinion since you need to have an understanding of everything covered in the course, which meant I learned a lot. The content for the course is very broad and give a strong foundation in software security."
- "Lectures were well explained and the labs were very interesting."
- "static analysis lab with Ahmed was fun"


Questions?