Overview of the Applied Crypto course

Daniel Bosk

KTH EECS

19th January 2026

Primitives

Constructions

Primitives

- Block ciphers
- Stream ciphers
- Hash functions
- Message authentication codes
- Digital signatures
- Public key encryption
- Zero-knowledge proofs
- Secure multi-party computation

Constructions

- Key exchange protocols
- Identification and authentication protocols
- Complex constructions from primitives

Example (Primitives: OTP)

- The One-Time Pad is perfectly secure.
- It is a stream cipher.
- It's also malleable—predictable bit flips!

Example (Primitives: RSA and ElGamal)

- RSA and ElGamal are public key encryption schemes.
- They secure the connection to our internet bank.
- Is it a good idea to send

  Enc("Transfer 1000 SEK from 1234 to 1235", k)

  to the bank?

Example (Construction: BankID)

- BankID is a construction based on public key encryption.
- It is used to authenticate users.
- It is also used to sign transactions.
- How should this be constructed to minimize attacks?

Example (Construction: Tor)

- Tor is a construction based on onion routing.
- It is used to anonymize internet traffic.
- It uses crypto a lot.
- How should this be constructed to minimize attacks?

Example (Example: The Signal Protocol)

- The Signal Protocol is used for secure messaging.
- It provides end-to-end encryption for messages.
- Key features include perfect forward secrecy and deniability.
- Widely used in applications like WhatsApp and Facebook Messenger.
- How to do group messaging securely?

Example (Example: Remote Attestation Protocols)

- Remote Attestation validates the integrity of a device.
- It leverages cryptographic proofs to verify software states.
- Commonly used in trusted computing environments.
- Helps in ensuring the device has not been tampered with.
- How can we use this and how can we do it correctly?

Example (Example: Keyless Entry to Cars)

- Keyless entry systems use wireless communication.
- Protocols allow unlocking and starting the vehicle without a physical key.
- Vulnerable to relay attacks if not properly secured.

Intended learning outcomes

After passing the course, the student should be able to:

- use basic terminology in computer security and cryptography correctly
- describe cryptographic concepts and explain their security properties
- find and use documentation of cryptographic libraries and standards
- identify and categorise threats against a cryptographic IT-system at a conceptual level, suggest appropriate countermeasures and present the reasoning to others

Purpose

. . . in order to

- as citizen and engineer be able to discuss applied cryptography in general, and risks of using/developing cryptography in particular
- in professional life and/or research and development project be able to evaluate challenges in software development related to cryptography.

In other words ...

- You should be able to write your own version of BankID.
- You should be able to make well-founded contributions to discussions about things like ChatControl.

In other words ...

- You should be able to write your own version of BankID.
- You should be able to make well-founded contributions to discussions about things like ChatControl.

Lecture format

- Most on Zoom and campus.
- Some flipped (watch videos, read material before class).

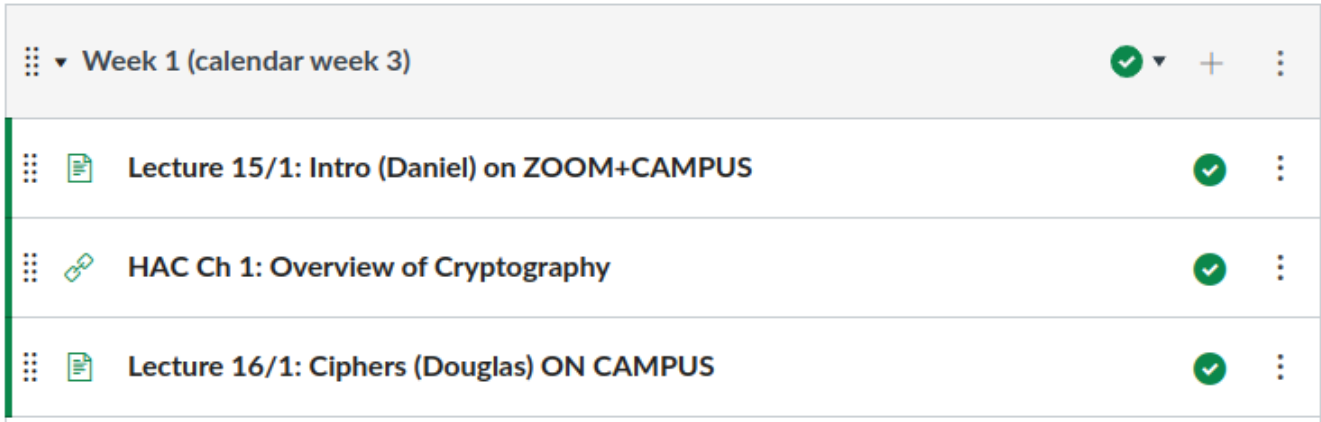

Figure: Screenshot from Canvas showing the lecture mode.

Lecture content

- Crypto primitives
- Practical applications
- Issues and higher level problems

Lecture content (cont.)

- The lectures are complemented by reading material.
- However, not all of them are published yet.

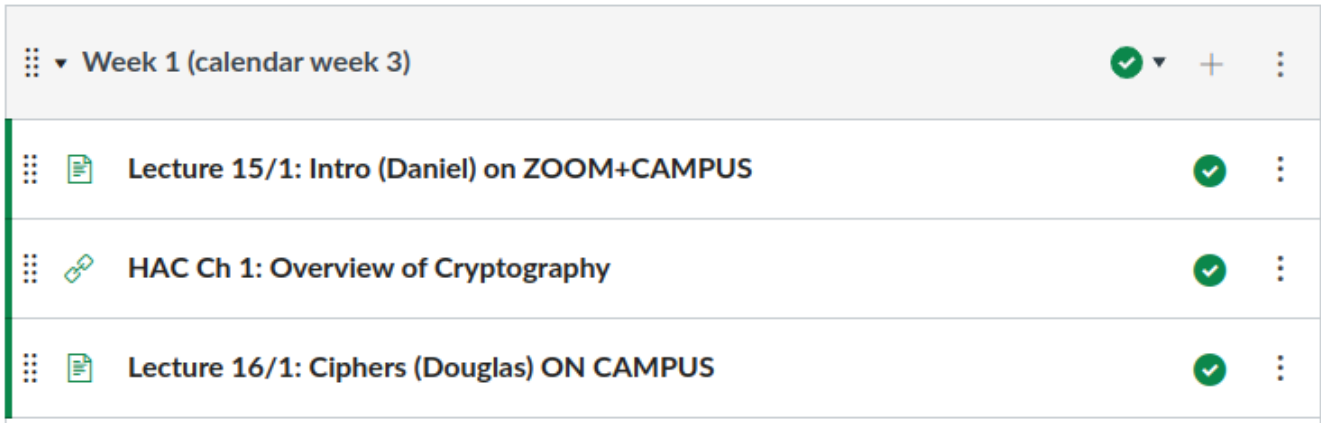

Figure: Screenshot from Canvas showing the lecture mode.

Lecture content (cont.)

- The lectures are complemented by reading material.
- However, not all of them are published yet.

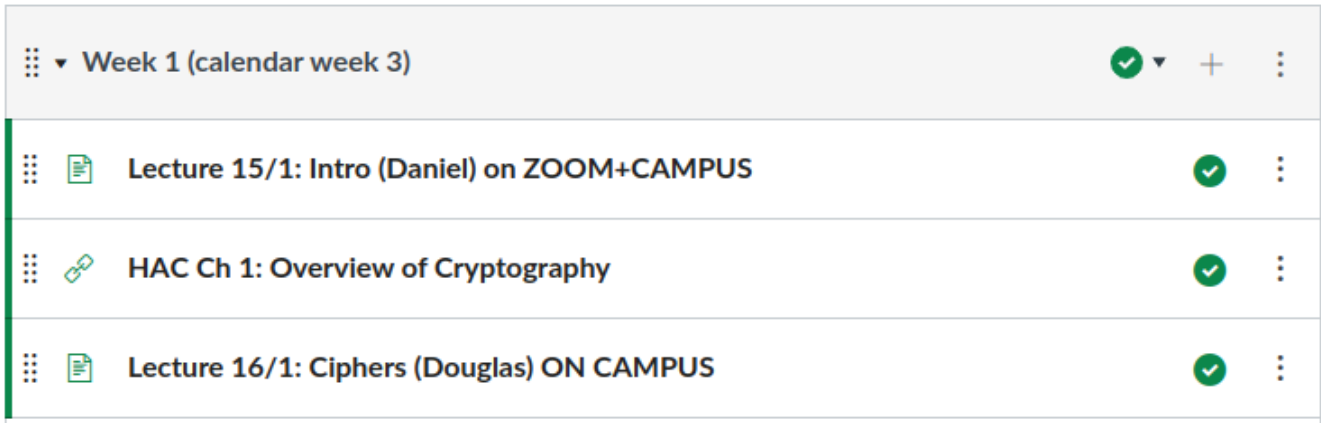

Figure: Screenshot from Canvas showing the lecture mode.

LADOK modules

- LAB1
- INL1

LAB1, mandatory

- Cryptanalysis of Ciphertexts
- Implement AES (Kattis Problem)
- AES presentation
- MANDATORY Seminar: usability (Sonja) ON CAMPUS
- MANDATORY Seminar: Impact considerations around crypto systems (Sonja) ON CAMPUS
- MANDATORY Design Considerations (after the impact considerations seminar)
- MANDATORY Lab: Introduction to ProVerif (Karl and Jesper) ON CAMPUS

LAB1, optional

- Optional: Cryptopals (C, B, A)
- Optional: Side channels (C, B, A)
- Optional: Secure multi-party computation (C, B, A)

Note: Higher grades

- To get a higher grade, you need to do some of the optional assignments.

INL1

- INL1Quiz Cryptographic Concepts, split per lecture
- INL1Written
- INL1Oral

Assignment format

- Most can be done at any time.
- Some have a specified lab session.
  - MANDATORY Seminar: usability (Sonja) ON CAMPUS
  - MANDATORY Seminar: Impact considerations around crypto systems (Sonja) ON CAMPUS
  - MANDATORY Lab: Introduction to ProVerif (Karl and Jesper) ON CAMPUS
- All assignments are individual.

Note: LabWeek

- If you miss, you can catch up in LabWeek in June.

Canvas

What you should be able to to do

- Let's look at INL1Written

Scenario: delivering parcels

- Parcels are delivered when recipients are away.
- A shared parcel box (a 'postbox') stores parcels temporarily.
- A backend coordinates access and notifications.

What must the system achieve?

- Only the right person can retrieve the right parcel.
- The delivery company can operate efficiently.
- Disputes can be resolved ('delivered or not?').
- The system should leak as little metadata as possible.

Example (What can go wrong?)

- Impersonation: attacker pretends to be customer or delivery person.
- Replay: attacker reuses a captured access message.
- Compromise: a box or backend is hacked or tampered with.
- Insider abuse: authorized actors misuse their privileges.
- Privacy leakage: pickup patterns reveal habits, locations, relationships.

Crypto toolbox you need to learn

- Hash functions: identifiers, integrity checks, commitments
- MACs and authenticated encryption: protect tokens and stored data
- Digital signatures: accountability and verifiable records
- Secure channels and key exchange: protect communication with backends
- Key management: provisioning, rotation, revocation, trust anchors
- Randomness and nonces: freshness, replay resistance, safe protocols

Course topics → scenario needs

Confidentiality parcel details and access data

Integrity tokens, logs, and device state

Authentication customer and delivery identities

Authorization who may open which compartment, when

Accountability audit trails and dispute handling

Privacy minimize metadata leakage and linkability