Lecture 1

- What is crypto?
- Example applications
- Encryption

Cryptography: communication and computation in potentially adversarial environments.

Encryption: \( \text{Enc}(0) \approx \text{Enc}(1) \)

Recipient has a SK that makes decryption possible

Authentication/Digital Signatures:
\[ G = \text{Sign}(m) \]
Everyone can verify \((\text{PK}, 6, m)\)
Need SK to issue 6

Zero-knowledge proofs (3-colorable graph example)
Secure
Multiparty computation

E.g., Voters

\[
\begin{align*}
V_1 & \quad \ldots \quad V_i & \quad \ldots \quad V_n \\
\text{Anna} & \quad \text{Forteini} & \quad \text{Forteini}
\end{align*}
\]

Running a proposal
\\
Voters only learn final result!

Application

Toy killer app for crypto: voting
- correctness and verifiability
- anonymity

Requirements

1. Define the problem. Define adversary's capabilities. (Definitional Step)
   Define what counts as a success.
2. Define a solution
3. Prove security of solution / under well-defined assumptions (if any)
   - computational (e.g., \( P \neq NP \))
   - about existing infrastructure (e.g., PKI)
Solution?

1. Voters submit encrypted (under PK_vo) votes by deadline.

2. VO's compute election outcome after the deadline.

\[ PK_vo, d_1 \oplus d_2 \oplus \ldots \oplus d_m = SK_{VO} \]

⇒ not correct! Voters could vote multiple times

\( f_{fix:} \) sign the votes (still anonymous since votes are encrypted)

other issues

• voters may be discriminated
• may add a receipt when a voter votes

Encryption

Alice                  Bob

\[ \text{TRY 0} \]

\[ \text{Alice} \rightarrow \text{Bob} \]

I LOVE YOU BOB

J KAWS ZPV CPC (shift by one)

Lesson 0: There need to be something that Eve doesn't know but A and B do. \( SK \)

\[ \text{TRY 1} \]

\[ SK=10 \]

S VYFO IYE LYL

Lesson 1: Eve can't enumerate over all keys efficiently (?)

\[ \text{TRY 2} \]

use some permutation M ZCLA LCS UCU

Lesson 2: Just because a key is too big to enumerate doesn't mean security!
Definition [Shannon]
Let keyGen, Enc, Dec be a cryptosystem.
Let M be a prob. distr. on the message space.
(keyGen, Enc, Dec) are a secure cryptosystem for M
if \( \forall m \in M, \forall c \)
\[
Pr_m = Pr_{\text{at. keyGen}}[m | c]
\]