Privacy Enhancing Technologies

1. Admin + commitments

Privacy Enhancing Technology

Your first and last class on privacy

Why Privacy?

how to control and choose who sees my data and how

The focus: how to compute on data with "privacy"

The setting: $y_1, \ldots, y_n \leftarrow f(x_1, \ldots, x_n)$

"privacy": the computation "leaks not too much" about the data x_i

Crypto | "Private statistics"

You are here

all parties learn $f(x)$ and nothing else

what if $f(x)$ itself leaks information?

$$\boxed{y_1, \dots, y_n \leftarrow f(x_1, \dots, x_n)}$$

- Encryption : Alice $\xrightarrow{m}$ Bob

  $x_1 = m$ $\qquad$ $x_2 = \perp$

  $$f(m, \perp) = (\perp, m)$$

- Zero knowledge proofs : $P(x, w)$ $\qquad$ $V(x)$

  $$f(\underset{V}{x}, \underset{P}{w}) \to (\underset{\text{"is the proof correct?"}}{0/1}, \perp)$$

- Private ML

  the model $\longleftarrow f\big((x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\big)$

  "train a ML model"

Things we will cover	won't cover
MPC, ZK, SNARKs	modern 2PC
Private reading & writing	Fully Homomorphic Enc
Data reconstructions	anonymous comm
Differential Privacy	private payments
Private ML	e-voting
	lots of stuff about Diff. Privacy

Why are these things not used in practice (yet)?

step 1) Build system to go fast
to be efficient
(no one cares about privacy)

step 2) We really need privacy now!

option 1 $\boxed{\text{Fast} \quad \text{no privacy}}$

option 2 $\boxed{\text{Privacy!} \quad \text{slow}}$

Logistics

spylab.ai/teaching/pets-f24

Grade: 4 homeworks
- at home
- covers ~ 3 lectures
- Latex
- Submit via Gradescope

Collaboration: write who you talked to on HWs

Exo sessions: fill out Moodle question

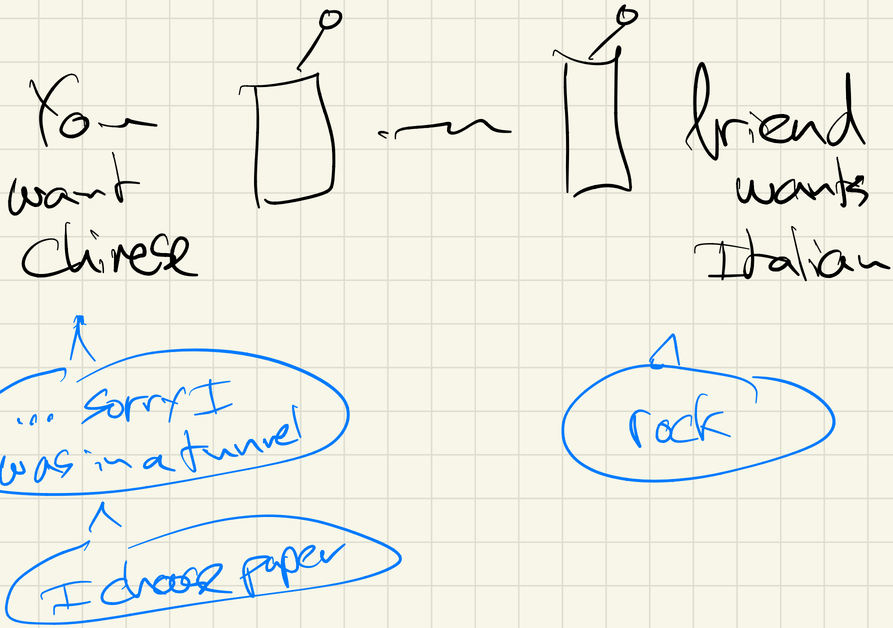
Feedback: google form for anon feedback

We make mistakes!
if anything looks odd or impossible, let me know!

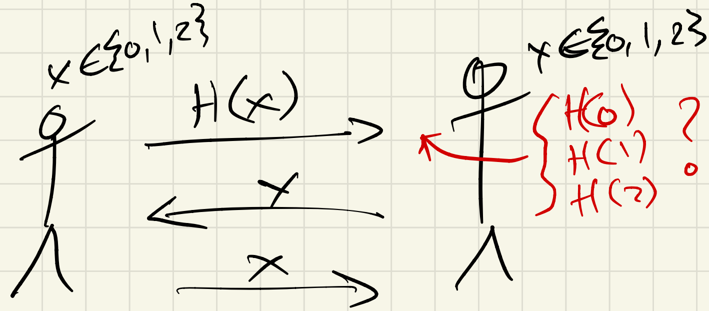
Rock paper scissors over the phone

- Commitment schemes

Setting

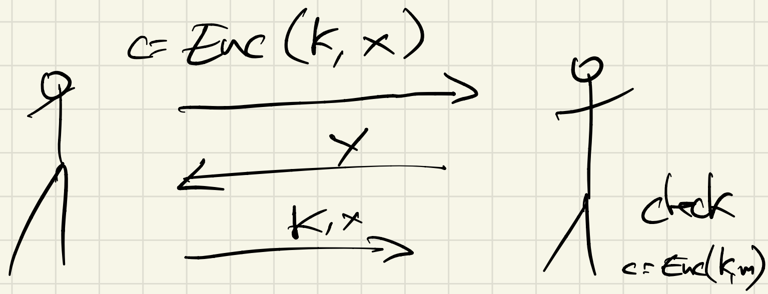


"Solution" 1: Hashing:



a Hash doesn't "hide" x

"Solution" 2: Encryption:



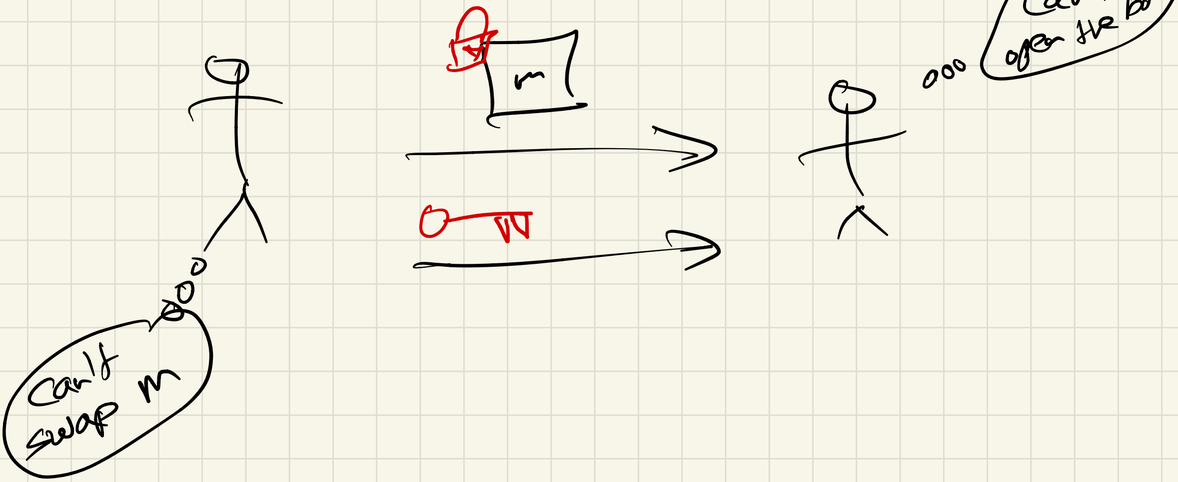
Encryption is not "binding"

$$Enc(k,m) = \underbrace{k \oplus m}_{\text{mod } 3}, \quad k \in \mathbb{Z}_3$$

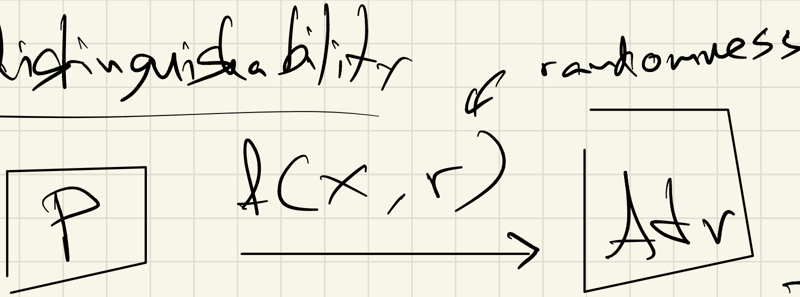
After seeing y, you choose x' and set $k' = c \oplus x'$

Commitment schemes

Informally: "a locked box"



Indistinguishability



$$\{ f(x,r) : r \xleftarrow{\$} R \}$$

↳ a distribution

↳ "the view" of the adversary

what we want: view indistinguishable from "something that is private"

- information-theoretic (statistical)

$$\underbrace{\{f(x,r) : r \xleftarrow{\$} R\}}\ \equiv\ \underbrace{\{f(x',r) : r \xleftarrow{\$} R\}}$$

these Dists are the same

- Computational: For all PPT adversary A:

$$\left| \Pr[A(y)=1 \mid y \leftarrow f(x,r)] - \Pr[A(y)=1 \mid y \leftarrow f(x',r)] \right|$$

$$\leq \mathrm{negl}(n)$$

NOT $\frac{1}{n^2}$, $\frac{1}{n^{100}}$

But $\frac{1}{2^n}$

$o(n^{-c})$ $\forall$ constants c

Definition Commitment scheme

An algorithm $\text{Commit} : \underset{\text{message}}{\mathcal{M}} \times \underset{\text{randomness}}{\mathcal{R}} \to \underset{\text{commitment}}{\mathcal{C}}$

$$\text{Commit}(m, r) = c$$

Properties

Statistical Hiding: $\forall m_0, m_1 \in \mathcal{M}$

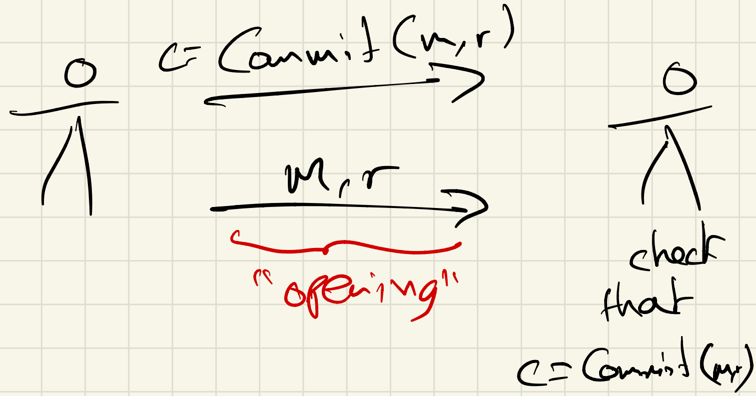
$$\{\text{Commit}(m_0, r) : r \xleftarrow{\$} \mathcal{R}\} \equiv \{\text{Commit}(m_1, r) : r \xleftarrow{\$} \mathcal{R}\}$$

Comp. Binding: No PPT adversary A can find (m_0, r_0), (m_1, r_1) s.t

$$\text{Commit}(m_0, r_0) = \text{Commit}(m_1, r_1)$$

and $m_0 \neq m_1$

Terminology



The simplest commitment scheme

$$\mathrm{Commit}(m, r) = H(m, r)$$

How to prove security?

The Random Oracle Model

1) take a hash function (say SHA-3)

2) "pretend" it is a random function

$$H : X \mapsto Y$$

What's a random function?

$\forall x \in X$, pick $y \xleftarrow{\$} Y$ and define $H(x) = y$

The "controversy": clearly, SHA-3 is not a random function

Why random oracles: proving security is "easy" when H is random
↳ very simple, efficient protocols

leap of faith: hope that using SHA-3 is still ok

Commit $(m, r) = H(m, r)$ ← random oracle

Hiding: if H is random, $H(m, r)$ is uniform over G
the chance Adv would even query $H(m, r)$ is negligible

Binding: a random function is collision resistant

Pedersen Commitments

Setup: let $\mathbb{G}$ be a group of prime order p → think subgroup of $\mathbb{Z}_q^*$ for a prime

$g, h \in \mathbb{G}$, s.t the discrete log between g and h ($g^x = h$) is unknown

Commit:

$$\text{Commit}(m, r) = g^m h^r \in \mathbb{G}$$

$m \in \mathbb{Z}_p$, $r \in \mathbb{Z}_p$

This scheme is linearly homomorphic

$$\text{Commit}(m_1, r_1) \cdot \text{Commit}(m_2, r_2)$$

$$= g^{m_1+m_2} \cdot h^{r_1+r_2}$$

$$= \text{Commit}(m_1+m_2, r_1+r_2)$$

We can compute linear functions over committed values