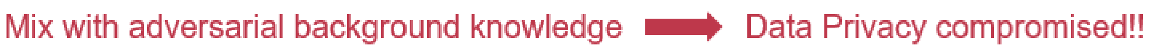
29 data-privacy

Common Attacks

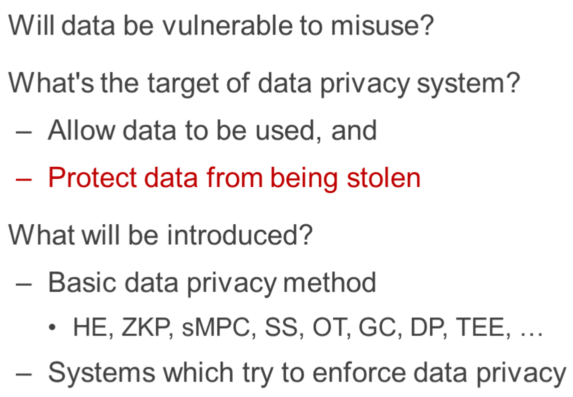
An adversary may learn about data:

- From ciphertext (ciphertext representation-based attack, e.g., order of values)如“是不是密文”、“密文大小”都可以作为信息
- From prior knowledge of data distribution (frequency-count attack) 数据是如何分布的
- From knowledge of frequency of queries (workload-skew attack)
- From the size of the output to a query (output-size attack)
- From the access pattern used by the mechanism in answering a query (access-pattern attack)
- From knowledge of queries that have executed (search-pattern attack)



Mix with adversarial background knowledge → Data Privacy compromised!!

Data Privacy



Will data be vulnerable to misuse?

What's the target of data privacy system?

– Allow data to be used, and

– Protect data from being stolen

What will be introduced?

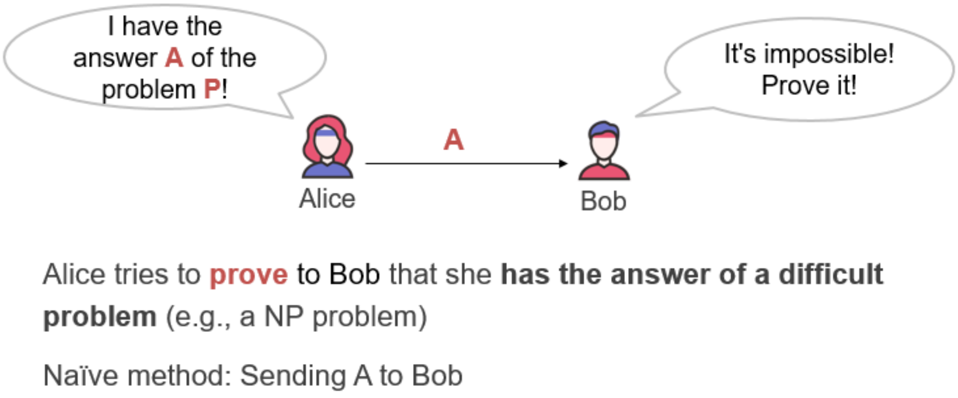
– Basic data privacy method

  • HE, ZKP, sMPC, SS, OT, GC, DP, TEE, …

– Systems which try to enforce data privacy

保护数据的机制有很多，要根据实际需要选择进行选择

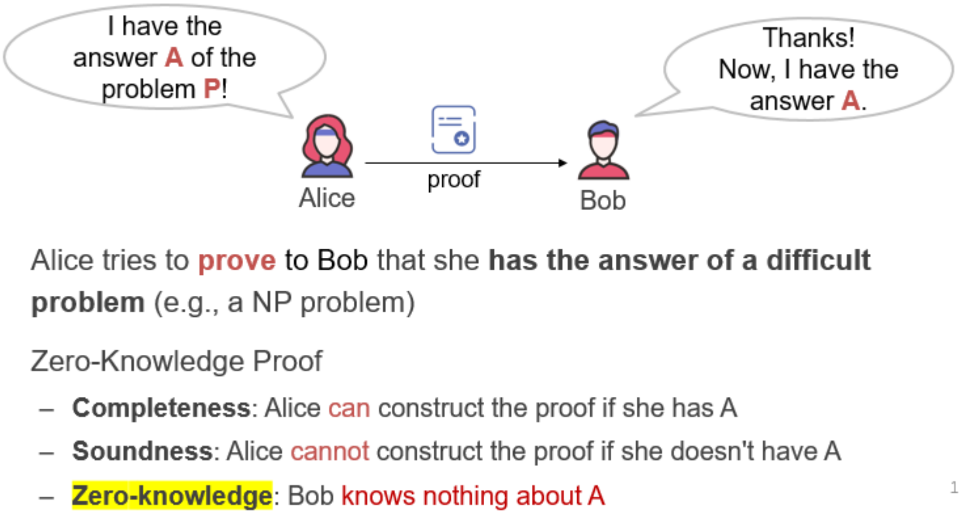
1. **Zero-Knowledge Proof (ZKP)**


I have the answer A of the problem P!
It's impossible! Prove it!
A
Alice
Bob

Alice tries to **prove** to Bob that she **has the answer of a difficult problem** (e.g., a NP problem)

Naïve method: Sending A to Bob

Alice想向Bob证明自己解出来了，但是Alice不能告诉Bob具体解法，于是Alice构造了一个proof给Bob来证明：


I have the answer A of the problem P!
Thanks! Now, I have the answer A.
proof
Alice
Bob

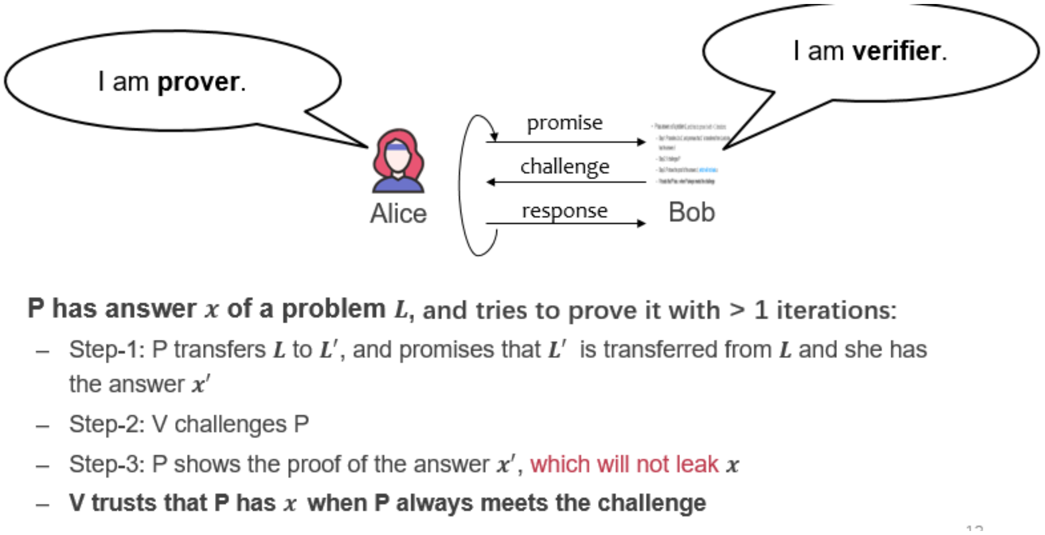
Alice tries to **prove** to Bob that she **has the answer of a difficult problem** (e.g., a NP problem)

Zero-Knowledge Proof

- **Completeness**: Alice can construct the proof if she has A
- **Soundness**: Alice cannot construct the proof if she doesn't have A
- **Zero-knowledge**: Bob knows nothing about A

1

交互式的ZKP



P has answer x of a problem L, and tries to prove it with > 1 iterations:

- Step-1: P transfers L to L', and promises that L' is transferred from L and she has the answer x'
- Step-2: V challenges P
- Step-3: P shows the proof of the answer x', which will not leak x
- **V trusts that P has x when P always meets the challenge**

2. **Private Information Retrieval (PIR)**

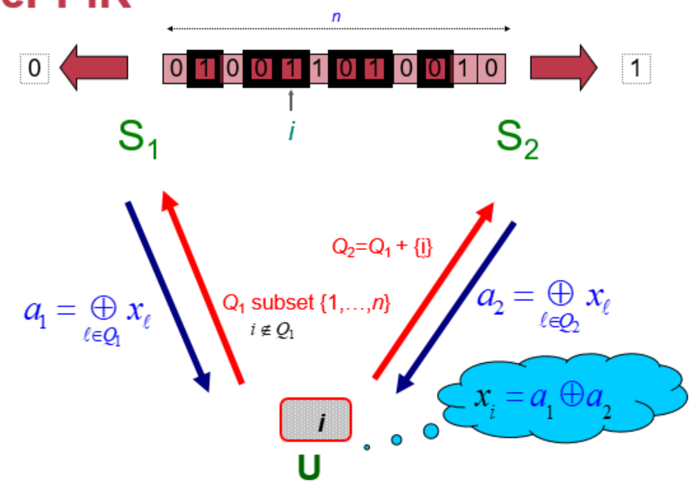
   Alice想向server查询信息，但是不想让server知道自己查了什么东西

   两种PIR：

   Information-Theoretic PIR

   - Replicate database among k servers
   - User queries all the servers

2-Server PIR


n
0
0 1 0 0 1 1 0 1 0 0 1 0
1
S_1
i
S_2
$Q_2=Q_1 + \{i\}$
$a_1 = \bigoplus_{\ell \in Q_1} x_\ell$
Q_1 subset $\{1,\ldots,n\}$
$i \notin Q_1$
$a_2 = \bigoplus_{\ell \in Q_2} x_\ell$
$x_i = a_1 \oplus a_2$
i
U

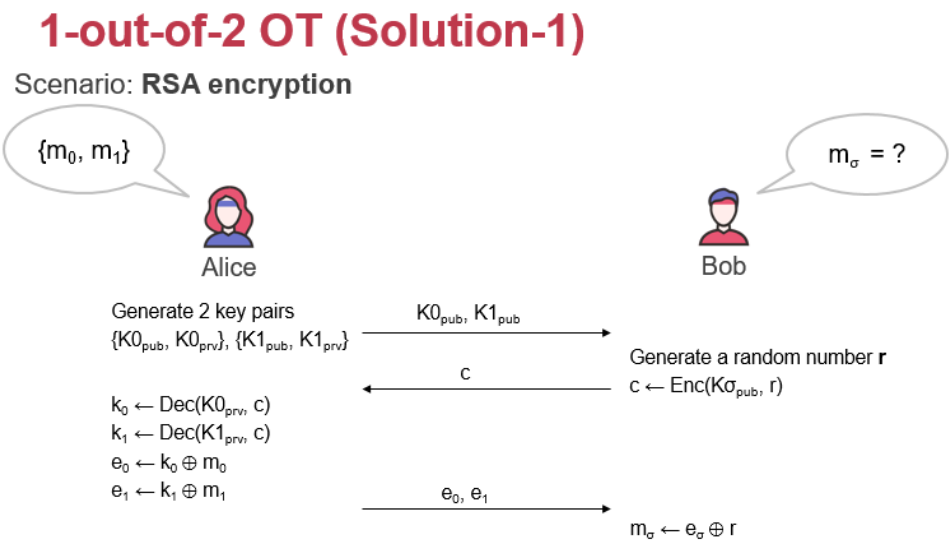
Notice: Servers should not collude!

Computational PIR

- Computational privacy, based on cryptographic assumptions

3. **OT**

向Bob推荐东西，让他在里面挑一个，但是又不能让Alice知道Bob拿了哪个



1-out-of-2 OT (Solution-1)

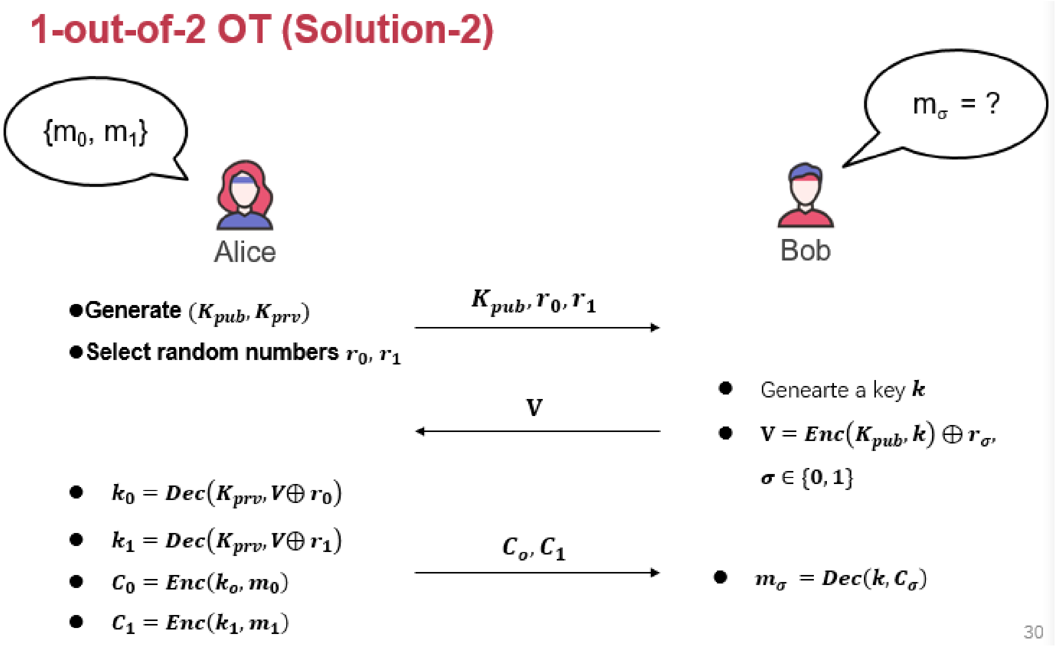
Scenario: **RSA encryption**

Generate 2 key pairs
$\{K0_{pub}, K0_{prv}\}$, $\{K1_{pub}, K1_{prv}\}$

Generate a random number **r**
$c \leftarrow Enc(K\sigma_{pub}, r)$

$k_0 \leftarrow Dec(K0_{prv}, c)$
$k_1 \leftarrow Dec(K1_{prv}, c)$
$e_0 \leftarrow k_0 \oplus m_0$
$e_1 \leftarrow k_1 \oplus m_1$

$m_\sigma \leftarrow e_\sigma \oplus r$



1-out-of-2 OT (Solution-2)

- **Generate** (K_{pub}, K_{prv})
- **Select random numbers** r_0, r_1

- Genearte a key $\boldsymbol{k}$
- $V = Enc(K_{pub}, k) \oplus r_\sigma$, $\sigma \in \{0, 1\}$

- $k_0 = Dec(K_{prv}, V \oplus r_0)$
- $k_1 = Dec(K_{prv}, V \oplus r_1)$
- $C_0 = Enc(k_o, m_0)$
- $C_1 = Enc(k_1, m_1)$

- $m_\sigma = Dec(k, C_\sigma)$

30

Alice分别用两个解密结果加密她的两份消息，将加密结果送给Bob （Ci=E(ki, mi) i=0,1）
Bob用k解密两份密文，得到需要的秘密 s

More OT Protocols:

- Different numbers of selected messages

  1-out-of-2 OT

  1-out-of-n OT

  k-out-of-n OT

- Implementation method

  Non-adaptive OT

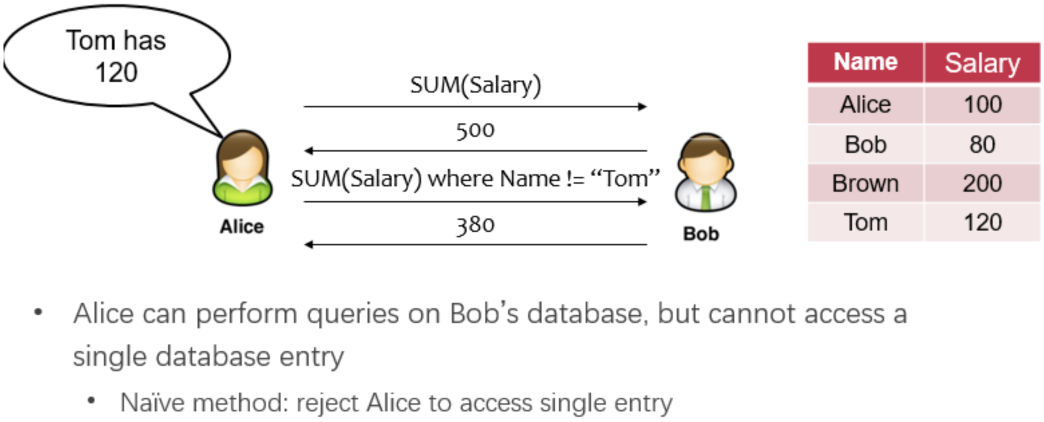
  Adaptive OT

  Publicly Verifiable OT

  …

4. **DP**

存在问题：虽然不能直接得到Tom的工资，但是可以根据多个平均数算出来

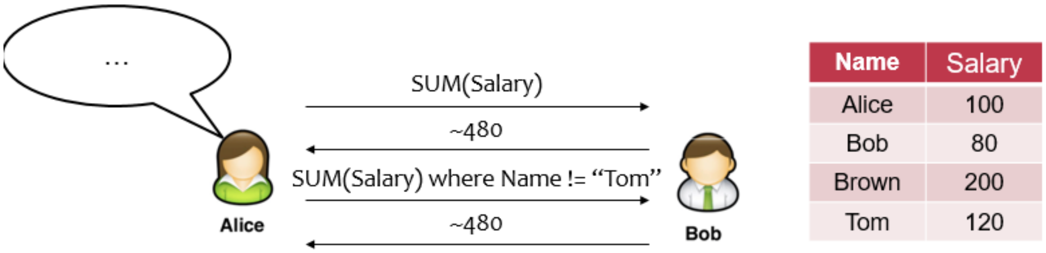


Name	Salary
Alice	100
Bob	80
Brown	200
Tom	120

- Alice can perform queries on Bob's database, but cannot access a single database entry
  - Naïve method: reject Alice to access single entry

解决方法：给计算函数加noisy（只给个大致范围）

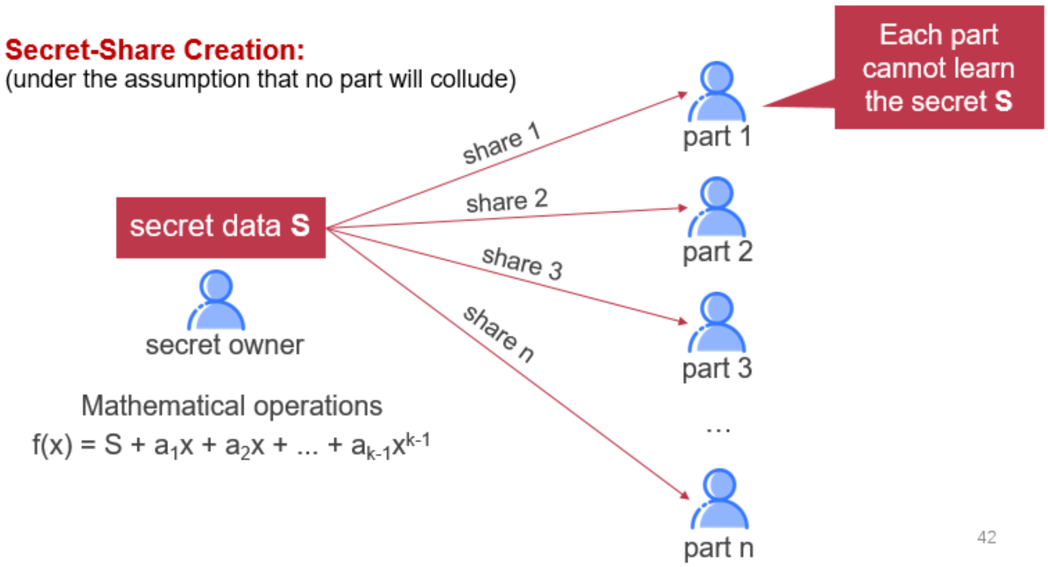
Existing Mechanism：Laplace mechanism、Gaussian mechanism



Name	Salary
Alice	100
Bob	80
Brown	200
Tom	120

5. **Secret Sharing**

将重要数据分区的好处：安全，一块数据丢了不会把全部数据丢了；容错



Secret-Share Creation:
(under the assumption that no part will collude)

secret data **S**

secret owner

Mathematical operations
$f(x) = S + a_1x + a_2x + \ldots + a_{k-1}x^{k-1}$

share 1
share 2
share 3
share n

part 1
part 2
part 3
…
part n

Each part cannot learn the secret **S**

42

6. secure Multi-Party Computation (sMPC) 多方安全计算

Semi-honest adversary

- Each party must **follow the protocol**

Generic protocol

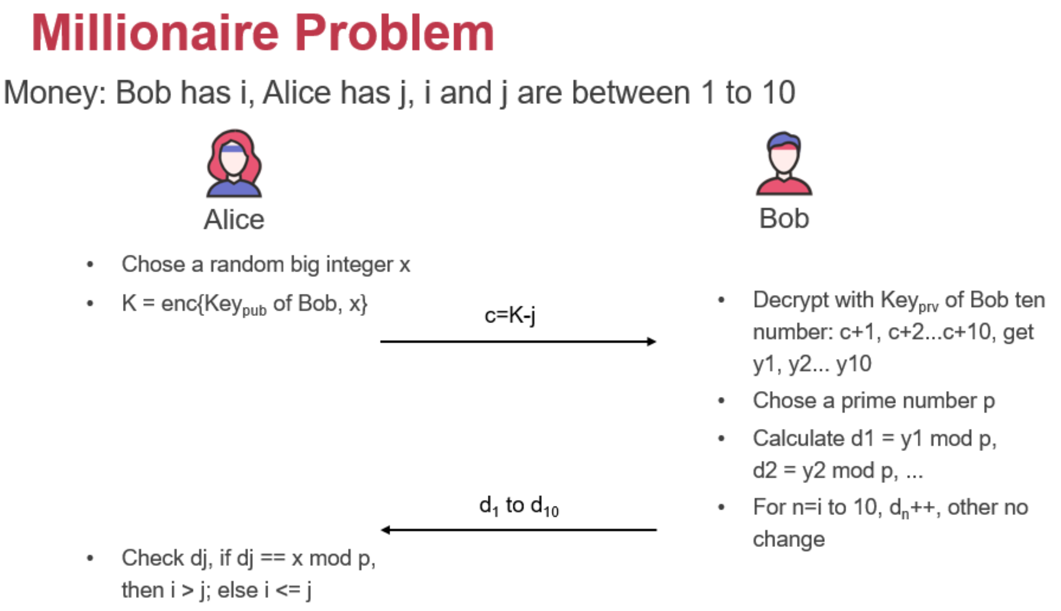
- Can securely compute **any functionality**

Multi-party computation

- Secret sharing

2-party computation

- GC(Garbled Circuits) + OT(Oblivious Transfer)



Millionaire Problem

Money: Bob has i, Alice has j, i and j are between 1 to 10

Alice

- Chose a random big integer x
- K = enc{Key_{pub} of Bob, x}

c=K-j

Bob

- Decrypt with Key_{prv} of Bob ten number: c+1, c+2...c+10, get y1, y2... y10
- Chose a prime number p
- Calculate d1 = y1 mod p, d2 = y2 mod p, ...
- For n=i to 10, d_n++, other no change

d_1 to d_{10}

- Check dj, if dj == x mod p, then i > j; else i <= j

下面Bob还要把p传给Alice的

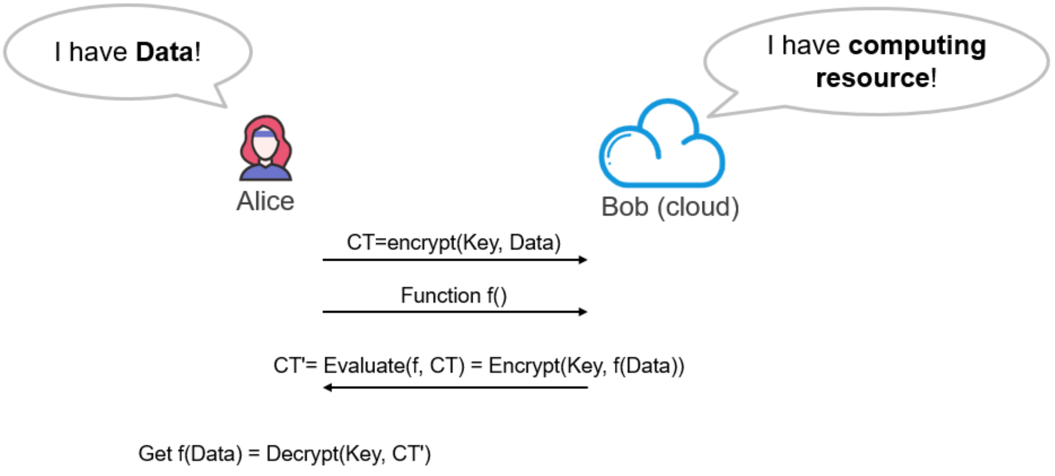
将财产数目一位一位PK，比谁钱多。这里K很大，y1~y10里面一定有一个是K。

7. Garbled Circuits（混淆电路）

   假设甲有数字，乙有数字 ，他们在不向对方披露自己数字的情况下，共同计算一个二元函数。主要过程：

   预处理：将函数转换成电路
   步骤一：将电路乱码化
   步骤二：忘性传输（OT）
   步骤三：执行乱码电路

8. Homomorphic Encryption(HE) 同态加密



CT=encrypt(Key, Data)

Function f()

CT'= Evaluate(f, CT) = Encrypt(Key, f(Data))

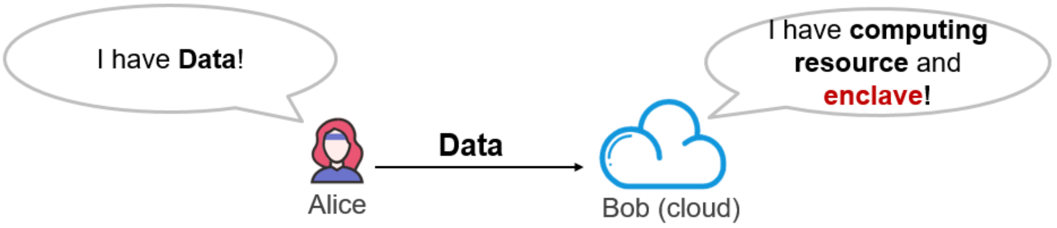
Get f(Data) = Decrypt(Key, CT')

SWHE and FHE

- HE: Homomorphic Encryption
  - $Enc(f(m_1, m_2)) = Eval_f(Enc(m_1), Enc(m_2))$
- SWHE: Somewhat Homomorphic Encryption
  - Support **limited** kinds and times of operation
  - $f(m_1, m_2) = m_1 \cdot m_2$ (e.g., RSA)
  - $f(m_1, m_2) = m_1 + m_2$ (e.g., Paillier)
- FHE: Full Homomorphic Encryption
  - Support all kinds of operations
  - Addition and multiplication

RSA是公私钥的加密算法；存在保序加密算法；全同态加密还未实现

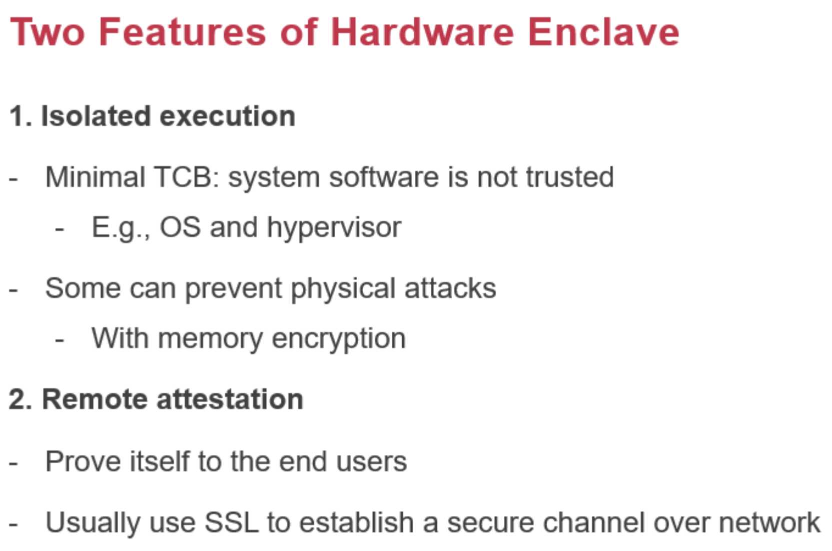
9. Hardware Enclave



Alice wants to ask Bob (e.g., a cloud) to perform calculation on her data

~~Naïve method: Sending Data to Bob~~

Bob cloud construct an enclave



Two Features of Hardware Enclave

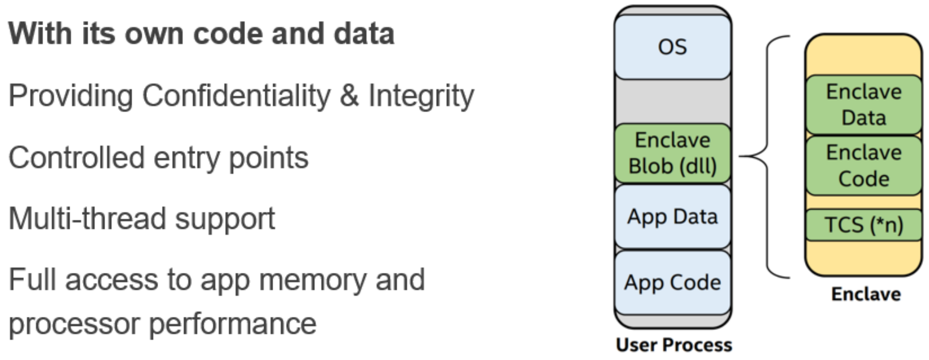
1. Isolated execution

- Minimal TCB: system software is not trusted
  - E.g., OS and hypervisor
- Some can prevent physical attacks
  - With memory encryption

2. Remote attestation

- Prove itself to the end users
- Usually use SSL to establish a secure channel over network

安全三个特点：CIA；HE只能做到CI，除了断电都不能阻止计算



With its own code and data

Providing Confidentiality & Integrity

Controlled entry points

Multi-thread support

Full access to app memory and processor performance

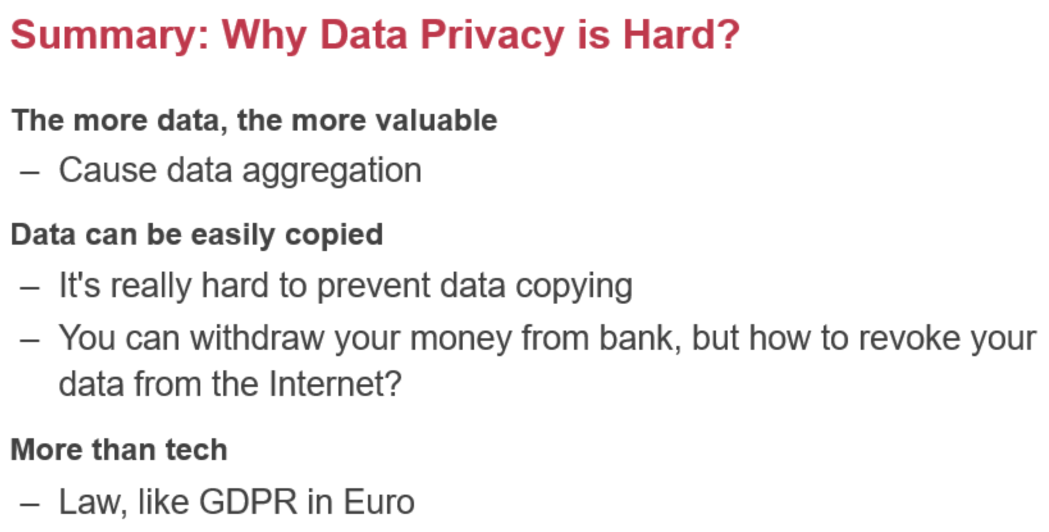
CPU和任何应用程序都是禁止访问Enclave区域的

内存里的数据进入CPU要进行解密，同样的，cache进入内存也要加密

CPU有一个独特的私钥而且不能改，Intel维护，云产商不能拿到CPU的密钥

Enclave是一个被保护的内容容器, 用于存放应用程序敏感数据和代码。SGX允许应用程序指定需要保护的代码和数据部分, 在创建enclave之前, 不必对这些代码和数据进行检查或分析, 但加载到enclave中去的代码和数据必须被度量，并保护它们不被外部软件所访问。Enclave可以向远程认证者证明自己的身份, 并提供必需的功能结构用于安全地提供密钥。用户也可以请求独有的密钥, 这个密钥通过结合enclave身份和平台的身份做到独一无二, 可以用来保护存储在enclave之外的密钥或数据。

总结



Summary: Why Data Privacy is Hard?

The more data, the more valuable

– Cause data aggregation

Data can be easily copied

– It's really hard to prevent data copying

– You can withdraw your money from bank, but how to revoke your data from the Internet?

More than tech

– Law, like GDPR in Euro