密码学进阶

第三课：可证明安全

授课老师：张秉晟

bingsheng@zju.edu.cn

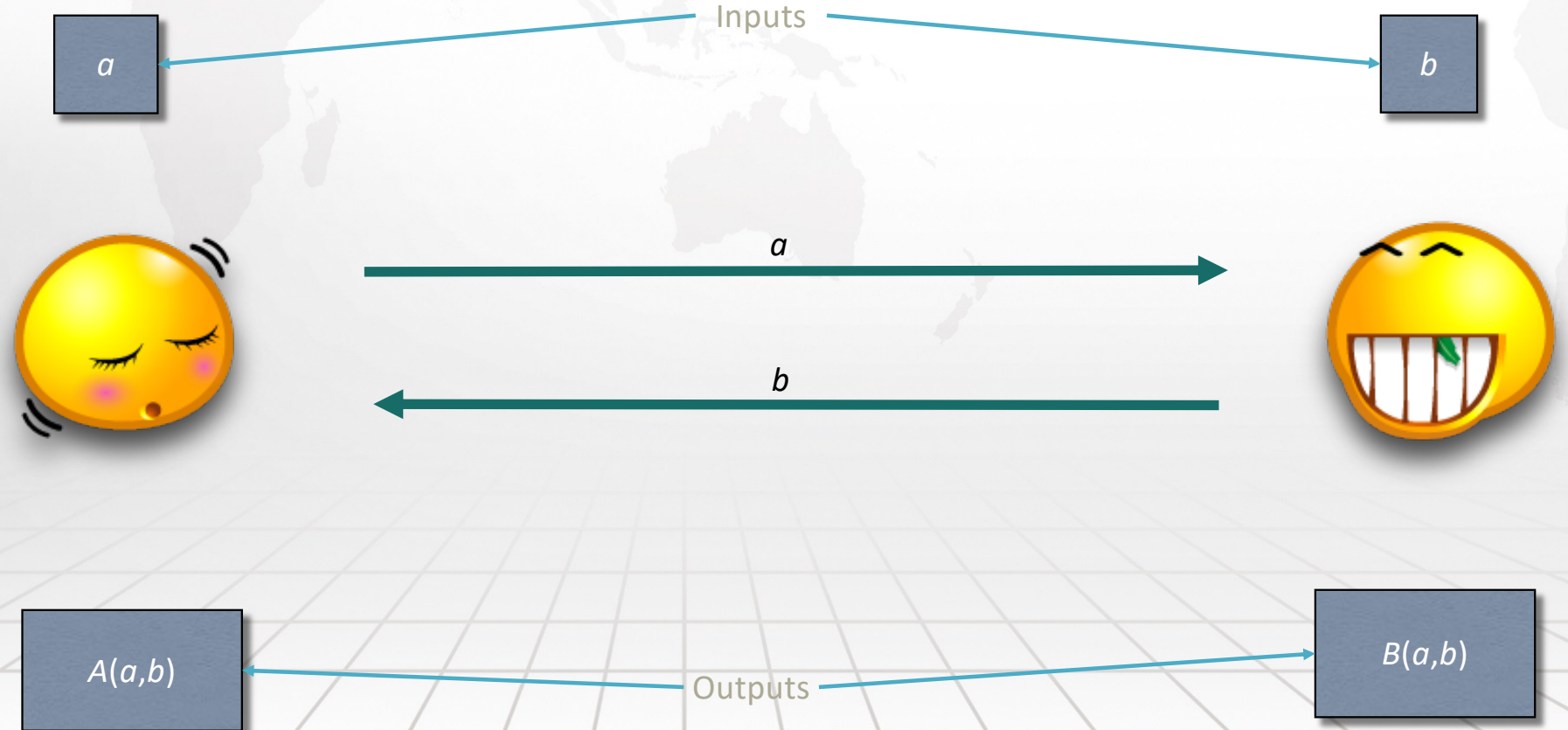
教学安排

- 课程安排：
  - 第一周：密码学基础（Cryptography Review）
  - 第二周：可证明安全（Provable Security）
  - 第三周：安全多方计算（MPC）
  - 第四周：零知识证明（ZK）
  - 第五周：隐私保护求交集（PSI）
  - 第六周：隐匿查询（PIR）
  - 第七周：加密数据库（CryptoDB）
  - 第八周：区块链共识（Consensus）

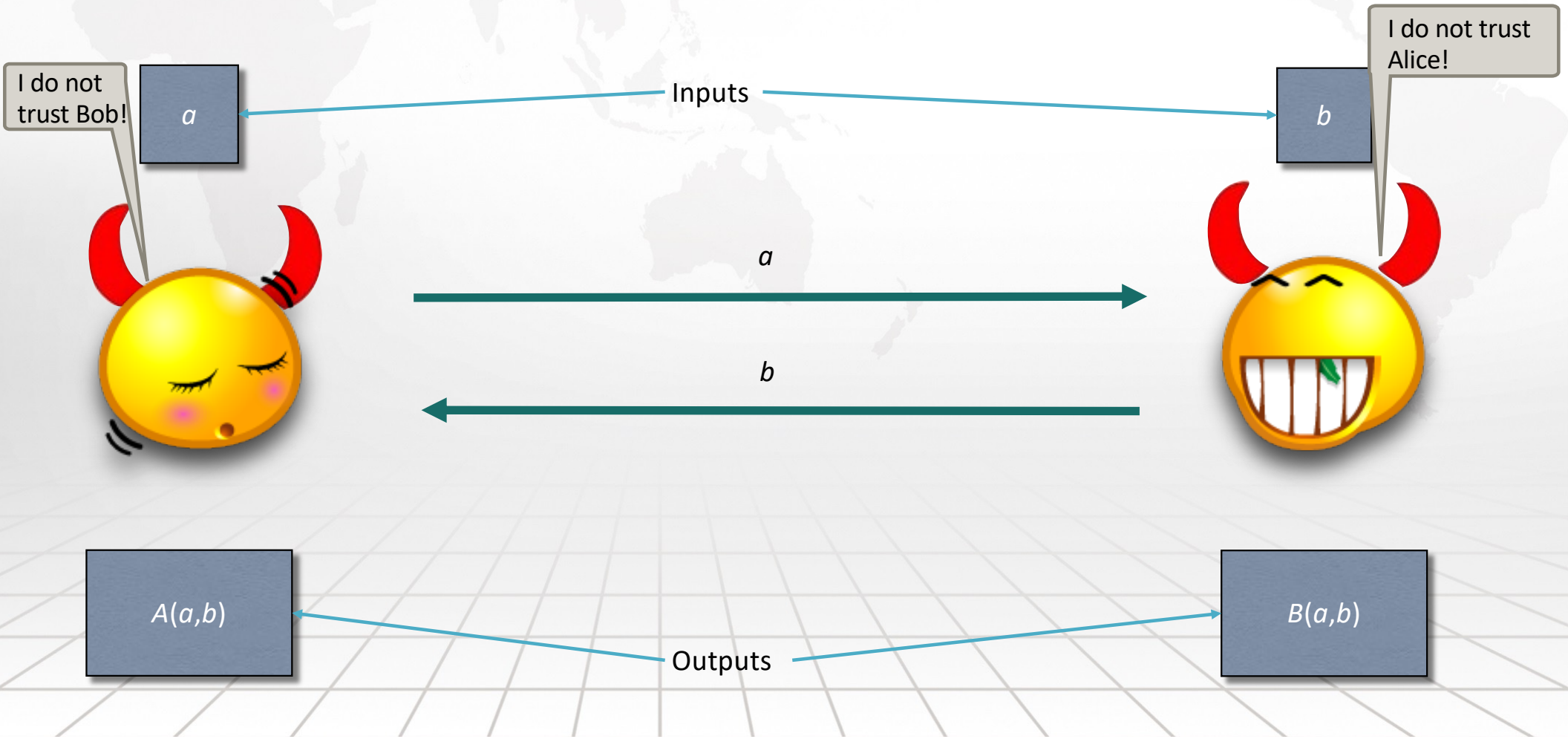
How to understand security and assumptions

protocol

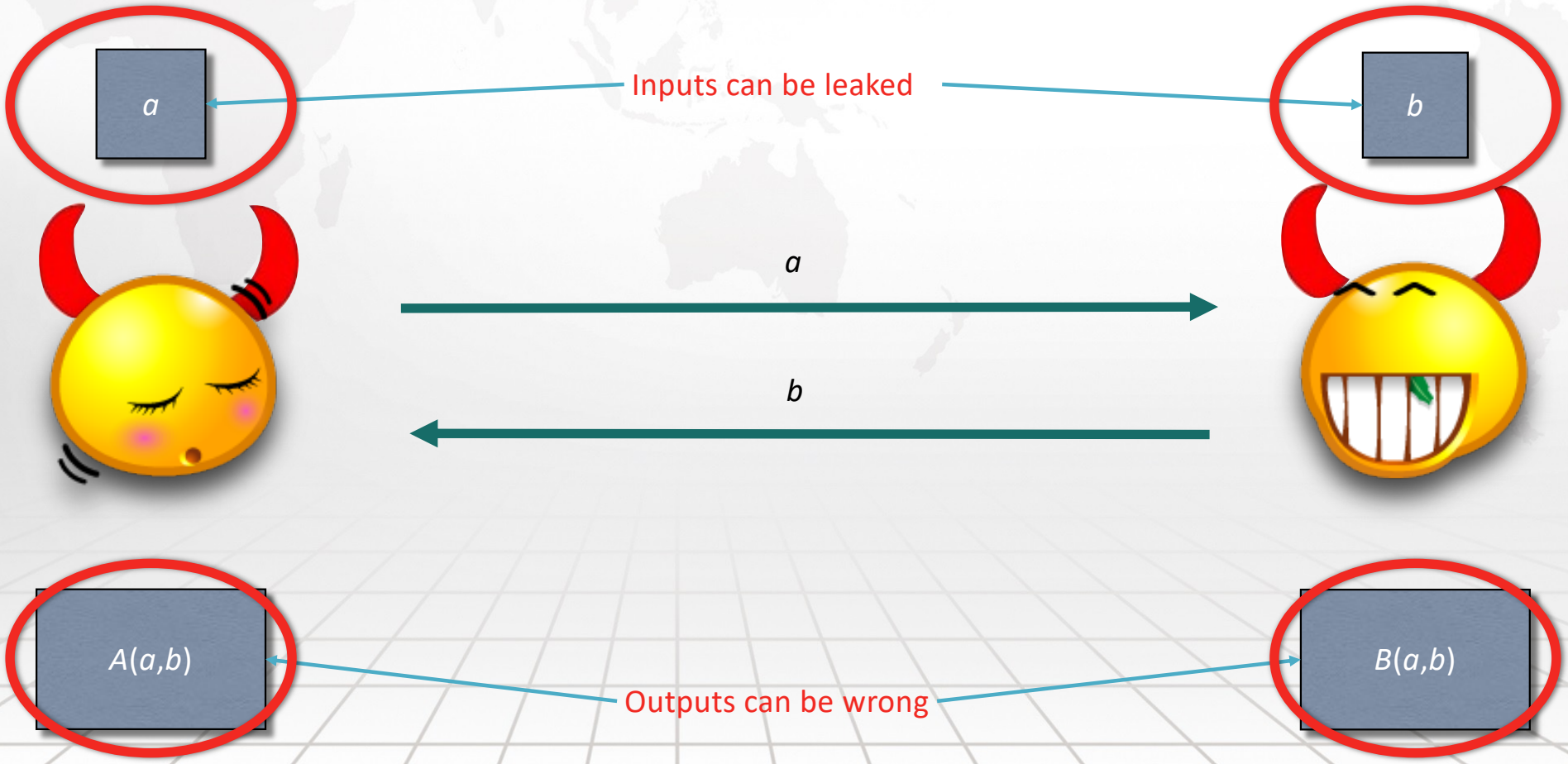
Easy if both are honest!



protocol


I do not trust Bob!
a
Inputs
b
I do not trust Alice!
a
b
A(a,b)
Outputs
B(a,b)

Quiz: what can go wrong?



Group

- Group G is a set with an operation $\cdot$ that satisfies:
  - (associativity) for each $a, b, c \in G, (a \cdot b) \cdot c = a \cdot (b \cdot c)$
  - (existence of unit element) there exists $1 \in G$ such that for each $a \in G, 1 \cdot a = a \cdot 1 = a$
  - (existence of inverse): for each $a \in G$, there exists $a^{-1} \in G$ such that $a \cdot a^{-1} = a^{-1} \cdot a = 1$

example about $\mathbb{Z}n$

- Consider $\mathbb{Z}_4 = \{0, 1, 2, 3\}$
- For example, $1 \cdot 1 = 1, 3 \cdot 3 = 9 \equiv 1 \pmod 4$
- Thus 1 and 3 are invertible, but 0 and 2 are not
- This means $\mathbb{Z}_4^* := \{1, 3\}$ consists of invertible elements
- Since $\mathbb{Z}_4^* \subset \mathbb{Z}_4$ then $\mathbb{Z}_4^*$ is a group

	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	0	2
3	0	3	2	1

exponentiation

- Let G be a multiplicative group
- Define $g^s := g \cdot g \ldots \cdot g$
  - **= exponentiation**
- Also $g^0 = 1$, and $g^{-s} = (g^{-1})^s$

Facts about groups

- For group G, its **order** $\text{ord}(G) = |G|$
- For $g \in G$, let its **order** $\text{ord}(g) =$ smallest positive $s \in \mathbb{Z}$ such that $g^s = 1$
- $\text{ord}(g)$ is well defined for finite groups:
  - g^s can take up to $|G|$ different values.
  - If $g^s = g^t$ and $s > t$ then $g^{s-t} = 1$, and thus $\text{ord}(g) \leq s - t$

Facts about groups

- For $g \in G$, $\langle g \rangle = \{h \in G: \exists s \text{ such that } h = g^s\} = \{g^s: s \in \mathbb{Z}\}$
  - $\langle g \rangle$ is group **generated by** g, g is **generator** of $\langle g \rangle$
  - (smallest) s is the **discrete logarithm** of h on basis g
- G is **cyclic** iff for some $g \in G$, $G = \langle g \rangle$
- In a cyclic group, every element has a unique discrete logarithm on basis a generator g

abstraction

- In the rest of the course, we talk about two possible cryptographically interesting instantiations of groups
- However, after that we just abstract details away and assume that we have a finite (mostly) cyclic group

Ring

- A **ring** is a set R with two operations, $+$ and $\cdot$ that satisfy the following requirements
  - $(R, +)$ is an (additive) group
  - $(R, \cdot)$ is a **monoid**: it is associative, has unit element 1
    - but every element is not required to have an inverse
  - distributivity: $a(b + c) = ab + ac$, and $(a + b)c = ac + bc$

Examples of rings

- $(\mathbb{Z}, \cdot, +)$
- $(\mathbb{Z}n, \cdot, +)$
- All univariate polynomials $f(x) = f_0 + f_1x + \ldots + f\mathrm{d}\ (x^{\mathrm{d}})$ with coefficients from either $\mathbb{Z}$ or $\mathbb{Z}n$
- etc, etc
- In all rings we encounter, both $\cdot$ and $+$ are commutative

Crypto in rings/groups

- Crypto in groups is much less useful than in rings
- **Basic reason:** we can only compute additions (and multiplications with scalars), so only implement affine functions / degree-1 polynomials $f(x) = ax + b$
- However, group-based crypto is much better known, so we mostly talk about this
- ... and it actually allows to do a lot of things

Provable Security

- Assume Alice designs a protocol
- How to make sure it is secure?
- **Approach 1:** proof by intimidation
  - "don't you trust me"
  - also "tautology"
  - **Theorem.** Assume Protocol X is secure. Then Protocol X is secure

motivation

- **Approach 2 (much better):**
  - prove that the protocol is secure

Unconditional security

- Problem:
  - only known how to do for a small number of protocols
  - need major advances in complexity theory
  - it is not known how to prove that any function takes more than a linear number of steps to compute

motivation

- **Approach 3 (mostly correct):**

Computational security

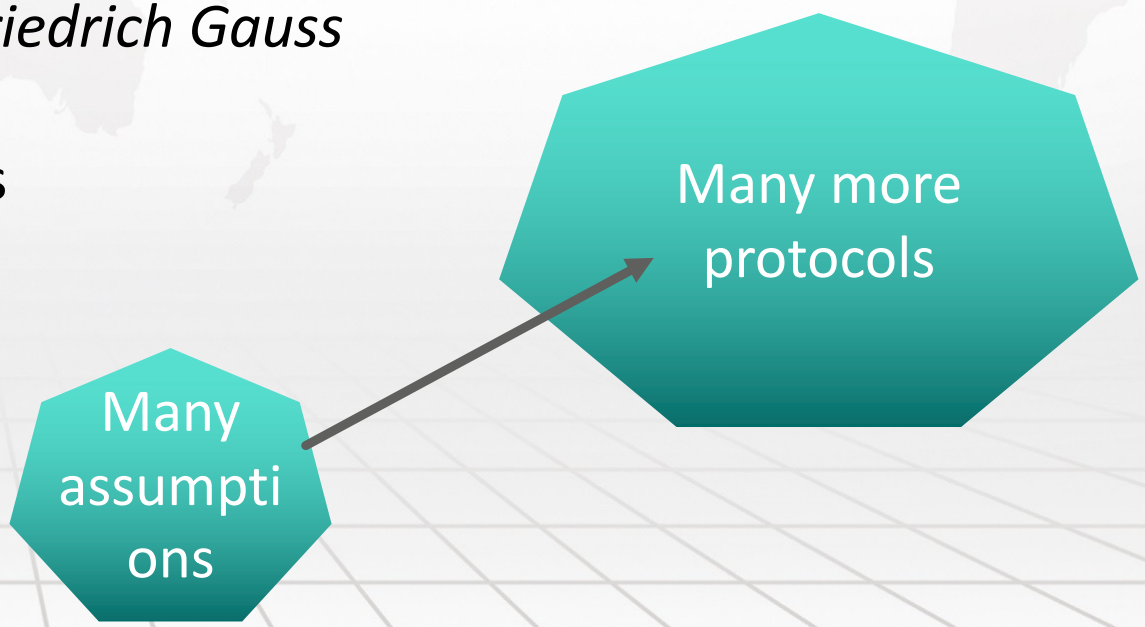
  - make an assumption:
    - assume that some well-known problem (say, factoring) is hard
  - prove that if that assumption holds, then the protocol is secure

security verification

	Protocol designer's task	Security verifier's task
proof by intimidation	Simpler: no need to prove anything	Spend years cryptanalyzing OR trust the protocol
proof by reduction	More complex: must reduce security to some assumption	Verify the usually short reduction. Trust the assumption

some known assumptions

- Factoring and friends
  - *The problem of distinguishing prime numbers from composite numbers and of resolving the latter into their prime factors is known to be one of the most important and useful in arithmetic.*
    ~~~~~~~ Carl Friedrich Gauss
- Discrete logarithm and friends
- Lattice assumptions
- Coding-theoretic assumptions
- …



Choice of assumption: tradeoff

- More security assurance or better efficiency?
- Quantum security?
- Number-theoretic flavor?
  - Any algebra needed by the goal?
- Compatibility with other protocols
- Some protocols are impossible/very inefficient with some assumptions

E.g., need to compute a sum of inputs

Assumption = everything

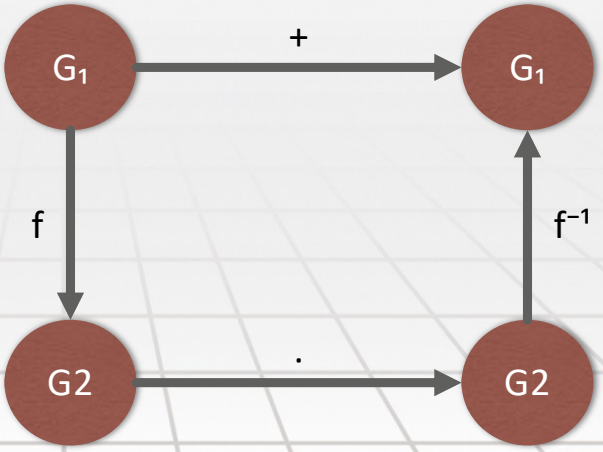
- Choice of the assumption is very important
- Seeing an assumption already gives a flavor of what can(not) be done efficiently
- So let's learn some

Recall: isomorphism

- Assume G_1 is additive and G_2 is multiplicative group
- $f : G_1 \rightarrow G_2$ is **isomorphism** if it agrees with group operations
  - $f(x + y) = f(x) f(y)$
  - $f(0) = 1$
  - $f(-x) = f(x)^{-1}$
- Two groups are **isomorphic** if there exists isomorphism between them

isomorphic = equal

- In mathematics, isomorphic groups are considered to be "essentially the same"
  - They have the same structure
- Instead of executing group operation in one group, you can map to another group, do group operation there, and then map back



f can be thought of as data representation

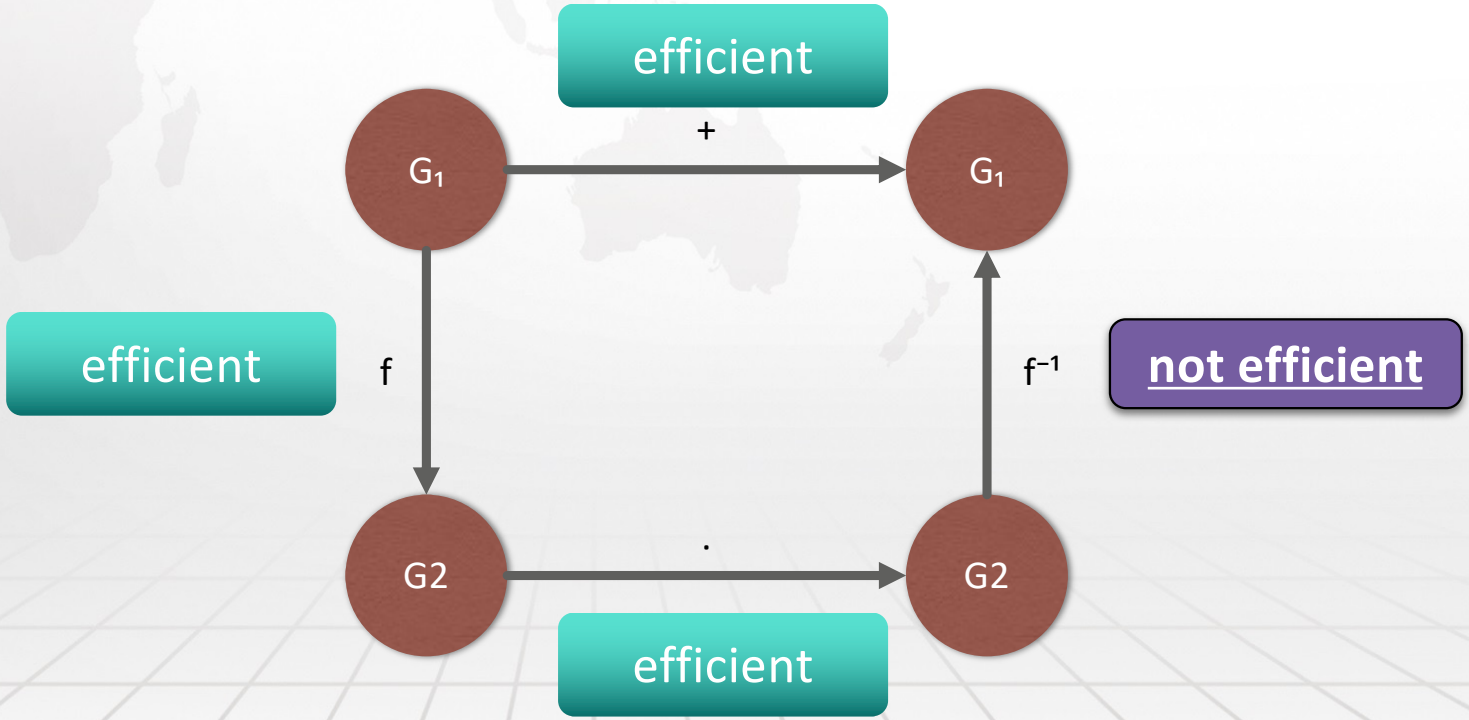
... assuming both *f* and f^{-1} can be computed efficiently

one-way isomorphisms

- $f : G_1 \to G_2$ is a **one-way isomorphism** if
  - f is an isomorphism
  - f can be computed efficiently
  - f^{-1} cannot be computed efficiently

assumption (not known how to prove such things)

One-way isomorphic ≠ equal



Recall: exponentiation

- $\exp_g : \mathbb{Z}q \rightarrow G$, $\exp g\ (m) = g^m$
- $\exp_g$ is isomorphism
  - $g^{m+n} = g^m\ g^n$
  - $g^0 = 1$
  - $g^{-m} = 1 / g^m$

QUIZ: Is Exp one-way?

- What do you think?
  - Depends on the group
- **Easy:**
  - $(\mathbb{R}^*, \cdot)$: the inverse of exp is logarithm
  - $(\mathbb{Z}, +)$, $(\mathbb{Z}q, +)$: exp = multiplication, inverse = division
- In finite groups, inverse of exp is called **discrete logarithm**

Email: bingsheng@zju.edu.cn