公開金鑰基礎結構

Public-key Infrastructure (PKI)

- The biggest challenge in PKC is ensuring the authenticity of public keys
- Certificates is used to help authenticate public keys6/12/2008
- PKI is a secure system that is used to manage and control certificates

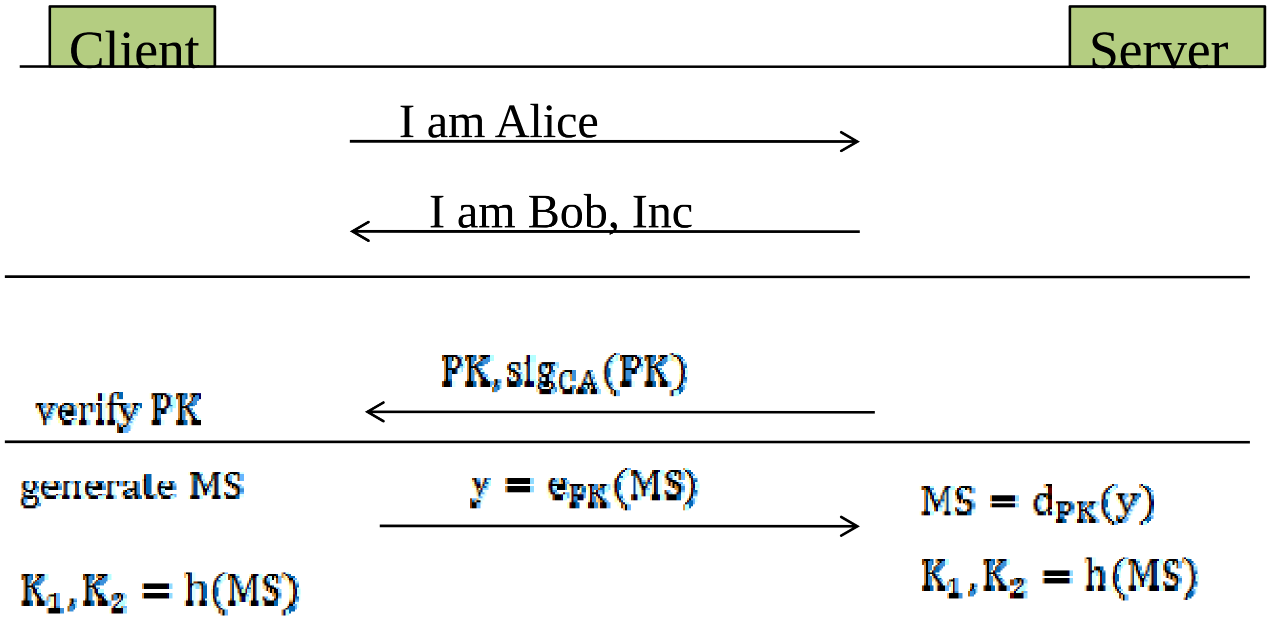
- A PKI is the basis of a pervasive security infrastructure whose services are implemented and delivered using public-key concepts and techniques
- It is an infrastructure
- It is a software on users' computers
- Users might not even be aware of the PKI-related procedures

Components of PKI

- Certificate issuance
- Certificate revocation
- Key backup/recovery/update
- Time stamping
- Secure communication
- Access control
- …

A practical Protocol : Secure Socket Layer (SSL)

- An SSL session is used to facilitate online purchases from a company's web page using a web browser



- K_1 is used to authenticate date using MAC
- K_2 is used to encrypt and decrypt data

- Only the serve is required to supply a certificate
- The client may not even have a pk

Certificates

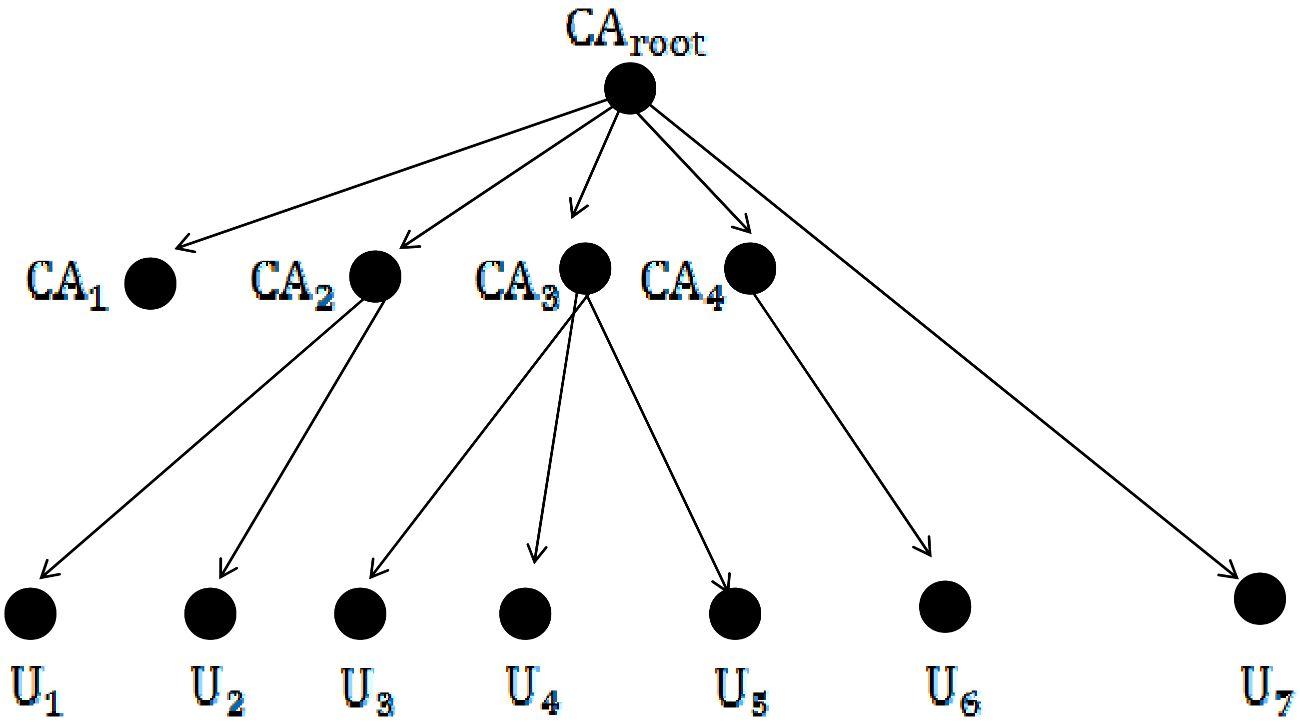
- A certificate binds an ID to a public key
- Using done by having a trusted authority (a certification authority, CA) sign the information on a certificate
- Everyone can access to the PK of the CA

PKI Trust Models

- Hierarch Model
- Networked PKIs
- The Web Browser Model
- PGP

Hierarchy Model

- The root CA has a self-signed, self-issued certificate
- The root CA issues certificates for lower-level CAs
- Any CA can issue certificates for end users



$x \longrightarrow y$: X signed a certificate for y

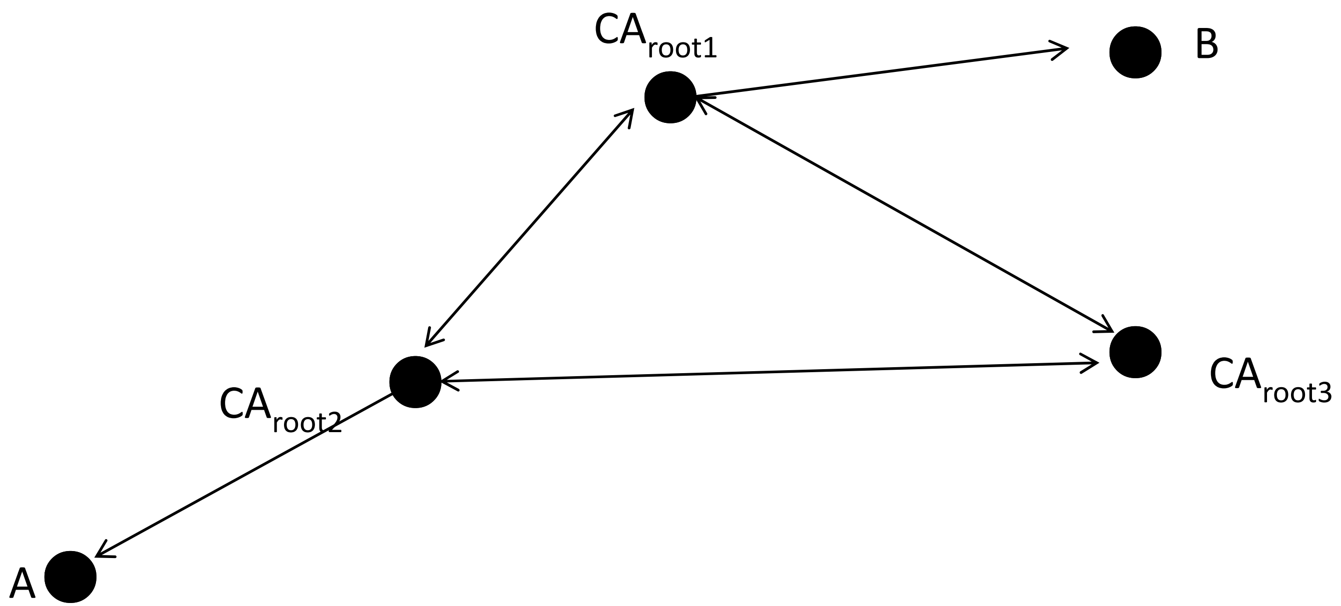
Verification

- Alice verifies Bob
  - Bob send all the certificates in the path
    $CA_{root} \rightarrow CA_1 \rightarrow Bob$

- Alice validates $Cert(CA_{root})$ using the key $ver_{CA\text{-}root}$
- Alice validates $Cert(CA_1)$ using the key $ver_{CA\text{-}root}$
- Alice extracts the key $ver_{CA\text{-}1}$ from $Cert(CA_1)$
- Alice validates Cert(Bob) using the key $ver_{CA\text{-}1}$
- Alice extracts Bob's public key from Cert(Bob)

Networked PKIs

- Connect root CAs of two or more different PKI domains by cross-certification



The Web Browser Model

- Most web browsers (e.g., Netscape of Internet Explorer) come preconfigured with a set of independent root CAs
- All of which are treated by the user of browse as trust CAs
- Ex. SSL

PGP Model

- A friend's friend is my friend
- Every user is his own CA
- Certificate Chain
  - if Alice trust Bob
  - if Bob trust Cindy
  - if I trust Alice, than
  - I trust Bob and Cindy
- It works best within a local community where most users know each other

秘密分割
Secret Splitting

前言

- 密碼是保護有價物品的技術的總稱，即在保護資訊的完整性(integrity)與私密性(confidentiality)
- 密鑰管理方式的安全與否，關係到整個密碼系統的安全性
- 秘密分享機制即為一個安全及有效的密鑰管理方案

Secret Splitting

Secret Splitting (There are ways to take a message and divide it up into pieces.)

Dealer can split a message between Alice and Bob:

- **Dealer generates a random-bit string, R, the same length as the message, M.**
- **Dealer XORs M with R to generate $S = M \oplus R$**
- **Dealer gives R to Alice and S to Bob.**
- **(To reconstruct the message, Alice and Bob have only one step to do as following:)**
- **Alice and Bob XOR their pieces together to reconstruct the message $M = R \oplus S$.**

Dealer divides up a message into n pieces

- **Dealer divides up a message into n pieces among U_1, …, and U_n.**
- **Dealer generates n-1 random-bit strings, R_1, …, and R_{n-1}, the same length as the message, M.**
- **Dealer XORs M with the n-1 strings to generate $R = M \oplus R_1 \oplus \ldots \oplus R_{n-1}$.**
- **Dealer gives R_1 to U_1, …, R_{n-1} to U_{n-1}, and R to U_n.**

- (U_1, …, and U_n, working together, can reconstruct the message as following.)
- U_1, …, and U_n get together and compute: $M = R \oplus R_1 \oplus \ldots \oplus R_{n-1}$.

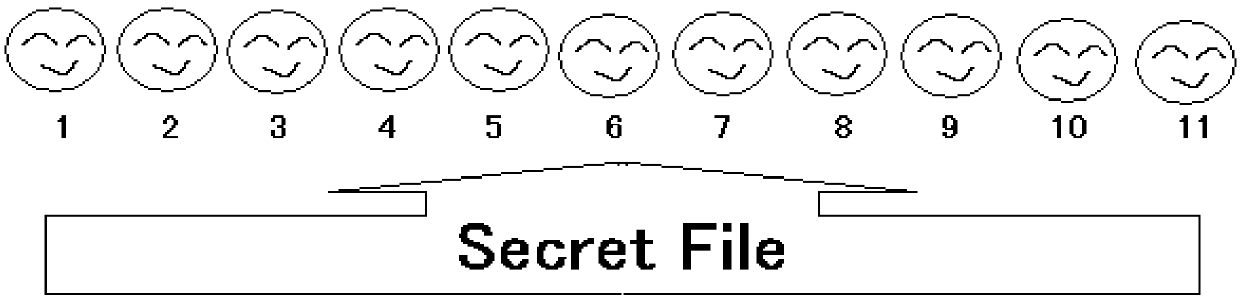
Real Applications

- In USB Tokens
- In PGP

秘密分享 (秘密分散)

Secret Sharing

What is Secret Sharing



If we want the secret file to be opened only when six or more then six of the scientists are to participate, then,

1. How many locks shall we install on the secret file at least?
2. How many keys should every scientist have at least?

1: C_6^{11} = **462..**

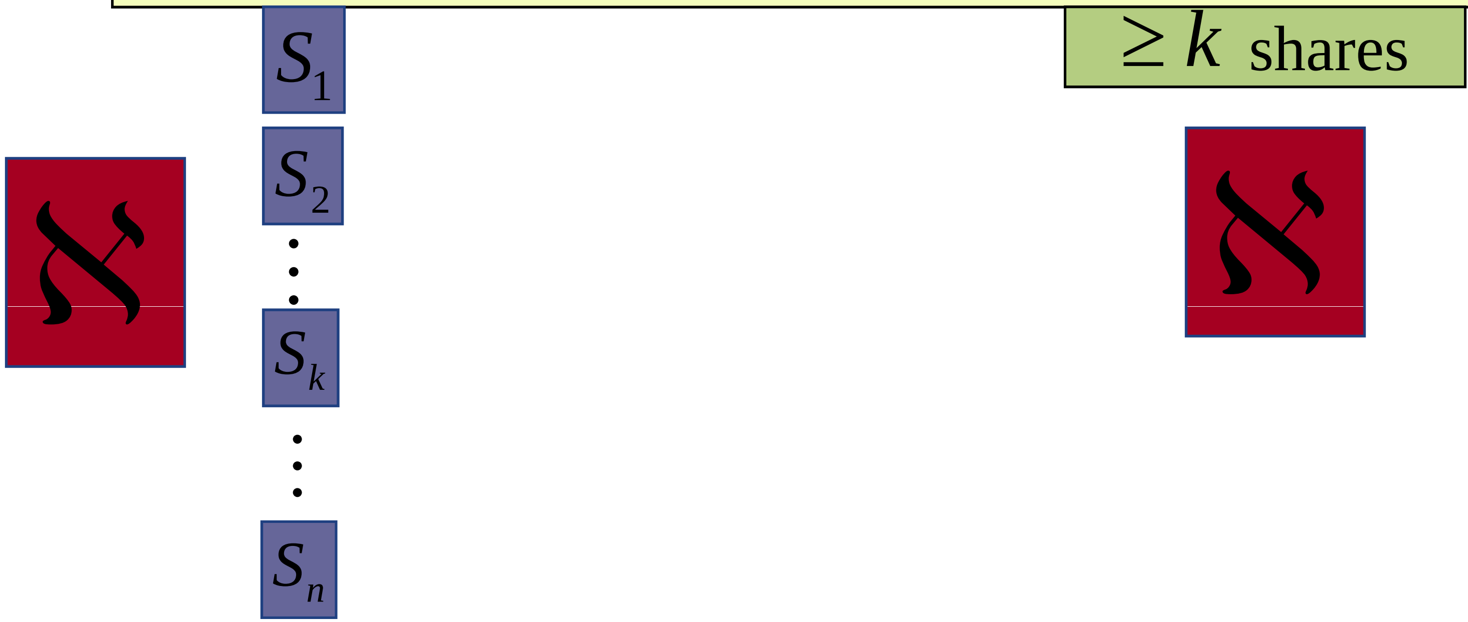
2: C_5^{10} = **252.**

Real Applications

- In banks
- In military (missile launch system)

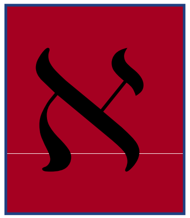
(k, n)-門檻秘密分享方案
(k, n)-threshold scheme

- A (k, n) threshold scheme is a method of breaking a secret $\aleph$ up into n different shares, $S_1, \ldots, S_n$, such that:
  - With the knowledge of any k or more shares $(k \le n)$, the secret $\aleph$ can be easily derived; and



(k, n)-threshold scheme

- A (k, n) threshold scheme is a method of breaking a secret $\aleph$ up into n different shares, $S_1, \ldots, S_n$, such that:
  - With the knowledge of any k-1 or fewer shares, it is impossible to derive the secret $\aleph$.



S_1

S_2

$\vdots$

S_k

$\vdots$

S_n

$< k$ shares

Shamir's (*k*,*n*)-threshold scheme*

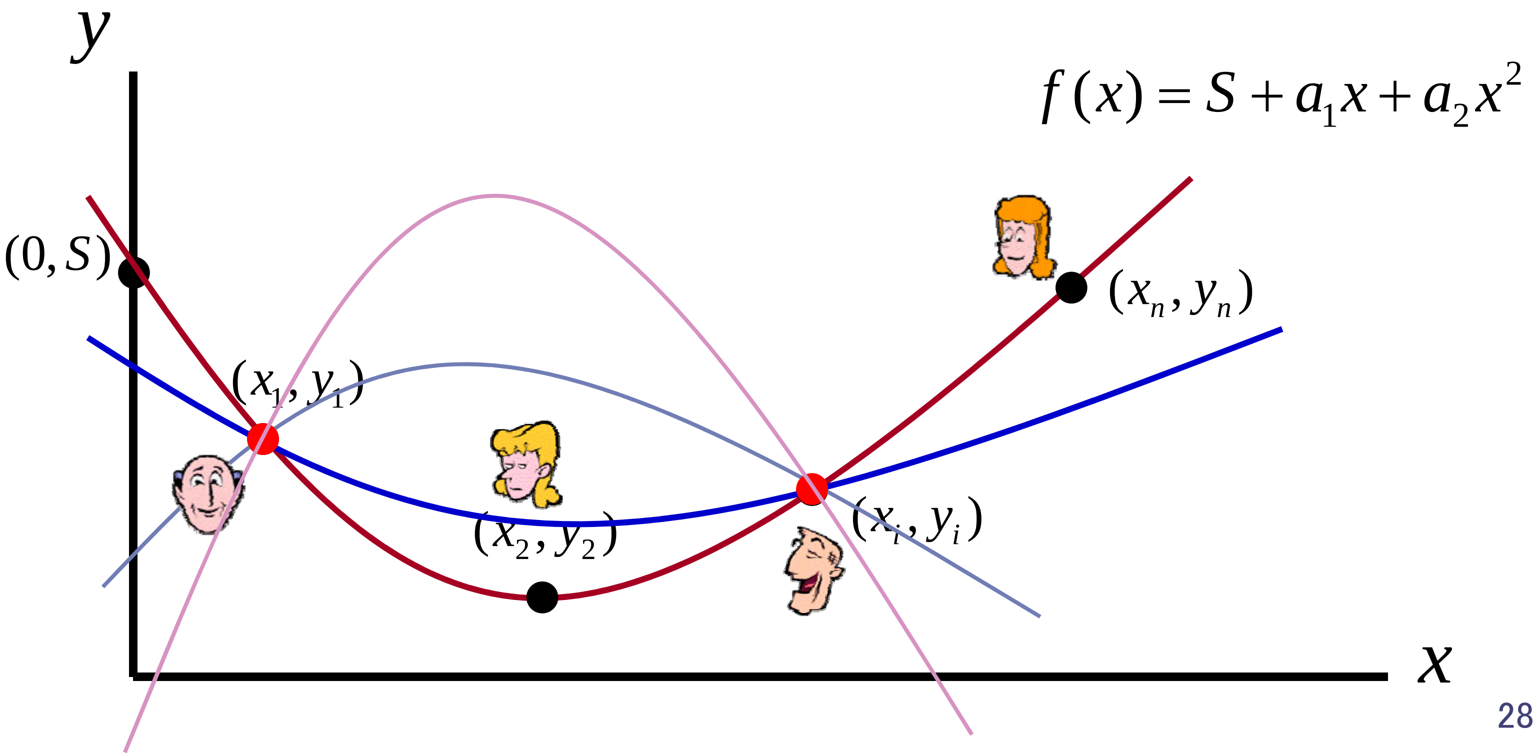
- $f(x) = a_0 + a_1 x + \cdots + a_{k-1} x^{k-1} \in Zp[x]$
- The secret : $f(0) = a_0$
- The set of *n* shares: $S = \{(x_i, y_i) : 1 \le i \le n, y_i = f(x_i)\}$
- By using the Lagrange interpolation formula, any *k* of the *n* participants pooling their shares can easily reconstruct the secret.

$$f(0) = a_0 = \sum_{i=1}^{k} y_i \times \prod_{\substack{1 \le l \le k \\ l \ne 1}} \frac{x_l}{x_l - x_i}$$

*A. Shamir, "How to share a secret", Comm. ACM 22 (1979), 612-613.

Shamir's (3,n)-threshold scheme

- The secret : $S = f(0)$
- The set of n shares: $(x_i, y_i), 1 \le i \le n$



The Shamir Threshold Scheme

- **Let t, w be positive integers, t ≦ w. A (t, w) threshold scheme is a method of sharing a key K among a set of w participants (denoted by ρ), in such a way that any t participants can compute the value of K, but no group of t-1 participants can do so.**

Shamir (t, w)-Threshold Scheme

- **There are two phases:**

  Initialization Phase

  Share Distrdibution Phase

Initialization Phase

- D choose w distinct, non-zero elements of Z_p, denoted x_i. $1 \leqq i \leqq w$ (this is where we require $p \geqq w+1$).
- For $1 \leqq i \leqq w$, D gives the value x_i to P_i
- The values x_i are public.

Share Distribution

- Suppose D wants to share a key $K \in Z_p$.
- D secretly chooses (indepently at random) t-1 elements of Z_p, which are denoted $a_1, \ldots, a_{t-1}$.
- For $1 \leqq i \leqq w$, D computes $y_i = a(x_i)$, where $a(x) = K + \sum_{j=1}^{t-1} a_j x^j \bmod p$.
- For $1 \leqq i \leqq w$, D gives the share y_i to P_i.

Description

- **K: secret.**
- **Polynomial: $a(x) = K + \sum_{j=1}^{t-1} a_j x^j$**
- **ρ= {Pj: 1≦ j ≦ w} : the set of w participants.**
- **Every participant P_j obtains a point (x_j, y_j) on this polynomial.**
- **Suppose that participants P1, …, Pt want to determine K. They know that $y_j = a(x_j)$, $1 \leqq j \leqq t$.**

Example

- Suppose that $p = 17$, $t = 3$, and $w = 5$; and the public x-coordinates are $x_j = j$, $1 \leqq j \leqq 5$.
- Suppose that $B = \{ P_1, P_3, P_5\}$ pool their shares, which are respectively 8, 10, 11.
- Write the polynomial $a(x) = a_0 + a_1x + a_2x^2$, and computing $a(1), a(3), a(5) \in \mathbb{Z}_{17}$.
- We will have $(a_0, a_1, a_2) = (13, 10, 2)$.
- The key is $K = a_0 = 13$.

Lagrange Interpolating Formula

- **Given (x_j, y_j), $1 \leqq j \leqq t$, there exists a polynomial $a(x)$ so that $y_j = a(x_j)$.**
- **The formula for $a(x)$ is as follows:**

$$a(x) = \sum_{i=1}^{t} y_i \prod_{j=1, j \neq i}^{t} \frac{x - x_j}{x_i - x_j} \bmod(p).$$

Key

- **The formula can be expressed as follows:**

  $$\mathbf{a(x) = a_0 + a_1x + a_2x^2 + \ldots + a_{t-1}x^{t-1}.}$$

- **The key K = $\mathbf{a_0}$ = a(0).**
- **K =**

$$a(x) = \sum_{i=1}^{t} y_i \prod_{j=1, j\neq i}^{t} \frac{x_j}{x_i - x_j} \bmod(p).$$

Simplified (t, t)-Threshold Scheme

- **D secretly chooses (independently at random) t-1 elements of Z_m, $y_1, \ldots, y_{t-1}$.**
- **D computes $y_t = K - \sum_{j=1}^{t-1} y_j \bmod m$.**
- **For $1 \leqq j \leqq t$, D gives the share y_j to P_j.**

Description

- **Observe that the t participants can compute K by the formula** $K = \sum_{j=1}^{t} y_j \bmod m$.

Example – Finding Key

- **Suppose that m = 10 and t = 4.**
- **Suppose that the shares for the four participants are $y_1 = 7$, $y_2 = 2$, $y_3 = 4$ and $y_4 = 2$.**
- **The key K = 7 + 2 + 4 + 2 mod 10 = 5.**

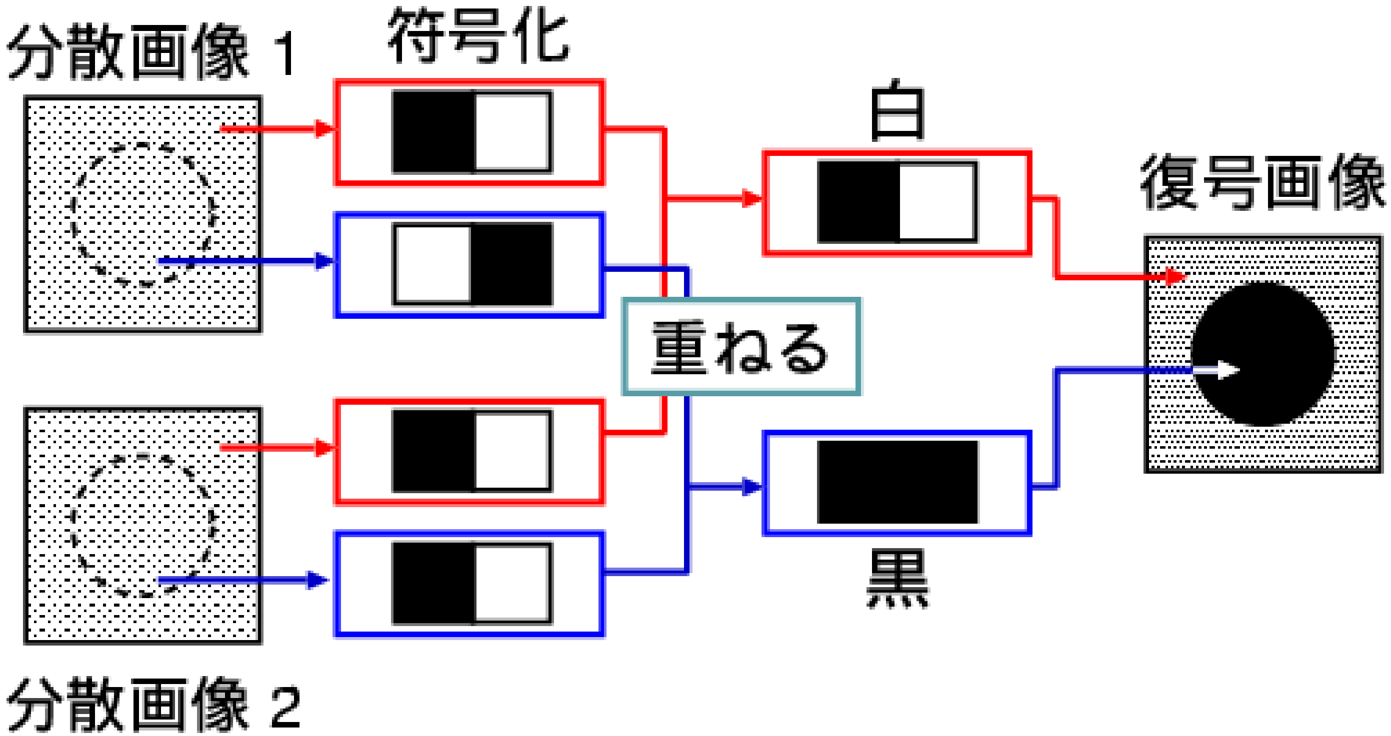
Example -- Security

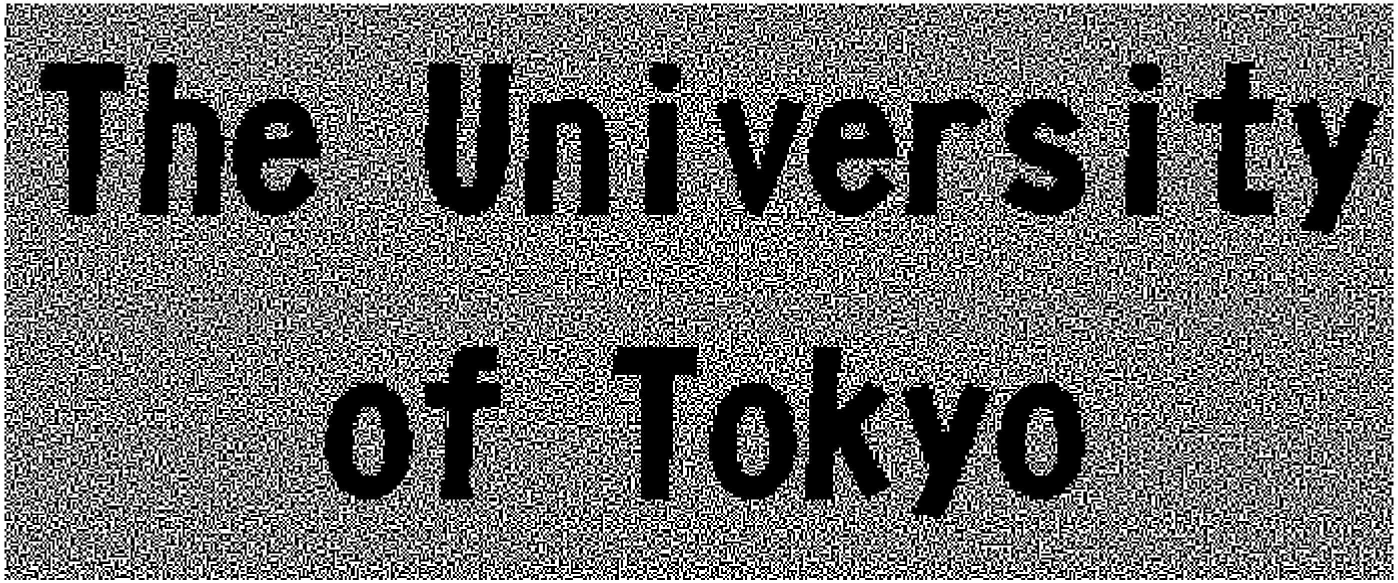
- **Suppose that the first three participants try to determine K.**
- **They know that $y_1 + y_2 + y_4 \bmod m = 3$, but they do not know the value y4.**
- **There is a one-to-one correspondence between the ten possible values of y_4 and the ten possible values of the key K.**

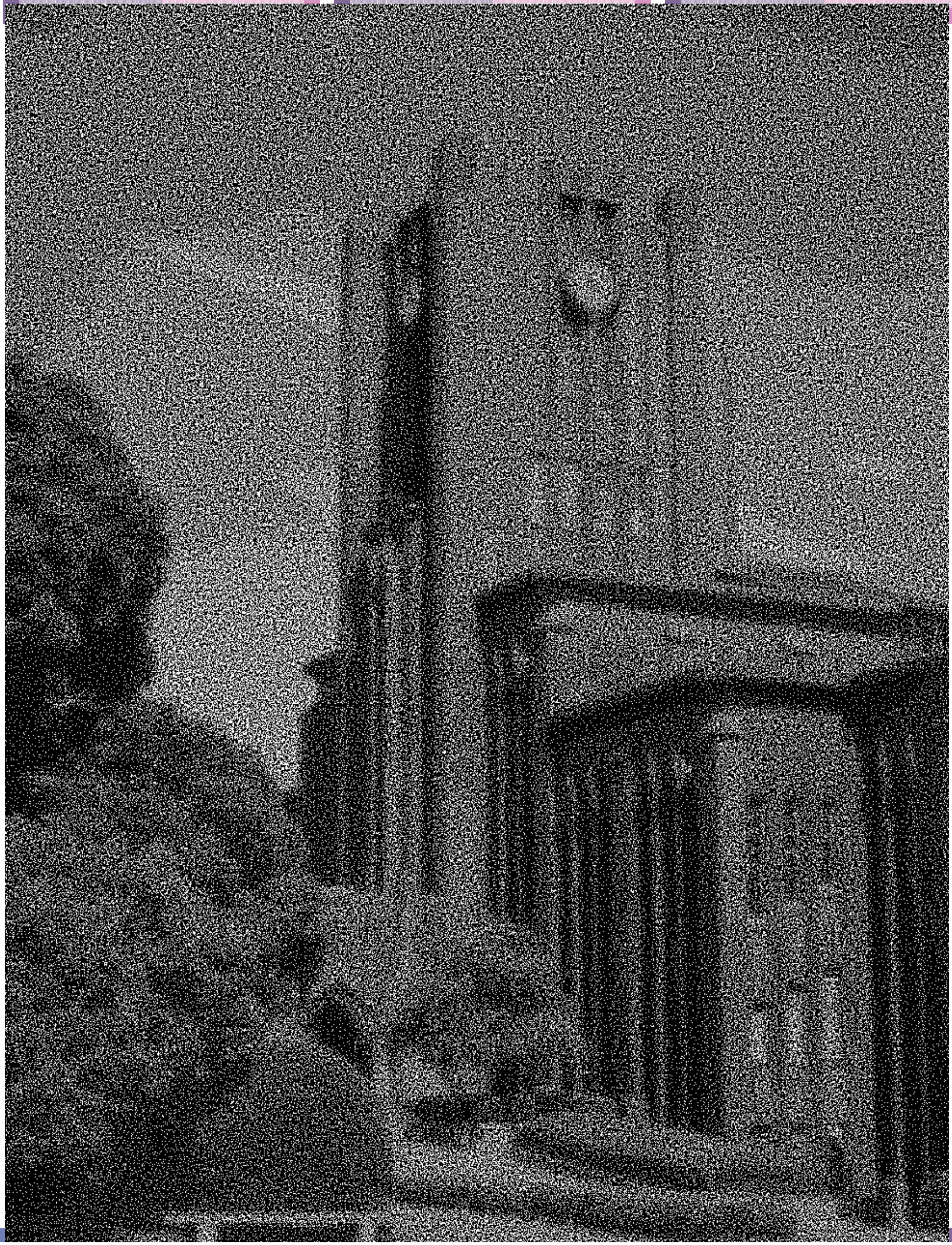
視覺型秘密分享機制

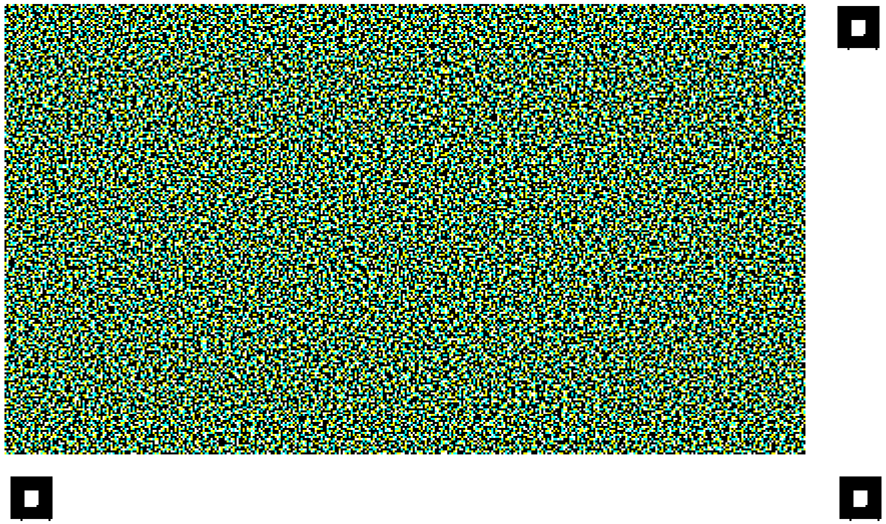
視覺型秘密分享機制

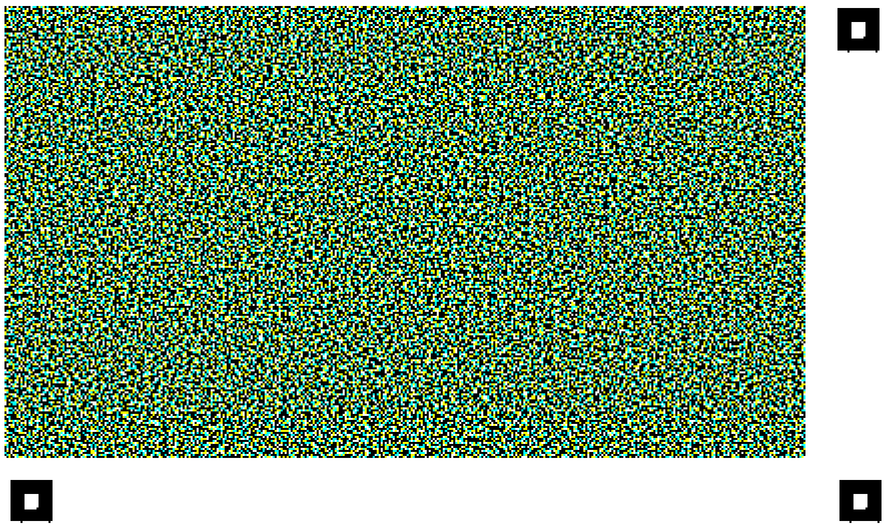
- V

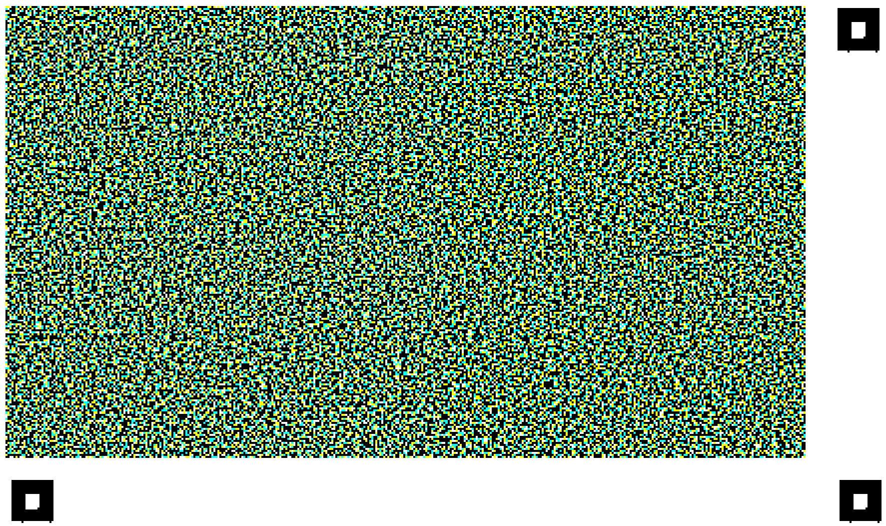


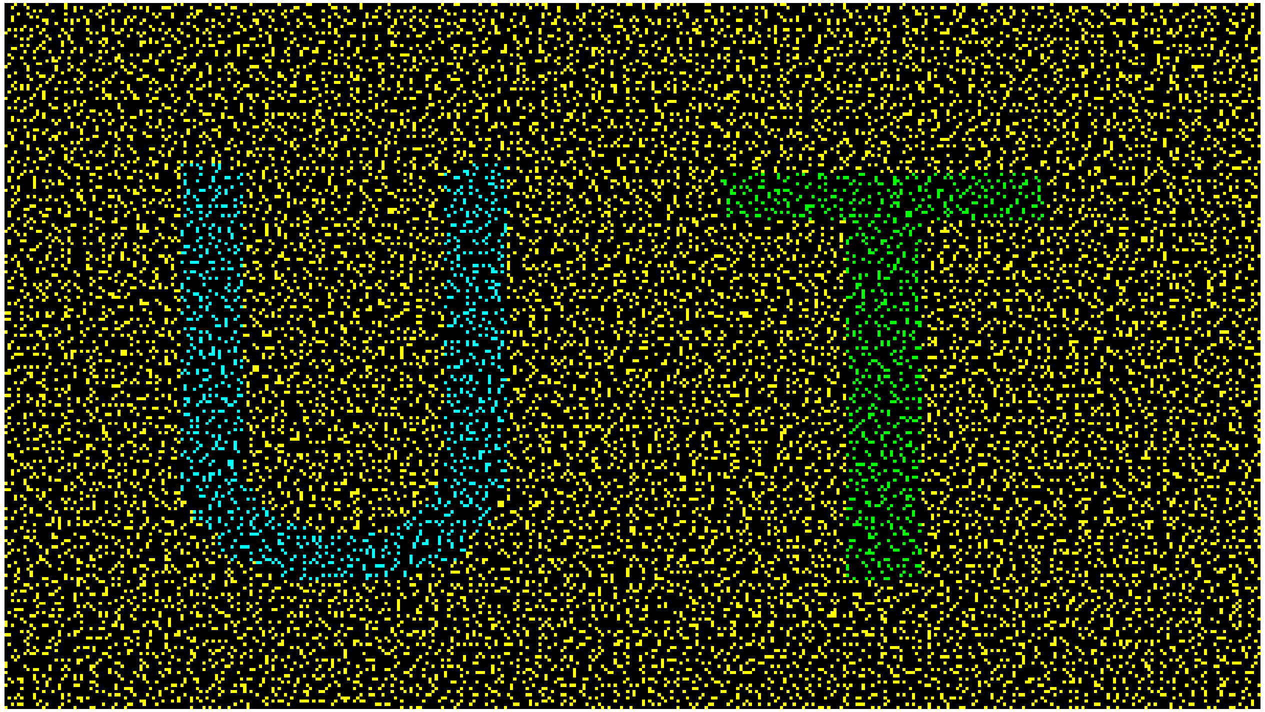

The University
of Tokyo

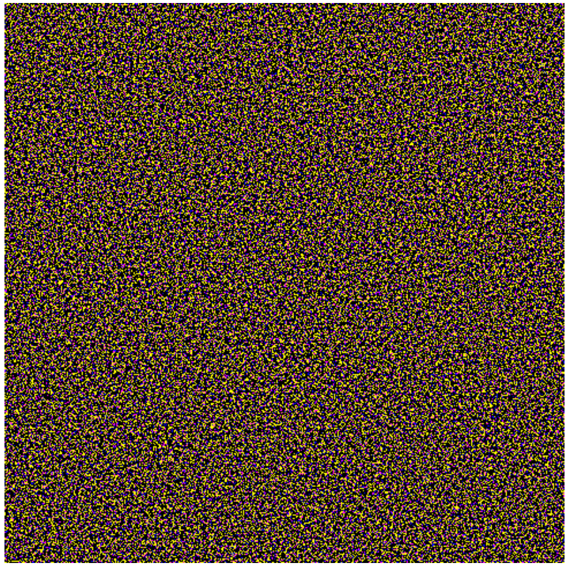


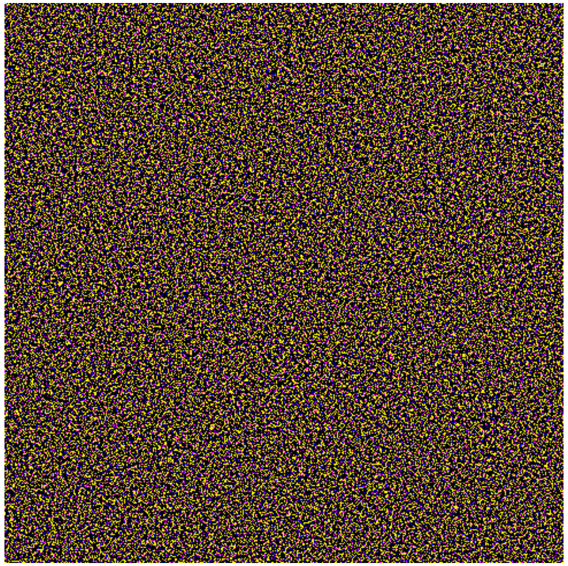


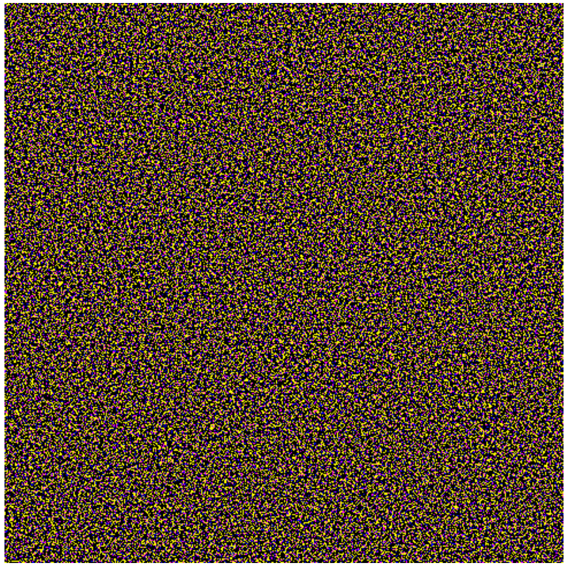


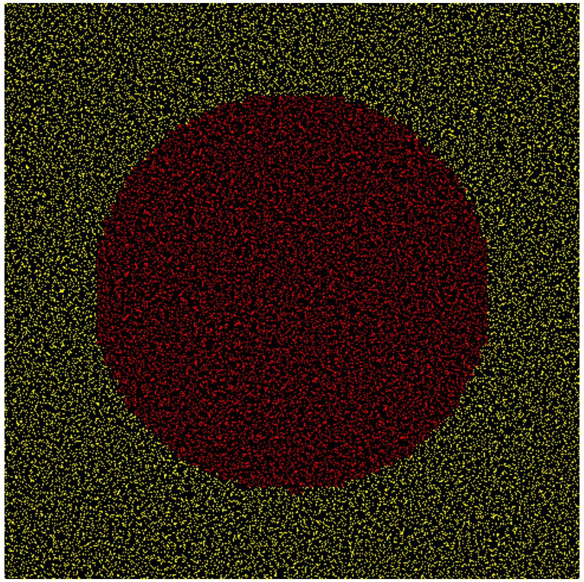


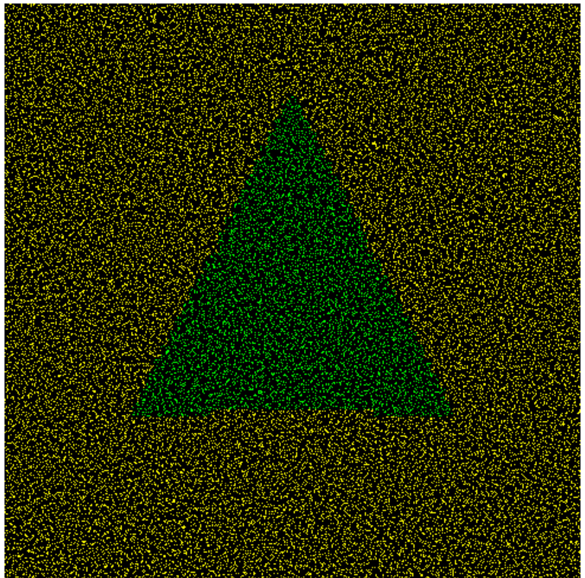

UT

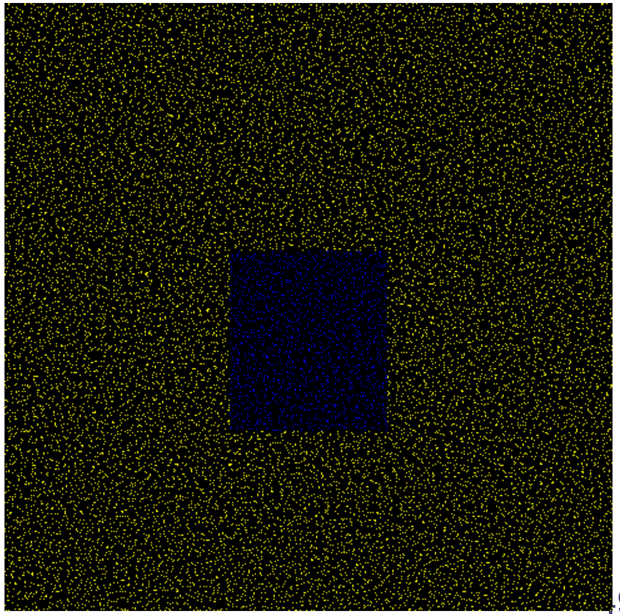












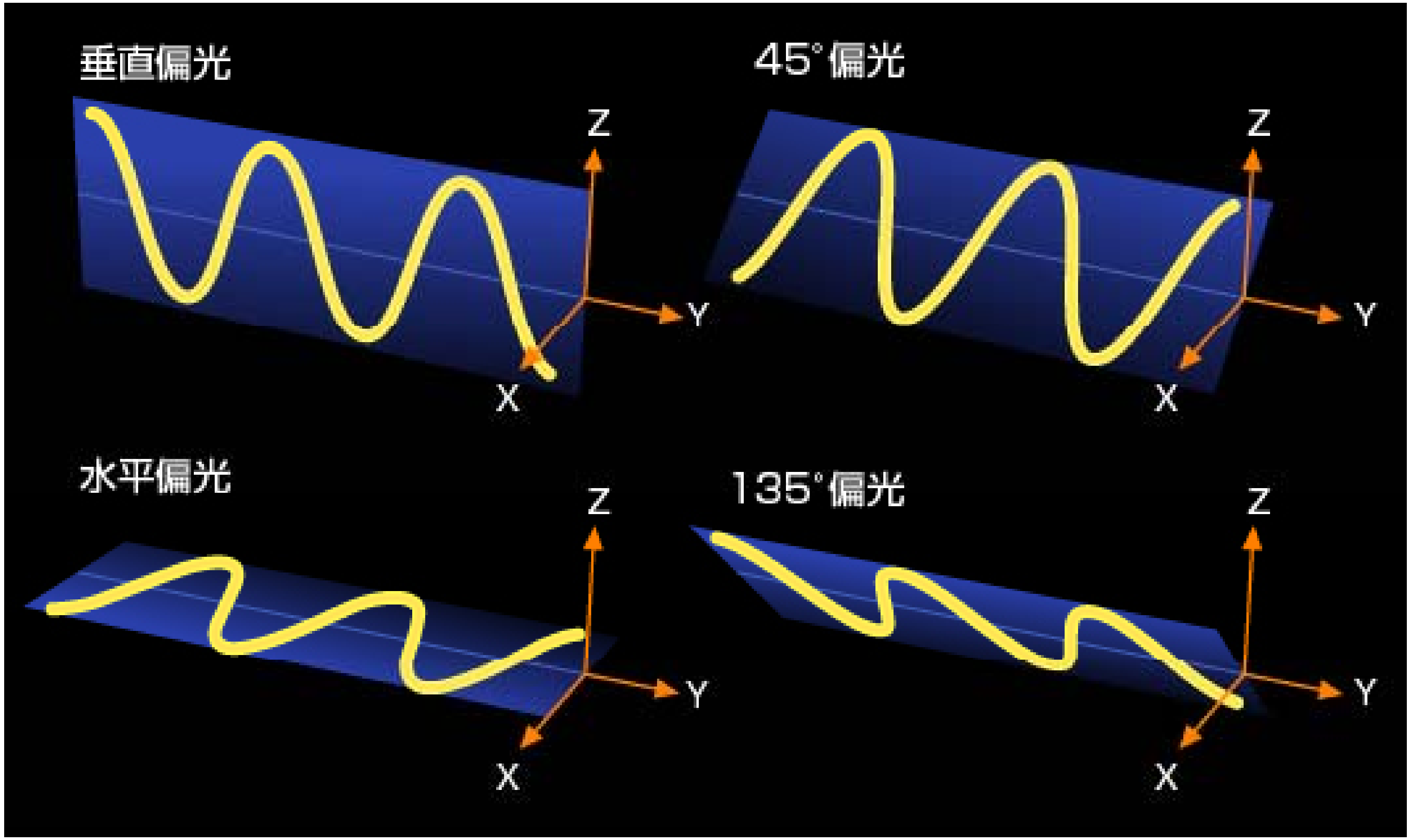
公開金鑰密碼的未來

- 公開金鑰密碼的安全性建立在許多假設性的難問題上(DL-Problem,CDH Problem, Prime Factoring)
- 針對這些難問題，如果發現新演算法，則安全性有可能一夕之間瓦解
- 量子電腦(Quantum)

量子密碼

- 絕對安全的密碼 (終極密碼)
- 終結了編碼者與解碼者的戰爭

量子的四種狀態（| — ／ ＼）

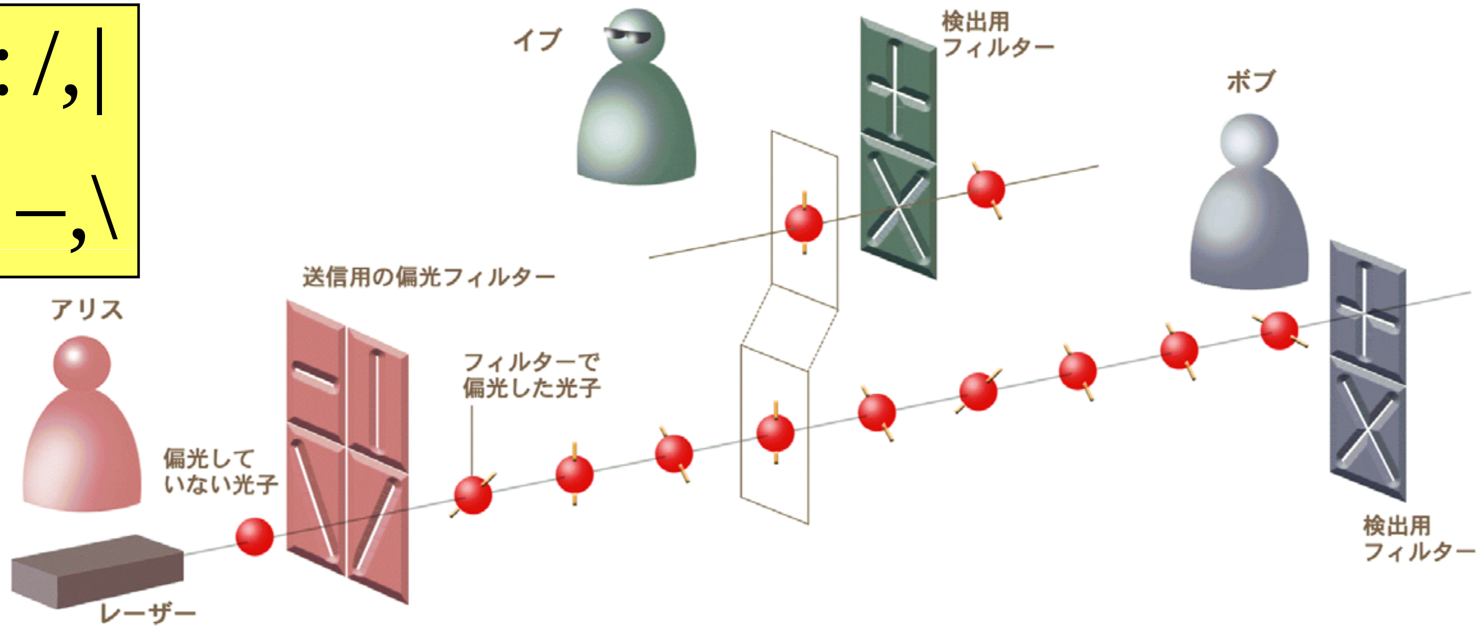


測不準原理

- 量子有四種狀態(| 一 / ＼)
- 利用兩種Filter (＋, ×)來觀測
  - ＋ filter 觀測時
    - | ⟶ |, 一 ⟶ 一
    - /or ＼ ⟶ | or 一
  - × filter 觀測時
    - / ⟶ / , ＼ ⟶ ＼
    - | or 一 ⟶ ＼or /

0 : /,|

1 : –,\



アリスが送った情報	0	0	1	0	1	0	1	1	1
アリスの送信用フィルター	/	|	\	|	\	/	\	\	—
ボブの検出用フィルター	+	+	+	+	×	+	+	×	+
ボブの検出結果	1	0	1	0	1	0	0	1	1
得られた暗号鍵	–	0	–	0	1	–	–	1	1