Lesson 6
Secret sharing

A. Sorokin (Input | Output)

Self paced course

July 31, 2025

What is secret sharing?

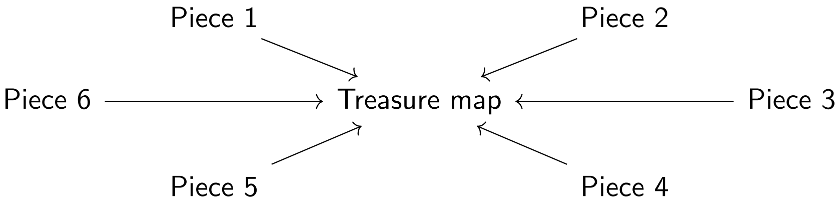
Secret sharing is a method used to protect important information by splitting it into pieces such that

(i) no single person holds the entire secret,

(ii) multiple people must cooperate to reconstruct it.

Real life example

Splitting a treasure map into several pieces and distributing them among trusted friends. Only when a sufficient number of them come together, the map can be reassembled and the treasure found.



Motivation

In traditional security, a single key or password protects sensitive data. But what if:

(i) the key is **lost**? (the data becomes inaccessible)

(ii) the key is **stolen**? (the data is compromised)

(iii) the key holder is **untrustworthy**? (they might misuse it)

Motivation

Secret sharing solves these problems by:

(i) avoiding reliance on one person;

(ii) preventing single-point attacks (no one has full power);

(iii) ensuring recovery if some parts are lost.

Bad example of secret sharing

Let we have a scheme in which the secret phrase

“password”

is divided into four shares

“pa - - - - - -”, “- - ss - - - -”, “- - - - wo - -”, “- - - - - - rd”

(0) A person with 0 shares knows only that the password consists of eight letters, and thus would have to guess the password from $26^8 = 208$ billion possible combinations.

(1) A person with one share would have to guess only the six letters, from $26^6 = 308$ million combinations, and so on.

This system is **not a secure secret sharing scheme**, because

a player with fewer than t secret shares is able to reduce the problem of obtaining the inner secret without first needing to obtain all of the necessary shares.

Shamir's secret sharing

The most famous secret sharing method was invented by Adi Shamir and is similar to the "Treasure Map" example.

The original secret (a password, encryption key, or sensitive number N) is split into shares: instead of storing N in one place, we generate multiple *shares*, like puzzle pieces.

Threshold scheme: only a certain number of shares are needed to reconstruct the secret. For example, in **3 out of 5** scheme

(i) if you have fewer than 3, you learn nothing;

(ii) if you have 3 or more, you can recover N.

Shamir's secret sharing: justification

It is based on the following math facts:

(1) through 2 points on plane passed exactly one line (polynomial of degree 1);

(2) through 3 points on plane passed exactly one parabola (polynomial of degree 2);

(n) through $n+1$ points on plane passed exactly one polynomial of degree n;

Moreover, through n points pass infinitely many polynomials of degree n, so without $(n+1)-$st point the polynomial cannot be determined uniquely.

Security of Shamir's secret sharing

(i) No single share reveals the secret (like having one piece of the treasure map).

(ii) Losing some shares is okay, if enough remain.

(iii) Unauthorized groups can't reconstruct it, unless they meet the threshold.

Other secret sharing schemes

(i) **Additive secret sharing**: Split a secret number into parts that add up to it.

Additive approach is simpler, but less secure since every share is needed: losing one means losing the secret.

Other secret sharing schemes

(ii) **Visual secret sharing**: a secret image is split into random-looking slides. Only when overlapped does the image appear.

This scheme is used in military, watermarking, and secure voting.

Other secret sharing schemes

(iii) **Secure multi-party computation**: multiple parties compute using their shares without revealing them.

This approach is useful for secure auctions and private data analysis.

Real-World Applications

Examples of secret sharing include:

(i) bank vaults: require multiple managers to open;

(ii) nuclear launch codes: split among officials to prevent misuse;

(iii) corporate secrets: no single employee knows the full key (e.g. a company's Bitcoin wallet requires 5 executives, but only 3 need to sign a transaction);

(iv) classified documents: split secret among agencies to prevent leaks.

(v) blockchain wallets: recovery phrases split among trusted parties.

(vi) cloud storage and backups: split recovery keys across devices so nobody can't steal them all.

Limitations and challenges

(i) **Trust in shareholders**: if too many shares are held by corrupt parties, they can collude.

(ii) **Secure storage of shares**: if shares are stored poorly (e.g., sticky notes), security fails.

(iii) **No protection against coercion**: attackers could force shareholders to reveal their parts.

(iv) **Complexity in large systems**: managing hundreds of shares can be cumbersome.

Advanced variations

(i) **Proactive secret sharing**: shares periodically refresh to resist long-term attacks.

(ii) **Verifiable secret sharing**: ensures that shares are valid, so no one cheats when distributing them.