CS 528 (Fall 2021) Data Privacy & Security

Yuan Hong
Department of Computer Science
Illinois Institute of Technology

Chapter 9-C Secret Sharing

SECRET SHARING SCHEMES

Imagine that you have made billions of £,$,€ from Internet stocks, and you wish to leave your estate to your 4 children. You like to divide it among them in such a way that two of them have to get together to reconstruct the real combination, i.e., someone who wants some of the inheritance must somehow cooperate with al least one of the other children.

(t,n)=(2,4) - threshold scheme

ANOTHER HISTORICAL PROBLEM

Liu in [1] considers the following:

Problem

Eleven scientists are working on a secret project. They wish to lock up the documents in a cabinet so that the cabinet can be opened if and only if six or more of the scientists are present. What is the smallest number of locks needed? What is the smallest number of keys to the locks each scientist must carry?

Answers

- 462 locks
- 252 keys per scientist.

DEFINITION

Let t ≦ n be positive integers.

A (t,n)-threshold scheme is a method of sharing a message M among a set of n participants such that any subset consisting of t participants can reconstruct the message M, but no subsets of smaller size can reconstruct M.

SHAMIR THRESHOLD SCHEME

Based on Lagrange interpolation polynomial
Also called Lagrange interpolation scheme

Choose a large prime p, the message M is represented as a number (mod p)

$s(x) \equiv M+s_1x+s_2x^2+\ldots+s_{t-1}x^{t-1} \pmod{p}$

(x_i,y_i), $i=1,2, \ldots, n$; $y_i \equiv s(x_i) \pmod{p}$

LAGRANGE INTERPOLATING POLYNOMIALS

Suppose that the function y=f(x) is known at the n+1 points (x_0,y_0), (x_1,y_1), $\cdots$, (x_n,y_n), where $a \leq x_0 < x_1 < x_2 \ldots < x_n \leq b$, then there is a unique polynomial $P_n(x_i)=y_i$, $0 \leq i \leq n$, as given below

$$P_n(x) = \sum_{k=0}^{n} y_k L_{n,k}(x) \equiv \sum_{k=0}^{n} y_k \left[\prod_{i=0, i \neq k}^{n} \frac{x - x_i}{x_k - x_i} \right] \bmod p$$

COMPUTING SECRET VALUE M

$$M \equiv \sum_{k=1}^{t} y_k \left[\prod_{i=1, i \neq k}^{t} \frac{-x_i}{x_k - x_i} \right] \bmod p$$

SIMPLE EXERCISES

You set up a (2,30) Shamir threshold scheme, working mod the prime 101. Two of the shares are (1,13) and (3,12). Another person received the share (2,*), what is the value of *?

M=?, *=?

$s(x) \equiv M + s_1 x$

In a (3,5) Shamir secret sharing scheme with modulus p=17, the following were given to Alice, Bob, Charles: (1,8), (3,10), (5,11). Calculate the corresponding Lagrange interpolating polynomial, and identify the secret.

$s(x) \equiv M + s_1 x + s_2 x^2$

(SECRET) VALUE SHARING

A (k, n) threshold secret sharing should satisfy the following requirements:

(1) A secret value M is used to generate n shadows.

(2) Any ≧k shadows can reconstruct the secret value M.

(3) Any ＜k shadows can not get sufficient information to reveal the secret value M.

SECRET SHARING (1/4)

A (k, n) threshold polynomial can be written by

$$s(x) \equiv M+s_1x+s_2x^2+\ldots+s_{k-1}x^{k-1} \pmod{p}$$

Select n distinct integers $x_1,x_2,\ldots,x_n$ from [0,p-1]

Deliver $(x_i,s(x_i))$ to the *i*-th participant

p= a (large) prime number

M: secret value

$s_1,\ldots,s_{k-1}$: randomly chosen from [0, p-1]

SECRET SHARING (2/4)

To reveal the secret value M, we must collect (at least k) ≧k shadows.

Without loss of generality, we use $(x_1,s(x_1)),\ldots,(x_k,s(x_k))$ as k shadows.

We can reveal the secret value M by using Lagrange interpolation.

$$s(x) \equiv \sum_{j=1}^{k} \left[s(x_j) \prod_{i=1, i \neq j}^{k} \frac{x - x_i}{x_j - x_i} \right] \bmod p$$

where $M=s(0)$

SECRET SHARING (3/4)

Example:

(k, n)=(2, 4)-threshold secret sharing

M=9, p=17

Given $x_1=1, x_2=2, x_3=3, x_4=4$

A polynomial equation can be defined as

$$s(x) \equiv 9+13x \bmod 17$$

Then s(1)=5, s(2)=1, s(3)=14, s(4)=10

Four shadows: (1,5), (2,1), (3,14), (4,10)

SECRET SHARING (4/4)

Example

We can get the equation by taking any two shares, e.g., (1,5), (2,1) by using Lagrange interpolation.

$$s(x) \equiv \sum_{j=1}^{k} \left[s(x_j) \prod_{i=1, i \neq j}^{k} \frac{x - x_i}{x_j - x_i} \right] \bmod p$$

s(0)=9

Secret Sharing Demo:
http://point-at-infinity.org/ssss/demo.html