

# Cryptography and Network Security

## Secret Sharing

Xiang-Yang Li

CS595-Cryptography and Network Security

## Threshold Scheme

- A  $(t,w)$ -threshold scheme
  - Sharing key  $K$  among a set of  $w$  users
  - Any  $t$  users can recover the key
  - Any  $t-1$  users can not do so
- Schemes
  - Shamir's scheme
  - Geometric techniques
  - Matroid theory

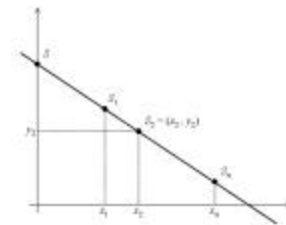
CS595-Cryptography and Network Security

## Shamir's Scheme

- Initialization phase
  - Dealer chooses a large prime number  $p$
  - Dealer chooses  $w$  distinct  $x_i$  from  $Z_p$
  - Gives value  $x_i$  to person  $p_i$
- Share distribution of key  $k$  from  $Z_p$ 
  - Dealer choose  $t-1$  random number  $a_i$
  - Dealer computes  $y_i=f(x_i)$ 
    - Here  $f(x)=k+\sum a_i x^i \pmod p$
  - Dealer gives share  $y_i$  to person  $p_i$

CS595-Cryptography and Network Security

## Geometry View



CS595-Cryptography and Network Security

## Simple $(t,t)$ Sharing

- Procedure
  - $D$  secretly chooses  $t-1$  random elements  $y_i$  from  $Z_n$
  - $D$  computes
    - Value  $y = K - \sum y_i \pmod n$
  - $D$  distributes  $y_i$  to person  $p_i$  for all  $i$
- It is secure and easy
  - Number  $n$  can be any number
  - Easy to recover the key
  - Only  $t$  persons together can do so, assume  $y_i$  random

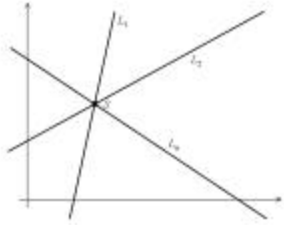
CS595-Cryptography and Network Security

## Blakley's Scheme

- Secret is a point in an  $t$ -dimensional space
- Dealer gives each user a hyper-plane passing the secret point
- Any  $t$  users can recover the common point

CS595-Cryptography and Network Security

## Geometry View



CS595-Cryptography and Network Security

## Avoid Cheating

- Two major distinct weaknesses
  - Bogus values are undetectable.
  - Participants need not reveal their true share.
- Even if a bogus value was detected, it would not necessarily give any information about the true value
- One participant did not reveal its true value after get the true values from other one

CS595-Cryptography and Network Security

## Ben-Or/Rabin Solution

- Using Checking Vectors
- For any two participants A and B
  - Dealer gives A  $(S_A, Y_{AB})$
  - Dealer gives B  $(B_{AB}, C_{AB})$
  - Here  $C_{AB} = B_{AB} Y_{AB} + S_A \pmod p$
  - $S_A$  is the secret share of A
  - A and B keep their values secret
  - B can use the value  $(S_A, Y_{AB})$  to verify A

CS595-Cryptography and Network Security

## Avoid Cheating

- Participant B can send A bogus value after receive A's value
- Solution: bit transfer
  - Dealer gives A  $(S_{Ai}, Y_{ABi})$
  - Dealer gives B  $(B_{ABi}, C_{ABi})$
  - Here  $C_{ABi} = B_{ABi} Y_{ABi} + S_{Ai} \pmod p$
  - $S_{Ai}$  is the  $i$ th bit of the secret share of A

CS595-Cryptography and Network Security

## Cont.

- Protocol
  - Participant A gives its value  $(S_{Ai}, Y_{ABi})$  to B
  - B verifies:  $C_{ABi} = B_{ABi} Y_{ABi} + S_{Ai} \pmod p$
  - B then sends its value  $(S_{Bi}, Y_{BAi})$  to A
  - A verifies:  $C_{BAi} = B_{BAi} Y_{BAi} + S_{Bi} \pmod p$
  - The protocol terminates whenever
    - One side detects cheating, or
    - All values transferred

CS595-Cryptography and Network Security

## Chinese Remainder Theorem

- Given a number  $m < n$ , and  $n = n_1 n_2 \dots n_k$ ,
  - Numbers  $n_i$  and  $n_j$  are coprimes
  - Let  $a_i = m \pmod{n_i}$
  - Number  $n$  is public
  - Dealer delivers  $a_i$  and  $n_i$  to the  $i$ th participant
  - Then all  $k$  users can recover the number  $m$
- Why it is not a good secret sharing scheme?
  - Is it computationally for any  $k-1$  users to recover the key if  $n$  is large?

CS595-Cryptography and Network Security

## Recover method

- Each user pre-computes
  - $N_i = n/n_i$
  - Inverse of  $N_i$ :  $y_i = N_i \text{ mod } n_i$
  - Compute the product  $s_i = a_i N_i y_i \text{ mod } n$
- Recover the secret  $m$ 
  - Each user submits  $s_i$
  - Computes  $s_1 + s_2 + \dots + s_k \text{ mod } n$

CS595-Cryptography and Network Security

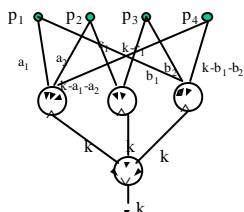
## Access Structure

- Threshold scheme allows any  $t$  users to recover key!
- Access structure allows some subsets to recover the key!
  - Example:  $\{\{p_1, p_2, p_4\}, \{p_1, p_3, p_4\}, \{p_2, p_3\}\}$  among  $p_1, p_2, p_3, p_4, p_5$  able to recover the key
  - Assume the accessing subset is minimized
    - No subset of any accessing subset is able to recover

CS595-Cryptography and Network Security

## Monotone Circuit

- Assign sharing for each accessing subset



CS595-Cryptography and Network Security

## Cont.

- Distribution
  - $(a_1, b_1)$  to  $p_1$
  - $(a_2, c_1)$  to  $p_2$
  - $(k - c_1, b_2)$  to  $p_3$
  - $(k - a_1 - a_2, k - b_1 - b_2)$  to  $p_4$
- The sharer needs know
  - The circuit used by dealer
  - Which shares corresponding to which wires
    - The shared value is secret

CS595-Cryptography and Network Security

## Visual Secret Sharing

- There is a secret picture to be shared among  $n$  participants.
  - The picture is divided into  $n$  transparencies (shares) such that
  - if any  $m$  transparencies are placed together, the picture becomes visible
  - but if fewer than  $m$  transparencies are placed together, nothing can be seen.

CS595-Cryptography and Network Security

## Visual Secret Sharing

- Such a scheme is constructed by viewing the secret picture as a set of black and white pixels and handling each pixel separately.
  - The schemes are perfectly secure and easily implemented without any cryptographic computation.
- A further improvement allows each transparency (share) to be an innocent picture
  - For example, a picture of a landscape or a picture of a building
  - thus concealing the fact of secret sharing

CS595-Cryptography and Network Security

## Interactive Proof

- ❑ *Interactive proof* is a protocol between two parties in which one party, called the *prover*, tries to prove a certain fact to the other party, called the *verifier*
- ❑ Often takes the form of a challenge-response protocol

CS595-Cryptography and Network Security

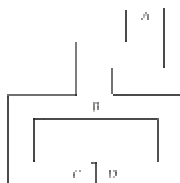
## Desired Properties

- ❑ Desired properties of interactive proofs
  - *Completeness*: The verifier always accepts the proof if the prover knows the fact and both the prover and the verifier follow the protocol.
  - *Soundness*: Verifier always rejects the proof if prover does not know the fact, and verifier follows protocol.
  - *Zero knowledge*: The verifier learns nothing about the fact being proved (except that it is correct) from the prover that he could not already learn without the prover. In a zero-knowledge proof, the verifier cannot even later prove the fact to anyone else.

CS595-Cryptography and Network Security

## An example

- ❑ Ali Baba's Cave



CS595-Cryptography and Network Security

## Cont.

- ❑ Alice wants to prove to Bob that
  - she knows the secret words to open the portal at CD
  - but does not wish to reveal the secret to Bob.
  - In this scenario, Alice's commitment is to go to C or D.

CS595-Cryptography and Network Security

## Proof Protocol

- ❑ A typical round in the proof proceeds as follows:
  - Bob goes to A, waits there while Alice goes to C or D.
  - Bob then asks Alice to appear from either the right side or the left side of the tunnel.
  - If Alice does not know the secret words
    - there is only a 50 percent chance that she will come out from the right tunnel.
  - Bob will repeat this round as many times as he desires until he is certain that Alice knows the secret words.
  - No matter how many times that the proof repeats, Bob does not learn the secret words.

CS595-Cryptography and Network Security