Randomized Component and Group Oriented (t,m,n)-Secret Sharing

Miao Fuyou
School of Computer Sci. & Tech.,USTC

2016.4.10

Outline

- (t,n)-Secret Sharing
- 2 Attacks Against (t,n)-SS
- Randomized Component
- (t,m,n)-Group Oriented Secret Sharing
- Asynchronous Group Authentication
- Key Agreement with Authentication
- *Ideal (t,m,n)-Group Oriented Secret Sharing*

What is (t,n)-Secret Sharing (SS)

- (t,n)-Secret Sharing ($t<=n$)
  - a secret s is divided into n shares such that:
    - (1) any t or more than t shares →s;
    - (2) less than t shares --\-->s;
- Applications
  - Threshold Encryption/Signature
  - Secure Multiparty Computation
  - …

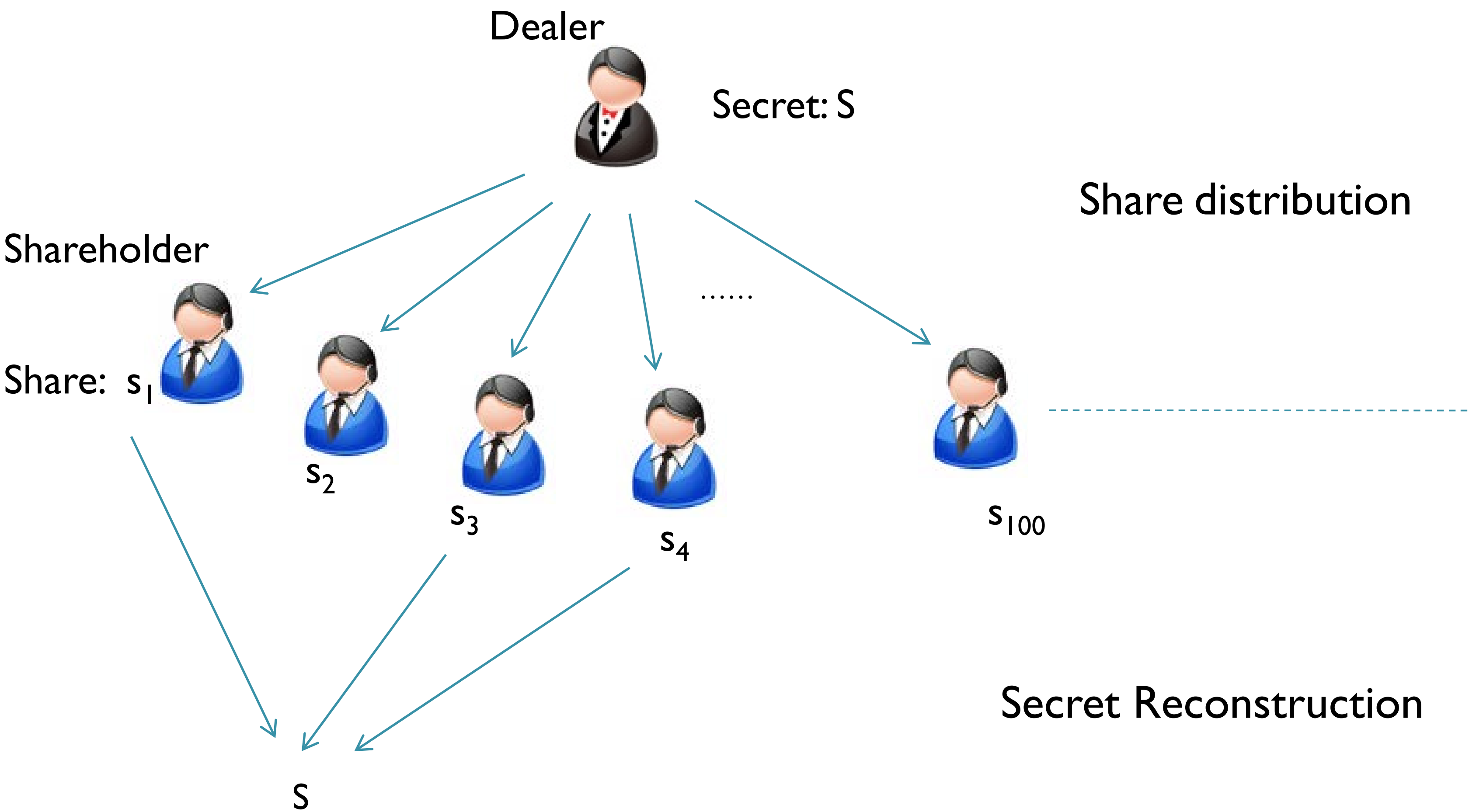


Fig 1. An example of (3,100)-SS

Typical (t,n)-SS

- Shamir's (t,n)-SS (1979)
  - Dealer D: $f(x) = a_0 + a_1 x + \ldots + a_{t-1} x^{t-1} \bmod p,$

    $s = f(0) = a_0 \quad a_j \in F_p, j = 0,1,2,\ldots,t-1$
  - Share Distribution:
    - n shareholders : $\{U_1, U_2, \ldots, U_m, \ldots, U_n\}$,
    - Dealer D: $f(x_i) \rightarrow U_i$, $f(x_i)$ is the share of the shareholder U_i. x_i is the public information.
  - Secret Reconstruction
    - m shareholders $U_{Jm} = \{U_1, U_2, \ldots, U_m\}$, (m>=t)

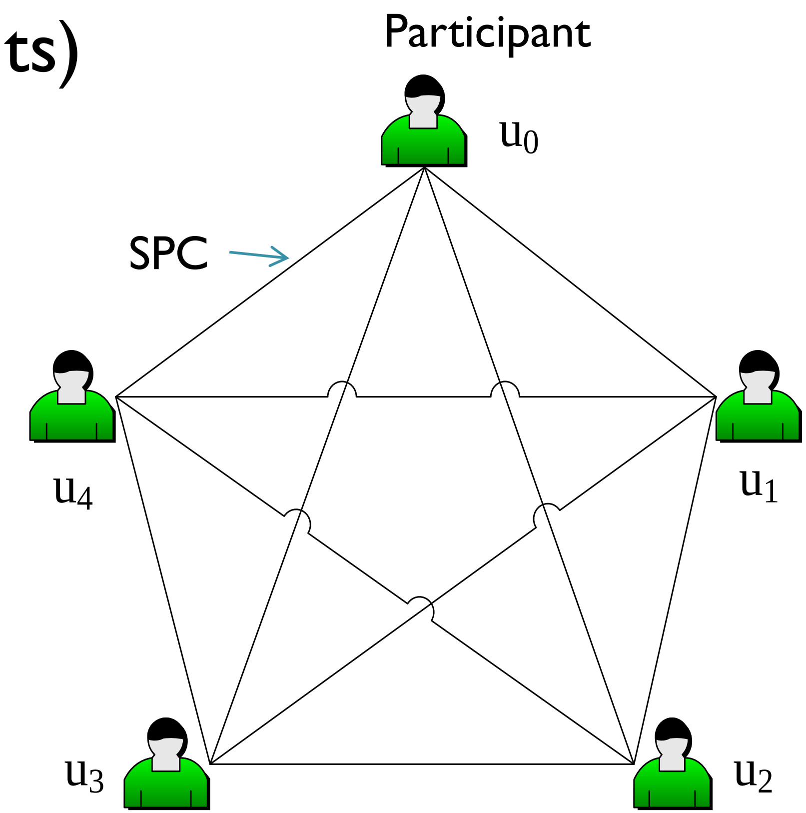
$$s = f(0) = \sum_{j \in J_m} s_j \prod_{r \in J_m, r \neq j} \frac{0 - x_r}{x_j - x_r} \bmod p.$$

Other (t,n)-SS

- Mignotte’s SS and Asmuth-Bloom's SS (CRT based)
- Blakely’s SS (Geometry based)
- Massey’s SS (Linear Code based)

Communication Model

Symmetrical Private Channel (or SPC) between each pair of shareholders (participants)



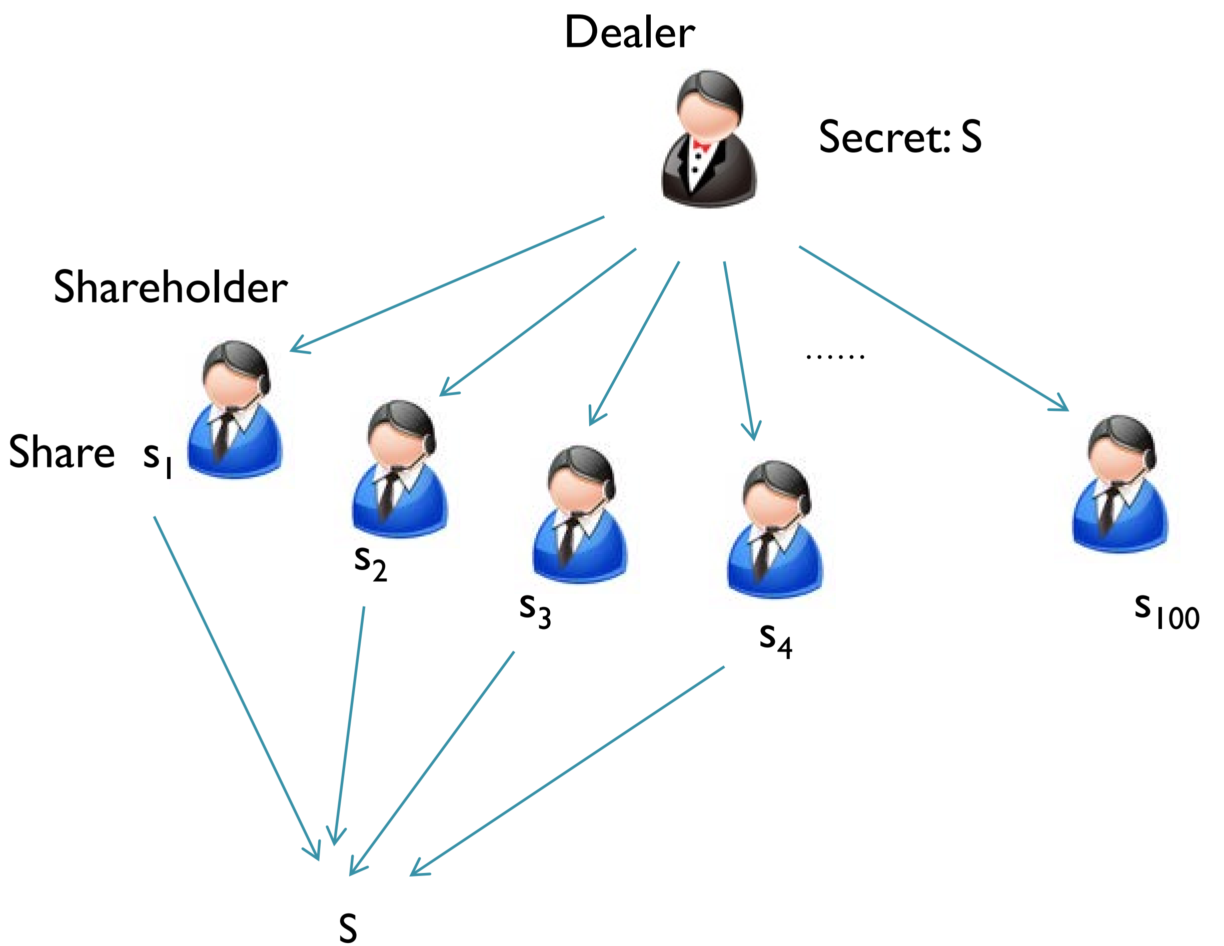


Fig 3. more than 3 participants recover the secret in (3,100)-SS

Illegal Participant Attack-IP

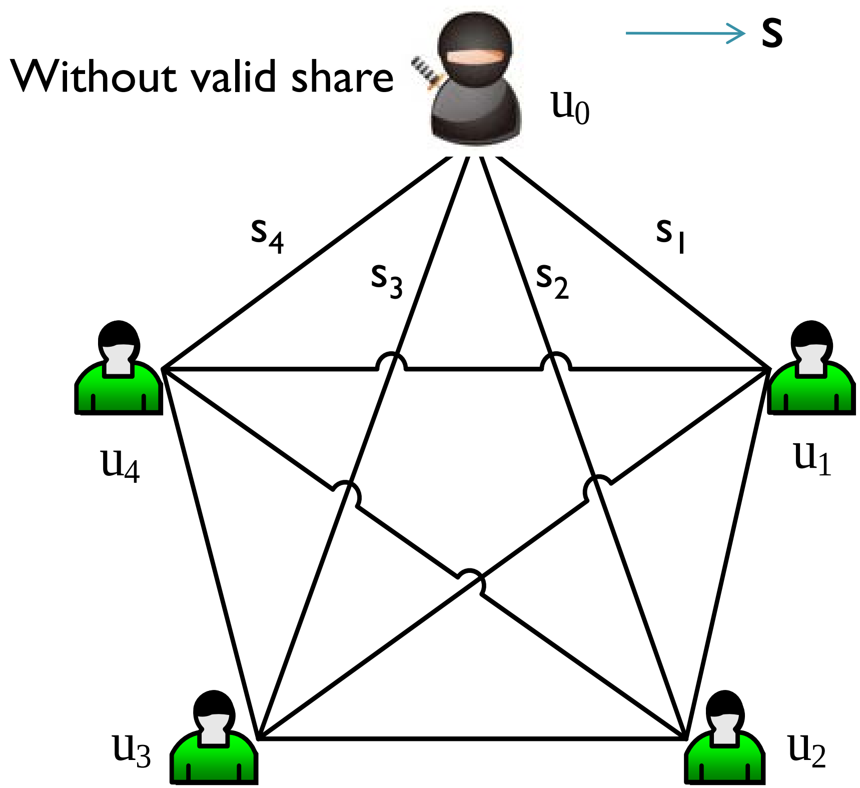


Fig.4 IP attack against (3,100)-SS

Private Channel Cracking Attack -PCC

Only to crack t/2 SPCs to obtain S even if much more than t participants recover S.

How to improve the robustness?

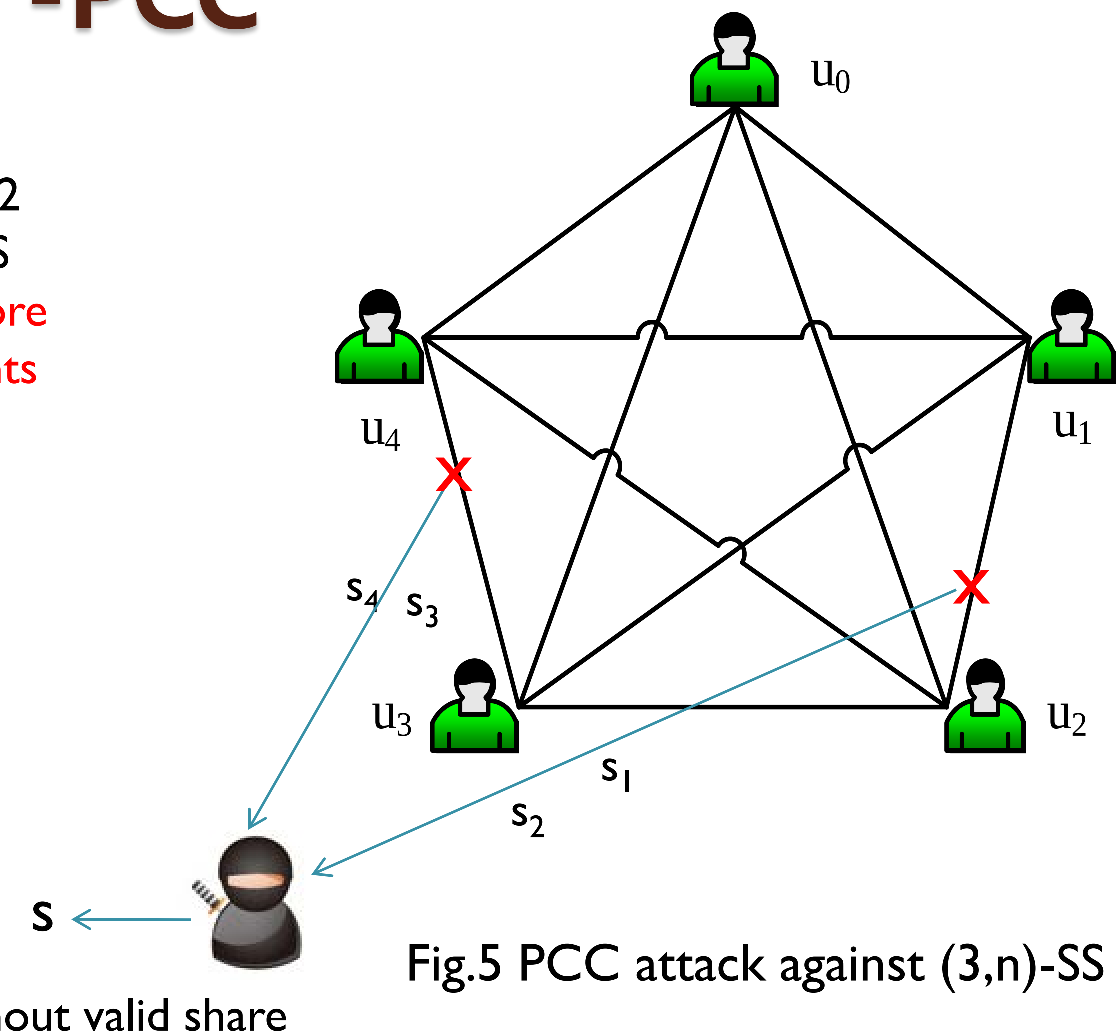


Fig.5 PCC attack against (3,n)-SS

How to

thwart IP attack and

improve the robustness

against PCC attack?

Existing Countermeasures against IP attack

- Verifiable Secret Sharing
  - verify each share of participant before secret reconstruction
- Participant Authentication
  - verify the identity of participants

Existing way to improve the robustness against PCC attack

- Full-shuffling

  each participant needs to exchange a random number with every other participant. m(m-1)/2 random numbers to exchange。 Lower bound: m/2 SPCs.

- partial-shuffling

  m random numbers need to be exchanged

  lower bound： t SPCs

Existing way to improve the robustness against PCC attack

- *Harn's secure secret reconstruction*
  - Use multiple polynomials +linear combination to bind all participant together. For example,
    - Secret s=af(0)+bg(1) mod p
    - Share s_i: $\{f(x_i), g(x_i)\}$

$$c_i = \{af(x_i) \prod_{r \in J_m, r \neq j} \frac{0 - x_r}{x_j - x_r} + bg(x_i) \prod_{r \in J_m, r \neq j} \frac{1 - x_r}{x_j - x_r}\} \bmod p.$$

$$s = af(0) + bg(1) = \sum_{j \in J_m} c_j \bmod p.$$

  - Solve the above 2 problems

Defect:

  - Multiple shares for each shareholder
  - Parameters restriction

Our Objects

- *1. thwart IP attack* and
  2. *improve the robustness against PCC attack*

- No need to exchange extra information
- Maximize the robustness

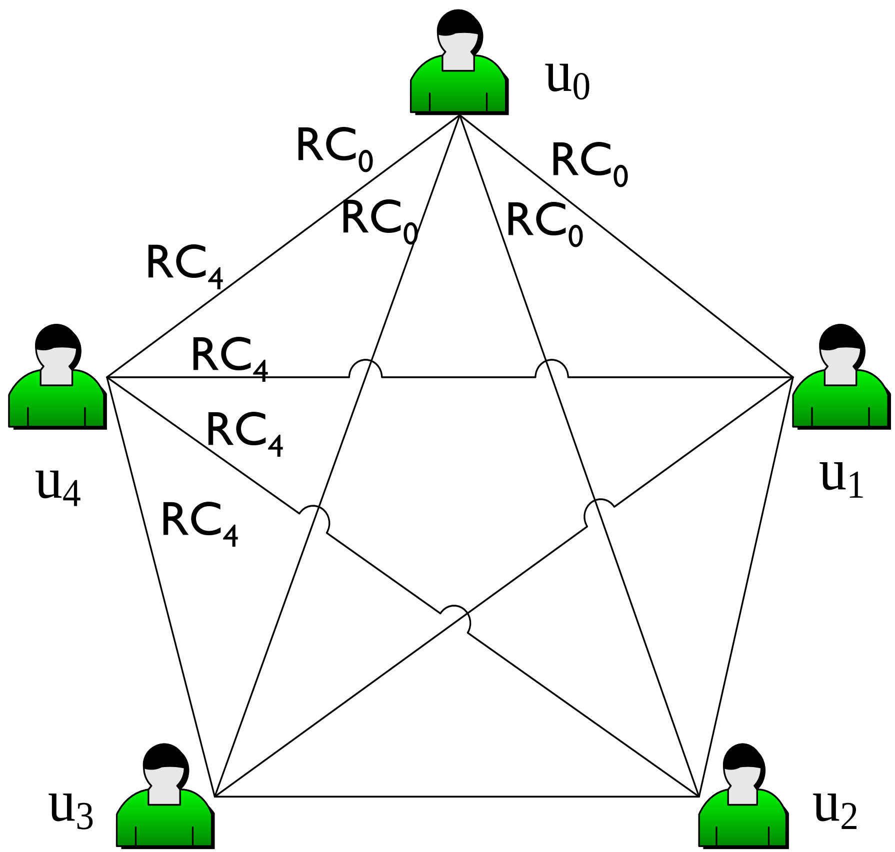
Randomized Component

- Functionality:
  - Protecting a share from exposure during secret reconstruction
  - Capability of secret reconstruction

$$s = f(0) = \sum_{j \in J_m} \boxed{s_j \prod_{r \in J_m, r \neq j} \frac{0 - x_r}{x_j - x_r}} \bmod p.$$

$$RC_j = (s_j \prod_{r \in J_m, r \neq j} \frac{0 - x_r}{x_j - x_r}) + ? \bmod p \longrightarrow \boldsymbol{S}$$

$$s = (\sum_{j=1}^{m} RC_j \bmod p) \bmod q$$



Randomized Component

- Basic Idea (hide the share)
  - If s is a secret value in $[0,9]$ to be hide, we can add a private random number r $\mathrm{in}[0,9]$, to obtain a number $\mathrm{c}=(\mathrm{s}+\mathrm{r}) \bmod 10$
  - Suppose you know $\mathrm{c}=3$, what is the probability to guess s?
  - Obviously it is $1/10$.
  - *c=3*
  - *s = 3 2 1 0 9 8 7 6 5 4*
  - *r = 0 1 2 3 4 5 6 7 8 9*
  - *Guessing r is as difficult as guessing s.*

Randomized Component

- Basic Idea (hide the share)
  - If x is a secret value in [0,99] to be hide, we add a private random number r in[0,9], to obtain a number c=(x+r) mod 100
  - Suppose you know c =33, what is the probability to guess x and s=(x mod 10) ?
  - Obviously it is 1/100 and 1/10 respectively.
  - *If c=33 then*
  - *x 33 32 31 30 29 28 27 26 25 24*
  - *r 0 1 2 3 4 5 6 7 8 9*
  - *Guessing r is easier than guessing x, but is as difficult as guessing s=x mod 10.*

Randomized Component

- $g : F_p \times F_p \times F_q \to F_p$ is a function, $c_i = g(s_k, INF_{I_m}, r_i)$ is called the Randomized Component of the participant U_k, where s_k is the share of U_k, INF_{I_m} is the public information related to the group of m participants in a secret reconstruction, r_i is a random integer uniformly distributed in F_q.

Randomized Component

- Object in design:
  - Make an adversary pay equal effort in guessing a share and the secret

Polynomial-based Randomized Component (PRC)

- If m participants, $\mathcal{U}_{A_m} = \{U_{a_1}, U_{a_2}, \ldots, U_{a_m}\}$, need to recover the secret , each participant, e.g., U_{a_i}, $(U_{a_i} \in \mathcal{U}_{A_m})$, constructs the RC as

$$c_i = \left(f(x_i) \prod_{v=1, v \neq i}^{m} \frac{-x_v}{x_i - x_v} + r_i q \right) mod\ p,$$

- $p > nq^2 + q$, r_i is uniformly distributed in F_q.
- p, q are primes.

Using PRC to protect the share

Given

$$c_i = \left(f(x_i) \prod_{v=1, v \neq i}^{m} \frac{-x_v}{x_i - x_v} + r_i q \right) \bmod p,$$

An adversary has the probability of 1/q to figure out the share $f(x_i)$.

Secret Reconstruction based PRC

- Each participant, e.g., $U_{i_j}, (1 \le j \le m)$, computes the secret as

$$s = \left(\sum_{j=1}^{m} c_{i_j} \bmod p \right) \bmod q$$

(t,m,n)-GOSS

- Group Oriented Secret Sharing with threshold t, m participants and totally n shareholders.

Set of n shareholders: $\mathcal{U} = \{U_1, U_2, \ldots, U_n\}$ with respective public information $\{x_1, x_2, \ldots, x_n\}$;

Group of m participants: $\{U_{i_1}, U_{i_2}, \ldots, U_{i_m}\} \subseteq \mathcal{U}, (m \geq t)$

Parameters:

Primes: p, q with $p > q + nq^2$;

Polynomial in F_p : $f(x) = a_0 + a_1 x + \ldots + a_{t-1} x^{t-1} \bmod p$,

$a_i \in F_p$, for $i = 1, \ldots, t-1$, $a_{t-1} \neq 0$, $a_0 \in F_q$;

Secret: $s = a_0$;

Algorithms:

A. Share Generation

D computes and sends share $s_i = f(x_i)$ to U_i secretly, for $i = 1, 2, \ldots, n$.

B. Randomized Component Construction

Given $\{U_{i_1}, U_{i_2}, \ldots, U_{i_m}\} \subseteq \mathcal{U}$, each participant, e.g., $U_{i_j}, (1 \leq j \leq m)$, constructs the RC, c_{i_j} and releases it to the others through private channels:

$$c_{i_j} = \left(f(x_{i_j}) \prod_{v=1, v \neq j}^{m} \frac{-x_{i_v}}{x_{i_j} - x_{i_v}} + r_{i_j} q \right) \bmod p$$

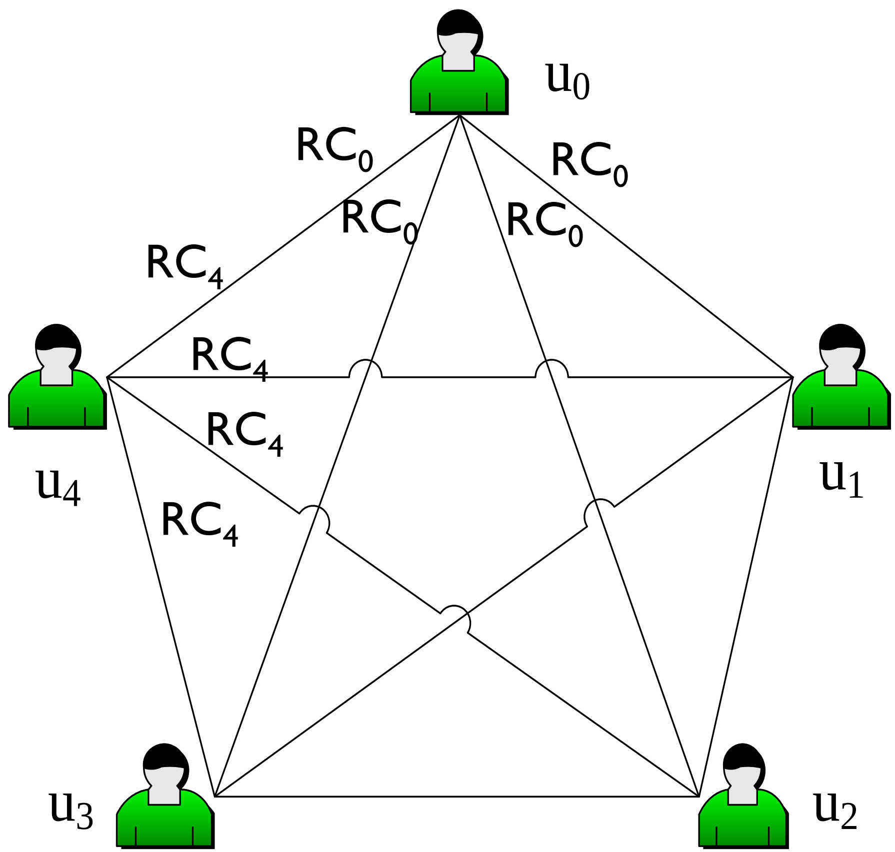
$$(r_{i_j} \in_R F_q, \text{ for } j = 1, 2, \ldots m)$$

C. Secret Reconstruction

Each participant, e.g., $U_{i_j}, (1 \leq j \leq m)$, computes the secret as $s = \left(\sum_{j=1}^{m} c_{i_j} \bmod p \right) \bmod q$.

Fig. 1. (t, m, n)-Group oriented SS based on PRC.

$$s = \left(\sum_{j=1}^{m} RC_j \bmod p\right) \bmod q$$



Correctness of (t,m,n)-GOSS

$$(\sum_{i_j \in I_m} c_{i_j} mod\ p) mod\ q$$

$$= \sum_{j=1}^{m} (f(x_{i_j}) \prod_{v=1, v\neq j}^{m} \frac{-x_{i_v}}{x_{i_j} - x_{i_v}} + r_{i_j} q) mod\ p\ mod\ q$$

$$= (f(0) + \sum_{j=1}^{m} r_{i_j} q) mod\ p\ mod\ q \qquad (4\text{-}1)$$

$$= (f(0) + \sum_{j=1}^{m} r_{i_j} q) mod\ q \qquad (4\text{-}2)$$

$$= f(0)$$

Step (4-1) is equivalent to step (4-2) because of $f(0) \in F_q, \sum_{j=1}^{m} r_{i_j} q \leq \sum_{j=1}^{n} r_{i_j} q < nqq = nq^2$ and thus $(f(0) + \sum_{j=1}^{m} r_{i_j} q) < q + nq^2 < p$.

Security Analysis

- An adversary without any valid share, almost has no information about the secret in (t,m,n)-GOSS even if it has up to m-1 PRCs .
  - (lower bound: m/2 SPCs)
- (t-1) Insiders almost obtains no information about the secret even if they conspire in (t,m,n)-GOSS.

Properties of (t,m,n)-GOSS

- ***Single share***
- Group oriented
- Unconditionally secure
- Without user authentication or share verification

- the secret can be reconstructed only if each participant has a valid share and releases its valid RC honestly.
- Information rate:

IR=(size of secret) / (size of share)

=log q/log p (1/3, 1/2)

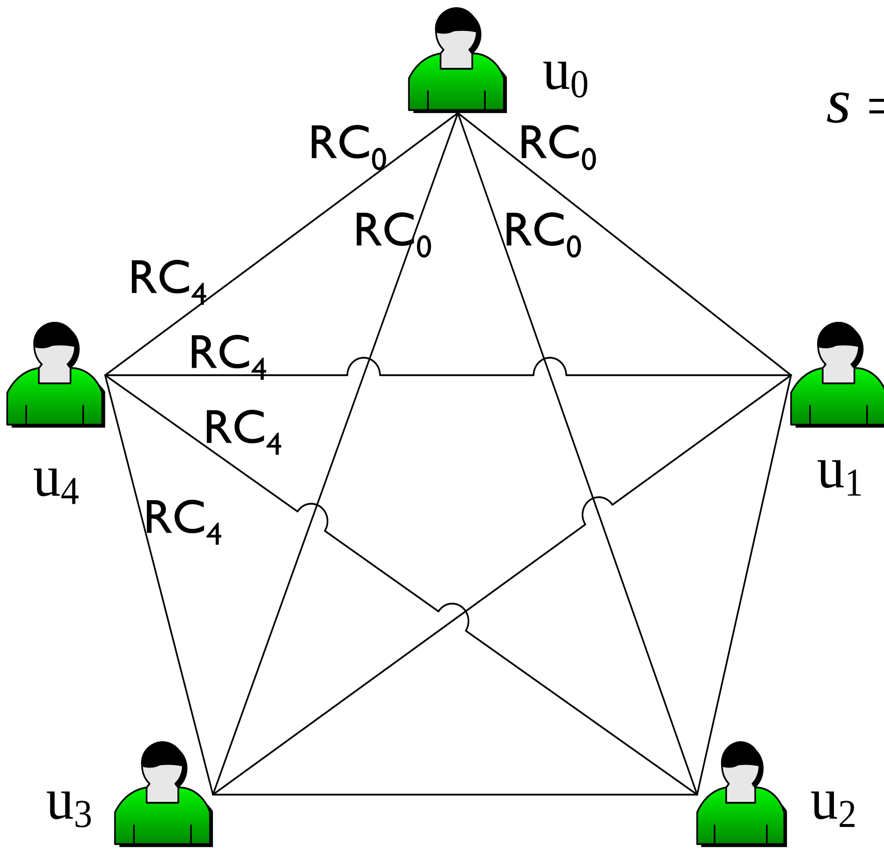
Conclusion of (t,n)-GOSS

- 2 attacks against (t,n)-SS
- Randomized Component
  - Protect shares
  - Bind a participant with all the others
- (t,m,n)-GOSS
  - Guarantees the secret can be recovered only if all m participants have valid shares and act honestly.

Application of GOSS

- Asynchronous Group Authentication
- Key agreement with authentication

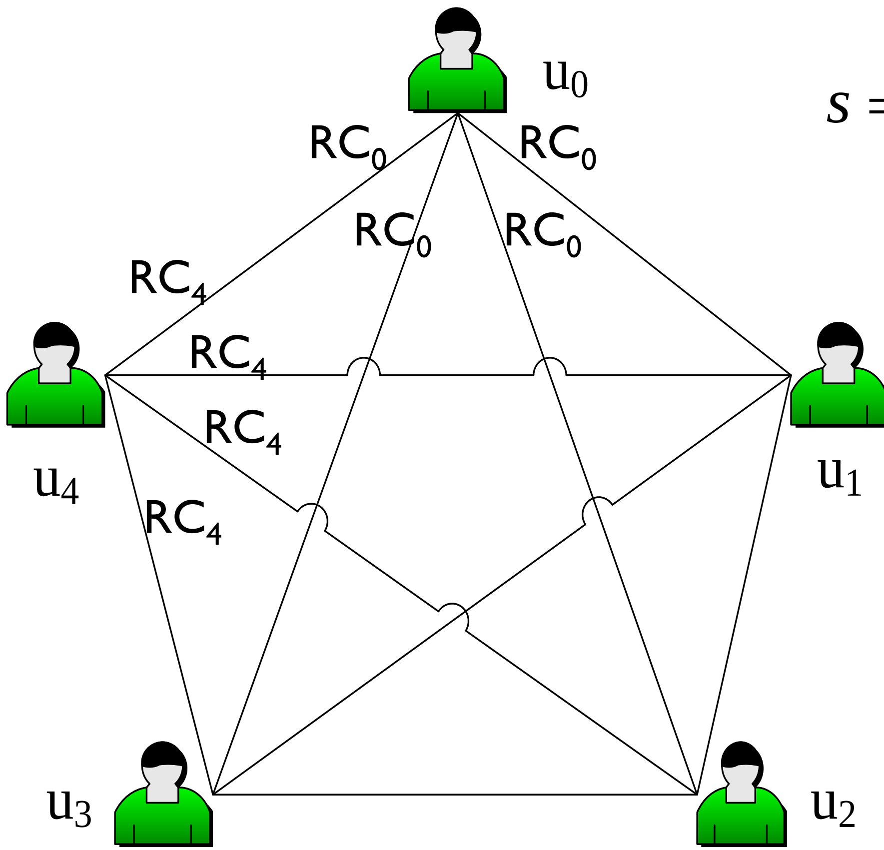
Asynchronous Group Authentication



$$s = (\sum_{j=1}^{m} RC_j \bmod p) \bmod q$$

1. Dealer: Publish H(s) in advance;
2. Users :use the secret s as the authentication token

Key agreement with authentication



$$s = \left(\sum_{j=1}^{m} RC_j \bmod p\right) \bmod q$$

1. Dealer: Publish H(s) in advance;
2. users: use the secret s as the authentication token

3. Shared Key:

$$k = \sum_{j=1}^{m} RC_j \bmod p$$

Support different groups, multiple key agreement

Ideal (t,m,n)-GOSS

- Ideal Secret Sharing
  - A share is the same as the secret in size;
  - Not information can be obtained about the secret with less than t shares
- (t,m,n)-GOSS
  - Secret is in Fq while shares are in Fp
  - Log q/log p <1/2

- How to construct an

  Ideal (t,m,n)-GOSS?

Randomized Component

- Functionality:
  - Protecting a share from exposure during secret reconstruction
  - Capability of secret reconstruction

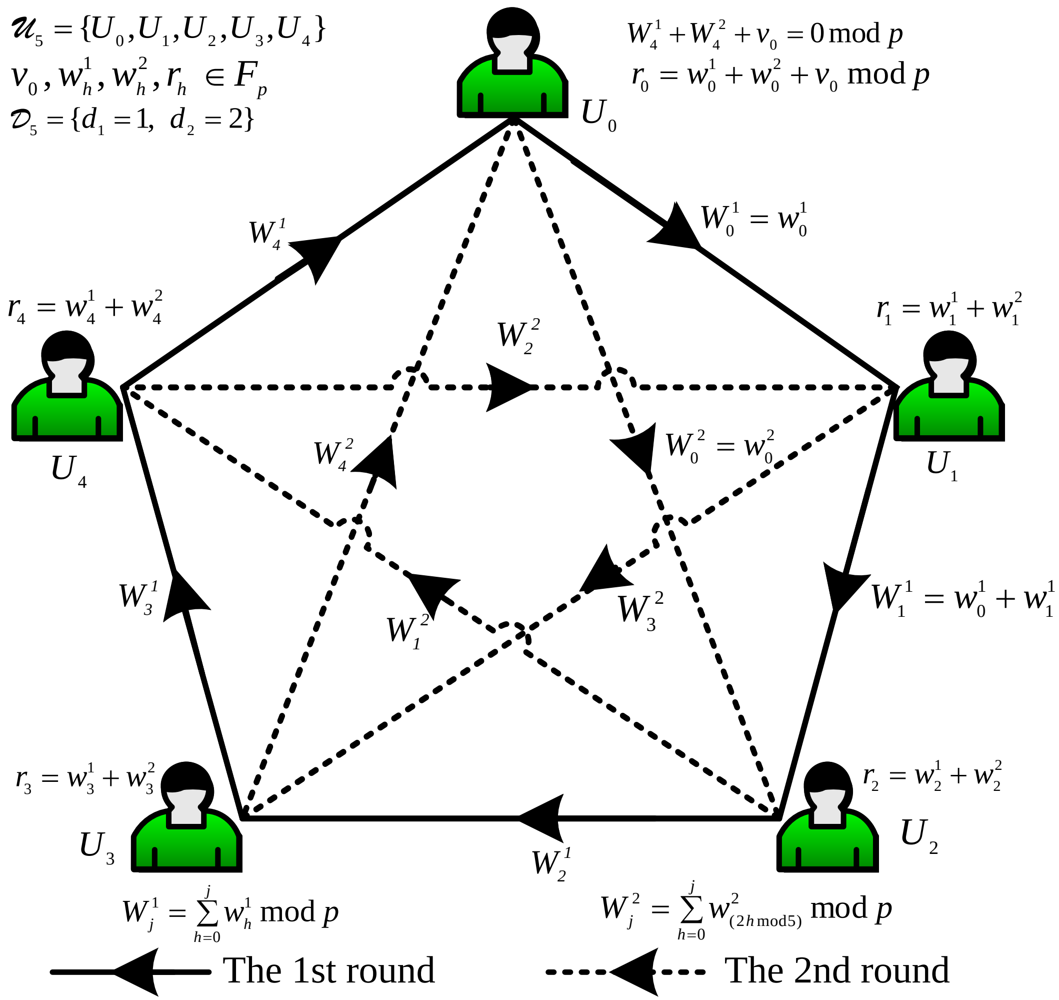
$$s = f(0) = \sum_{j \in J_m} \boxed{s_j \prod_{r \in J_m, r \neq j} \frac{0 - x_r}{x_j - x_r}} \bmod p.$$

$$RC_j = (s_j \prod_{r \in J_m, r \neq j} \frac{0 - x_r}{x_j - x_r}) + r_j \bmod p \longrightarrow \boldsymbol{S}$$

1. $\boldsymbol{r}_j$ has the same range as s_j

2. All r can be removed finally in secret reconstruction

Ideal (t,m,n)-GOSS



Randomized Component

$$RC_j = s_j \prod_{r \in J_m, r \neq j} \frac{0 - x_r}{x_j - x_r} + r_j \bmod p$$

$$s = \sum_{j=1}^{m} RC_j \bmod p$$

Thanks!

Any Question?