Discrete Structures (Monsoon 2022)

Ashok Kumar Das

Associate Professor
IEEE Senior Member
Center for Security, Theory and Algorithmic Research
International Institute of Information Technology, Hyderabad (IIIT Hyderabad)

E-mail: *ashok.das@iiit.ac.in*
URL: `http://www.iiit.ac.in/people/faculty/ashokkdas`
`https://sites.google.com/view/iitkgpakdas/`

Hierarchical Access Control

Hierarchical Access Control

Overview of Hierarchical Access Control

- Hierarchical access control is a fundamental problem in computer and network systems.
- In a hierarchical access control, a user of higher security level class has the ability to access information items (such as message, data, files, etc.) of other users of lower security classes.
- A user hierarchy consists of a number n of disjoint security classes, say, SC_1, SC_2, ..., SC_n. Let this set be $SC = \{SC_1, SC_2, \ldots, SC_n\}$.
- A binary partially ordered relation $\geq$ is defined in SC as $SC_i \geq SC_j$, which means that the security class SC_i has a security clearance higher than or equal to the security class SC_j.

Hierarchical Access Control

Overview of Hierarchical Access Control

- In addition the relation $\geq$ satisfies the following properties:
  - **[Reflexive property]** $SC_i \geq SC_i, \forall SC_i \in SC$.
  - **[Anti-symmetric property]** If $SC_i, SC_j \in SC$ such that $SC_i \geq SC_j$ and $SC_j \geq SC_i$, then $SC_i = SC_j$.
  - **[Transitive property]** If $SC_i, SC_j, SC_k \in SC$ such that $SC_i \geq SC_j$ and $SC_j \geq SC_k$, then $SC_i \geq SC_k$.
- If $SC_i \geq SC_j$, we call SC_i as the predecessor of SC_j and SC_j as the successor of SC_i. If $SC_i \geq SC_k \geq SC_j$, then SC_k is an intermediate security class. In this case SC_k is the predecessor of SC_j and SC_i is the predecessor of SC_k.
- In a user hierarchy, the encrypted message by a successor security class is only decrypted by that successor class as well as its all predecessor security classes in that hierarchy.

Hierarchical Access Control

Overview of Hierarchical Access Control

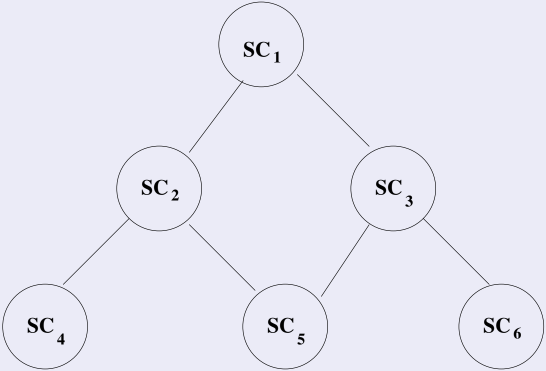

Figure: A small sample of poset in a user hierarchy.

Hierarchical Access Control

Applications of Hierarchical Access Control

- Military
- Government schools and colleges
- Private corporations
- Computer network systems
- Operating systems
- Database management systems

Chung et al.'s User Hierarchical Access Control Scheme

Reference

- Y. F. Chung, H. H. Lee, F. Lai and T. S. Chen, "Access control in user hierarchy based on elliptic curve cryptosystem", Information Sciences (Elsevier), vol. 178, no. 1, pp. 230-243, 2008 (2021 SCI Impact Factor: 8.233). [Research Paper Link: `https://www.sciencedirect.com/science/article/pii/S0020025507003763`]

Chung et al.'s User Hierarchical Access Control Scheme

Relationship Building Phase

- CA (central authority) builds a hierarchical structure for controlling access according to the relationships among the nodes in the hierarchy.
- Let $U = \{SC_1, SC_2, \ldots, SC_n\}$ be a set of n security classes in the hierarchy. Assume that SC_i is a security class with higher clearance and SC_j a security class with lower clearance, that is, $SC_i \geq SC_j$.
- A legitimate relationship $(SC_i, SC_j) \in R_{i,j}$ between two security classes SC_i and SC_j exists in the hierarchy if SC_i can access SC_j.

Chung et al.'s User Hierarchical Access Control Scheme (Continued...)

Key Generation Phase

CA performs the following steps:

- **Step 1:** Randomly selects a large prime p.
- **Step 2:** Selects an elliptic curve $E_p(a, b)$ defined over Z_p such that the order of $E_p(a, b)$ lies in the interval $[p + 1 - 2\sqrt{p}, p + 1 + 2\sqrt{p}]$.
- **Step 3:** Selects a one-way function $h(\cdot)$ to transform a point into a number and a base point G_j from $E_p(a, b)$ for each security class SC_j $1 \leq j \leq n$.
- **Step 4:** For each security class SC_j $(1 \leq j \leq n)$, selects a secret key sk_j and a sub-secret key s_j.
- **Step 5:** For all $\{SC_i | (SC_i, SC_j)\} \in R_{i,j}$, computes the followings:
  $s_i G_j = (x_{j,i}, y_{j,i})$,
  $h(x_{j,i} || y_{j,i})$, where $||$ is a bit concatenation operator.

Chung et al.'s User Hierarchical Access Control Scheme (Continued...)

Key Generation Phase (Continued...)

- **Step 6:** Finally, computes the public polynomial $f_j(x)$ using the values of $h(x_{j,i}||y_{j,i})$ as

$$f_j(x) = \prod_{SC_i \geq SC_j} (x - h(x_{j,i}||y_{j,i})) + sk_j \pmod{p}$$

- **Step 7:** Sends sk_j and s_j to the security class SC_j via a secret channel.
- **Step 8:** Announces $p, h(\cdot), G_j, f_j(x)$ as public.

Chung et al.'s User Hierarchical Access Control Scheme (Continued...)

Key Derivation Phase

In order to compute the secret keys sk_j of all successors, SC_j, the predecessor SC_i, for which the relationships $(SC_i, SC_j) \in R_{i,j}$ between SC_i and SC_j hold, proceeds as follows:

- Step 1: For $\{SC_i | (SC_i, SC_j)\} \in R_{i,j}$, computes the followings:
  $s_i G_j = (x_{j,i}, y_{j,i})$,
  $h(x_{j,i} || y_{j,i})$.
- Step 2: Computes the secret key sk_j using $h(x_{j,i} || y_{j,i})$ as follows:

$$
\begin{aligned}
f_j(x) &= \prod_{SC_i \geq SC_j} (x - h(x_{j,i} || y_{j,i})) + sk_j \pmod{p}, \\
f_j(h(x_{j,i} || y_{j,i})) &= sk_j \pmod{p}.
\end{aligned}
$$

Chung et al.'s User Hierarchical Access Control Scheme (Continued...)

Key Derivation Phase (Continued...)

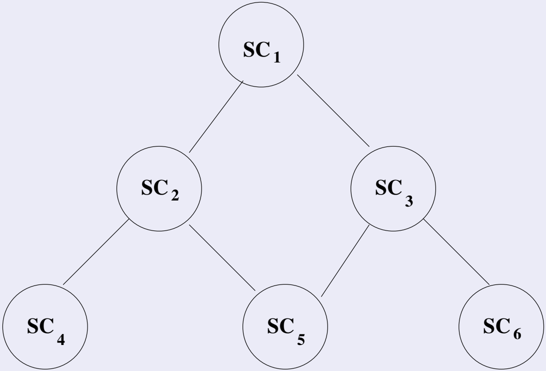

Figure: A small sample of poset in a user hierarchy.

Chung et al.'s User Hierarchical Access Control Scheme (Continued...)

Key Derivation Phase (Continued...)

$$
\begin{aligned}
f_j(x) &= \prod_{SC_i \geq SC_j} [x - h(x_{j,i}||y_{j,i})] + sk_j \pmod{p}, \\
SC_1 : f_1(x) &= [x - h(x_{1,0}||y_{1,0})] + sk_1 \pmod{p}, \text{ where } s_0 \text{ is given} \\
& \text{by CA} \\
SC_2 : f_2(x) &= [x - h(x_{2,1}||y_{2,1})] + sk_2 \pmod{p}, \\
SC_3 : f_3(x) &= [x - h(x_{3,1}||y_{3,1})] + sk_3 \pmod{p}, \\
SC_4 : f_4(x) &= [x - h(x_{4,1}||y_{4,1})][x - h(x_{4,2}||y_{4,2})] + sk_4 \pmod{p}, \\
SC_5 : f_5(x) &= [x - h(x_{5,1}||y_{5,1})][x - h(x_{5,2}||y_{5,2})][x - h(x_{5,3}||y_{5,3})] \\
& + sk_5 \pmod{p}, \\
SC_6 : f_6(x) &= [x - h(x_{6,1}||y_{6,1})][x - h(x_{6,3}||y_{6,3})] + sk_6 \pmod{p}
\end{aligned}
$$

Chung et al.'s Scheme (Continued...)

Key Derivation Phase (Continued...)

To derive the secret key sk_5 of SC_5 by its predecessor class SC_2, SC_2 needs to do following:

- Computes $s_2 G_5 = (x_{5,2}, y_{5,2})$ and then $h(x_{5,2}||y_{5,2})$.
- Determines sk_5 using $h(x_{5,2}||y_{5,2})$ from the public polynomial $f_5(x) = [x - h(x_{5,1}||y_{5,1})][x - h(x_{5,2}||y_{5,2})][x - h(x_{5,3}||y_{5,3})] + sk_5 \pmod{p}$ as $sk_5 = f_5(h(x_{5,2}||y_{5,2})) \pmod{p}$.

Chung et al.'s Scheme (Continued...)

Inserting New Security Classes Phase

If a new security class SC_k is inserted into the hierarchy such that $SC_i \geq SC_k \geq SC_j$, then the relationships $(SC_i, SC_k) \in R_{i,k}$ for $SC_i \geq SC_k$ and $(SC_k, SC_j) \in R_{k,j}$ for $SC_k \geq SC_j$ need to be updated into the hierarchy. CA needs the following steps to manage the accessing priority of SC_k in the hierarchy.

- Step 1: Updates the partial relationships R that follows when the security class SC_k joins the hierarchy.
- Step 2: Randomly selects the secret key sk_k, the sub-secret key s_k and the base point G_k for the class SC_k.
- Step 3: For all $\{SC_i | (SC_i, SC_k)\} \in R_{i,k}$ that satisfies $SC_i \geq SC_k$ when the new class SC_k is inserted in the hierarchy, computes $s_i G_k = (x_{k,i}, y_{k,i})$, and $h(x_{k,i} || y_{k,i})$.

Chung et al.'s User Hierarchical Access Control Scheme (Continued...)

Inserting New Security Classes Phase (continued...)

- Step 4: Computes the public polynomial $f_k(x)$ as follows:

$$f_k(x) = \prod_{SC_i \geq SC_k} (x - h(x_{k,i}||y_{k,i})) + sk_k \pmod{p}$$

- Step 5: For all $\{SC_i|(SC_i, SC_k)\} \in R_{i,k}$ and $\{SC_k|(SC_k, SC_j)\} \in R_{k,j}$ that satisfy $SC_i \geq SC_k \geq SC_j$ when the new class SC_k is inserted in the hierarchy, computes
  $s_k G_j = (x_{j,k}, y_{j,k})$,
  $s_i G_j = (x_{j,i}, y_{j,i})$,
  $h(x_{j,k}||y_{j,k})$ and $h(x_{j,i}||y_{j,i})$.

Chung et al.'s User Hierarchical Access Control Scheme (Continued...)

Inserting New Security Classes Phase (continued...)

- Step 6: Computes the public polynomial $f'_j(x)$ as follows:

$$f'_j(x) = \prod_{SC_i \geq SC_k \geq SC_j} (x - h(x_{j,i} || y_{j,i}))(x - h(x_{j,k} || y_{j,k})) + sk_j \pmod{p}$$

- Step 7: Replaces $f_j(x)$ with $f'_j(x)$, and sends sk_k and s_k to SC_k via a secure channel, and announces publicly G_k, $f_k(x)$ and $f'_j(x)$.

Chung et al.'s User Hierarchical Access Control Scheme (Continued...)

Inserting New Security Classes Phase (continued...)

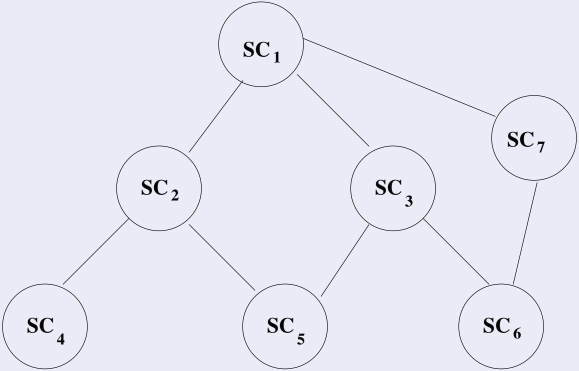

Figure: A small sample of poset in a user hierarchy: when a new security class SC_7 is added into the hierarchy.

Chung et al.'s User Hierarchical Access Control Scheme (Continued...)

Inserting New Security Classes Phase (continued...)

$$\begin{aligned}
SC_1 : f_1(x) &= [x - h(x_{1,0}||y_{1,0})] + sk_1 \pmod p, \text{ where } s_0 \text{ is given by CA} \\
SC_2 : f_2(x) &= [x - h(x_{2,1}||y_{2,1})] + sk_2 \pmod p, \\
SC_3 : f_3(x) &= [x - h(x_{3,1}||y_{3,1})] + sk_3 \pmod p, \\
SC_4 : f_4(x) &= [x - h(x_{4,1}||y_{4,1})][x - h(x_{4,2}||y_{4,2})] + sk_4 \pmod p, \\
SC_5 : f_5(x) &= [x - h(x_{5,1}||y_{5,1})][x - h(x_{5,2}||y_{5,2})][x - h(x_{5,3}||y_{5,3})] \\
&\quad + sk_5 \pmod p, \\
SC_6 : f_6'(x) &= [x - h(x_{6,1}||y_{6,1})][x - h(x_{6,3}||y_{6,3})][x - h(x_{6,7}||y_{6,7})] \\
&\quad + sk_6 \pmod p \\
SC_7 : f_7(x) &= [x - h(x_{7,1}||y_{7,1})] + sk_7 \pmod p
\end{aligned}$$

Chung et al.'s User Hierarchical Access Control Scheme (Continued...)

Removing Existing Security Classes Phase

If an existing member SC_k, such that the relationship $SC_i \geq SC_k \geq SC_j$ breaks up, wants to leave from a user hierarchy, then CA not only directly revokes information related to SC_k, but also alters the accessing relationship between the involved ex-predecessor SC_i and ex-successor SC_j of SC_k. In this phase, CA executes the following steps.

- Step 1: Updates the partial relationship R that follows when SC_k is removed.

Chung et al.'s User Hierarchical Access Control Scheme (Continued...)

Removing Existing Security Classes Phase (Continued...)

- Step 2: For all $\{SC_k | (SC_k, SC_j)\} \in R_{k,j}$ does the followings:
  - Step 2.1: Renews the secret key sk_j as sk_j' and the base point G_j as G_j' of SC_j.
  - Step 2.2: For all $\{SC_i | (SC_i, SC_j)\} \in R_{i,j}$ does the followings:
    - Step 2.2.1: Renews $\{SC_i | (SC_i, SC_j)\} \in R_{i,j}$ after removing SC_k.
    - Step 2.2.2: Computes $s_i G_j' = (x_{j,i}, y_{j,i})$.
    - Step 2.2.3: Computes $h(x_{j,i}, y_{j,i})$.
    - Step 2.2.4: Computes the public polynomial $f_j'(x)$ as

$$f_j'(x) = \prod_{SC_i \geq SC_j} (x - h(x_{j,i} || y_{j,i})) + sk_j' \pmod p$$

    - Step 2.2.5: Replaces $f_j(x)$ with $f_j'(x)$.
- Step 3: Sends sk_j' to SC_j via a secret channel and announces G_j' and $f_j'(x)$ as public.

Chung et al.'s Scheme (Continued...)

Removing Existing Security Classes Phase (continued...)

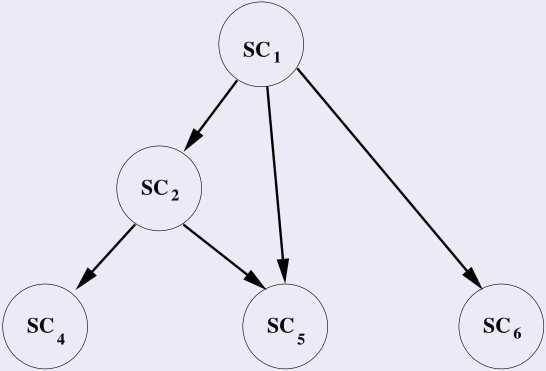

Figure: A small sample of poset in a user hierarchy: when an existing security class SC_3 is removed into the hierarchy.

Chung et al.'s Scheme (Continued...)

Removing Existing Security Classes Phase (continued...)

- Before deleting SC_3, $f_5(x)$ and $f_6(x)$ are formed as
  $f_5(x) = [x - h(x_{5,1}||y_{5,1})][x - h(x_{5,2}||y_{5,2})][x - h(x_{5,3}||y_{5,3})] + sk_5 \pmod{p}$
  $f_6(x) = [x - h(x_{6,1}||y_{6,1})][x - h(x_{6,3}||y_{6,3})] + sk_6 \pmod{p}$.
- After deleting SC_3, $f'_5(x)$ and $f'_6(x)$ are formed as
  $f'_5(x) = [x - h(x_{5,1}||y_{5,1})][x - h(x_{5,2}||y_{5,2})] + sk'_5 \pmod{p}$
  $f'_6(x) = [x - h(x_{6,1}||y_{6,1})] + sk'_6 \pmod{p}$.

Chung et al.'s Scheme (Continued...)

Creating New Relationships

- Suppose we want to create a new relationship between SC_5 and SC_6 in the hierarchy (Figure 1) such that $SC_2 \geq SC_5 \geq SC_6$.

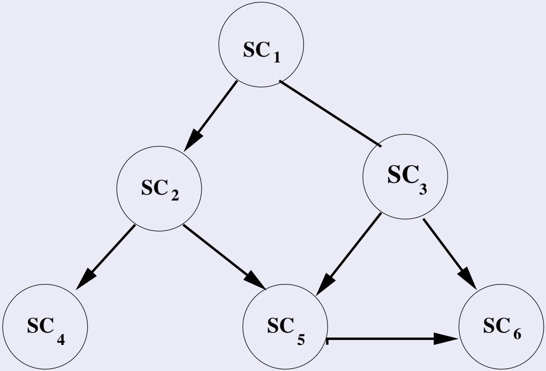

Figure: The consequent poset after creating $SC_5 \geq SC_6$ in Figure 1.

Chung et al.'s Scheme (Continued...)

Creating New Relationships

- Before creating the relationship $SC_2 \geq SC_5 \geq SC_6$, $f_6(x)$ is formed as follows:

$$f_6(x) = [x - h(x_{6,1}||y_{6,1})][x - h(x_{6,3}||y_{6,3})] + sk_6 \pmod{p}.$$

- After creating the relationship $SC_2 \geq SC_5 \geq SC_6$, updated public polynomial $f_6'(x)$ is formed as follows:

$$\begin{aligned} f_6'(x) = {} & [x - h(x_{6,1}||y_{6,1})][x - h(x_{6,3}||y_{6,3})] \\ & [x - h(x_{6,2}||y_{6,2})][x - h(x_{6,5}||y_{6,5})] + sk_6 \pmod{p}. \end{aligned}$$

Chung et al.'s Scheme (Continued...)

Revoking Existing Relationships

- Suppose we want to revoke the existing relationship $\{SC_2 | (SC_2, SC_5) \in R_{2,5}\}$ in the following figure such that $\{SC_2 | (SC_2, SC_5) \notin R_{2,5}\}$

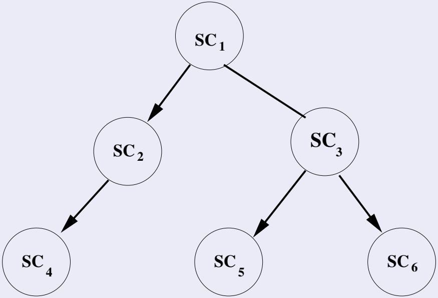

Figure: The consequent poset after revoking $SC_2 \geq SC_5$ in Figure 1.

Chung et al.'s Scheme (Continued...)

Revoking Existing Relationships

- Before revoking $\{SC_2 | (SC_2, SC_5) \in R_{2,5}\}$, $f_5(x)$ is formed as follows:

$$\begin{aligned} f_5(x) &= [x - h(x_{5,1}||y_{5,1})][x - h(x_{5,2}||y_{5,2})] \\ &\quad [x - h(x_{5,3}||y_{5,3})] + sk_5 \pmod{p}. \end{aligned}$$

- After revoking $\{SC_2 | (SC_2, SC_5) \in R_{2,5}\}$, $f_5(x)$ is replaced with the updated $f_5'(x)$ as follows:

$$f_5'(x) = [x - h(x_{5,1}||y_{5,1})][x - h(x_{5,3}||y_{5,3})] + sk_5' \pmod{p}.$$

after renewing the secret key sk_5' in place of sk_5.

Chung et al.'s Scheme (Continued...)

Changing Secret Keys

- A secret key must be changeable to maximize security.
- To change a secret key sk_j to sk_j', CA must replace the base point G_j with G_j' and the public polynomial $f_j(x)$ with $f_j'(x)$ as follows.
  - Step 1: Replace the secret key sk_j with sk_j' and the base point G_j with G_j'.
  - Step 2: For all $\{SC_i | (SC_i, SC_j)\} \in R_{i,j}$:
    - Step 2.1: Determine $s_i G_j' = (x_{j,i}, y_{j,i})$
    - Step 2.2: Determine $h(x_{j,i} || y_{j,i})$, where $||$ is a bit concatenation operator

Chung et al.'s Scheme (Continued...)

Changing Secret Keys (Continued...)

- Step 3: Determine the public polynomial $f'_j(x)$ as follows

$$f'_j(x) \quad = \quad \prod_{SC_i \geq SC_j} [x - h(x_{j,i} || y_{j,i})] + sk'_j \pmod p$$

- Step 4: Replace $f_j(x)$ with $f'_j(x)$
- Step 5: Send sk'_j to SC_j via a secret channel, and announce G'_j and $f'_j(x)$

Cryptanalysis and Improvement of Chung et al.'s Scheme

- **Ashok Kumar Das, Nayan Ranjan Paul, and Laxminath Tripathy. "Cryptanalysis and improvement of an access control in user hierarchy based on elliptic curve cryptosystem," in *Information Sciences (Elsevier)*, Vol. 209, No. C, pp. 80 - 92, 2012, doi: http://dx .doi.org/10.1016/ j.ins.2012.04.036. (2021 SCI Impact Factor: 8.233)** [Research Paper Link: `https://www.sciencedirect.com/science/article/pii/S0020025512003155`]