Key-Policy Attribute-Based Encryption

Present by Xiaokui Shu
09/08/2011
Application Scenario

Basic KP-ABE

Advanced Topics

Related Work

Identity-Based Encryption

Secret-Sharing Scheme

Fine-grained Access Control

Identity-Based Encryption

Secret-Sharing Scheme

Fine-grained Access Control
How to share encrypted data?

- Alice decrypt data for Bob
- Alice give its key to Bob
Log need to be encrypted

- Cannot be known to everyone

Collaborative forensic analysis

- Someone need to known it
- “Someone” is not fixed

Attributes

- User name
- Permitted time
Scenario

- A broadcaster broadcasts different items
- Each user is subscribed to a different “package”

Broadcast Encryption Schemes

- Encrypting to an arbitrary subset of users
- Efficiency dependents on the size of users

Attributes

- E.g. sports, news, financial
ABE is not enough.
Fine-grained access control needed.
Key-Policy Attribute-Based Encryption

Ts == 2

d

admin

Alice

Bob

Pro.

time

CIA

spec

CS

copro

OR

OR

OR
All You Need to Know About KP-ABE

KP-ABE

- IBE
- SSS Tree
  - Secret-Sharing Scheme
  - Fine-grained Access Control
Identity-based Encryption

Private Key Generator (PKG)

Setup

master-key

Extract

params

Sender

Encrypt

M

Receiver

d_{ID}

Decrypt

M

ID

C

ID

Cite from http://courses.cs.vt.edu/cs6204/Privacy-Security/Presentations/Identity-Based-Encryption.pptx
Key-Policy Attribute-Based Encryption :: IBE

- **Setup**
  - master-key

- **Key Generation**
  - A

- **Receiver**
  - d
  - Decrypt
  - C

- **Encrypt**
  - M

- **Sender**
  - att
  - s

- **Key Generator**
  - params

- **Encrypt**
  - master-key

- **Decrypt**
  - M
Play with math.
Bilinear map

\[ e(u^a, v^b) = e(u, v)^{ab} \]
Identity-Based Encryption

- Public parameters
  \[ Y = e(g, g)^y \]

- Encryption
  \[ E' = MY^s \]

- Decryption
  \[ e(g^{q_x(0)}t_i, g^{s \cdot t_i}) = e(g, g)^{s \cdot q_x(0)} = e(g, g)^{ys} = Y^s \]

Key to each attribute
Clue
Secret
Identity-Based Encryption

- Public parameters

\[ T_1 = g^{t_1}, \ldots, T_{|U|} = g^{t_{|U|}}, Y = e(g, g)^y \]

- Encryption

\[ E = (\gamma, E' = MY^s, \{E_i = T_i^s\}_{i \in \gamma}) \]

- Decryption

\[ e\left(g^{\frac{q_x(0)}{t_i}}, g^{s \cdot t_i}\right) = e(g, g)^{s \cdot q_x(0)} = e(g, g)^{y^s} = Y^s \]
Secret-Sharing Scheme

\[ P_2(x) \]
\[ P_3(x) \]
\[ P_4(x) \]
\[ P_5(x) \]
Secret of lower level is share of higher level
Secret-Sharing Scheme

Π
\[ F_x = \prod_{z \in S_x} \Delta_{i, S'_x}^{(0)} \]
where

\[ i = \text{index}(z) \]
\[ S'_x = \{ \text{index}(z) : z \in S_x \} \]

\[ = \prod_{z \in S_x} \left( e(g, g)^{s \cdot q_{z(0)}} \right)^{\Delta_{i, S'_x}^{(0)}} \]

Output: leaves'

\[ = \prod_{z \in S_x} \left( e(g, g)^{s \cdot q_{\text{parent}(z)(\text{index}(z))}} \right)^{\Delta_{i, S'_x}^{(0)}} \]

Output: current node
AND, OR and Threshold

- n: number of children of a node
- k: number of shares to unlock a secret of a node

- AND: \( k = n \)
- OR: \( k = 1 \)
- Threshold: \( k > 1 \) && \( k < n \)

- Leaf: \( k = 1 \)
Sketch is done.
Collusion

CS, admin

ECE, dean

CS, dean?
Collusion resistance

- not set, but access tree
From limited attributes to any arbitrary strings

- Unchanged: Setup, Encryption
- Modified: Key Generation, Decryption

\[ H : \{0, 1\}^* \rightarrow \mathbb{Z}_p^* \]
Advanced Topics: Delegation of Private Keys
Manipulations

- Adding a new trivial gate to $T$
  - blue-green -> blue-red-green with 0-degree poly

- Manipulating an existing $(t, n)$-gate
  - Converting a $(t, n)$-gate to a $(t + 1, n)$-gate
  - Converting a $(t, n)$-gate to a $(t + 1, n + 1)$-gate
  - Converting a $(t, n)$-gate to a $(t, n-1)$-gate

- Re-randomizing the obtained key
  - New random poly

"The given set of operations is complete"
Related Work: CP-ABE

- KP-ABE
  - ciphertexts are associated with sets of attributes
  - user secret keys are associated with policies

- CP-ABE
  - user keys are associated with sets of attributes
  - ciphertexts are associated with policies
Thank you!
Decisional Bilinear Diffie-Hellman Assumption

\[(A = g^a, B = g^b, C = g^c, e(g, g)^{abc})\]
**Manipulations details**

- Adding a new trivial gate to $T$
- Converting a $(t, n)$-gate to a $(t + 1, n)$-gate
  \[ q'_x(X) = (X + 1)q_x(X) \]
- Converting a $(t, n)$-gate to a $(t + 1, n + 1)$-gate
  \[ q'_x(X) = (aX + 1)q_x(X) \text{ where } a = \frac{-1}{v}, \text{ index}(z) = v \]
- Converting a $(t, n)$-gate to a $(t, n-1)$-gate