廣播加密法
Broadcast Encryption

吳宗成教授
國立台灣科技大學資訊管理系

E-mail: tcwu@cs.ntust.edu.tw
TEL: (02) 27376780

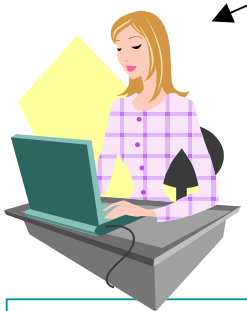
Broadcast Encryption

- Monumental problem:
  - Pirate copying of digital content via Internet, such as MPEG 3, video, or software
- Functional goals
  - Broadcast/multicast
  - Fingerprinting for broadcasted content
- Security requirements
  - Key management
  - Revocation
  - Traitor tracing

General System Model for Broadcast Encryption

content provider

broadcast the encrypted digital content



subscriber 1

subscriber 2

...

subscriber *n*

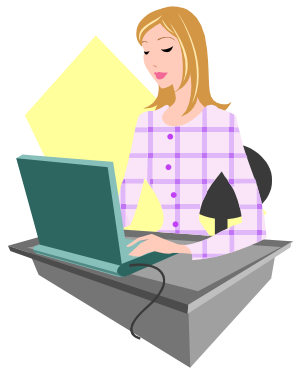
Simple Model for Broadcast Encryption

1. subscribe information

2. decryption key

(secure channel)

4. broadcast the encrypted digital content



content provider

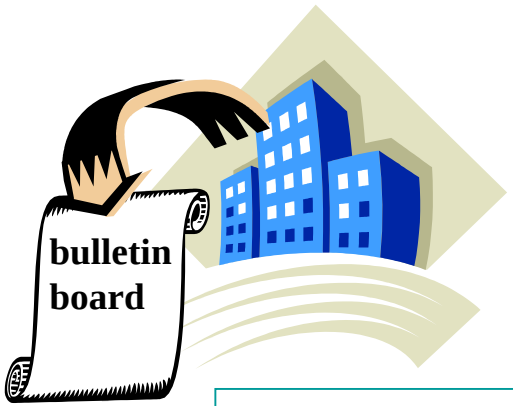
subscriber *i*

3. encrypt the digital content

5. use the decryption key to decrypt the ciphertext

What can we do if we want that **any third party is able to send secure messages** to the set of subscribers without producing their decryption keys for each broadcast?

Improved Model for Broadcast Encryption

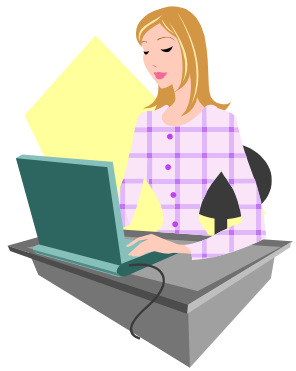


system center

2. subscribe information

3. decryption key

(secure channel)



subscriber *i*

1. publish the public key information

7. use the decryption key to decrypt the ciphertext

4. read the public key information

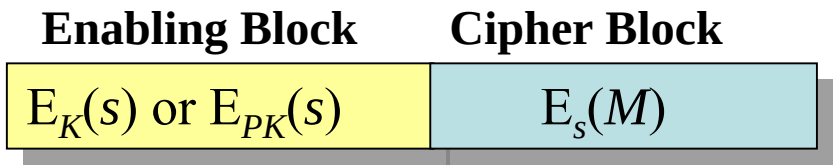
6. broadcast the encrypted digital content

5. encrypt the digital content

content provider

Data Structure for Broadcast Encryption

- The cipher block (CB) contains the actual message (or fragment) *M* encrypted by the session key *s* using symmetric encryption algorithm



- The enabling block (EB) contains the session key *s* encrypted by the random keys *K* or the public encryption key *PK* using symmetric or asymmetric encryption algorithm

Key Management

- Enable a content provider to broadcast digital content to a *large* set of privileged users
- The set of privileged users can be arbitrary, size-limited, or with hierarchical structure
- Each user only holds a single master key or a small amount of secret keys
- Content provider only holds a single master key, without knowing the secret keys held by the users, and dynamically generates random key for securing broadcast
- Trade-off : Minimize *the size of enabling block* and *the number (or the size) of keys per user*

References for Key Management

- Key Management for Multicast: Issues and Architectures (Wallner et. al, 1997)
- Multicast Security: A Taxonomy and Some Efficient Constructions (Canetti et. al, 1999)
- Long-Lived Broadcast Encryption (Garay et. al, 2000)
- Revocation and Tracing Schemes for Stateless Receivers (Naor et. al, 2001)
- The LSD Broadcast Encryption Scheme (Halevy and Shamir, 2002)

Revocation

- Enable content provider to let only legitimate users obtain the broadcast content, but exclude illegitimate users (called revoked users)
- Revoked users can be re-joined in the set of legitimate for later broadcast if necessary
- Process of revocation should not interfere with the issued private/secret keys for existing users
- Trade-off: Simplify the *process* of revocation, and further be compliant with key management and traitor tracing mechanisms

References for Revocation

- How to Broadcast a Secret (Berkovits,1991)
- Broadcast Encryption (Fiat and Naor, 1993)
- A Quick Group Key Distribution Scheme with Entity Revocation (Anzai et. al, 1999)
- Coding Constructions for Blacklisting Problems without Computational Assumptions (Kumar et. al, 1999)
- Self-Healing Key Distribution with Revocation (Staddon et. al, 2002)
- A Revocation Scheme with Minimal Storage at Receivers (Asano, 2002)

Traitor Tracing

- Malicious or collusive privileged users (called traitors) might attempt to construct a *pirate decoder* and pass it to some unauthorized users for illegal access to the broadcast content
- Enable the content provider to detect or identify the traitor(s)
- Trade-off: Simplify the *process* of traitor tracing, increase the *acceptance rate* for detecting/identifying the traitors, and further be compliant with key management and revocation mechanisms

References for Traitor Tracing

- Tracing Traitors (Chor, Fiat and Naor, 1994)
- Threshold Traitor Tracing (Naor and Pinkas, 1998)
- An Efficient Public Key Traitor Tracing Scheme (Boneh and Franklin, 1999)
- Dynamic Traitor Tracing (Fiat and Tassa, 1999)
- Sequential Traitor Tracing (Safavi and Wang, 2000)
- Traitor Tracing with Constant Transmission Rate (Kiyaias and Yung ,2002)
- New Traitor Tracing Schemes Using Bilinear Map (Tô et. al, 2003)

References for Integration of Tracing and Revocation

- Efficient Trace and Revoke Schemes (Naor and Pinkas, 2000)
- Revocation and Tracing Schemes for Stateless Receivers (Naor et. al, 2001)
- A Public-Key Traitor Tracing Scheme with Revocation Using Dynamic Shares (Tzeng and Tzeng, 2001)
- Public Key Trace and Revoke Scheme Secure against Adaptive Chosen Ciphertext Attack (Dodis and Fazio, 2003)
- Time-Bound Broadcast Encryption Mechanism (Wu and Tseng, 2004)

On the Design of Secret Broadcast (Key Management) Scheme

- **How to Broadcast a Secret** (Berkovits, 1991)
  - Apply the secret sharing scheme on the design of the secret broadcasting scheme
  - Storage for each user: O(1)
  - Message length for broadcasting: O(n)
  - Same secret to the same privileged users can not be broadcasted twice

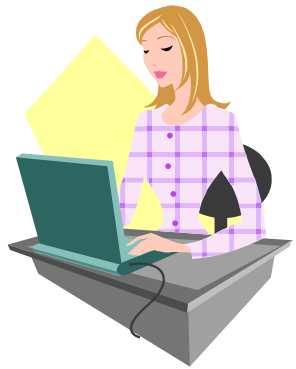
Berkovits' System Model

1. subscribe information

2. pseudoshare (x_i, y_i)

(secure channel)

4. broadcast other distinct n shares of the polynomial



content provider

subscriber *i*

3. construct a n-degree polynomial with k privileged users' shares, $(n\text{-}k)$ dummy shares, and $(0,S)$

5. reconstruct the polynomial with n broadcasted shares and pseudoshare (x_i, y_i)
6. recover the secret S from the polynomial

For the revoked user j, he has not enough shares to reconstruct the polynomial

Example for Berkovits' Scheme

- **Initiation phase**
  - Let the modulo p=11, n=3, k=2, the secret S=9
  - Secrets for users: U_1:(1, 3), U_2:(2, 5), U_3:(3, 2)

- **Broadcast phase**
  - Broadcast the secret S=9 to the privileged users U_1, U_2
    - Construct an interpolation polynomial

      (0, 9),(1, 3),(2, 5),(5,3) $\rightarrow 2x^3+9x^2+5x+9 \bmod 11$
    - Broadcast other distinct shares (6,3), (7,5), (8,9)

Example of Berkovits' Scheme

- **Decryption phase**
  - For privileged user U_1

    (6,3),(7,5),(8,9),(1,3) → $2x^3+9x^2+5x+9 \bmod 11$

    recover the secret 9
  - For privileged user U_2

    (6,3),(7,5),(8,9),(2,5) → $2x^3+9x^2+5x+9 \bmod 11$

    recover the secret 9
  - For revoked user U_3

    (6,3),(7,5),(8,9),(3,2) → *unknown* ($\neq 9$)

Example of Broadcast+Revocation
Simple Method 1 -- Fiat & Noar

- **Initiation phase**

$$U_1 : \{K_1\},\ U_2 : \{K_2\},\ U_3 : \{K_3\},\ U_4, \{K_4\}$$

- **Broadcast phase (Broadcasting message is *M*)**
  - If U_1, U_2 are privileged users, the broadcast message is

$$\{[E_{K_1}(SK), E_{K_2}(SK)], E_{SK}(M)\}$$

- **Decryption phase**
  - For the privileged user U_i $\left\{ \begin{array}{l} SK = D_{K_i}(E_{K_i}(SK)) \\ M = D_{SK}(E_{SK}(M)) \end{array} \right\}$
    ??
  - For the revoked user U_j, $SK = D_{K_j}(E_{K_i}(SK))$

Example of Broadcast+Revocation
Simple Method 2 -- Fiat & Noar

- **Initiation phase** ($n = 3$)
  - Generate all possible subsets and assign a key to each subset

$$\{1\}: K_1, \quad \{2\}: K_2, \quad \{3\}: K_3,$$
$$\{1,2\}: K_{1,2}, \quad \{1,3\}: K_{1,3}, \quad \{2,3\}: K_{2,3}, \quad \{1,2,3\}: K_{1,2,3}$$

  - Assign each users the keys corresponding to the subset he belongs to

$$U_1: \{K_1, K_{1,2}, K_{1,3}, K_{1,2,3}\}$$
$$U_2: \{K_2, K_{1,2}, K_{2,3}, K_{1,2,3}\}$$
$$U_3: \{K_3, K_{1,3}, K_{2,3}, K_{1,2,3}\}$$

Example of Broadcast+Revocation
Simple Method 2 -- Fiat & Noar

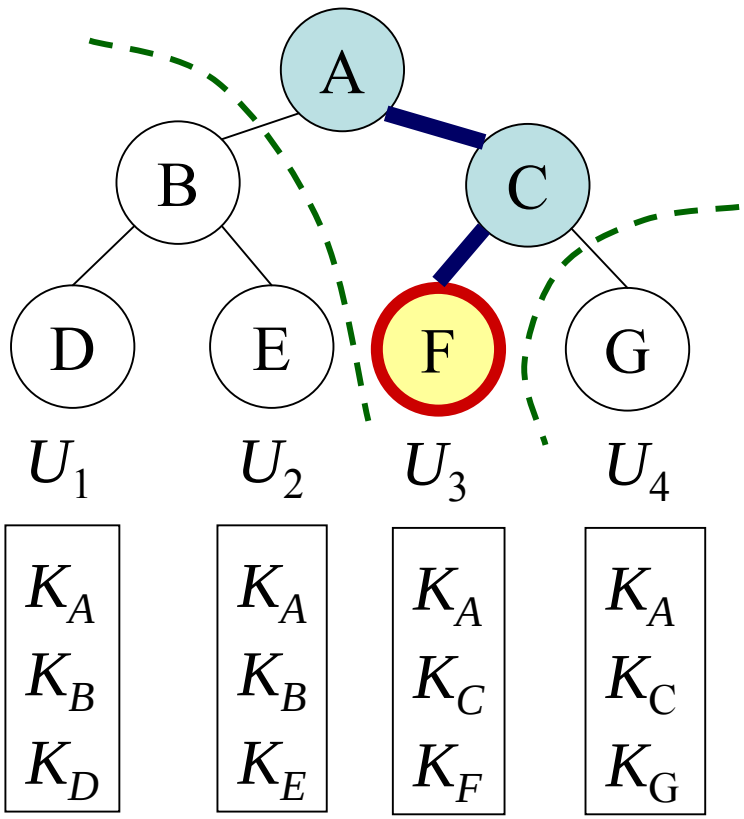
- **Broadcast phase**
  - Broadcast a message M to the privileged users
  - If U_1, U_2 are privileged users, the broadcast message is

$$\{[E_{K_{1,2}}(SK)], E_{SK}(M)\}$$

- **Decryption phase**
  - For the privileged user U_i, find the key $K_{1,2}$ corresponding to subset $\{1,2\}$

$$SK = D_{K_{1,2}}(E_{K_{1,2}}(SK))$$
$$M = D_{SK}(E_{SK}(M))$$

Example of Broadcast+Revocation
Subset-Cover Method – Noar et al.



- If no one be revoked
  - Use K_A to broadcast message M

$$\{[A, E_{K_A}(SK)], E_{SK}(M)\}$$

- If U_3 is revoked
  - Use K_B, K_G to broadcast message M

$$\{[B, G, E_{K_B}(SK), E_{K_G}(SK)], E_{SK}(M)\}$$

Storage for each user : O(log$n$)
Message length : O($r$log$n$)

Comparison of Revocation Schemes

Items / Schemes	Storage	Message Length
Berkovits's Scheme	$O(1)$	$O(n)$
Fiat & Naor's Simple Method 1	$O(1)$	$O(n\text{-}r)$
Fiat & Naor's Simple Method 2	$O(2^n)$	$O(1)$
Fiat & Naor's Unconditional Method	$O(n^r)$	$O(1)$
Fiat & Naor's Cryptographic Method	$O(r(\log r)(\log n))$	$O(r^2(\log^2 r)(\log n))$
Naor et. al, CS Method	$O(\log n)$	$O(r\log n)$
Naor et. al, SD Method	$O(\log^2 n)$	$O(r)$

n: number of users
r: number of revoked users

Traitor Tracing Scheme
-- Chor-Fiat-Naor, 1994

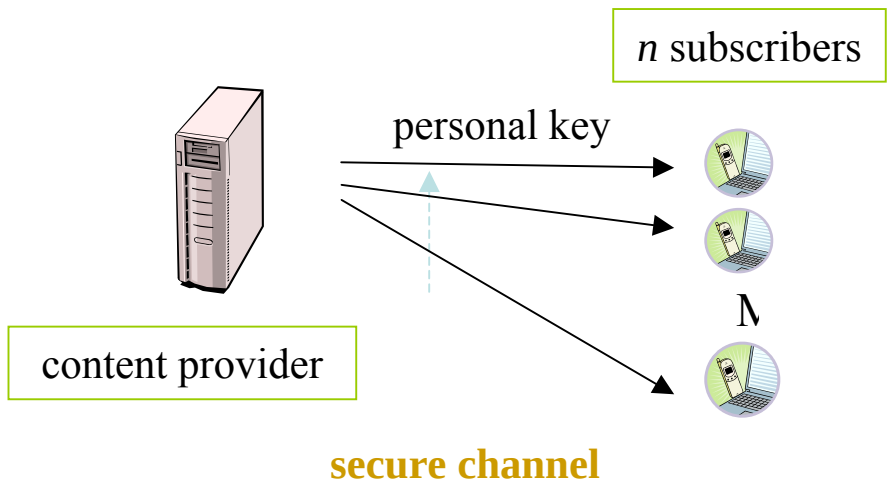
- **Notation**

- n the number of authorized users
- k the upper bound of colliding traitors
- h_i a set of hash function ($1 \le i \le w$), where $w = 4k^2 \log n$
- A_i a set of $2k^2$ random keys ($1 \le i \le w$), where $A_i = \{a_{i,1}, a_{i,2}, \text{L}, a_{i,2k^2}\}$
- $P(j)$ the personal key of user u_j, j is identity for u_j
- SK the session key $SK = s_1 \oplus s_2 \oplus \text{L} \oplus s_w$
  where s_i ($1 \le i \le w$) are randomly chosen
- E_i a set of $2k^2$ ciphertext of s_i ($1 \le i \le w$) that encrypted by $2k^2$ random keys of A_i, where $E_i = \{e_{i,1}, e_{i,2}, \text{L}, e_{i,2k^2}\}$

Traitor Tracing Scheme
-- Chor-Fiat-Naor, 1994

- **Initialization** (done by content provider)
  - Randomly generate $2k^2$ keys $\{a_{i,1}, a_{i,2}, \text{L}, a_{i,2k^2}\}$ for each A_i

$$A_1 = \{a_{1,1}, a_{1,2}, \ldots, a_{1,2k^2}\}$$
$$A_2 = \{a_{2,1}, a_{2,2}, \ldots, a_{2,2k^2}\}$$
$$\text{M}$$
$$A_w = \{a_{w,1}, a_{w,2}, \ldots, a_{w,2k^2}\}$$

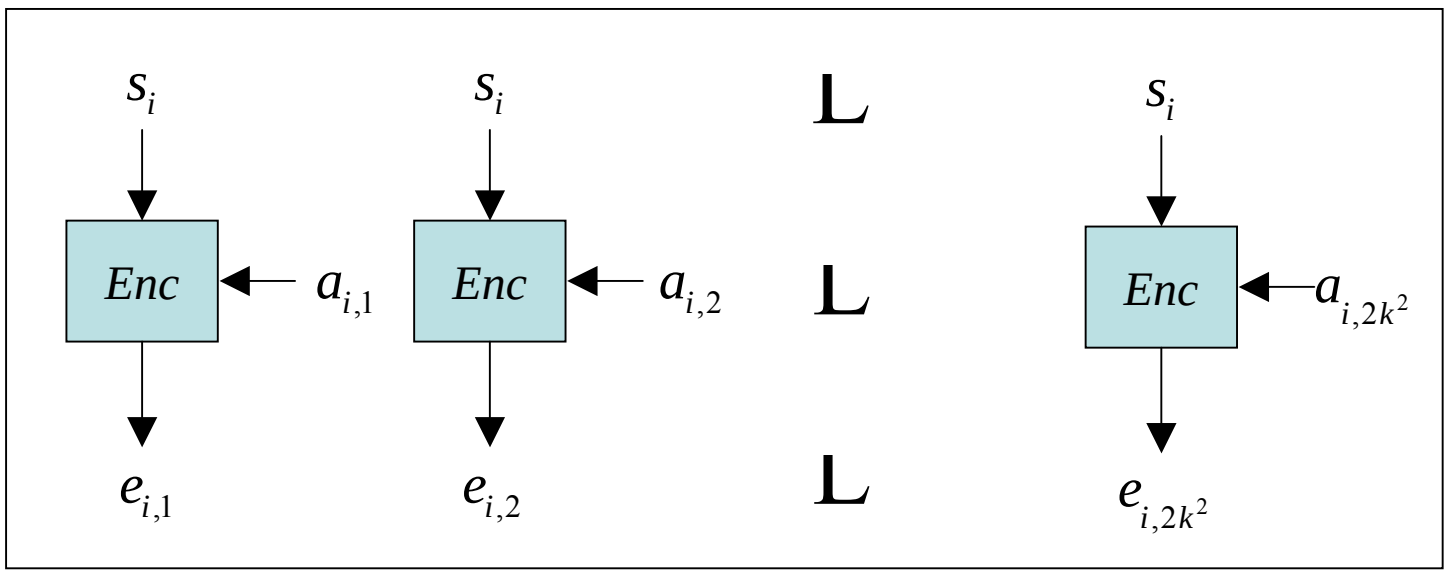


  - Determine personal key for u_j :
    $P(j) = \{a_{1,h_1(j)}, a_{2,h_2(j)}, \text{L}, a_{w,h_w(j)}\}$, where h_j maps
    $j (1 \le j \le n)$ into $\{1, 2, \text{L}, 2k^2\}$

Traitor Tracing Scheme
-- Chor-Fiat-Naor, 1994

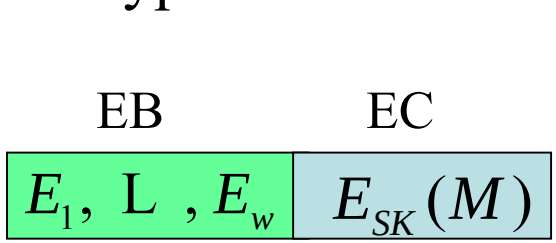
- **Broadcast**(done by content provider)
  - Randomly generate keys $\{s_1, s_2, \text{L}, s_w\}$
  - Generate enabling block $E_i (1 \le i \le w)$, where

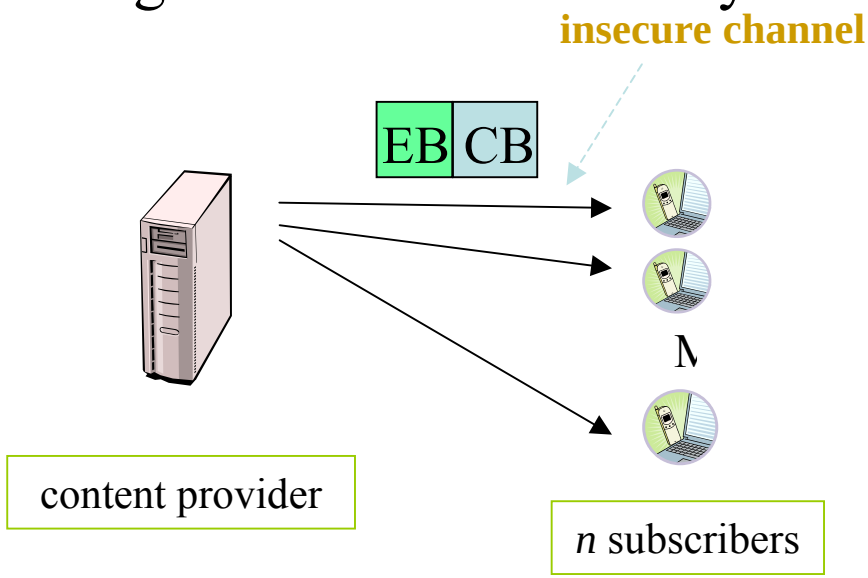
    $E_i = \{e_{i,1}, e_{i,2}, \text{L}, e_{i,2k^2}\}$, and $e_{i,c} = E_{a_{i,c}}(s_i), c = 1, 2, \text{L}, 2k^2$



Traitor Tracing Scheme
-- Chor-Fiat-Naor, 1994

- **Broadcast** (cont.)
  - Compute session key $SK = s_1 \oplus s_2 \oplus \mathrm{L} \oplus s_w$
  - Construct the cipher block $E_{SK}(M)$ which is the symmetric encryption of the actual message M under session key SK





  - Broadcast enabling and cipher blocks to subscribers

Traitor Tracing Scheme
-- Chor-Fiat-Naor, 1994

- **Decryption** (done by each subscribers)
  - Use personal key $P(j) = \{a_{1,h_1(j)}, a_{2,h_2(j)}, \text{L}, a_{w,h_w(j)}\}$ to recover $s_i = D_{a_i,h_i(j)}(e_{a_i,h_i(j)}) = D_{a_i,h_i(j)}(E_{a_i,h_i(j)}(s_i))$ $(1 \le i \le w)$
  - Computer session key $SK = s_1 \oplus s_2 \oplus \text{L} \oplus s_w$
  - Recover message M : $M = D_{SK}(E_{SK}(M))$

Traitor Tracing Scheme
-- Chor-Fiat-Naor, 1994

- **Fraud**
  - The k traitors get together and combine their personal keys
  - w keys are put together for a pirate decoder which can decrypt every E_i
- **Detection of Traitors**
  - Identify and mark $h_i^{-1}(a_i)$
  - The user with largest number of marks is traitor

Summary & Conclusions

- All currently available cryptographic algorithms, such as encryption, hash functions, and key management, are applicable
- Notes for practical considerations
  - *Broadcasting channel*
  - *Performance acceptance* (*requirements for real time and multimedia data*)
  - *Cost effectiveness* (*esp. for additional devices*)
  - *Legitimate issues or legal systems*