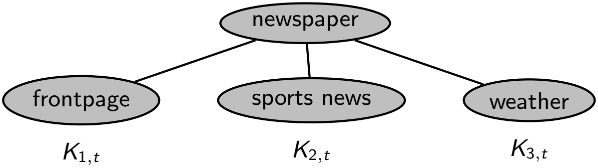
An Efficient Time-bound Hierarchical Key Management Scheme for Secure Broadcasting

Elisa Bertino, Ning Shang, Sam Wagstaff
Purdue University

January 8, 2008

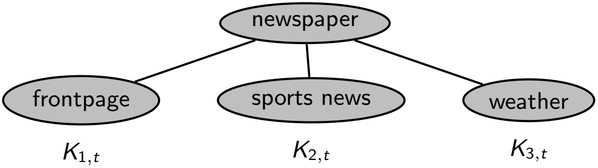
Motivation: Problem to Solve

In e-subscription and pay TV systems, data can be organized and encrypted using symmetric key algorithms according to predefined time periods and privileges, then broadcast to users.



Motivation: Problem to Solve

In e-subscription and pay TV systems, data can be organized and encrypted using symmetric key algorithms according to predefined time periods and privileges, then broadcast to users.



Problem: need an efficient way to manage the encryption keys.

Key Management Scheme Needs to Achieve

Key Management Scheme Needs to Achieve

1. User in a higher class can access keys of its lower classes; but not vice versa

Key Management Scheme Needs to Achieve

1. User in a higher class can access keys of its lower classes; but not vice versa
2. User can access keys/data only in assigned time periods

Key Management Scheme Needs to Achieve

1. User in a higher class can access keys of its lower classes; but not vice versa
2. User can access keys/data only in assigned time periods
3. Low time and space complexity

Key Management Scheme Needs to Achieve

1. User in a higher class can access keys of its lower classes; but not vice versa
2. User can access keys/data only in assigned time periods
3. Low time and space complexity

Schemes (Tzeng, 2002; Chien, 2004) were proposed; they were unable to achieve the above requirements completely.

Components of Key Scheme

1. Intractability of Elliptic Curve Discrete Logarithm Problem (ECDLP)
   In short: given P, $[n]P$, hard to find n.
   Used to achieve hierarchical property.

Components of Key Scheme

1. Intractability of Elliptic Curve Discrete Logarithm Problem (ECDLP)
   In short: given P, $[n]P$, hard to find n.
   Used to achieve hierarchical property.
2. Oneway property of cryptographic hash function
   In short: given hash value (digest), hard to find pre-image.
   Used to achieve time-bound property.

Components of Key Scheme

1. Intractability of Elliptic Curve Discrete Logarithm Problem (ECDLP)
   In short: given P, $[n]P$, hard to find n.
   Used to achieve hierarchical property.
2. Oneway property of cryptographic hash function
   In short: given hash value (digest), hard to find pre-image.
   Used to achieve time-bound property.
3. Tamper-resistant device (e.g. TPM)
   In short: by its name.
   Used to store important sensitive info.

Key Scheme in 4 Phases

- Initialization
- Encrypting Key Generation
- User Subscription
- Decrypting Key Derivation

Initialization Phase

Step 1: hierarchical parameters

Vendor chooses

- Elliptic curve E over a finite field $\mathbb{F}_q$; DLP hard
- Point $Q \in E(\mathbb{F}_q)$ with large prime order p
- $2n$ integers n_i, g_i; $n_i g_i$ all different mod p for $1 \le i \le n$

Vendor computes

- $P_i = [n_i]Q$ on $E(\mathbb{F}_q)$
- h_i such that $g_i h_i \equiv 1 \pmod{p}$
- Class key $K_i = [g_i]P_i$ for class $\mathcal{C}_i$
- Points $R_{i,j} = [g_i]K_j + (-K_i)$, when $\mathcal{C}_j \prec \mathcal{C}_i$.

Initialization Phase

Step 2: temporal parameters

Vendor chooses

- Random integers a, b
- keyed-hash message authentication code (HMAC) $H_K(-)$ built with hash function $H(-)$ and a fixed secret key K

Initialization Phase

Step 2: temporal parameters

Vendor chooses

- Random integers a, b
- keyed-hash message authentication code (HMAC) $H_K(-)$ built with hash function $H(-)$ and a fixed secret key K

Step 3: parameter publication

Vendor publishes $R_{i,j}$ on an authenticated board, whereas the integers g_i, h_i, a and b are kept secret.

Encrypting Key Generation Phase

$[1, Z]$: system's life time
$K_{i,t}$: temporal class key; $t \in [1, Z]$

The generation process for $K_{i,t}$

$$K_{i,t} = H_K\left((K_i)_Y \oplus H^t(a) \oplus H^{Z-t}(b) \oplus ID_i\right)$$

$(K_i)_Y$: y-coordinate of K_i
$H^m(x)$: m-fold iteration of $H(-)$ applied to x
ID_i: identity of $\mathcal{C}_i$
$\oplus$: bitwise XOR

User Subscription Phase

In this phase, a tamper-resistant device storing important information is issued to the subscriber.

Step 1: encryption information

Vendor matches a subscription request with a policy configuration, assigns to class $\mathcal{C}_i$. **Encryption information**

$$EncInf_i = \left\{ \left(H^{t_1}(a), H^{Z-t_2}(b) \right) \right\},$$

defined for all acceptable time intervals $[t_1, t_2]$ defined in the policy.

User Subscription Phase

Step 2: delivery of info

- Vendor distributes the class key K_i to the subscriber through a secure channel
- Vendor issues the subscriber a tamper-resistant device storing H_K (thus H, K), E, $\mathbb{F}_q$, ID_i, h_i and $EncInf_i$.
- Secure clock embedded in the device

Decrypting Key Derivation Phase

In this phase the temporal keys for a class and the classes below it are reconstructed by the tamper-resistant device.

Step 1: user input

To access data for class $\mathcal{C}_j \preceq \mathcal{C}_i$, user in class $\mathcal{C}_i$ inputs into the device

- only K_i, if $\mathcal{C}_j = \mathcal{C}_i$
- $R_{i,j}$, obtained from the authenticated public board; class identity ID_j of $\mathcal{C}_j$; its own secret class key K_i, otherwise if $\mathcal{C}_j \prec \mathcal{C}_i$

Decrypting Key Derivation Phase

Step 2

- If K_j is the only input, the next step is executed directly
- Otherwise, the tamper-resistant device computes the secret class key of $\mathcal{C}_j$:

$$K_j = [h_i](R_{i,j} + K_i).$$

Decrypting Key Derivation Phase

Step 3

If $t \in [t_1, t_2]$ for some acceptable time interval $[t_1, t_2]$ of the access control policy, the tamper-resistant device computes

$$H^t(a) = H^{t-t_1}(H^{t_1}(a)), \qquad H^{Z-t}(b) = H^{t_2-t}(H^{Z-t_2}(b)),$$

and $K_{j,t} = H_K((K_j)_Y \oplus H^t(a) \oplus H^{Z-t}(b) \oplus ID_j)$.

Decrypting Key Derivation Phase

Step 3

If $t \in [t_1, t_2]$ for some acceptable time interval $[t_1, t_2]$ of the access control policy, the tamper-resistant device computes

$$H^t(a) = H^{t-t_1}(H^{t_1}(a)), \qquad H^{Z-t}(b) = H^{t_2-t}(H^{Z-t_2}(b)),$$

and $K_{j,t} = H_K((K_j)_Y \oplus H^t(a) \oplus H^{Z-t}(b) \oplus ID_j)$.

Step 4

At time granule t, the protected data belonging to class $\mathcal{C}_j$ can be decrypted by applying the key $K_{j,t}$.

Achievements of Scheme

- Hierarchical key management supporting time-bound
- Resistant to the collusion attacks that break earlier schemes
- Efficient in terms of time and space requirements
- Class keys can be changed without re-issuing new devices

Thank you!

Questions?