CoverCrypt

1 KEM with Subset cover

Let be given a CPA-secure KEM scheme defined by the 3 algorithms, $\mathsf{KEM.KeyGen}$, $\mathsf{KEM.Encaps}$ and $\mathsf{KEM.Decaps}$, the broadcast encryption scheme will be defined as follows :

- Setup : $\lambda \rightarrow (\mathsf{msk}, \mathsf{mpk})$
  takes the security parameter. It first defines the partition of subsets S_i that covers the set S with respect to the target users' rights.
  And for each S_i, it invokes ($\mathsf{KEM.KeyGen}$ which outputs $(\mathsf{pk}_i, \mathsf{sk}_i)$. It defines $\mathsf{mpk} = (\mathsf{pk}_i)_i$ and $\mathsf{msk} = (\mathsf{sk}_i)_i$ the master public key and master secret key.
- Join : $(\mathsf{msk}, U) \rightarrow \mathsf{sk}_U$
  For a user U, defines sk_U as the set of secret keys sk_i for each i such that $U \in S_i$,
- Encaps : $(\mathsf{mpk}, T) \rightarrow C = (K, C_i = (K_i \oplus K, E_i)_{i \in A})$
  takes as input mpk and target set T of rights, definied as the union of subsets S_i. It first samples a random key K and expresses T as a set of covering subsets, i.e $T = \cup_{i \in A} S_i$.
  Then for each $i \in A$, it invokes $\mathsf{KEM.Encaps}$ which $C_i = (K_i, E_i)_{i \in A}$. It finally returns $(K, C = (K_i \oplus K, E_i)_{i \in A})$.
- Decaps: $(\mathsf{sk}_U, C) \rightarrow K$
  Let $R = \cup_{j \in B} S_j$ such that the secret key $\mathsf{sk}_U = \{\mathsf{sk}_j\}_{j \in B}$ and let T the target set associated to C.

  If there exists an index $j \in B$ such that $S_j \subseteq T$, it invokes $\mathsf{KEM.Decaps}(\mathsf{sk}_j, E_j)$ which gives K_j. Then using the corresponding ciphertext C_j parsed as K'_j, E_j, it obtains the session key as $K = K'_j \oplus K_j$.

2 Examples

The Setup phase first partitions the sets of rights as a union of subsets S_i so that:

- A right with FN and security level LW is associated with set S_1. A user joining the system with these rights obtains $(\mathsf{sk}_1, \mathsf{pk}_1)$.
- A right with FN and security level LW is associated with set $S_2 \cup S_1$. A user joining the system with these rights obtains $(\mathsf{sk}_1, \mathsf{pk}_1)$ and $(\mathsf{sk}_2, \mathsf{pk}_2)$.
- A right with FN and security level LW is associated with set $S_3 \cup S_2 \cup S_1$. A user joining the system with these rights obtains $(\mathsf{sk}_1, \mathsf{pk}_1)$, $(\mathsf{sk}_2, \mathsf{pk}_2)$ and $(\mathsf{sk}_3, \mathsf{pk}_3)$.

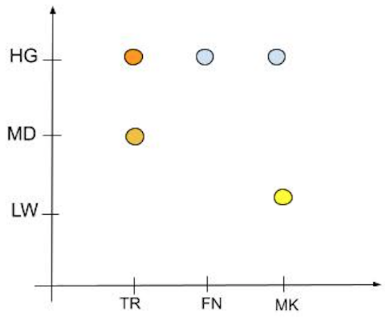

Figure 1: Hierarchical policies where domains are in abiscissa: TR (treasury), FN (finance), MK (market); and security level in increase order: HG (high), MD (medium) and LW (low).

3 Updates

A new user joining the system will receive secret keys associated to the rights he has; these rights have possibly evovled and the policy can be enriched over time.

A first option would be to add timestamps to the policy so that the description will be defined in a three-dimensional space of ”attributes”.
A new user in the system will be given secret keys associated to a given time period. In such a case, dummy keys won’t be useful anymore.

If any secret keys becomes dummy, but the policy remains unchanged, then a new value is generated for the dummy secrets key.