A Note on Broadcast Encryption Key Management with Applications to Large Scale Emergency Alert Systems

Guoqiang Shu[1], David Lee[1] and Mihalis Yannakakis[2]

[1]The Ohio State University
Dept. of Computer Science and Engineering
Columbus, OH 43210 USA
{shug,lee}@cse.ohio-state.edu

[2]Columbia University
Dept. of Computer Science
New York, NY 10027 USA
mihalis@cs.columbia.edu

Abstract

Emergency alerting capability is crucial for the prompt response to natural disasters and terrorist attacks. The emerging network infrastructure and secure broadcast techniques enable prompt and secure delivery of emergency notification messages. With the ubiquitous deployment of alert systems, scalability and heterogeneity pose new challenges for the design of secure broadcast schemes. In this paper we discuss the key generation problem with the goal of minimizing the total number of keys which need to be generated by the alert center and distributed to the users. Two encryption schemes, zero message scheme and extended header scheme, are modeled formally. For both schemes we show the equivalence of the general optimal key generation (OKG) problem and the bipartite clique cover (BCC) problem, and show that OKG problem is NP-Hard. The result is then generalized to the case with resource constraints, and we provide a heuristic algorithm for solving the restricted BCC (and OKG) problem.

Keywords: *Broadcast Cryptography, Key Management, Emergency Alert, Security, Bipartite Clique Cover*

1 Motivation

Emergency alert system, in its many different forms, has been one of the most crucial social infrastructures for more than centuries. Recent years have witnessed an exponentially increasing demand for nation wide effective emergency alert system that allows us to quickly respond to the terrorist threats. Most modern alert systems use preexisting mature communication infrastructure to broadcast alert messages. Moreover, it is very common and effective that an alert message is dispatched simultaneously through different networks. Since the alerting message is transmitted through shared media, the major goal of building the alert system is to ensure confidential and authenticated message delivery. On one hand, multiple alert systems may coexist in the same infrastructure and their alert messages are usually sensitive, therefore it is important that only intended receivers can read the content of the message. On the other hand, since fraud of alert message will bring disastrous impact, it is also desirable that the receiver is capable of authenticating the message source. Cryptographic broadcast encryption technique enables secure message delivery by protecting the message with a set of secrets (keys) in such a way that only a selected subset of receivers is capable of recovering the original content from the broadcast message. There have been substantial research results [22, 9, 7, 17] about broadcast encryption schemes. While the existing broadcast encryption schemes have been proved efficient in many areas such as pay TV program and Internet software distribution,the application to nation-wide emergency alert system poses many new challenges.

Key generation and management are becoming critical, expensive and sometimes the bottleneck for a super large scale alert system. Generating, maintaining and delivering keys for a large user group is time consuming. Therefore it is natural to require the number of keys that we have to generate to be minimal. Many existing solutions optimize the number of keys stored by each receiver [17, 5], but the total number of keys existing in broadcast system is still in proportion to the

[1]This work was supported in part by the U.S. National Science Foundation (NSF) under grant awards CNS-0403342 and CNS-0548403. [2]This work was supported in part by the U.S. National Science Foundation (NSF) under grant awards CCF-0430946.

1-4244-0054-6/06/$20.00

number of users. This derives from the requirement that any subset of users could potentially be the legitimate user group. Indeed, one can prove that to achieve such full flexibility, the number of keys used is at least equal to the number of users. A fact of typical emergency alert system that we might take advantage of is the way legitimate receiver groups are formed. The nature of alert system implies the subset of intended users for any alert message is solely predetermined and maintained relatively stable by the administrator, and normally users do not have the choice of arbitrarily joining and leaving a group. Under this circumstance, it is attractive to explore the possibility of minimizing the total number of keys that the administrator has to generate.

Another interesting characteristic for distributed alert system is its heterogeneity, which is reflected in at least two dimensions. First, the intended user group size varies largely for different alert messages. Thus the assumption made by some existing methods that only a small portion of all users will be excluded for a message becomes inappropriate here. For example, in the classic Set Difference (SD) scheme[17], the transmission overhead of a message grows linearly with the number of excluded users, which is not practical for high confidential alert messages only targeting few receivers. Second, the communication infrastructures may have different constraints on the broadcast encryption scheme. For example, in a cellular phone network both communication bandwidth and computation power is low so the message length and number of encryption should be limited; similarly if we use terminals with small amount of memory, then the number of keys required to store should be limited. Ideally, the broadcast scheme could be easily customized with regard to those constraints.

In this paper we primarily investigate the first challenge of key generation while also taking the system heterogeneity into account. We present our theoretical result on minimizing the total number of keys needed for an alert system. We first formally model a typical broadcast system and two broadcast encryption paradigms: zero message scheme and extended header scheme, both using multiple keys to protect a message. We define first a general optimal key generation problem (without constraints) and prove the NP-Hardness of the problem for both schemes. The bipartite graph representation of a broadcast system is utilized and we establish the equivalence of the key generation problem and bipartite clique cover(BCC) problem. We also define a variant of BCC problem (Half Bounded BCC) that corresponds to the key generation problem under one type of resource constraint. A general heuristic algorithm is presented to produce approximate optimal key generation and distribution scheme.

2 System Model

The broadcast scenario in general consists of a message source (sender) and a set of users (receivers) to which the message is to be disseminated. From the networking perspective, the source could be a special node in the network and the users are all or a subset of the remaining nodes. Each message has a legitimate receiver set, representing the subset of all users to whom the message is intended to deliver. We do not distinguish users associated with the exact same set of messages, and use the term "user group" to represent such a collection. Similarly, two messages with the same receiver set could be treated as the same, since they will be protected in the exact same way.

Formally speaking, the entities involved in the scenario are

- A single source of message: Src
- A set of alert messages: $M = \{M_i, i = 1, ..., m\}$
- A set of user group or receiver group : $U = \{U_j, j = 1, ..., n\}$, where each U_j is a set of users.

Each U_i represents a user group inside which all users share the same messages. A receiver function is defined as $Rcv : M \rightarrow 2^U$. With each message M_i, a set of user groups $Rcv(M_i) \subseteq U$ is associated.

$$Rcv(M_i) = \{U_{i^1}, U_{i^2}, ..., U_{i^k}\}, 0 \leq k \leq n$$

$Rcv(M_i)$ is the set of user groups that are authorized to receive the message M_i. Two messages M_i, M_j are regarded as different if and only if $Rcv(M_i) \neq Rcv(M_j)$.

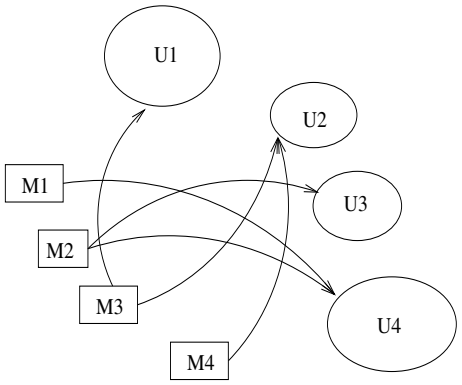

Figure 1. Messages and User Groups

Figure 1 shows one such broadcast scenario which consists of four messages $\{M_1, M_2, M_3, M_4\}$, and four user groups $\{U_1, U_2, U_3, U_4\}$. Message M_1 is to be received only by user group U_4, so $Rcv(M_1) = \{U_4\}$.

Similarly $Rcv(M_2) = \{U_3, U_4\}$, $Rcv(M_3) = \{U_1, U_2\}$ and $Rcv(M_4) = \{U_2\}$. Note that because of the way we define user group, any two different user groups U_i and U_j are disjoint, i.e $\forall_{i,j} :: U_i \cap U_j = \phi$, therefore the number of message types is bounded by the total number of possible subsets of the user set U. Thus the maximum cardinality of the set M is $2^{|U|}$.

We consider the secure broadcast in the above scenario using cryptographic approach, and we focus primarily on the requirement of confidentiality. A set of keys will be generated and distributed to the user groups. Each message M_i from the source is encrypted using some keys in such a way that all and only users in $Rcv(M_i)$ are capable of obtaining the original message using their keys. The encrypted message is broadcast to all users. Although it is technically feasible to encrypt the whole message M_i with many keys sequentially (in the form of $E_{K_{i^1}}(E_{K_{i^2}}(\cdots E_{K_{i^j}}(M_i))))$, it is not practical when its size is large (e.g. multimedia content). A commonly adapted scheme is to use a freshly generated session key. The message is encrypted by the session key, and the session key is encrypted using the distributed user keys and attached to the message. Therefore the expensive operation on huge amount of data only needs to be done once. We will follow this scheme and do not count session keys in total number of keys used since session keys do not need to be distributed. Furthermore, we assume public key cryptography is used although the theoretical result derived by this paper will not depend on the choice of encryption scheme. For a generated key pair $K_i = (K_i^u, K_i^r)$ with an identifier, the public key (K_i^u) is used for the sender and the private key (K_i^r) is distributed to the users. For sake of simplicity, we use K_i to represent the key pair. A **user key distribution** is defined as a function $f_{KU} : U \rightarrow 2^K$, where K is set of key pairs. Similarly a **message key distribution** based on the same set of keys is defined as $f_{KM} : M \rightarrow 2^K$.

Given a message key distribution(i.e. keys used for each message), a message M_i could be encoded and broadcast to all users following either of the following two encryption schemes.

Zero Message Scheme. In zero message scheme, the message M_i is encrypted by the session key, and the message header is a cascading encryption of session key with all the keys in $f_{KM}(M_i)$. The length of message header is small and constant. Formally, the sender broadcasts to all the following message:

$$[MSGID||E_{K_s}(M_i)||E_{K_{i^1}}(E_{K_{i^2}}(\cdots E_{K_{i^j}}(K_s)))]$$

where K_s is the session key and $K_{i^1}, K_{i^2} \cdots K_{i^j} \in f_{KM}(M_i)$. $MSGID$ is a preamble of the message that contains the public identifier of each key used. We use $E_k()$ and $D_k()$ to denote encryption and decryption with a key k respectively, we use "||" to represent concatenation of two messages. A user upon receiving the key, reads $MSGID$ and decides the keys (and their order) needed to decrypt it, and recovers the original message if and only if it possesses all the keys in $f_{KM}(M_i)$. In order for all legitimate users in $Rcv(M_i)$ to recover the message and others not capable of doing so, the key distribution scheme must satisfy

$$f_{KM}(M_j) \subseteq f_{KU}(U_i) \Leftrightarrow U_i \in Rcv(M_j) \qquad (1)$$

That is, a user has all keys required to decrypt one message if and only if it is in the receiver set of the message. This scheme is also referred to as "AND" scheme.

Extended Header Scheme. In extended header scheme the message M_i is encrypted by the session key, the message header contains a sequence of encrypted session key (called enabling blocks), using the keys in $f_{KM}(M_i)$. The length of message header is larger and different for each message. Formally, the sender broadcasts to all the following message:

$$[MSGID||E_{K_s}(M_i)||E_{K_{i^1}}(K_s)||E_{K_{i^2}}(K_s)||\cdots||E_{K_{i^j}}(K_s)]$$

A user can recover the original message if and only if it possesses one of the keys in $f_{KM}(M_i)$ (the user can deduce from $MSGID$ the position of each enabling block). For this scheme the key distribution scheme must satisfy

$$f_{KM}(M_j) \cap f_{KU}(U_i) \neq \phi \Leftrightarrow U_i \in Rcv(M_j) \qquad (2)$$

That is, a user has at least one key required to decrypt the message if and only if it is in the receiver set of the message. This scheme is also referred to as "OR" scheme. While the previous scheme uses multiple encryption to exclude invalid users, this scheme uses a symmetric form to include valid users.

	Msg/Usr	Zero Message	Ext. Header
f_{KM}	M_1	K_1,K_2	K_1
	M_2	K_1	K_3
	M_3	K_3	K_2,K_4
	M_4	K_2,K_3	K_4
f_{KU}	U_1	K_3	K_2
	U_2	K_2,K_3	K_4
	U_3	K_1	K_3
	U_4	K_1,K_2	K_1,K_3

Table 1. Message and User Key Distribution

For the model in Figure 1, Table 1 shows the optimal solution of key distribution for both schemes. Particularly, in zero message scheme we only need 3 keys

although the number of both user and message is 4. In extended header scheme 4 keys are needed.

The adaption and implementation of a secure broadcast scheme is guided and sometimes restricted by the resource constraints of the target distributed infrastructure. As discussed in the previous section, in a heterogeneous alerting system, we might be concerned with many characteristics of the broadcast encryption and key distribution method. Below we list some of the important issues.

Total number of keys ($|K|$). This is a primary concern of this paper. We want to minimize the number of keys we have to generate and distribute while retaining the security properties. A obvious simple scheme could have $|K| = m$, i.e each message has one special key. However, in case of large scale emergency alert system, this might not satisfactory. We show the possibility and hardness of constructing schemes where $|K| < min(m, n)$.

Number of keys for each message ($max_{1\leq i\leq m}|f_{KM}(M_i)|$). In zero message scheme, this is the number of decryption operations done for a message by the receiver (a computation resource constraint). In extended header scheme, this number decides the length of message header (communication overhead constraint).

Number of keys for each user ($max_{1\leq j\leq n}|f_{KU}(U_j)|$). The number of keys distributed to each user is a constraint when the device of the receiver has limited storage.

Since the main goal of this work is to optimize total number of keys needed, we will present a result without any other constraints listed above (i.e. the only requirement is data confidentiality). Then we consider the affect of adding those limitations on the overall goal of minimizing total number of keys.

3 Optimal Key Generation Problem

In this section we address the problem of minimizing number of keys the sender generates. We have shown in Table 1 that if one message (actually the session key) is encrypted using more than one key, then the total number of keys could be less than number of messages and users. Here the question to ask is what the minimum number of keys needed is, and how to construct such a key distribution scheme. Formally speaking, given a system model $< M, U, Rcv >$, the optimal key generation scheme is a set of keys K with minimum cardinality, and functions f_{KM} and f_{KU} defined on K such that for zero message scheme (1) is satisfied or for extended header scheme (2) is satisfied. As we will see the construction of K suggests the construction of f_{KM} and f_{KU}, hence we shall focus on calculating the minimum number of keys required. We present the main result of this paper: Optimal Key Generation (OKG) problem for both encryption schemes is NP-Hard.

3.1 Bipartite Clique Coverage Problem and Bipartite Graph Representation

We prove this result by reducing Bipartite Clique Cover(BCC) problem to OKG. In fact we show that OKG is equivalent to BCC problem, in the sense that reductions go both ways and there is 1-1 correspondence between instances and solutions (hence the reduction preserves optimality and approximation).

Definition 1. Bipartite clique cover (BCC) problem*: Given a bipartite graph $G =< V, W, E >$, where V and W are the set of vertices and E is the set of edges $E \subseteq V \times W$, find the minimum number of complete bipartite subgraphs (a.k.a. bicliques) such that every edge in E is included in at least one of these subgraphs.*

First, it is straightforward to represent a broadcast system model by a bipartite graph. We define two representations of a model.

Definition 2. *The bipartite graph representation of $I =< M, U, Rcv >$ is defined as a bipartite graph $G =< V, W, E >$, where $V = M$ and $W = U$, $E = \{(M_i, U_j)|U_j \in Rcv(M_i)\}$.*

Definition 3. *The complement bipartite graph representation of $I =< M, U, Rcv >$ is defined as a bipartite graph $G =< V, W, E >$, where $V = M$ and $W = U$, $E = \{(M_i, U_j)|U_j \notin Rcv(M_i)\}$.*

The two graph representations are disjoint and complementary. Furthermore, either of these representation relationships defines an one-to-one mapping between system models and bipartite graphs.

3.2 General OKG Problem

We assume there is not any constraint on the key distribution scheme. Based on definitions in the previous section, we provide a result for zero message scheme first. The following theorem establishes the correspondence relationship between the OKG problem (decision form) and the BCC problem.

Theorem 1. *Optimal key generation problem for zero message scheme is NP-Hard.*

Proof. We prove the theorem by showing $BCC \leq_p OKG$. Given an instance of BCC $G =< V, W, E >$, $V_1 = V_1, \cdots, V_m$, $W = W_1, \cdots, W_n$, construct a unique instance of $I =< M, U, Rcv >$ such that G is the

complement bipartite graph representation of I. Then we show the minimum number of keys required in I is equal to the minimum number of complete bipartite subgraphs needed to cover E in G.

Suppose first that we have a cover with r bipartite cliques: $C_1, ..., C_r$ where $C_t =< V^t, W^t, E^t = V^t \times W^t >$, $V^t \subseteq V$, $W^t \subseteq W$,$t = 1, ..., r$. We show that r keys suffice for a zero message scheme. First generate a set K of r keys (one for each bipartite clique), and define f_{KM} and f_{KU} as follows. For each message M_i, let $f_{KM}(M_i) = \{K_t | V_i \in V^t\}$ and for each receiver U_j set $f_{KU}(U_j) = \{K_t | W_j \notin W^t\}$; that is a key K_t is used in the encoding of the message M_i *iff* the corresponding bipartite subgraph C_t includes the corresponding node i of V^t, and K_t is given to receiver U_j *iff* C_t does not include the corresponding node j of W^t. In figure 2, this process is applied on the complement bipartite graph representation of the model in Figure 1. C_1 is one of the cliques, and the corresponding key K_1 is assigned to $\{M_1, M_2\}$ and $\{U_3, U_4\}$.

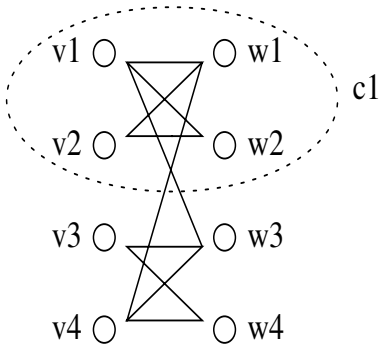

Figure 2. Construction of f_{KM} and f_{KU}

Now it is straightforward to prove that this scheme satisfies (1). For any user $U_j \in Rcv(M_i)$, let $K_t \in f_{KM}(M_i)$. Since $(V_i, W_j) \notin E$, any bipartite clique including V_i must not include W_j (otherwise it can not be a bipartite clique). Therefore C_t must not include W_j. By the way we construct f_{KU} we know $K_t \in f_{KU}(U_j)$. Because we choose K_t arbitrarily, U_j can decode M_i. On the other hand, for any user $U_j \notin Rcv(M_i)$, $(V_i, W_j) \in E$. This edge must be covered by one of the bipartite cliques C_t, and by the similar argument as above we have $K_t \in f_{KM}(M_i)$, but $K_t \notin f_{KU}(U_j)$.

Conversely, if we can solve the key generation problem with r keys, then we can cover the graph with r complete bipartite subgraphs $C_t = (V^t, W^t, E^t), t = 1, ..., r$. Let the keys be $K_1, ..., K_r$, the broadcast scheme and key distribution scheme be f_{KM} and f_{KU} respectively, we define $V^t = \{V_i | K_t \in f_{KM}(M_i)\}$ and $W^t = \{W_j | K_t \notin f_{KU}(U_j)\}$, and E^t to be the edges induced by them. Now we need to show that all bipartite graphs are complete and cover all the edges. If a subgraph C_t is not complete, then there is a node $i \in V^t$ and a node $j \in W^t$ such that $(V_i, W_j) \notin E$. This means U_j is in $Rcv(M_i)$, but at the same time U_j does not have K_t needed to decrypt M_i. This is a contradiction because the scheme should satisfy (1). On the other hand, if there is one edge (V_i, W_j) not covered by any of the cliques. Then $U_j \notin Rcv(M_i)$. However, for each $K_t \in f_{KM}(M_i)$, C_t must not cover W_j, i.e $W_j \notin W^t$. Therefore $K_t \in f_{KU}(U_j)$. Now we have a contradiction again because now U_j has all the keys needed to decrypt M_i. To complete this proof, it is obvious that the reduction could be done in polynomial time. □

We can use a similar approach to prove the following result for extended header scheme. The difference will be that we use the bipartite graph representation instead of the complement bipartite graph representation. Then we can again show the minimum number of keys used in the broadcast scheme is equal to minimum number of cliques that cover the graph. Due to space limitation, we omit the details of proof.

Theorem 2. *Optimal key generation problem for extended header scheme is NP-Hard.*

The bipartite clique cover problem is known to be NP-hard [12], and furthermore it is hard to approximate within any constant or any polynomial like n^c where c is a constant [16]. The variants of this problem are also studied. [2] discovers a subclass of bipartite graph for which BCC could be solved in polynomial time (unfortunately the subclass is not suitable for our broadcast model). Also [6] presents the result on bipartite cliques with restricted degree. While in general we prove OKG problem is hard, any heuristic for BCC could be used to build approximate key distribution solution. For example, it follows immediately from the proof that m and n are both upper bounds on the number of keys. A tighter upper bound is the vertex cover solution, which for bipartite graph is solvable in polynomial time. Also note that this proof in fact suggests the construction of message key distribution and user key distribution. Once one identifies the clique cover from bipartite or complement bipartite representation of a broadcast model, it is trivial to construct f_{KM} and f_{KU}. The last remark is the extreme case where we have 2^n messages, i.e. one for each subset of users, then it can be proved that for both schemes we need at least n keys.

3.3 Restricted OKG Problem

Having shown the complexity of the general problem, we now consider a slightly more complicated

model where we have constraints. In our definition of zero message scheme, the number of keys assigned to a message should be restricted if the receiver has computation resource constraint. For the same reason, in extended header scheme the number of keys assigned to a user should be restricted if the receiver has storage constraint. Formally, we define one type of restricted OKG problem as follows. Given a system model $< M, U, Rcv >$ and an upper bound b of the number of keys for each user, the optimal key generation scheme is $< K, f_{KM}, f_{KU} >$ that satisfies the requirement of original OKG problem, and in addition $\forall i : |f_{KU}(U_i)| \leq b$. Similarly we could define restricted OKG problem with upper bound of number of keys for each message. It is important to note that the main goal of optimization is still the total number of keys.

It is instructive to consider this restricted model together with its (complement) bipartite graph representation. For zero message scheme, from the proof of Theorem 1 we see that the cardinality of set $f_{KM}(M_i)$ is equal to the number of cliques that M_i is associated with. This observation sets up a correspondence between the restricted OKG problem and the following derivative of BCC problem.

Definition 4. Half Bounded Bipartite Clique Cover (HB-BCC) Problem*: Given a bipartite graph $G =< V, W, E >$ and an integer b, find the minimum number of complete bipartite subgraphs such that every edge in E is included in at least one of these subgraphs, and each node in V is included in at most b different subgraphs.*

Clearly for any $b > 0$ the solution to HB-BCC problem always exists since any bipartite graph could be covered by $|V|$ cliques and each node in V is only associated with one clique. The original BCC problem could be reduced to HB-BCC problem by a trivial mapping. Therefore, HB-BCC problem is also NP-Hard. Note here b is a parameter of the problem. If we fix b as a constant, the problem might become easy. The investigation of such problem is included in our future work. Another issue is to bound nodes in both sides of the graph. In this case, an edge coverage may not exist. The corresponding scenario in broadcast encryption model is that in extended header scheme we want to limit both number of keys per user and the message header length.

4 Heuristic Key Generation and Distribution Scheme

In this section we investigate the approximate solution for optimal key generation problem. First we present a polynomial time general heuristic algorithm for calculating optimal bipartite clique coverage, and then it is adapted to fit the bounded problem. The heuristic procedure contains two steps. Given a bipartite graph $G =< V, W, E >$ with $|V| = m, |W| = n$ and assume $m > n$, we first identify a set of no more than $O(m)$ cliques that altogether form a full coverage of E. This step could be probabilistic and could be implemented using many algorithms. The only guideline is that no one should be the subgraph of another, and all cliques should be expanded maximally. In the second step, we run a greedy algorithm that repeats selecting the "best" candidate and adding it to the solution until we get a full coverage. The algorithm is briefly described as follows.

Algorithm 1 General Heuristic for HB-BCC

INPUT: Bipartite Graph $G =< V, W, E >$, Integer b
OUTPUT: A Set S of Cliques

$S_0 = \phi$, $H = \phi$
{STEP 1: Find a set of cliques}
for $i = 1$ to $m + n$ **do**
  Pick a node alternatively from V and W ($i.e. v_1, w_1, v_2, w_2 \cdots$). Suppose we pick v_k
  if $neighbor(v_k) \cap H = \phi$ **then**
    Create a clique $< \{v_k\}, \phi >$ and add it to S_0
  else
    Insert v_k to each clique in S_0 if applicable
    Create r cliques to cover all the rest nodes in $neighbor(v_k) \cap H$ (r is a small constant)
  end if
  $H = H \cup \{v_k\}$
end for
Expand each clique in S_0
$S = S \cup < nb(w_j), \{w_j\} >, j = 1 \cdots m$
Remove subcliques from S_0
{STEP 2: Greedy selection of cliques}
$S = \phi$
for $i = 1$ to n **do**
  $deg[i] = b$
end for
repeat
  for all clique $C \in S_0$ **do**
    $C' = C - \{w_i|$ all edges with w_i in C are covered$\}$
    Calculate a metric $f(C')$
  end for
  Find the clique C'_{max} with largest metric value
  $S = S \cup \{C'_{max}\}$, $S_0 = S_0 - \{C'_{max}\}$
  for all $v_i, w_j \in C'_{max}$ **do**
    $deg[w_i] = deg[w_i] - 1$
  end for
until S covers all edges in G

In the first step our algorithm to select the first set

of cliques is inspired by the heuristic of Clique Cover Problem proposed by [13]. The main idea is to add one node in each round (pick from V and W alternatively), and update the current set of clique to incorporate the new node such that all edges between it and the existing nodes are covered. The algorithm guarantees in each round only constant number of new cliques need to be inserted, therefore the total number of cliques is bounded by $O(m)$. Then all the cliques are expanded if possible, and we remove cliques (Line 13) that are subgraph of another clique (Line 15).

We have to define for the second step the metric function for a clique $f(C)$. For general BCC problem, we simply use the number of uncovered edges in C as $f(C)$ to measure the priority of adding C to the solution. The case for HB-BCC problem is more complicated. We maintain for each node in W the number of selected cliques associated with it. For a clique C, we first remove all the nodes in it that do not contribute to covering new edges, so that if we selection C, no node will be included unnecessarily. Furthermore if there is one node w with $deg[w] = 1$, and C does not include all the uncovered edges with w, then C should not be selected because otherwise the solution will not satisfy the restriction. On the other hand, if we could select C, then we simply set $f(C)$ as in the general BCC problem. In order to guarantee that all edges will be eventually covered therefore the algorithm terminates, we include m stars (Line 14, one for each node in W) to the set of cliques.

To analyze the complexity of this algorithm let us assume the clique is implemented using set with perfect hashing. Step 1 of the algorithm requires $m + n$ steps. In each step trying to insert the new node costs $O(m^2)$ time, since there are $O(m)$ existing cliques and testing subset relationship costs $O(m)$. Creating r new cliques requires $O(rm)$. So the total cost of step 1 is $O(m^3)$ and we end up with $O(m)$ cliques. Expanding each clique costs $O(m^2)$, and finding all subcliques costs $O(m^3)$. Finally, step 2 requires at most $m \times b$ steps. in each step calculating all the metrics needs $O(m^2)$. Therefore, the total cost of the heuristic algorithm is $O(bm^3)$.

5 Related Work

The problem of secure broadcast of messages has been studied extensively in the past in varied contexts. The first works on broadcast confidentiality include [3, 7]. The well known schemes that try to optimize number of keys stored in each user and transmission overhead include the SD method proposed in [17] and many subsequent improvements. [9] and [4] consider resilience against user collusion and propose elegant techniques to achieve k-resilience using 1-resilience scheme as building block. The authors establish quantitative result on the number of keys and the length of messages. Another influential result on the bound and tradeoff of number of keys and messages is [15]. The authors apply set theory to the broadcast model and they provide a sufficient and necessary condition under which using K keys total is feasible. Also they derive lower bound of number of keys hold by each user, assuming bound of the number of excluded users. Garay et al address resilience from another perspective [11], and bring the long-lived broadcast encryption method that identifies colluded "super user" and discard the compromised keys. Some other work on single source secure broadcast include [8] [10] [14] [20]. The major difference between our approach and all work described above lies in the fact that we exploit multiple encryption for each message, making the total number of keys in the system possibly less than both the number of user and number of message. [19] provides a detailed study of many issues besides data confidentiality in wired and wireless networks. Here focus is on the message authentication and key distribution in case of dynamic membership update. The basic authentication mechanism is the TESLA protocol [18], which is based on the non-disclosure of one way chain function. Efficient key distribution schemes taking into account dynamic join and leave events are described in [1], which applies a central key server and [21] which uses a logical key tree (LKH) structure to decrease the transmission required.

6 Conclusion and Future Work

The main contribution of this work is that we address the optimization problem for total number of keys in the multi-message broadcast system. Based on a formal definition of broadcast model and two encryption schemes, we prove the NP-hardness of general optimal key generation problem. Then resource constraints in a practical alert system are considered and we define a generalized model and present a heuristic algorithm.

Our focus in this paper is the static configuration of alert messages, while to make the research complete we will have to in our future work extend the model to include dynamic message and user group update and we shall analyze the problem of maintaining minimal number of keys upon the event of adding or removing messages/users. Furthermore, so far in the restricted OKG problem we only consider one restriction. It is interesting and more challenging to see how the trade off between two parameters interferes with the minimization of total number of keys.

References

[1] Multicast security (MSEC) working group within the Internet Engineering Task Force (IETF). `http://www.ietf.org/html.charters/msec-charter.html`, 2002.

[2] J. Amilhastre, P. Janssen, and M.-C. Vilarem. Computing a minimum biclique cover is polynomial for bipartite domino-free graphs. In *SODA '97*, pages 36–42, Philadelphia, PA, USA, 1997. Society for Industrial and Applied Mathematics.

[3] S. Berkovits. How to broadcast a secret. In *Advances in Cryptology - EuroCrypt '91*, pages 535–541, Berlin, 1991. Springer-Verlag. Lecture Notes in Computer Science Volume 547.

[4] C. Blundo and A. Cresti. Space requirements for broadcast encryption. In *EUROCRYPT*, pages 287–298, 1994.

[5] D. Boneh and B. Waters. Collusion resistant broadcast encryption with short ciphertexts and private keys (in submission). `http://citeseer.ist.psu.edu/article/boneh05collusion.html`.

[6] J. Chen and I. A. Kanj. Constrained minimum vertex cover in bipartite graphs: complexity and parameterized algorithms. *J. Comput. Syst. Sci.*, 67(4):833–847, 2003.

[7] G.-H. Chiou and W.-T. Chen. Secure broadcasting using the secure lock. *IEEE Trans. Softw. Eng.*, 15(8):929–934, 1989.

[8] Y. Desmedt, Y. Frankel, and M. Yung. Multi-receiver/multi-sender network security: efficient authenticated multicast/feedback. In *IEEE INFOCOM*, pages 2045–2054, Los Alamitos, CA, USA, 1992. IEEE Computer Society Press.

[9] A. Fiat and M. Naor. Broadcast encryption. In D. R. Stinson, editor, *Advances in Cryptology - Crypto '93*, pages 480–491, Berlin, 1993. Springer-Verlag. Lecture Notes in Computer Science Volume 773.

[10] H. Fujii, W. Kachen, and K. Kurosawa. Combinatorial bounds and design of broadcast authentication. *TIEICE: IEICE Transactions on Communications/Electronics/Information and Systems*, 1996.

[11] J. A. Garay, J. Staddon, and A. Wool. Long-lived broadcast encryption. In *CRYPTO '00*, pages 333–352, London, UK, 2000. Springer-Verlag.

[12] M. R. Garey and D. S. Johnson. *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA, 1990.

[13] L. T. Kou, L. J. Stockmeyer, and C. K. Wong. Covering edges by cliques with regard to keyword conflicts and intersection graphs. *Commun. ACM*, 21(2):135–139, 1978.

[14] K. Kurosawa and S. Obana. Characterisation of (k, n) multi-receiver authentication. In *ACISP '97: Proceedings of the Second Australasian Conference on Information Security and Privacy*, pages 204–215, London, UK, 1997. Springer-Verlag.

[15] M. Luby and J. Staddon. Combinatorial bounds for broadcast encryption. In *EUROCRYPT*, pages 512–526, 1998.

[16] C. Lund and M. Yannakakis. On the hardness of approximating minimization problems. In *STOC' 93*, pages 286–293, New York, NY, USA, 1993. ACM Press.

[17] D. Naor, M. Naor, and J. B. Lotspiech. Revocation and tracing schemes for stateless receivers. In *CRYPTO '01*, pages 41–62, London, UK, 2001. Springer-Verlag.

[18] A. Perrig, R. Canetti, J. D. Tygar, and D. Song. The TESLA broadcast authentication protocol. *RSA CryptoBytes*, 5(Summer):2–13, 2002.

[19] A. Perrig and J. D. Tygar. *Secure Broadcast Communication in Wired and Wireless Networks*. Kluwer Academic Publishers, Norwell, MA, USA, 2002.

[20] R. Safavi-Naini and H. Wang. New results on multi-receiver authentication codes. *Lecture Notes in Computer Science*, 1403, 1998.

[21] D. Wallner, E. Harder, and R. Agree. Key management for multicast: Issues and architectures. Intenet Request for Comment, RFC 2627, Internet Engineering Task Force, 1999.

[22] A. Wool. Key management for encrypted broadcast. *ACM Trans. Inf. Syst. Secur.*, 3(2):107–134, 2000.