Non-constructive Existence Results (summary)

- define
$$\rho = \frac{t}{t+w}$$
and
$$p_{t,w} = 1 - \frac{t^t w^w}{(t+w)^{t+w}}$$
- recall that $\mathsf{Exp}[X] < n^{t+w}(p_{t,w})^v$
- if
$$v > \frac{(t+w)\log_2 n}{-\log_2(p_{t,w})}$$
then $\mathsf{Exp}[X] < 1$, and hence a (t,w)-CFF(v,n) exists
- this is an existence proof, so we do not have an explicit formula or an efficient deterministic algorithm to construct the CFF
- we will consider two possible approaches to making this method practical:

Error Probabilities

1. generate a random M (as described in the existence proof), and then check to see if it is a CFF
2. choose v such that a random M will be a CFF with high probability, and don't check it

- what is the probability that a random M is not a CFF?
- we have that

$$\begin{aligned} \mathsf{Exp}[X] &= \sum_{i=0}^{\infty} (\mathsf{Pr}[X = i] \times i) \\ &\geq \sum_{i=1}^{\infty} \mathsf{Pr}[X = i] \\ &= \mathsf{Pr}[X > 0] \\ &= \mathsf{Pr}[M \text{ is not a CFF}] \end{aligned}$$

A Practical Algorithm?

- let $s \geq 1$ (s is an error probability parameter)
- it is easy to determine v so that $\mathsf{Exp}[X] \leq 2^{-s}$
- if
$$v > \frac{(t+w)\log_2 n + s}{-\log_2(p_{t,w})}$$
then $\mathsf{Exp}[X] < 2^{-s}$
- then a randomly constructed M will be a (t,w)-CFF(v,n) with probability at least $1 - 2^{-s}$
- if $s = 1$, then it will take (on average) two tries until a CFF is constructed
- if $s = 20$, then almost surely a random M will be a CFF (and hence we might not even bother to verify that M is indeed a CFF)

Verification

- how do we verify that a given M is a (t,w)-CFF(v,n)?
- if M is constructed "randomly", it is necessary to check all possible (P,F) to see that the CFF property is satisfied
- the complexity of this algorithm is $O\left(\binom{n}{t}\binom{n-t}{w}(t+w)v\right)$
- for fixed t,w, this simpifies to $O\left(n^{t+w}\log n\right)$
- verification is reasonably fast if $t+w$ is small, or if n is small

Applications of (t, w)-CFF to Multicast Security

group key predistribution

A (t, w)-CFF can be used to construct a KDP in which predefined keys for groups of size at most t exist, that are secure against coalitions of size at most w.

group session key distribution

The TA can use an existing group key (from the above-described KDP) to encrypt a group session key, which is then broadcasted to the group.

blacklisting (broadcast exclusion)

A $(1, w)$-CFF allows the TA to broadcast a group (session) key, K, to everyone in the network *except* for a specified set of at most w excluded users.

This technique can be used for key revocation in a network.

Applications of (t, w)-CFF to Multicast Security

anti-jamming

A $(1, w)$-CFF allows a *TA* to broadcast a message using a set of carrier fequencies so that no coalition of at most w users can "jam" the communication to any intended receiver.

network source authentication

A $(1, w)$-CFF enables everyone in a network to verify that a given message was transmitted by the *TA* (and not by someone else); the scheme is secure against coalitions of size at most w.

Blacklisting

- for every
$$i \notin \bigcup_{U_j \in F} keys(U_j),$$
the TA computes $y_i = e_{k_i}(K)$ using a secure secret-key encryption scheme

- the following properties hold:
  1. no user $U_j \in F$ can compute K (even if the users in F combine all their information)
  2. if $|F| \leq w$, then every user $U_h \notin F$ can decrypt K, because
  $$keys(U_h) \not\subseteq \bigcup_{U_j \in F} keys(U_j)$$
  (this is the $(1, w)$ cover-free property).

Anti-jamming

- a $(1, w)$-CFF is used to distribute *carrier frequencies* to the network users
- every user knows a subset of the v carrier frequencies
- the TA broadcasts v copies of the same message, using all v carrier frequencies
- a bad user can "jam" a frequency, destroying the ability of that frequency to transmit a message
- a coalition of bad users, F, can jam all the frequencies that they know jointly
- if $|F| \leq w$, then every user not in F still receives at least one copy of the message (because of the $(1, w)$ cover-free property)

Source Authentication

- a $(1, w)$-CFF is used to distribute keys for a MAC
- the TA broadcasts a message x, along with v MACs for this message: $MAC_{k_1}(x), \ldots, MAC_{k_v}(x)$
- each user U_j verifies the MACs using all keys that he possesses
- he accepts the message as being valid only if all the MACs that he can compute are valid
- a coalition F of size at most w cannot create a list of forged MACs for a new message x' which will be accepted by any user not in F

Broadcast Encryption to Arbitrary Subsets of Users

- we have seen that a (t, w)-CFF enables broadcast encryption to a subset of at most t users, and a $(1, w)$-CFF enables broadcast encryption to a subset of at least $n - w$ users
- in a general broadcast encryption scheme (BES), a TA wants to broadcast an encrypted message to an *arbitrary* subset P of the n users
- for example, a pay-TV movie, $\mathcal{M}$, might be encrypted with a key K, i.e., $y = e_K(\mathcal{M})$
- the BES is used to encrypt K in such a way that only the members of P can determine K
- the BES is used to encrypt a short key, rather than a long movie, because of the message expansion that is typically required in a BES

The Trivial BES

- suppose the TA gives each user U_i in the network a different key, k_i
- for every $U_i \in P$, the TA can encrypt K with the key k_i, by computing $y_i = e_{k_i}(K)$
- the broadcast $b_P = (y_i : U_i \in P)$ is of length $|P|$, so the *broadcast message expansion* is $|P|$
- this scheme has low storage requirements (one key per user) and high security (no coalition can compute K), but it has a high message expansion
- in general, we want to find good tradeoffs between the parameters of a BES

Designing Efficient BES: A High-Level View

- suppose $P \subseteq \mathcal{U}$, and let w denote the maximum size of a coalition (security parameter)
- many BES involve two steps:
  1. split K into shares $s_1, \ldots, s_v$ using an (r, v) threshold scheme (for some positive integer $r \leq v$)
  2. encrypt every share s_i with a key k_i, in such a way that
     (a) every user $U_j \in P$ can compute at least r of the keys $k_1, \ldots, k_v$ (hence they can decrypt r shares of K, and then reconstruct K)
     (b) any coalition F, such that $F \cap P = \emptyset$ and $|F| \leq w$, can compute at most $r - 1$ of the keys (hence they can decrypt at most $r - 1$ shares of K, and therefore they cannot obtain any information about K)
- the keys k_i are obtained from key predistribution scheme(s)

Efficient BES

- efficient BES can be constructed from Fiat-Naor KDPs
- we construct a collection of v different Fiat-Naor 1-KDPs, each of which is constructed on a certain subset of participants
- a v by n incidence matrix, M, having entries from $\{0, 1\}$, is used to define which users are associated with which schemes
- the v schemes are denoted $\mathcal{F}_1, \dots, \mathcal{F}_v$
- user U_j is given keys from the KDP $\mathcal{F}_i$ if and only if $M[i, j] = 1$
- for $1 \leq i \leq v$, let

$$users(i) = \{U_j : M[i, j] = 1\}$$

and for $1 \leq j \leq n$, let

$$schemes(j) = \{i : M[i, j] = 1\}$$

Encryption Steps

- suppose that $|schemes(j)| = r$ for every j, $1 \leq j \leq n$ (i.e., every user is associated with exactly r of the v schemes)
- for $1 \leq i \leq v$, let k_i denote the group key for $P \cap users(i)$ in the scheme $\mathcal{F}_i$ (if $P \cap users(i) = \emptyset$, then k_i is chosen randomly by the TA)
- proceed to encrypt K, as in the "high-level" description, using an (r, v) threshold scheme
- in order for this BES to be secure, it must be the case that the coalition F cannot compute more than $r - 1$ of the keys $k_1, \ldots, k_v$
- this will be done by ensuring that M satisfies certain additional properties (to be discussed later)

Example of a BES

Suppose $n = 7$, $v = 7$, and the incidence matrix is as follows:

$$M = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Note that every user is associated with $r = 3$ schemes.

Example (cont.)

In the set-up phase, a total of nine keys will be given to each user, as indicated below:

U_1	U_2	U_3	U_4	U_5	U_6	U_7
ℓ_1	ℓ_1	ℓ_2	ℓ_1	ℓ_2	ℓ_3	ℓ_4
$\ell_{1,2}$	$\ell_{1,1}$	$\ell_{2,2}$	$\ell_{1,1}$	$\ell_{2,2}$	$\ell_{3,3}$	$\ell_{4,4}$
$\ell_{1,4}$	$\ell_{1,4}$	$\ell_{2,5}$	$\ell_{1,2}$	$\ell_{2,3}$	$\ell_{3,4}$	$\ell_{4,5}$
ℓ_5	ℓ_2	ℓ_3	ℓ_3	ℓ_4	ℓ_5	ℓ_6
$\ell_{5,5}$	$\ell_{2,3}$	$\ell_{3,4}$	$\ell_{3,3}$	$\ell_{4,4}$	$\ell_{5,1}$	$\ell_{6,2}$
$\ell_{5,6}$	$\ell_{2,5}$	$\ell_{3,6}$	$\ell_{3,6}$	$\ell_{4,7}$	$\ell_{5,5}$	$\ell_{6,6}$
ℓ_7	ℓ_6	ℓ_7	ℓ_4	ℓ_5	ℓ_6	ℓ_7
$\ell_{7,3}$	$\ell_{6,6}$	$\ell_{7,1}$	$\ell_{4,5}$	$\ell_{5,1}$	$\ell_{6,2}$	$\ell_{7,1}$
$\ell_{7,7}$	$\ell_{6,7}$	$\ell_{7,7}$	$\ell_{4,7}$	$\ell_{5,6}$	$\ell_{6,7}$	$\ell_{7,3}$

Example (cont.)

- suppose that the TA wants to broadcast a message to the set $P = \{U_1, U_2, U_3\}$
- the following will be the keys used in the seven Fiat-Naor schemes:

i	$users(i) \cap P$	k_i
1	$\{U_1, U_2\}$	$\ell_1 + \ell_{1,4}$
2	$\{U_2, U_3\}$	$\ell_2 + \ell_{2,5}$
3	$\{U_3\}$	$\ell_3 + \ell_{3,4} + \ell_{3,6}$
4	$\emptyset$	random
5	$\{U_1\}$	$\ell_5 + \ell_{5,5} + \ell_{5,6}$
6	$\{U_2\}$	$\ell_6 + \ell_{6,6} + \ell_{6,7}$
7	$\{U_1, U_3\}$	$\ell_7 + \ell_{7,7}$

Example (cont.)

- $r = 3$ and $v = 7$, so the TA sets up a $(3, 7)$ threshold scheme in $\mathbb{Z}_p$
- let $s_1, \ldots, s_7$ be the seven shares of the secret K, where $K \in \mathbb{Z}_p$
- the broadcast $b_P = (b_1, \ldots, b_7)$ consists of the following values:

$$\begin{aligned}
b_1 &= e_{\ell_1+\ell_{1,4}}(s_1)\\
b_2 &= e_{\ell_2+\ell_{2,5}}(s_2)\\
b_3 &= e_{\ell_3+\ell_{3,4}}(s_3)\\
b_4 &= \text{irrelevant}\\
b_5 &= e_{\ell_5+\ell_{5,5}+\ell_{5,6}}(s_5)\\
b_6 &= e_{\ell_6+\ell_{6,6}+\ell_{6,7}}(s_6)\\
b_7 &= e_{\ell_7+\ell_{7,7}}(s_7)
\end{aligned}$$