Lower Bounds for Subset Cover Based Broadcast Encryption

Per Austrin Gunnar Kreitz

KTH – Royal Institute of Technology

Africacrypt '08, June 13

KTH Computer Science and Communication

Three Usage Scenarios

- Pay-per-view Euro 2008 on your cell phone
- BluRay copy protection
- Military radio communication

What is Broadcast Encryption?

- The problem of establishing secure communication with a changing group of receivers
- One trusted sender, multiple receivers
- Network is a broadcast medium
- Berkovits 1991, Fiat and Naor 1994

KTH Computer Science and Communication

The Basic Principle

- Initialize the system by giving each user some private information
- Establish a *message key* (sometimes called *group key*), K_m
- Broadcast content encrypted with K_m

KTH Computer Science and Communication

The Basic Principle

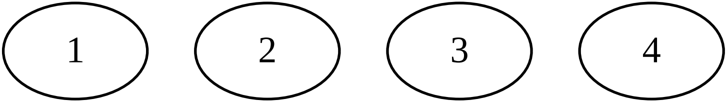
- Initialize the system by giving each user some private information
- Establish a *message key* (sometimes called *group key*), K_m
- Broadcast content encrypted with K_m
- Updating the message key (depending on application)
  - When some number of members (possibly 1) have left/joined
  - At timed intervals
  - A combination of the above

KTH Computer Science and Communication

Notation and Terminology

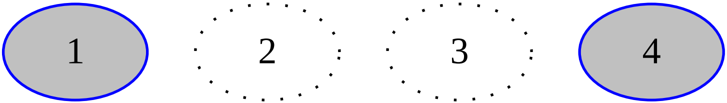
- m is the number of *members*
- r is the number of *revoked users*
- $n = r + m$ is the number of *users*

The Naive Scheme



- One symmetric key for each user
- To establish message key K_m, broadcast K_m encrypted with each member's key

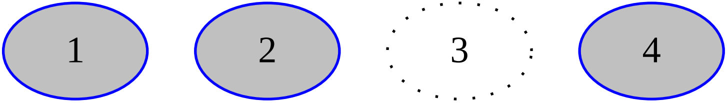
The Naive Scheme



- One symmetric key for each user
- To establish message key K_m, broadcast K_m encrypted with each member's key
- Example: $\mathcal{M} = \{1, 4\}$
- Broadcast: $\mathrm{E}_{K_1}(K_m)$, $\mathrm{E}_{K_4}(K_m)$

KTH Computer Science and Communication

The Naive Scheme



- One symmetric key for each user
- To establish message key K_m, broadcast K_m encrypted with each member's key
- Example: $\mathcal{M} = \{1, 2, 4\}$
- Broadcast: $\mathrm{E}_{K_1}(K'_m), \mathrm{E}_{K_2}(K'_m), \mathrm{E}_{K_4}(K'_m)$

Scheme parameters

- b is the *bandwidth* overhead
- s is the *space* required at users
- (We will ignore *time* to decrypt for this talk)

KTH Computer Science and Communication

Scheme parameters

- b is the *bandwidth* overhead (m for naive)
- s is the *space* required at users (1 for naive)
- (We will ignore *time* to decrypt for this talk)

Subset Cover-based Broadcast Encryption

- Subset Cover is a principle for constructing Broadcast Encryption schemes
- Static family of sets of users
- Each set is associated with a key
- Only users in the subset know the key
- Naor, Naor, Lotspiech 2001

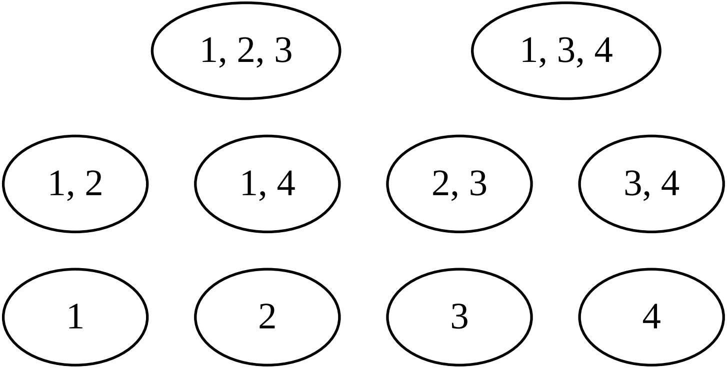
KTH Computer Science and Communication

Subset Cover (cont'd)

- To distribute a new group key
  1. Compute a cover of the members (avoiding revoked users), using the subsets
  2. Encrypt message key K_m with subset key for each subset in cover
- Bandwidth is equal to cover size

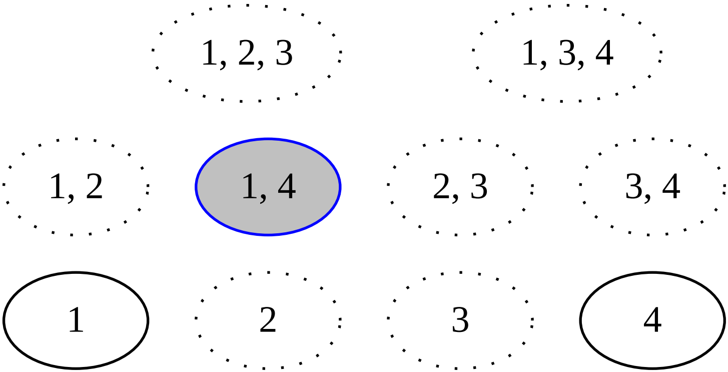
KTH Computer Science and Communication

Subset Cover Example



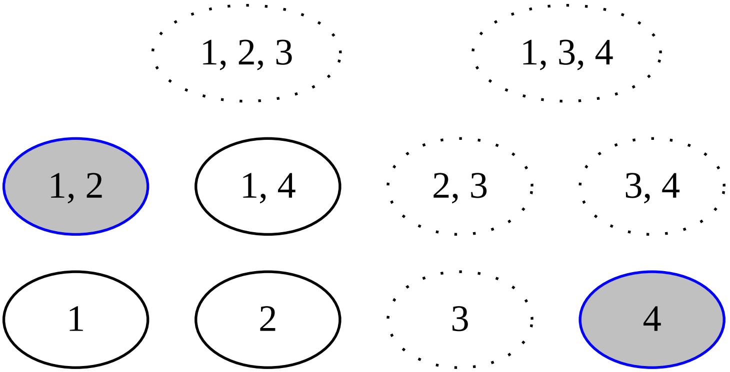
- Each node is a key shared between users named in node

Subset Cover Example



- Each node is a key shared between users named in node
- Example: $\mathcal{M} = \{1, 4\}$
- Broadcast: $\mathrm{E}_{K_{1,4}}(K_m)$

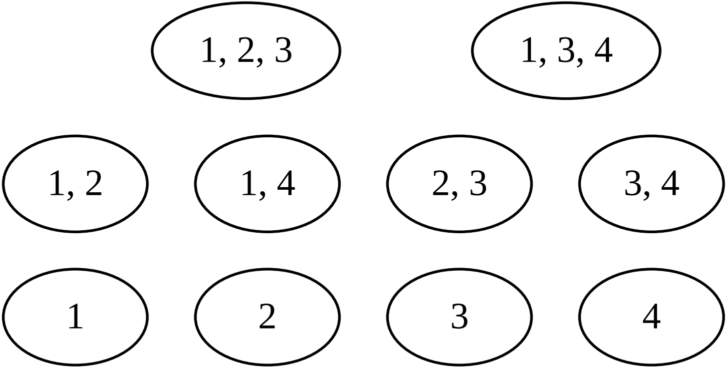
Subset Cover Example



- Each node is a key shared between users named in node
- Example: $\mathcal{M} = \{1, 2, 4\}$
- Broadcast: $\mathrm{E}_{K_{1,2}}(K'_m), \mathrm{E}_{K_4}(K'_m)$

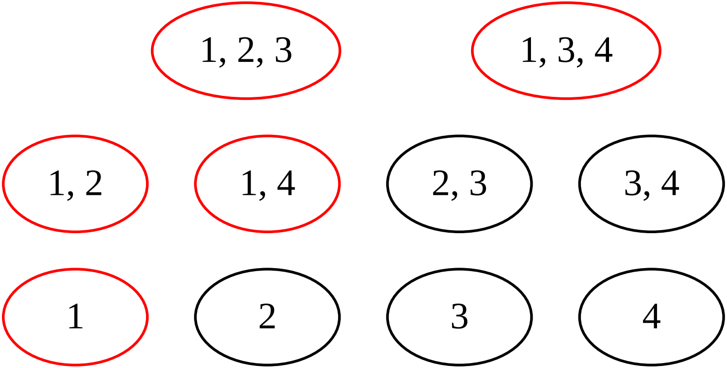
KTH Computer Science and Communication

Key Derivation



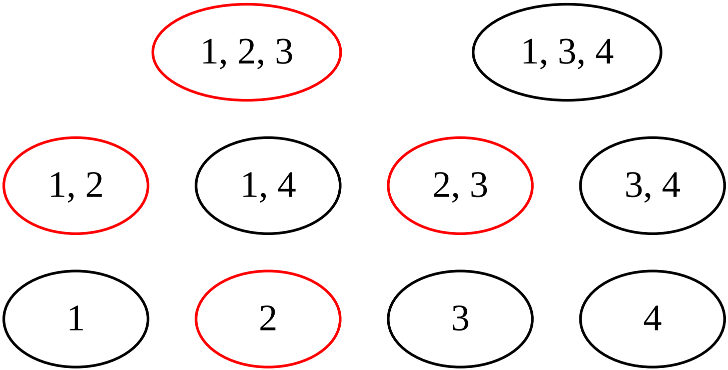
- Bandwidth of this scheme is 2

Key Derivation



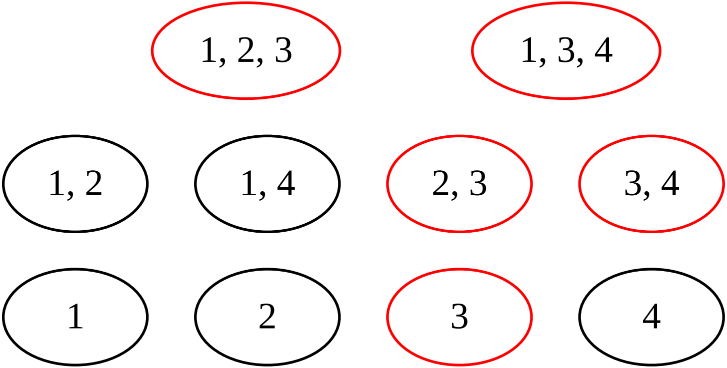
- Bandwidth of this scheme is 2
- Space of this scheme is

Key Derivation



- Bandwidth of this scheme is 2
- Space of this scheme is

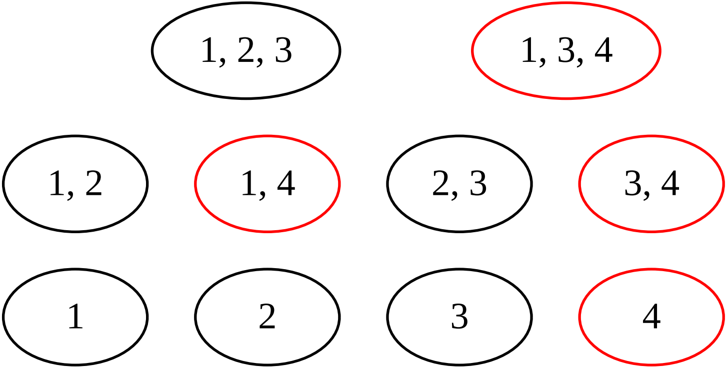
Key Derivation



- Bandwidth of this scheme is 2
- Space of this scheme is

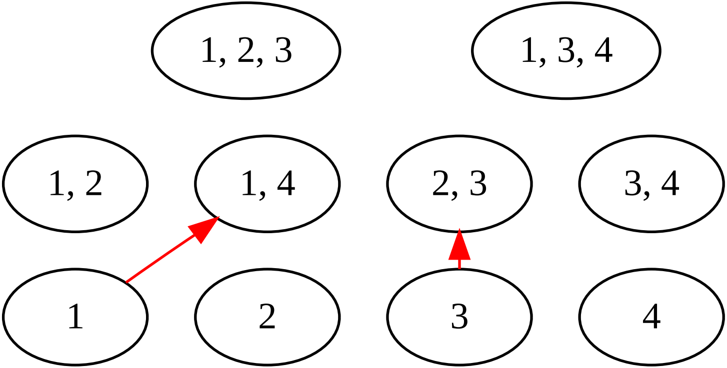
KTH Computer Science and Communication

Key Derivation



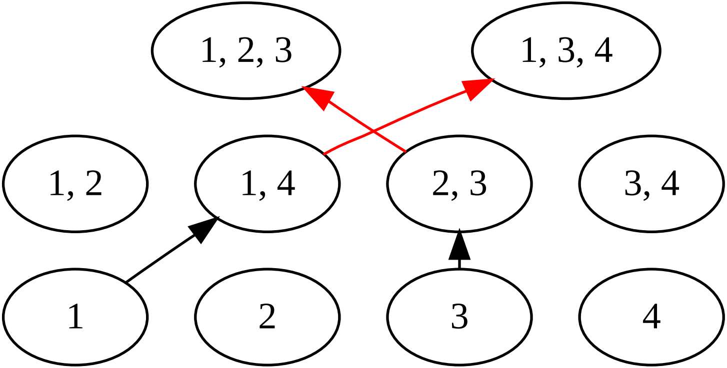
- Bandwidth of this scheme is 2
- Space of this scheme is 5

Key Derivation



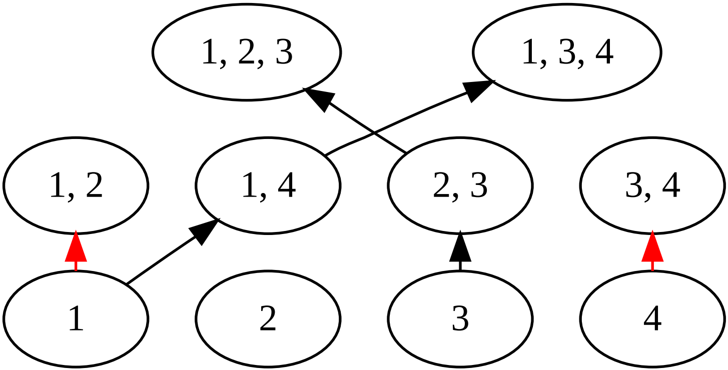
- Bandwidth of this scheme is 2
- Space of this scheme is 4
- $K_{1,4} = \mathrm{PRG}(K_1)$, $K_{2,3} = \mathrm{PRG}(K_3)$

Key Derivation



- Bandwidth of this scheme is 2
- Space of this scheme is 3
- $K_{1,4} = \mathrm{PRG}\,(K_1)$, $K_{2,3} = \mathrm{PRG}\,(K_3)$

Key Derivation



- Bandwidth of this scheme is 2
- Space of this scheme is 3
- $K_{1,4} = \mathrm{PRG}(K_1),\ K_{2,3} = \mathrm{PRG}(K_3)$

KTH Computer Science and Communication

Unique Predecessor-schemes

- Unique Predecessor (UP) schemes
- The indegree of each node is at most 1
- Models natural key derivation with PRG
- Called Sequential Key Derivation Pattern in Attrapadung, Komara, Imai 2003

Unique Predecessor-schemes (cont'd)

- Allow any outdegree
- Allow any depth (inherently $\leq n$)
- Every singleton node must be present
- Key derivation graph will be a forest
- After *normalization*, there will be exactly n trees

KTH Computer Science and Communication

An Easy Key Lemma

Lemma

Any Unique Predecessor scheme will have at most ns distinct subsets.

An Easy Key Lemma

Lemma

Any Unique Predecessor scheme will have at most ns distinct subsets.

Proof-sketch

Adding a new subset means at least one user needs to store one more key, proof by induction

KTH Computer Science and Communication

Performance of PRG-based Subset Cover schemes

Scheme	Bandwidth	Space	Authors
Subset Difference	$2r-1$	$\mathcal{O}\left(\log^2 n\right)$	NNL01
Layered SD	$4r-2$	$\mathcal{O}\left(\log^{3/2} n\right)$	HS02
Stratified SD	$2r-1$	$\mathcal{O}(\log n)$	GST04
Punctured Intervals (π)	$r/c+\mathcal{O}(1)$	$\mathcal{O}(\text{poly}\, n)$	JHCKLY05

KTH Computer Science and Communication

Performance of PRG-based Subset Cover schemes

Scheme	Bandwidth	Space	Authors
Subset Difference	$2r-1$	$\mathcal{O}\left(\log^2 n\right)$	NNL01
Layered SD	$4r-2$	$\mathcal{O}\left(\log^{3/2} n\right)$	HS02
Stratified SD	$2r-1$	$\mathcal{O}(\log n)$	GST04
Punctured Intervals (π)	$r/c+\mathcal{O}(1)$	$\mathcal{O}(\text{poly}\, n)$	JHCKLY05

- Is $\mathcal{O}(r)$ the best we can do?

Lower Bounds for Schemes Without Key Derivation

- $s \geq \left(\frac{\binom{n}{r}^{1/b}}{b} - 1 \right) / r$ by Luby and Staddon 98
- $s \geq (\binom{n}{r}^{1/b} - 1)/r$ by Gentry et al. 06
- Proofs using the Sunflower lemma

KTH Computer Science and Communication

Generic Lower Bound

- All broadcast encryption scheme need to encode the revoked subset
- (It is possible to test which users can decrypt correctly)
- This gives lower bound of $\approx r \log n$ bits

KTH Computer Science and Communication

Few Revoked Users

- For BluRay players, we can expect few revoked users (players)

Few Revoked Users

- For BluRay players, we can expect few revoked users (players)
- With polynomial space, worst case bandwidth will be $\Omega(r)$ for "small" r

KTH Computer Science and Communication

A Lower Bound for Small r

Theorem

Let $c \geq 0$ and $0 \leq \delta < 1$. Then, any UP-scheme with n users and space $s \leq n^c$ will, when the number of revoked users $r \leq n^\delta$, require bandwidth

$$b \geq \frac{1-\delta}{c+1} \cdot r \tag{1}$$

KTH Computer Science and Communication

A Lower Bound for Small r

Theorem

Let $c \geq 0$ and $0 \leq \delta < 1$. Then, any UP-scheme with n users and space $s \leq n^c$ will, when the number of revoked users $r \leq n^\delta$, require bandwidth

$$b \geq \frac{1-\delta}{c+1} \cdot r \tag{1}$$

Proof-sketch

Count the number of ways to pick up to b subsets, and compare to the number of ways to choose r revoked users

KTH Computer Science and Communication

How tight is the bound?

- Subset Difference has $s = \log^2 n,\ b = 2r - 1$
- For $r = \sqrt{n}$ our bound gives $b \geq \frac{r}{2(1+o(1))}$
- Within a factor $4 + o(1)$

KTH Computer Science and Communication

How strong is the bound?

- Our bound only applies to UP-schemes
- We do not place any restriction on decryption time
- Our bound applies when space is polynomial
- Generally, logarithmic or poly-logarithmic space is acceptable

KTH Computer Science and Communication

Many revoked users

- In pay-per-view scenarios, we can expect (relatively) few members

Many revoked users

- In pay-per-view scenarios, we can expect (relatively) few members
- For $m < n/6s$, worst case bandwidth will be m (same as Naive scheme)

A Lower Bound for Large r

Theorem

For any UP-scheme $\mathcal{S}$ and $m \leq \frac{n}{6s}$, there is a member set $\mathcal{M}$ of size $|\mathcal{M}| = m$ requiring bandwidth $b = |\mathcal{M}|$.

KTH Computer Science and Communication

A Lower Bound for Large r

Theorem

For any UP-scheme $\mathcal{S}$ and $m \leq \frac{n}{6s}$, there is a member set $\mathcal{M}$ of size $|\mathcal{M}| = m$ requiring bandwidth $b = |\mathcal{M}|$.

Proof-sketch

- Revoke users with to high outdegree of their singleton node
- Pick a non-revoked user to keep as member, revoke users so that only her singleton key is usable
- This step will revoke at most $3s$ users each time

KTH Computer Science and Communication

How tight is the bound?

- There is a scheme with $b < m$ when $m > \lceil n/s \rceil$
- Partition the users into blocks of size $\leq s$ and let each pair in a block share a key
- Bound is tight within a factor 6

KTH Computer Science and Communication

What's in the middle?

- For military communication, the number of members will vary

What's in the middle?

- For military communication, the number of members will vary
- For some r, bandwidth is at least $n/1.89 \log_2 s$

A Lower Bound for Arbitrarily Many Revoked Users

Theorem

Let $\delta \in (0,1]$ and $\epsilon > 0$. Then for every UP-scheme $\mathcal{S}$ with $n > \frac{2\delta(1-\delta)}{\epsilon^2}$ there exists a set of users M of size $\delta - 3\epsilon \leq |M|/n \leq \delta + \epsilon$ which requires bandwidth $b \geq |M| \frac{\log(1/\delta)}{\log(s/\epsilon)}$

KTH Computer Science and Communication

A Lower Bound for Arbitrarily Many Revoked Users

Theorem

Let $\delta \in (0, 1]$ and $\epsilon > 0$. Then for every UP-scheme $\mathcal{S}$ with $n > \frac{2\delta(1-\delta)}{\epsilon^2}$ there exists a set of users M of size $\delta - 3\epsilon \leq |M|/n \leq \delta + \epsilon$ which requires bandwidth $b \geq |M| \frac{\log(1/\delta)}{\log(s/\epsilon)}$

Proof-sketch

- If the largest usable sets have size k, bandwidth will be $\geq m/k$
- Revoke each user with probability $1 - \delta$
- From each subset of size $k + 1$, revoke one more user
- Show that with positive probability, a sufficiently large number of members will remain

KTH Computer Science and Communication

How tight is the bound?

- There is a scheme with bandwidth b at most $\left\lceil \frac{n}{\log_2(s)} \right\rceil$
- Partition the users into blocks of size $\leq \log_2(s)$
- In each block, let every subset of users share a key
- Our bound is tight within a factor 1.89

Summary

- Have shown lower bounds on bandwidth for a class of broadcast encryption schemes
- Bounds seem hard to sidestep without using more expensive key derivation techniques
- Bounds are tight up to small constants

KTH Computer Science and Communication

Thank you!

KTH Computer Science and Communication