Key Management For Restricted Multicast Using Broadcast Encryption

Michel Abdalla, Yuval Shavitt & Avishai Wool

Presented by:

Karthik Narayan

24th Sep 2002

Area Of Usage

- Internet secure Multicast via M-Bone (Multicast Backbone)
- Pay TV domain

Assumptions made :

- Secure memory that cannot be cracked.
- A system able to handle multiple calculations.
- The system has a large number of users.

Related Work

- Fiat and Naor (Impractical Number of Keys)
- Luby and Staddon (Fixed size target sets)
- "Tracing Traitors" (Identifying cloned receivers)
- Iolus Project (Tree structure. Group servers using secure multicast at higher levels and Group members communicating at lower levels)
- Wong et al. Key Management for multicast in the internet. (Balanced tree / nodes are group keys and leaves are group members. New tree per program has high overhead.)

Purpose of this paper

- Reduction of the number of transmissions.
- Limiting the number of free riders.
- Not be restricted by the target size, make an equally efficient structure for both large and small groups.
- Reduce the complexity of computation.

Measures of quality

- Number of transmissions T
- Proportion of free riders outside the target set – Opportunity
- Number of subscribers n
- Number of keys necessary would be O(log n) as compared to n keys in a normal situation.

How stuff works

- Split the population into partitions.
- Each partition has its own establishment key allocation.
- A new partition is created at the beginning of each phase.
- Virtual users are used to fill in the new partition.
- Real users replace the virtual users once they join.
- Real users get the virtual keys established.
- Phase ends when all virtual users are replaced.

When a user leaves

- Once a user disconnects he is marked as a nonexistent entity.
- Once the number of nonexistent entities drops below a threshold a new partition is created.
- Existing users are rekeyed to a new partition.
- It is the same as giving a new key to all existing users.

Transmission of decryption keys in a dynamic environment

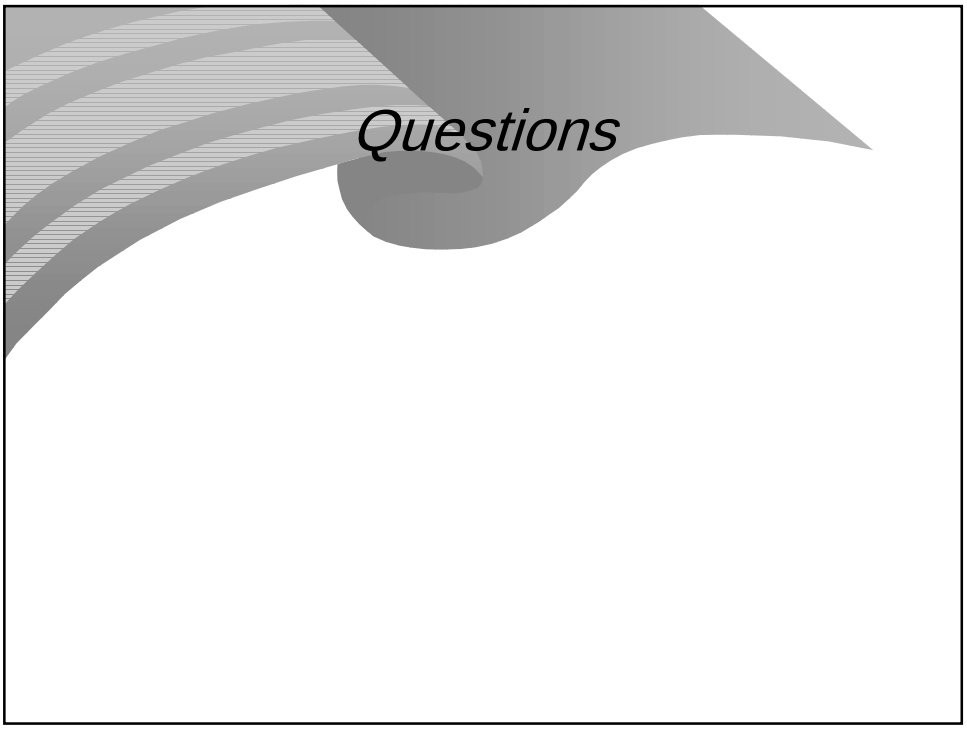
- Divide the programs transmission time into slots
- Each slot has a different encryption key
- Pick I continuous slots
- Generate a key after collecting all join leave operation that would be valid for the next I slots of the program.

Advantage of this method

- As the number of users are high the percentage of free riders are always low
- New keys are generated only when a group falls below the threshold value
- For a dynamic situation, no user would be able to view the program free for more than I+1 sessions.
- Since the group is split into partitions, key changes / updating keys are easily done.

Things in the dark

- Paper fails to mention the encryption and decryption scheme used.
- Accepts free runners are a part of the scheme.
- Relies on the fact that the keys are changed every I sessions.


Questions