Key Management Protocols
STR Protocol (Distributed GKMP)

- It is totally decentralized and based on equal contributions from all the members.

- It provides basic requirements like forward secrecy, backward secrecy and key independence.

- It also requires smaller number of unicasts and multicasts to compute a new group key after a member leave or join.
STR Protocol

- Unbalanced tree of height n-1 where n is current group size.
- Final group key:
  \[ K_n = g^{rn} \cdot g^{r_{n-1} \cdot r_2 \cdot r_1} \]
- Important Recurrence
  \[ k_i = (bk_{i-1})^{ri} \mod p = (br_i)^{ki-1} \mod p \]
- All \( br_i \)'s and \( bk_i \)'s are known to all members.

\[ br_i = g^{ri} \mod p \]
\[ bk_i = g^{ki} \mod p \]
Initialization:
- $M_1$ computes
  
  $k_2 = (br_2)^{r_1} \mod p = g^{r_1 r_2} \mod p, bk_2 = g^{k_2} \mod p$
  
  $k_3 = (br_3)^{k_2} \mod p, bk_3 = g^{k_3} \mod p$
  
  $k_n = (br_n)^{k_{n-1}} \mod p$

- $M_1$ broadcasts all $bk_i$’s to members
- Each member then computes the group key from $bk_i$’s.
- For ex:
  - $M_3$ computes $k_3 = (bk_2)^{r_3}$
  - $br_i = g^{ri} \mod p$
  - $bk_i = g^{ki} \mod p$

  $k_i = (bk_{i-1})^{ri} \mod p = (br_i)^{ki-1} \mod 4p$
STR Protocol (Cont.)

Join:

- M_5 broadcasts br_5 to all members.
- Each member then computes a new key.
- M_4 sends all previous br_i’s and bk_i’s to M_5 who then computes a new key k_5.

\[ br_i = g^{ri} \mod p \]
\[ bk_i = g^{ki} \mod p \]
\[ k_i = (bk_{i-1})^{ri} \mod p = (br_i)^{ki-1} \mod^5 p \]
STR Protocol (Cont.)

Leave:
- If $M_n$ leaves then $M_{n-1}$ takes responsibility to create new key.
- Everybody renumbers the node.
- $M_{n-1}$ selects new random key $r'_{n-1}$. Computes all $bk_i$'s and broadcasts it to all members.

\[
br_i = g^{ri} \mod p \\
bk_i = g^{ki} \mod p \\
k_i = (bk_{i-1})^{ri} \mod p = (br_i)^{ki-1} \mod p
\]
TGDH protocol

- The TGDH protocol uses binary trees; every node is either a leaf or a parent of two nodes.
- The nodes are denoted as \( < l, v > \), where \( 0 < v < 2l - 1 \) since each level \( l \) hosts at most \( 2l \) nodes.
- Each node \( < l, v > \) is associated with the key \( K < l, v > \) and the public blinded key (bkey) \( BK < l, v > = f(K < l, v >) \)
- where the function \( f(\cdot) \) is modular exponentiation in prime order groups, i.e. \( f(K) = g^{k \mod p} \).
- Computing a key at \( < l, v > \) requires the knowledge of the key of one of the two child nodes and the bkey of the other child node.
TGDH

Diagram with levels:
- $l = 0$
- $l = 1$
- $l = 2$
- $l = 3$

Levels:
- $h = 3$
- $N = 6$

Nodes:
- $<0,0>$
- $<1,0>$
- $<1,1>$
- $<2,0>$
- $<2,1>$
- $<2,2>$
- $<2,3>$
- $<3,0>$
- $<3,1>$
- $<3,6>$
- $<3,7>$

Nodes labeled:
- $M_1$, $M_2$, $M_3$, $M_4$, $M_5$, $M_6$
The final group key $K < 0, 0 >$ is: $K < 0, 0 >=$

$$g \left( g^{r_3( g^{r_1 r_2})} \right) (g^{r_4(g^{r_5 r_6})})$$