Secure Group Communications using Twisted Group Rings

M.D. Gómez Olvera J.A. López Ramos B. Torrecillas Jover

March 2, 2020

1 Introduction

In recent years, new hard problems have been proposed in public key cryptography, since those that we are using might be not secure soon. When two parties want to communicate through an insecure channel, they need to do a key agreement, which consist on agreeing on a secret shared key by exchanging information that does not compromise the common key.

The first widely used protocol that allows this to happen was proposed in 1976 by W. Diffie and M. Hellman [2], and works as follows:

> Let two users, Alice and Bob, who want to agree on a common key through an insecure channel. Let p a prime number, $\mathbb{Z}_p^*$ the multiplicative group of integers modulo p, and g a primitive root modulo p public.
>
> 1. Alice chooses a secret integer a, and sends Bob $p_A = g^a (\bmod\ p)$.
> 2. Bob chooses a secret integer b, and sends Alice $p_B = g^b (\bmod\ p)$.
> 3. Alice computes $p_B^a (\bmod\ p)$, and Bob computes $p_A^b (\bmod\ p)$, so both obtain the same value, which is the secret shared key $K = g^{ab} (\bmod\ p)$.

Information shared does not compromise the shared key since the underlying problem an attacker would need to solve, the so-called Discrete Logarithm Problem (DLP) is believed to be hard. This key agreement can be seen as an example of this generalization by Maze et al [9]:

> Let S be a finite set, G an abelian semigroup, ϕ a $G-$action on S, and a public element $s \in S$.
>
> 1. Alice chooses $a \in G$, and sends Bob $p_A = \phi(a, s)$.
> 2. Bob chooses $b \in G$, and sends Alice $p_B = \phi(b, s)$.
> 3. Alice computes $\phi(a, p_B)$, and Bob computes $\phi(b, p_A)$, so both obtain the secret shared key $K = \phi(a, \phi(b, s)) = \phi(b, \phi(a, s))$.

whose underlying problem is called the Semigroup Action Problem (SAP).

> **Semigroup Action Problem.** Given a semigroup action ϕ of the group G on a set S and elements $x \in S$ and $y \in G$, find $g \in G$ such that $\phi(g, x) = y$.

In the context of SAP, we proposed in [4] a new setting, and some protocols. In our case, the platform is a twisted group ring, a new proposal in the context of group rings, that have also been recently used in cryptography in works like [3, 6, 5, 7]. And the action proposed is the two-sided multiplication in a twisted group ring, so the problem is a variation in the twisted case of the so-called Decomposition Problem (DP), which is a generalization of the Conjugate Search Problem (CSP).

> **Decomposition Problem.** Given a group G, $(x, y) \in G \times G$ and $S \subset G$, the problem is to find $z_1, z_2 \in S$ such that $y = z_1 x z_2$.

A natural extension is how to extend this kind of schemes to more than two users. In the classic Diffie-Hellman protocol, a solution is proposed in [10]. And in the more case of SAP, this solution can be found in [8]. In both cases, it is shown that the extra information shared in the case of a n users key exchange does not imply information leakage for an attacker compared to the 2-users case.

Our aim in this work is to show that in our setting, that differs from those above given the non-commutativity of twisted group rings, and which could work better against problems that threat current communications, this is also true: the extra information shared between n users does not imply information leakage, so if the 2-users key exchange is computationally secure, then the extension to n users is also secure.

2 Algebraic Setting

In this section, twisted group rings are defined, and we also show some properties that make the key exchange possible.

Definition 2.1. *Let K be a ring, G be a multiplicative group, and α be a cocycle in $U(K)$, the units of K. The group ring $K^\alpha G$ is defined to be the set of all finite sums of the form*

$$\sum_{g_i \in G} r_i g_i,$$

where $r_i \in K$ and all but a finite number of r_i are zero.

The sum of two elements in $K^\alpha G$ is given by

$$\Big(\sum_{g_i \in G} r_i g_i\Big) + \Big(\sum_{g_i \in G} s_i g_i\Big) = \sum_{g_i \in G} (r_i + s_i) g_i.$$

And multiplication, which is twisted by a cocycle, is given by

$$\Big(\sum_{g_i \in G} r_i g_i\Big) \cdot \Big(\sum_{g_i \in G} s_i g_i\Big) = \sum_{g_i \in G} \Big(\sum_{g_j g_k = g_i} r_j s_k \ \alpha(g_j, g_k)\Big) g_i.$$

As an example, consider the finite field K, a primitive element t, and the dihedral group of $2m$ elements, $D_{2m} =< x, y : x^m = y^2 = 1, yx^a = x^{m-a} y >$. The group ring $R = K^\alpha D_{2m}$, where α is

$$\alpha : D_{2m} \times D_{2m} \to K^*$$

with $\alpha(x^i, x^j y^k) = 1$ and $\alpha(x^i y, x^j y^k) = t^j \ i, j = 1, ..., 2m-1$, is a twisted group ring.

Now we establish some useful properties that will allow us to make our key exchange possible.

Definition 2.2. *Let $R = K^\alpha D_{2m}$, where t is the primitive root of unity that generates K and α is the cocycle defined above. Given $h \in R$,*

$$h = \sum_{\substack{0 \le i \le m-1 \\ k=0,1}} r_i x^i y^k,$$

where $r_i \in K$ and $x, y \in D_{2m}$. We define $h^* \in K^\alpha D_{2m}$:

$$h^* = \sum_{\substack{0 \le i \le m-1 \\ k=0,1}} r_i t^{-i} x^i y^k,$$

where $r_i \in K$ and $x, y \in D_m$.

Note that $R = K^\alpha D_{2m}$ can be written as vector space as

$$R = R_1 \oplus R_2,$$

where $R_1 = KC_m$ and $R_2 = K^\alpha C_m y$, and C_m is a cyclic group of order m. In this context, we can define $A_j \le R_j$ as

$$A_j = \Big\{ \sum_{i=0}^{m-1} r_i x^i y^k \in R_j : r_i = r_{m-i} \Big\}.$$

where $j = 1, 2$.

Proposition 2.3. *Given $h_1, h_2 \in R$,*

- *If $h_1, h_2 \in R_1$, then $h_1 h_2 = h_2 h_1$;*
- *If $h_1, h_2 \in A_2$, then $h_1 h_2^* = h_2 h_1^*$, and $h_1^* h_2 = h_2^* h_1$;*
- *If $h_1 \in A_1, h_2 \in A_2$, then $h_1 h_2 = h_2 h_1^*$.*

A proof of this proposition can be found in [4].

3 Key management over twisted group rings

In this section, we explain the protocols proposed in [4], over the twisted group ring $R = K^\alpha D_{2m}$ defined above.

Let $h \in R$ be a random public element. The key exchange between two users, Alice and Bob, is as follows:

1. Alice selects a secret pair $s_A = (g_1, k_1)$, where $g_1 \in R_1, k_1 \in A_2 \le R_2$.
2. Bob selects a secret pair $s_B = (g_2, k_2)$, where $g_2 \in R_1, k_2 \in A_2 \le R_2$.
3. Alice sends Bob $p_A = g_1 h k_1$, and Bob sends Alice $p_B = g_2 h k_2$.
4. Alice computes $K_A = g_1 p_B k_1^*$, and Bob computes $K_B = g_2 p_A k_2^*$, and they get the same secret shared key.

This protocol works, it was shown in [4]. Let the underlying decisional problem be the following:

Let $R = K^\alpha D_{2m} = R_1 \oplus R_2$, $A_2 \leq R_2$, given $(h, g_1hk_1, g_2hk_2, r_1hr_2)$, decide whether $(r_1, r_2) = (g_2g_1, k_1k_2^*)$ or not, where $h \in R$, $g_i, r_1 \in R_1$, $k_i \in A_2, r_2 \in A_1$.

It means that if someone breaks this problem, then the key exchange above can also be broken.

To define the general protocol for n users, let us define the action $\phi : (R_1 \times A_2) \times R \longrightarrow R$,

$$\phi(s_i, h) = g_ihk_i$$

where $s_i = (g_i, k_i)$. Note that

$$\phi(s_i\phi(s_j, h)) = \phi(s_is_j, h)$$

We will sometimes write $\phi(s_is_j, h)$ to refer to $\phi(s_i, \phi(s_j, h))$, to make some definitions more readable.

Let $h \in R$ be a random public element, and $h \in R = R_1 \oplus R_2$, described before. For $i = 1, ..., n$, user U_i has a secret pair $s_i = (g_i, k_i)$, where $g_i \in R_1$ and $k_i \in A_2 \leq R_2$. Let $\phi(s_i, h) = g_ihk_i$, 2-sided multiplication. We will denote $s_i^* = (g_i, k_i^*)$. The key establishment for n is as follows:

1. For $i = 1, ..., n$, user U_i sends to user U_{i+1} the message

$$\{C_i^1, C_i^2, ..., C_i^{i+1}\},$$

   where $C_1^1 = h$, $C_1^2 = g_1hk_1$ and

   - for $i > 1$ even, $C_i^j = \phi(s_i, C_{i-1}^j)$, when $j < i$, $C_i^i = C_{i-1}^i$, $C_i^{i+1} = \phi(s_i^*, C_{i-1}^i)$,
   - for $i > 1$ odd, $C_i^j = \phi(s_i^*, C_{i-1}^j)$, when $j < i$, $C_i^i = C_{i-1}^i$, $C_i^{i+1} = \phi(s_i, C_{i-1}^i)$.

2. User U_n computes $\phi(s_n, C_{n-1}^n)$ if n is odd and $\phi(s_n^*, C_{n-1}^n)$ if n is even.

3. User U_n broadcasts

$$\{C_n^1, C_n^2, ..., C_n^n\}.$$

4. User U_i computes $\phi(s_i, C_n^i)$ if n is odd or $\phi(s_i^*, C_n^i)$ if n is even, and gets the shared key.

This protocol allows all users to obtain a common shared key, as shown in Proposition 3 of [4]. In this case, the underlying decisional problem is the following:

- *(n even) Let $R = K^\alpha D_{2m} = R_1 \oplus R_2$, $A_2 \leq R_2$, given r_1hr_2, and*

$$\left\{\phi(s_{i_1}s_{i_2}^*s_{i_3}...s_{i_{m-2}}^*s_{i_{m-1}}s_{i_m}^*, h) : \{i_1, ..., i_m\} \subsetneq \{1, ..., n\}, m \in \{1, ..., n-1\}\right\}$$

  decide whether $(r_1, r_2) = (g_1g_2g_3...g_{n-1}g_n, k_1k_2^*k_3...k_{n-1}k_n^*)$ or not, where $h \in R$, $g_i, r_1 \in R_1$, $k_i \in A_2, r_2 \in A_1$.

- *(n odd) Let $R = K^\alpha D_{2m} = R_1 \oplus R_2$, $A_2 \leq R_2$, given r_1hr_2, and*

$$\left\{\phi(s_{i_1}s_{i_2}^*s_{i_3}...s_{i_{m-2}}s_{i_{m-1}}^*s_{i_m}, h) : \{i_1, ..., i_m\} \subsetneq \{1, ..., n\}, m \in \{1, ..., n-1\}\right\}$$

  decide whether $(r_1, r_2) = (g_1g_2g_3...g_{n-1}g_n, k_1k_2^*k_3...k_{n-1}^*k_n)$ or not, where $h \in R$, $g_i, r_1 \in R_1$, $k_i, r_2 \in A_2$.

We have described the so-called Initial Key Agreement (IKA), but another important process in group communication is key refreshment through the Auxiliary Key Agreement (AKA), which takes advantage of the information that was sent before to create a new key in a group when necessary, and is more computationally efficient than IKA. There exist three situations: the members of the group stay the same, a member leaves the group, or someone new joins it.

In the first situation, very user U_i has the information C_n^i received from the user U_n. The rekeying process can be carried out by any of them. We call this user U_c. He chooses a new element $\widetilde{s_c} = (\widetilde{g_c}, \widetilde{k_c})$, where $\widetilde{g_c} \in R_1$ and $\widetilde{k_c} \in A_2$. If n is odd, he changes his private key to $\widetilde{s_c}^* s_c$ and broadcasts the message

$$\{\phi(\widetilde{s_c}^*, C_n^1), \phi(\widetilde{s_c}^*, C_n^2), ..., \phi(\widetilde{s_c}^*, C_n^{c-1}), C_n^c, \phi(\widetilde{s_c}^*, C_n^{c+1}), ..., \phi(\widetilde{s_c}^*, C_n^n)\}.$$

If n is even, he changes his private key to $\widetilde{s_c} s_c^*$ and broadcasts the message

$$\{\phi(\widetilde{s_c}, C_n^1), \phi(\widetilde{s_c}, C_n^2), ..., \phi(\widetilde{s_c}, C_n^{c-1}), C_n^c, \phi(\widetilde{s_c}, C_n^{c+1}), ..., \phi(\widetilde{s_c}, C_n^n)\}.$$

Then every user recovers the common key using the private key s_i if n is even, and s_i^* if n is odd. A proof can be found in [4].

In the second case, when some user leaves the group, the corresponding position in the rekeying message is omitted.

In the last case, when a new user U_{n+1} joins the group, if n is odd, then U_c adds the element $\phi(\widetilde{s_c}, C_n^n)$ and sends the following to the new user:

$$\{\phi(\widetilde{s_c}, C_n^1), \phi(\widetilde{s_c}, C_n^2), ..., \phi(\widetilde{s_c}, C_n^{c-1}), C_n^c, \phi(\widetilde{s_c}, C_n^{c+1}), ..., \phi(\widetilde{s_c}, C_n^{n-1}), \phi(\widetilde{s_c}, C_n^n)\}.$$

If n is even, U_c adds the element $\phi(\widetilde{s_c}^*, C_n^n)$ and sends to U_{n+1} the following:

$$\{\phi(\widetilde{s_c}^*, C_n^1), \phi(\widetilde{s_c}^*, C_n^2), ..., \phi(\widetilde{s_c}^*, C_n^{c-1}), C_n^c, \phi(\widetilde{s_c}^*, C_n^{c+1}), ..., \phi(\widetilde{s_c}^*, C_n^{n-1}), \phi(\widetilde{s_c}^*, C_n^n)\}.$$

Finally, user U_{n+1} proceeds to step 3 of the group key protocol and sends the other users the information to obtain the shared key using their private keys.

4 Secure Group Key Management

In this section, we show that the extra information sent in the protocol of n users does not implies aditional information leakage for an attacker respect to the 2-users case. For this purpouse, we define the following random variables, choosing X randomly from $(R_1 \times A_2)^n$:

$$A_n = \Big(view(n, X), y\Big), \text{ for } y \in R \text{ randomly chosen.}$$

$$D_n = \begin{cases} \Big(view(n, X), \phi(s_n^* s_{n-1} s_{n-2}^* ... s_3 s_2^* s_1, h), h)\Big), & \text{if } n \text{ is even.} \\ \Big(view(n, X), \phi(s_n s_{n-1}^* s_{n-2} ... s_3 s_2^* s_1, h)\Big), & \text{if } n \text{ is odd.} \end{cases}$$

where

- $view(n, X)$:= the ordered set of all $\phi(s_{i_1} s_{i_2}^* s_{i_3} ... s_{m-2}^* s_{m-1} s_m^*, h)$, for all proper subsets $\{i_1, ..., i_m\}$ of $\{1, ..., n\}$; $m \in \{1, ..., n-1\}$.

when n is even, and

- $view(n, X)$:= the ordered set of all $\phi(s_{i_1} s^*_{i_2} s_{i_3} ... s_{m-2} s^*_{m-1} s_m, h)$, for all proper subsets $\{i_1, ..., i_m\}$ of $\{1, ..., n\}$; $m \in \{1, ..., n-1\}$.

when n is odd.

Also note that $\phi(s^*_n s_{n-1} s^*_{n-2} ... s_3 s^*_2 s_1, h), h)$, or $\phi(s_n s^*_{n-1} s_{n-2} ... s_3 s^*_2 s_1, h)$, is the common secret key, is case n is even or odd respectively.

Let the relation $\sim$ be polynomial indistinguishability, as defined in [10]. In this context, it means that no polynomial-time algorithm can distinguish between a key and a random value with probability significantly greater than $\frac{1}{2}$.

Proposition 4.1. *The relation $\sim$ is an equivalence relation.*

A proof of this proposition can be found in [1]. Before we prove the main result, let us show that

Lemma 4.2. *We can write $view(n, \{s_1, s_2\} \cup X)$, with $X = \{s_3, ..., s_n\}$ as a permutation of*

$$V = \Big(view(n-1, \{s_1\} \cup X), \phi(s_n s^*_{n-1} ... s_2, h), view(n-1, \{s_2\} \cup X), \phi(s_n s^*_{n-1} s_{n-2} ... s^*_3 s_1, h), view(n-1, \{s^*_2 s_1\} \cup X)\Big)$$

when n is even, and as a permutation of

$$V = \Big(view(n-1, \{s_1\} \cup X), \phi(s^*_n s_{n-1} s^*_{n-2} ... s_2, h), view(n-1, \{s_2\} \cup X), \phi(s^*_n s_{n-1} ... s^*_3 s_1, h), view(n-1, \{s_1 s^*_2\} \cup X)\Big)$$

when n is odd.

Proof

Now we show that both sets are equal. First, we prove that $view(n, \{s_1, s_2\} \cup X) \subset V$: Let an element $a \in view(n, \{s_1, s_2\} \cup X)$:

- If n is even:
  1. If a contains $s^*_2 s_1 (= s^*_1 s_2)$, then it belongs to $view(n-1, \{s^*_2 s_1\} \cup X) \subset V$.
  2. If a does not contain s_1 (or s^*_1),
     - but it contains all the remaining elements, $s^{(*)}_2, ..., s^{(*)}_n$, then it belongs to $\phi(s_n s^*_{n-1} ... s^*_3 s_2, h) \subset V$.
     - and if it does not contain all the remaining elements, then it belongs to $view(n-1, \{s_2\} \cup X) \subset V$.
  3. If a does not contain s_2 (or s^*_2),
     - but it contains all the remaining elements, $s^{(*)}_1, s^{(*)}_3, ..., s^{(*)}_n$, then it belongs to $\phi(s_n s^*_{n-1} ... s^*_3 s_1, h) \subset V$.
     - and if it does not contain all the remaining elements, then it belongs to $view(n-1, \{s_1\} \cup X) \subset V$.
  4. Finally, if a does not contain s_1 neither s_2, it belongs to any of the following $view(n-1, \{s_1\} \cup X), view(n-1, \{s_2\} \cup X), view(n-1, \{s_1 s^*_2\} \subset V$.

- If n is odd:

  1. If a contains $s_2^* s_1 (= s_1^* s_2)$, then it belongs to $view(n-1, \{s_2^* s_1\} \cup X) \subset V$.
  2. If a does not contain s_1 (or s_1^*),
     - but it contains all the remaining elements, $s_2^{(*)}, ..., s_n^{(*)}$, then it belongs to $\phi(s_n^* s_{n-1} ... s_3^* s_2, h) \subset V$.
     - and if it does not contain all the remaining elements, then it belongs to $view(n-1, \{s_2\} \cup X) \subset V$.
  3. If a does not contain s_2 (or s_2^*),
     - but it contains all the remaining elements, $s_1^{(*)}, s_3^{(*)}, ..., s_n^{(*)}$, then it belongs to $\phi(s_n^* s_{n-1} ... s_3^* s_1, h) \subset V$.
     - and if it does not contain all the remaining elements, then it belongs to $view(n-1, \{s_1\} \cup X) \subset V$.
  4. Finally, if a does not contain s_1 neither s_2, it belongs to any of the following $view(n-1, \{s_1\} \cup X), view(n-1, \{s_2\} \cup X), view(n-1, \{s_1 s_2^*\} \subset V$.

The reverse inclusion, $V \subset view(n, \{s_1, s_2\})$ is true since all the elements in V belong to $view(n, \{s_1, s_2\} \cup X)$ by definition.

■

Let us finally prove, following the idea of [10], that if the 2-users underlying decisional problem is hard, then the n-users is hard as well, or equivalently:

Theorem 4.3. *For any $n > 2$, $A_2 \sim D_2$ implies that $A_n \sim D_n$.*

Proof

We show this is true by induction on n. Assume that $A_2 \sim D_2$ and $A_i \sim D_i$, $i \in \{3, ..., n-1\}$. Thus, we have to show that $A_n \sim D_n$. We define the random variables B_n, C_n, and show that $A_n \sim B_n \sim C_n \sim D_n$, and since $\sim$ is a equivalence relation, by transitivity, this implies that $A_n \sim D_n$.

We split the proof in two cases:

a) Assume n is even:

   We redefine A_n, D_n using Lemma 4.2, and define B_n, C_n as follows:

   - $A_n = \Big(view(n-1, \{s_1\} \cup X), \phi(s_n s_{n-1}^* ... s_2, h), view(n-1, \{s_2\} \cup X), \phi(s_n s_{n-1}^* s_{n-2} ... s_3^* s_1, h), view(n-1, \{s_2^* s_1\} \cup X), y \Big)$
   - $B_n = \Big(view(n-1, \{s_1\} \cup X), \phi(s_n s_{n-1}^* ... s_2, h), view(n-1, \{s_2\} \cup X), \phi(s_n s_{n-1}^* s_{n-2} ... s_3^* s_1, h), view(n-1, \{c\} \cup X), y \Big)$
   - $C_n = \Big(view(n-1, \{s_1\} \cup X), \phi(s_n s_{n-1}^* ... s_2, h), view(n-1, \{s_2\} \cup X), \phi(s_n s_{n-1}^* s_{n-2} ... s_3^* s_1, h), view(n-1, \{c\} \cup X), \phi(s_n^* s_{n-1} ... s_4^* s_3 c, h) \Big)$
   - $D_n = \Big(view(n-1, \{s_1\} \cup X), \phi(s_n s_{n-1}^* ... s_2, h), view(n-1, \{s_2\} \cup X), \phi(s_n s_{n-1}^* s_{n-2} ... s_3^* s_1, h), view(n-1, \{s_2^* s_1\} \cup X), \phi(s_n^* s_{n-1} ... s_4^* s_3 s_2^* s_1, h) \Big)$

choosing $s_1, s_2 \in R_1 \times A_2$, $c \in R_1 \times A_1$; and $X \in (R_1 \times A_2)^{n-2}, y \in R_1 h A_1$ randomly. Note that only the last two components vary.

$\underline{A_2 \sim D_2 \Longrightarrow A_n \sim B_n}$

Suppose, for the sake of contradiction, that an adversary Eve distinguishes A_n and B_n. We produce an instance of $A_n \not\sim B_n$ for Eve

$$\begin{aligned}
A_n &= \Big(view(n-1, \{s_1\} \cup X), \phi(s_n s_{n-1}^* ... s_2, h), view(n-1, \{s_2\} \cup X), \phi(s_n s_{n-1}^* s_{n-2} ... s_3^* s_1, h), \\
&\quad view(n-1, \{s_2^* s_1\} \cup X), y \Big) \\
&= \Big(\boldsymbol{g_1 h k_1}, ..., g_n g_{n-1} ... g_4 g_3 h k_3 k_4^* ... k_{n-1} k_n^*, g_n g_{n-1} ... g_3 g_1 h k_1 k_3^* k_4 ... k_{n-1}^* k_n, \\
&\quad \boldsymbol{g_2 h k_2}, ..., g_{n-1} ... g_3 g_2 h k_2 k_3^* ... k_{n-2}^* k_{n-1}, g_n g_{n-1} ... g_3 g_2 h k_1 k_2^* k_4 ... k_{n-1}^* k_n, \\
&\quad \boldsymbol{g_2 g_1 h k_1 k_2^*}, ..., g_{n-1} g_{n-2} ... g_3 (g_2 g_1) h (k_1 k_2^*) k_3 ... k_{n-2}^* k_{n-1}, y \}
\end{aligned}$$

$$\begin{aligned}
B_n &= \Big(view(n-1, \{s_1\} \cup X), \phi(s_n s_{n-1}^* ... s_2, h), view(n-1, \{s_2\} \cup X), \phi(s_n s_{n-1}^* s_{n-2} ... s_3^* s_1, h), \\
&\quad view(n-1, \{c\} \cup X), y \Big) \\
&= \Big(\boldsymbol{g_1 h k_1}, ..., g_n g_{n-1} ... g_4 g_3 h k_3 k_4^* ... k_{n-1} k_n^*, g_n g_{n-1} ... g_3 g_1 h k_1 k_3^* k_4 ... k_{n-1}^* k_n, \\
&\quad \boldsymbol{g_2 h k_2}, ..., g_{n-1} ... g_3 g_2 h k_2 k_3^* ... k_{n-2}^* k_{n-1}, g_n g_{n-1} ... g_3 g_2 h k_1 k_2^* k_4 ... k_{n-1}^* k_n, \\
&\quad \boldsymbol{c_1 h c_2}, ..., g_{n-1} g_{n-2} ... g_3 (c_1) h (c_2) k_3 ... k_{n-2}^* k_{n-1}, y \}
\end{aligned}$$

if Eve distinguishes A_n and B_n, then in particular, she distinguishes $g_2 g_1 h k_1 k_2^*$ from $c_1 h c_2$ (given $g_1 h k_1$ and $g_2 h k_2$), which means that she distinguishes

$$\begin{aligned}
A_2 &= \Big(view(2, \{s_1, s_2\}), y \Big) \\
&= (g_1 h k_1, g_2 h k_2, y)
\end{aligned}$$

$$\begin{aligned}
D_2 &= \Big(view(2, \{s_1, s_2\}), \phi(s_2^* s_1, h) \Big) \\
&= (g_1 h k_1, g_2 h k_2, g_2 g_1 h k_1 k_2^*)
\end{aligned}$$

which contradicts our hypothesis.

$\underline{A_{n-2} \sim D_{n-2} \Longrightarrow B_n \sim C_n}$

Suppose towards the sake of contradiction that an adversary Eve distinguishes B_n and C_n. We produce and instance of $B_n \not\sim C_n$ for Eve

$$
\begin{aligned}
B_n &= \Big(view(n-1,\{s_1\}\cup X), \phi(s_n^* s_{n-1}...s_3^* s_2, h), view(n-1,\{s_2\}\cup X), \phi(s_n^* s_{n-1}...s_3^* s_1, h),\\
&\quad view(n-1,\{c\}\cup X), y\Big)\\
&= \Big(g_1 h k_1, ..., g_n g_{n-1}...g_4 g_3 h k_3 k_4^* ... k_{n-1} k_n^*, g_n g_{n-1}...g_3 g_1 h k_1 k_3^* k_4 ... k_{n-1}^* k_n,\\
&\quad g_2 h k_2, ..., g_{n-1}...g_3 g_2 h k_2 k_3^* ... k_{n-2}^* k_{n-1}, g_n g_{n-1}...g_3 g_2 h k_1 k_2^* k_4 ... k_{n-1}^* k_n,\\
&\quad \boldsymbol{c_1 h c_2, ..., g_{n-1}...g_5 g_4 (g_3 c_1) h (c_2 k_3) k_4^* k_5 ... k_{n-2} k_{n-1}^*, y}\}
\end{aligned}
$$

$$
\begin{aligned}
C_n &= \Big(view(n-1,\{s_1\}\cup X), \phi(s_n^* s_{n-1}...s_3^* s_2, h), view(n-1,\{s_2\}\cup X), \phi(s_n s_{n-1}^*...s_3^* s_1, h),\\
&\quad view(n-1,\{c\}\cup X), \phi(s_n s_{n-1}^* ... s_5 s_4^* s_3 c, h)\Big)\\
&= \Big(g_1 h k_1, ..., g_n g_{n-1}...g_4 g_3 h k_3 k_4^* ... k_{n-1} k_n^*, g_n g_{n-1}...g_3 g_1 h k_1 k_3^* k_4 ... k_{n-1}^* k_n,\\
&\quad g_2 h k_2, ..., g_{n-1}...g_3 g_2 h k_2 k_3^* ... k_{n-2}^* k_{n-1}, g_n g_{n-1}...g_3 g_2 h k_1 k_2^* k_4 ... k_{n-1}^* k_n,\\
&\quad \boldsymbol{c_1 h c_2, ..., g_{n-1}...g_5 g_4 (g_3 c_1) h (c_2 k_3) k_4^* k_5 ... k_{n-2} k_{n-1}^*, g_n ... g_4 (g_3 c_1) h (c_2 k_3) k_4^* k_5 ... k_n}\}
\end{aligned}
$$

if Eve distinguishes B_n and C_n in polynomial time, in particular, she distinguishes y and $\phi(s_n^* s_{n-1}...s_4^*(s_3 c), h)$ (given $view(n-1,\{c\}\cup X)$). Let $\Big((view(n-2,\{cs_3, s_4, s_5, ..., s_{n-1}, s_n\}), y\Big)$ be an instance of A_{n-2}, D_{n-2}:

$$
\begin{aligned}
A_{n-2} &= \Big((view(n-2,\{s_3 c, s_4, s_5, ..., s_{n-1}, s_n\}), y\Big)\\
&= ((g_3 c_1) h (c_2 k_3), g_4 h k_4, ..., g_n h k_n, g_4 (g_3 c_1) h (c_2 k_3) k_4^* ..., g_n (g_3 c_1) h (c_2 k_3) k_n^*,\\
&\quad g_5 g_4 (g_3 c_1) h (c_2 k_3) k_4^* k_5, ..., g_n g_{n-1} ... g_4 g_3 h k_3 k_4^* ... k_{n-1} k_n, y)
\end{aligned}
$$

$$
\begin{aligned}
D_{n-2} &= \Big(view(n-2,\{s_3 c, s_4, s_5, ..., s_{n-1}, s_n\}), \phi(s_n^* s_{n-1}...s_4^*(s_3 c), h)\Big)\\
&= ((g_3 c_1) h (c_2 k_3), g_4 h k_4, ..., g_n h k_n, g_4 (g_3 c_1) h (c_2 k_3) k_4^* ..., g_n (g_3 c_1) h (c_2 k_3) k_n^*,\\
&\quad g_5 g_4 (g_3 c_1) h (c_2 k_3) k_4^* k_5, ..., g_n g_{n-1} ... g_5 g_4 h k_4 k_5^* ... k_{n-1} k_n, g_n g_{n-1} ... g_4 (g_3 c_1) h (c_2 k_3) k_4^* ... k_{n-1} k_n)
\end{aligned}
$$

since Eve can distinguish y and $\phi(s_n^* s_{n-1}...s_4^*(s_3 c), h)$ given $view(n-1,\{c\}\cup X)$, then in particular

she distinguishes y and $\phi(s_n^* s_{n-1}...s_4^*(s_3 c), h)$ given $view(n-2,\{s_3 c, s_4, s_5, ..., s_{n-1}, s_n\}) \subset view(n-1,\{c\}\cup X)$, and this means $A_{n-2} \not\sim D_{n-2}$, but this contradicts our hypothesis.

$\underline{A_2 \sim D_2 \Longrightarrow C_n \sim D_n}$

Suppose, for the sake of contradiction, that an adversary Eve distinguishes C_n and D_n. We produce and instance of $C_n \not\sim D_n$ for Eve

$$\begin{aligned}
C_n &= \Big(view(n-1,\{s_1\}\cup X), \phi(s_n s_{n-1}^*...s_2,h), view(n-1,\{s_2\}\cup X), \phi(s_n s_{n-1}^* s_{n-2}...s_3^* s_1,h), \\
&\quad view(n-1,\{c\}\cup X), \phi(s_n^* s_{n-1}...s_4^* s_3 c,h)\Big) \\
&= \Big(g_1hk_1, ..., g_ng_{n-1}...g_4g_3hk_3k_4^*...k_{n-1}k_n^*, g_ng_{n-1}...g_3g_1hk_1k_3^*k_4...k_{n-1}^*k_n, \\
&\quad g_2hk_2, ..., g_{n-1}...g_3g_2hk_2k_3^*...k_{n-2}^*k_{n-1}, g_ng_{n-1}...g_3g_2hk_1k_2^*k_4...k_{n-1}^*k_n, \\
&\quad \boldsymbol{c_1hc_2}, ..., g_{n-1}g_{n-2}...g_3c_1hc_2k_3...k_{n-2}^*k_{n-1}, g_ng_{n-1}...g_4g_3c_1hc_2k_3k_4^*...k_{n-1}k_n\}
\end{aligned}$$

$$\begin{aligned}
D_n &= \Big(view(n-1,\{s_1\}\cup X), K(n-1,\{s_1\}\cup X), view(n-1,\{s_2\}\cup X), K(n-1,\{s_2\}\cup X), \\
&\quad view(n-1,\{s_2^* s_1\}\cup X), \phi(s_n^* s_{n-1}...s_4^* s_3 s_2^* s_1,h)\Big) \\
&= \Big(g_1hk_1, ..., g_ng_{n-1}...g_4g_3hk_3k_4^*...k_{n-1}k_n^*, g_ng_{n-1}...g_3g_1hk_1k_3^*k_4...k_{n-1}^*k_n, \\
&\quad g_2hk_2, ..., g_{n-1}...g_3g_2hk_2k_3^*...k_{n-2}^*k_{n-1}, g_ng_{n-1}...g_3g_2hk_1k_2^*k_4...k_{n-1}^*k_n, \\
&\quad \boldsymbol{g_2g_1hk_1k_2^*}, ..., g_{n-1}g_{n-2}...g_3(g_2g_1)h(k_1k_2^*)k_3...k_{n-2}^*k_{n-1}, g_ng_{n-1}...g_3(g_2g_1)h(k_1k_2^*)k_3...k_{n-1}k_n^*\}
\end{aligned}$$

as in the first case, if Eve distinguishes A_n and B_n, then in particular, she distinguishes $g_2g_1hk_1k_2^*$ from c_1hc_2 (given g_1hk_1 and g_2hk_2), which means that she distinguishes

$$\begin{aligned}
A_2 &= \Big(view(2,\{s_1,s_2\}), y\Big) \\
&= (g_1hk_1, g_2hk_2, y)
\end{aligned}$$

$$\begin{aligned}
D_2 &= \Big(view(2,\{s_1,s_2\}), \phi(s_2^* s_1,h)\Big) \\
&= (g_1hk_1, g_2hk_2, g_2g_1hk_1k_2^*)
\end{aligned}$$

which contradicts our hypothesis.

b) Similarly, if n is odd:

We redefine A_n, D_n using Lemma 4.2, and define B_n, C_n as follows:

- $A_n = \Big(view(n-1,\{s_1\}\cup X), \phi(s_n^* s_{n-1}...s_3^* s_2,h), view(n-1,\{s_2\}\cup X), \phi(s_n^* s_{n-1}...s_3^* s_1,h),$ $view(n-1,\{s_2^* s_1\}\cup X), y\Big)$
- $B_n = \Big(view(n-1,\{s_1\}\cup X), \phi(s_n^* s_{n-1}...s_2,h), view(n-1,\{s_2\}\cup X), \phi(s_n^* s_{n-1}...s_3^* s_1,h),$ $view(n-1,\{c\}\cup X), y\Big)$
- $C_n = \Big(view(n-1,\{s_1\}\cup X), \phi(s_n^* s_{n-1}...s_2,h), view(n-1,\{s_2\}\cup X), \phi(s_n^* s_{n-1}...s_3^* s_1,h),$ $view(n-1,\{c\}\cup X), \phi(s_n s_{n-1}^*...s_5 s_4^* s_3 c,h)\Big)$
- $D_n = \Big(view(n-1,\{s_1\}\cup X), \phi(s_n^* s_{n-1}...s_2,h), view(n-1,\{s_2\}\cup X), \phi(s_n^* s_{n-1}...s_3^* s_1,h),$ $view(n-1,\{s_2^* s_1\}\cup X), \phi(s_n s_{n-1}^*...s_5 s_4^* s_3 s_2^* s_1,h)\Big)$

choosing $s_1, s_2 \in R_1 \times A_2$, $c \in R_1 \times A_1$; and $X \in (R_1 \times A_2)^{n-2}, y \in R_1 h A_2$ randomly.

$\underline{A_2 \sim D_2 \Longrightarrow A_n \sim B_n}$.

Suppose towards the sake of contradiction that an adversary Eve distinguishes A_n and B_n. We produce an instance of $A_n \not\sim B_n$ for Eve

$$\begin{aligned}
A_n &= \Big(view(n-1, \{s_1\} \cup X), \phi(s_n^* s_{n-1} ... s_2, h), view(n-1, \{s_2\} \cup X), \phi(s_n^* s_{n-1} ... s_3^* s_1, h), \\
&\quad view(n-1, \{s_2^* s_1\} \cup X), y\Big) \\
&= \Big(\boldsymbol{g_1 h k_1}, ..., g_n g_{n-1} ... g_4 g_3 h k_3 k_4^* ... k_{n-1} k_n^*, g_n g_{n-1} ... g_3 g_1 h k_1 k_3^* k_4 ... k_{n-1} k_n^*, \\
&\quad \boldsymbol{g_2 h k_2}, ..., g_{n-1} ... g_3 g_2 h k_2 k_3^* ... k_{n-2}^* k_{n-1}, g_n g_{n-1} ... g_3 g_2 h k_1 k_2^* k_4 ... k_{n-1} k_n^*, \\
&\quad \boldsymbol{g_2 g_1 h k_1 k_2^*}, ..., g_{n-1} g_{n-2} ... g_3 (g_2 g_1) h (k_1 k_2^*) k_3 ... k_{n-2} k_{n-1}^*, y\}
\end{aligned}$$

$$\begin{aligned}
B_n &= \Big(view(n-1, \{s_1\} \cup X), \phi(s_n^* s_{n-1} ... s_2, h), view(n-1, \{s_2\} \cup X), \phi(s_n^* s_{n-1} ... s_3^* s_1, h), \\
&\quad view(n-1, \{c\} \cup X), y\Big) \\
&= \Big(\boldsymbol{g_1 h k_1}, ..., g_n g_{n-1} ... g_4 g_3 h k_3 k_4^* ... k_{n-1} k_n^*, g_n g_{n-1} ... g_3 g_1 h k_1 k_3^* k_4 ... k_{n-1} k_n^*, \\
&\quad \boldsymbol{g_2 h k_2}, ..., g_{n-1} ... g_3 g_2 h k_2 k_3^* ... k_{n-2}^* k_{n-1}, g_n g_{n-1} ... g_3 g_2 h k_1 k_2^* k_4 ... k_{n-1} k_n^*, \\
&\quad \boldsymbol{c_1 h c_2}, ..., g_{n-1} g_{n-2} ... g_3 (c_1) h (c_2) k_3 ... k_{n-2} k_{n-1}^*, y\}
\end{aligned}$$

if Eve distinguishes A_n and B_n, then in particular, she distinguishes $g_2 g_1 h k_1 k_2^*$ from $c_1 h c_2$ (given $g_1 h k_1$ and $g_2 h k_2$), which means that she distinguishes

$$\begin{aligned}
A_2 &= \Big(view(2, \{s_1, s_2\}), y\Big) \\
&= (g_1 h k_1, g_2 h k_2, y)
\end{aligned}$$

$$\begin{aligned}
D_2 &= \Big(view(2, \{s_1, s_2\}), \phi(s_2^* s_1, h)\Big) \\
&= (g_1 h k_1, g_2 h k_2, g_2 g_1 h k_1 k_2^*)
\end{aligned}$$

which contradicts our hypothesis.

$\underline{A_{n-2} \sim D_{n-2} \Longrightarrow B_n \sim C_n}$.

Suppose, for the sake of contradiction, that an adversary Eve distinguishes B_n and C_n. We produce and instance of $B_n \not\sim C_n$ for Eve

$$
\begin{aligned}
B_n &= \Big(view(n-1,\{s_1\}\cup X), \phi(s_n s_{n-1}^*...s_2,h), view(n-1,\{s_2\}\cup X), \phi(s_n s_{n-1}^* s_{n-2}...s_3^* s_1,h),\\
&\quad view(n-1,\{c\}\cup X), y\Big)\\
&= \Big(g_1hk_1,...,g_ng_{n-1}...g_4g_3hk_3k_4^*...k_{n-1}k_n^*, g_ng_{n-1}...g_3g_1hk_1k_3^*k_4...k_{n-1}^*k_n,\\
&\quad g_2hk_2,...,g_{n-1}...g_3g_2hk_2k_3^*...k_{n-2}^*k_{n-1}, g_ng_{n-1}...g_3g_2hk_1k_2^*k_4...k_{n-1}^*k_n,\\
&\quad \boldsymbol{c_1hc_2,...,g_{n-1}...g_5g_4(g_3c_1)h(c_2k_3)k_4^*k_5...k_{n-2}^*k_{n-1}, y}\}
\end{aligned}
$$

$$
\begin{aligned}
C_n &= \Big(view(n-1,\{s_1\}\cup X), \phi(s_n s_{n-1}^*...s_2,h), view(n-1,\{s_2\}\cup X), \phi(s_n s_{n-1}^* s_{n-2}...s_3^* s_1,h),\\
&\quad view(n-1,\{c\}\cup X), \phi(s_n s_{n-1}^*...s_4^* s_3 c,h)\Big)\\
&= \Big(g_1hk_1,...,g_ng_{n-1}...g_4g_3hk_3k_4^*...k_{n-1}k_n^*, g_ng_{n-1}...g_3g_1hk_1k_3^*k_4...k_{n-1}^*k_n,\\
&\quad g_2hk_2,...,g_{n-1}...g_3g_2hk_2k_3^*...k_{n-2}^*k_{n-1}, g_ng_{n-1}...g_3g_2hk_1k_2^*k_4...k_{n-1}^*k_n,\\
&\quad \boldsymbol{c_1hc_2,...,g_{n-1}...g_5g_4(g_3c_1)h(c_2k_3)k_4^*k_5...k_{n-2}^*k_{n-1}, g_n...g_4(g_3c_1)h(c_2k_3)k_4^*k_5...k_n^*}\}
\end{aligned}
$$

if Eve distinguishes B_n and C_n in polynomial time, in particular, she distinguishes y and $\phi(s_n s_{n-1}^*...s_5 s_4^*(s_3c),h)$ (given $view(n-1,\{c\}\cup X)$). Let $\Big((view(n-2,\{cs_3,s_4,s_5,...,s_{n-1},s_n\}),y\Big)$ be an instance of A_{n-2}, D_{n-2}:

$$
\begin{aligned}
A_{n-2} &= \Big((view(n-2,\{s_3c,s_4,s_5,...,s_{n-1},s_n\}),y\Big)\\
&= ((g_3c_1)h(c_2k_3), g_4hk_4,...,g_nhk_n, g_4(g_3c_1)h(c_2k_3)k_4^*..., g_n(g_3c_1)h(c_2k_3)k_n^*,\\
&\quad g_5g_4(g_3c_1)h(c_2k_3)k_4^*k_5,...,g_ng_{n-1}...g_4g_3hk_3k_4^*...k_{n-1}^*k_n, y)
\end{aligned}
$$

$$
\begin{aligned}
D_{n-2} &= \Big(view(n-2,\{s_3c,s_4,s_5,...,s_{n-1},s_n\}), \phi(s_n s_{n-1}^*...s_5 s_4^*(s_3c),h)\Big)\\
&= ((g_3c_1)h(c_2k_3), g_4hk_4,...,g_nhk_n, g_4(g_3c_1)h(c_2k_3)k_4^*..., g_n(g_3c_1)h(c_2k_3)k_n^*,\\
&\quad g_5g_4(g_3c_1)h(c_2k_3)k_4^*k_5,...,g_ng_{n-1}...g_5g_4hk_4k_5^*...k_{n-1}^*k_n, g_ng_{n-1}...g_4(g_3c_1)h(c_2k_3)k_4^*...k_{n-1}^*k_n)
\end{aligned}
$$

since Eve can distinguish y and $\phi(s_n s_{n-1}^*...s_5 s_4^*(s_3c),h)$ given $view(n-1,\{c\}\cup X)$, then in parti-

cular she distinguishes y and $\phi(s_n^* s_{n-1}...s_4^*(s_3c),h)$ given $view(n-2,\{s_3c,s_4,s_5,...,s_{n-1},s_n\}) \subset view(n-1,\{c\}\cup X)$, and this means $A_{n-2} \not\sim D_{n-2}$, but this contradicts our hypothesis.

$\underline{A_2 \sim D_2 \Longrightarrow C_n \sim D_n}$.

Suppose towards the sake of contradiction that an adversary Eve distinguishes C_n and D_n.

We produce and instance of $C_n \not\sim D_n$ for Eve

$$\begin{aligned}
C_n &= \Big(view(n-1,\{s_1\}\cup X), \phi(s_n s_{n-1}^* ... s_2, h), view(n-1,\{s_2\}\cup X), \phi(s_n s_{n-1}^* s_{n-2} ... s_3^* s_1, h),\\
&\quad view(n-1,\{c\}\cup X), \phi(s_n s_{n-1}^* ... s_4^* s_3 c, h)\Big)\\
&= \Big(g_1hk_1, ..., g_n g_{n-1} ... g_4 g_3 hk_3 k_4^* ... k_{n-1} k_n^*, g_n g_{n-1} ... g_3 g_1 hk_1 k_3^* k_4 ... k_{n-1}^* k_n,\\
&\quad g_2hk_2, ..., g_{n-1} ... g_3 g_2 hk_2 k_3^* ... k_{n-2}^* k_{n-1}, g_n g_{n-1} ... g_3 g_2 hk_1 k_2^* k_4 ... k_{n-1}^* k_n,\\
&\quad \boldsymbol{c_1hc_2}, ..., g_{n-1} g_{n-2} ... g_3 c_1 hc_2 k_3 ... k_{n-2}^* k_{n-1}, g_n g_{n-1} ... g_4 g_3 c_1 hc_2 k_3 k_4^* ... k_{n-1} k_n\}
\end{aligned}$$

$$\begin{aligned}
D_n &= \Big(view(n-1,\{s_1\}\cup X), \phi(s_n s_{n-1}^* ... s_2, h), view(n-1,\{s_2\}\cup X), \phi(s_n s_{n-1}^* s_{n-2} ... s_3^* s_1, h),\\
&\quad view(n-1,\{s_2^* s_1\}\cup X), \phi(s_n s_{n-1}^* ... s_4^* s_3 s_2^* s_1, h)\Big)\\
&= \Big(g_1hk_1, ..., g_n g_{n-1} ... g_4 g_3 hk_3 k_4^* ... k_{n-1} k_n^*, g_n g_{n-1} ... g_3 g_1 hk_1 k_3^* k_4 ... k_{n-1}^* k_n,\\
&\quad g_2hk_2, ..., g_{n-1} ... g_3 g_2 hk_2 k_3^* ... k_{n-2}^* k_{n-1}, g_n g_{n-1} ... g_3 g_2 hk_1 k_2^* k_4 ... k_{n-1}^* k_n,\\
&\quad \boldsymbol{g_2g_1hk_1k_2^*}, ..., g_{n-1} g_{n-2} ... g_3 (g_2g_1) h (k_1k_2^*) k_3 ... k_{n-2}^* k_{n-1}, g_n g_{n-1} ... g_3 (g_2g_1) h (k_1k_2^*) k_3 ... k_{n-1} k_n^*\}
\end{aligned}$$

as in the first case, if Eve distinguishes A_n and B_n, then in particular, she distinguishes $g_2g_1hk_1k_2^*$ from c_1hc_2 (given g_1hk_1 and g_2hk_2), which means that she distinguishes

$$\begin{aligned}
A_2 &= \Big(view(2,\{s_1,s_2\}), y\Big)\\
&= (g_1hk_1, g_2hk_2, y)
\end{aligned}$$

$$\begin{aligned}
D_2 &= \Big(view(2,\{s_1,s_2\}), \phi(s_2^* s_1, h)\Big)\\
&= (g_1hk_1, g_2hk_2, g_2g_1hk_1k_2^*)
\end{aligned}$$

which contradicts our hypothesis.

■

So in the Initial Key Agreement the n-users underlying decisional problem is as hard as the 2-users decisional problem. This is also true in the Auxiliary Key Agreement. We can say the protocol provides on forward and backward security, i.e. any former or future users cannot distinguish future or past distributed keys, as it is shown in the following result.

Corollary 4.4. *The AKA provides on forward and backward security.*

Proof

Let Eve be a powerful adversary, that knows all the information of a past user or a future user. She would know a subset of $view(k,\varepsilon)$, where k is the number of current users, and ε the secret keys.

In the first case, when the members of the group stay the same, note that the key update adds a new secret key (and we consider it as a new user). Then we substitute n with $k = n+1$, $\phi(s_n^* s_{n-1} ... s_4^* s_3 s_2^* s_1, h)$ (or $\phi(s_n s_{n-1}^* ... s_3 s_2^* s_1, h)$) with $\phi(\widetilde{s_c} s_n^* s_{n-1} ... s_3 s_2^* s_1, h)$ (resp. $\phi(\widetilde{s_c}^* s_n s_{n-1}^* ... s_3 s_2^* s_1, h)$) if n is even (if n is odd), and X with $\varepsilon = \{s_1, s_2, ..., s_{c-1}, s_c, s_{c+1}, ..., s_{n-1}, s_n, s_c'\}$ in Theorem 4.3. It follows that

$$A_k = \Big(view(k,\varepsilon), y\Big), \text{ for } y \in R \text{ randomly chosen.}$$

$$D_k = \begin{cases} \Big(view(k,\varepsilon), \phi(\widetilde{s_c} s_n^* s_{n-1} ... s_3 s_2^* s_1, h)\Big), \text{ if } k \text{ is odd.} \\ \Big(view(k,\varepsilon), \phi(\widetilde{s_c}^* s_n s_{n-1}^* ... s_3 s_2^* s_1, h))\Big), \text{ if } k \text{ is even.} \end{cases}$$

and it still verifies that if $A_2 \sim D_2$, then $A_k \sim D_k$.

When a user leaves, the key update also adds a new secret key, so we replace n with $k = n+1$ (the user left, but we suppose that Eve had access to the communications before that happened, and that private key is still part of the common secret key). The rest is the same, so we get again the first case, and the AKA benefits form the same security benefits in this case.

When a new users joins the group, we need to replace $k = n+2$ (the new secret key and the key update), $\phi(s_n^* s_{n-1} ... s_4^* s_3 s_2^* s_1, h)$ (or $\phi(s_n s_{n-1}^* ... s_3 s_2^* s_1, h)$) with $\phi(s_{n+1}^* \widetilde{s_c} s_n^* s_{n-1} ... s_3 s_2^* s_1, h)$ (resp. $\phi(s_{n+1} \widetilde{s_c}^* s_n s_{n-1}^* ... s_3 s_2^* s_1, h))$ if n is even (if n is odd), and X with $\varepsilon = \{s_1, s_2, ..., s_{n-1}, s_n, s_{n+1}, s_c'\}$ in Theorem 4.3. It follows that

$$A_k = \Big(view(k,\varepsilon), y\Big), \text{ for } y \in R \text{ randomly chosen.}$$

$$D_k = \begin{cases} \Big(view(k,\varepsilon), \phi(s_{n+1}^* \widetilde{s_c} s_n^* s_{n-1} ... s_3 s_2^* s_1, h)\Big), \text{ if } k \text{ is even.} \\ \Big(view(k,\varepsilon), \phi(s_{n+1} \widetilde{s_c}^* s_n s_{n-1}^* ... s_3 s_2^* s_1, h))\Big), \text{ if } k \text{ is odd.} \end{cases}$$

and it still verifies that if $A_2 \sim D_2$, then $A_k \sim D_k$, so the Auxiliary Key Agreement benefits from the same security properties.

■

Note that we could also consider D_k as

$$D_k = \begin{cases} \Big(view(k,\varepsilon), \phi(\widetilde{s_c}, K_p))\Big), \text{ if } k \text{ is odd.} \\ \Big(view(k,\varepsilon), \phi(\widetilde{s_c}^*, K_p))\Big), \text{ if } k \text{ is even.} \end{cases}$$

where K_p would be the previous key, when the number of users stay the same or someone left, and

$$D_k = \begin{cases} \Big(view(k,\varepsilon), \phi(s_{n+1}^* \widetilde{s_c}, K_p))\Big), \text{ if } k \text{ is even.} \\ \Big(view(k,\varepsilon), \phi(s_{n+1} \widetilde{s_c}^*, K_p))\Big), \text{ if } k \text{ is odd.} \end{cases}$$

when a new user joins the group.

Also note that in the key refresh, we consider $k = n+1$ in the first two cases, but the set of secret keys are $\{s_1, s_2, ..., s_{c-1}, \widetilde{s_c}^* s_c, s_{c+1}, ..., s_{n-1}, s_n\}$ when n is odd, and $\{s_1, s_2, ..., s_{c-1}, \widetilde{s_c} s_c^*, s_{c+1}, ..., s_n\}$ when n is even, i.e. the number of stored keys stay the same, and the private key of the user U_c is $\widetilde{s_c^*} s_c$ or $\widetilde{s_c} s_c^*$ depending on whether the number of users is even or odd. Finally when $k = n+2$, the set of secret keys has just one new key, from the new user U_{n+1}, so it is $\{s_1, s_2, ..., s_{c-1}, \widetilde{s_c}^* s_c, s_{c+1}, ..., s_{n-1}, s_n, s_{n+1}\}$ when n is odd, and $\{s_1, s_2, ..., s_{c-1}, \widetilde{s_c} s_c^*, s_{c+1}, ..., s_n, s_{n+1}\}$ when n is even

References

[1] Barak, B. *Computational indistinguishability, pseudorandom generators*, Lecture Notes, Princeton University, https://www.cs.princeton.edu/courses/archive/fall07/cos433/lec4.pdf (2007).

[2] Diffie, W.; Hellman, M. *New Directions in Cryptography*, IEEE Transactions on Information Theory, IT-22, n. 6, 644-654 (1976).

[3] Eftekhari, M. *A Diffie-Hellman key exchange protocol using matrices over group rings*, Groups Complex. Cryptol., vol. 4, n. 1, 167-176 (2012).

[4] Gómez Olvera, M.D.; López Ramos, J.A.; Torrecillas Jover, B. *Public Key Protocols over Twisted Dihedral Group Rings*, Symmetry, vol. 11, n. 8, 1019 (2019).

[5] Gupta I.; Pandey A.; Kant Dubey U. *A Key Exchange Protocol using Matrices over Group Rings*, Asian-European Journal of Mathematics, vol. 5, n. 5 (2018).

[6] Habeeb M.; Kahrobaei D.; Koupparis C.; Shpilrain V. *Public key exchange using semidirect product of (semi)groups*, Lecture Notes Comp. Sc., Springer, vol. 7954, 475-486 (2013).

[7] Kahrobaei D.; Koupparis C.; Shpilrain V. *Public key exchange using matrices over group rings*, Groups Complex. Cryptol., vol. 5, n. 1, 97-115 (2013).

[8] López Ramos, J.A.; Rosenthal, J.; Schipani, D.; Schnyder, R. *Group key management based on semigroup actions*, Journal of Algebra and its Applications, vol. 16, n. 8 (2017).

[9] Maze, G.; Monico, C.; Rosenthal J. *Public key cryptography based on semigroup actions*, Advances in Mathematics of Communications, vol. 1, n. 4, 489-507 (2007).

[10] Steiner, M.; Tsudik, G.; Waidner, M. *Key Agreement in Dynamic Peer Groups*. IEEE Trans. Parallel Distrib. Syst., vol. 11, 769-780 (2000).