SECURITY ANALYSIS OF SEAGKAP

1. INTRODUCTION

In this section, we prove the correctness of the protocol and provide a formal security proof in the Random Oracle Model. Authentication is proven secure under the EUF-CMA, and confidentiality is proven secure under the ROR-CPA. In addition, we conduct an informal security analysis on replay resistance, forward and backward secrecy.

2. FORMAL PROOF

2.1. Authentication Proof

Assumptions. We work over a cyclic group G of prime order q with generator G. All secret keys $sk_i \in \mathbb{Z}_q^*$ (hence invertible). The hash $H : \{0,1\}^* \to \mathbb{Z}_q$ is modeled as a random oracle. The reduction has access to a *single-signature Schnorr* signing oracle for the chosen target identity.

Lemma 1. Equivalence of adjacent double-signature verification and sequential verification.

Proof. Upon receiving the first-round messages from neighbors P_{i-1} and P_{i+1}, participant P_i computes

$$S_i = s_{i+1} - s_{i-1},$$

$$X_i = r_i\big(R_{i+1} - R_{i-1}\big),$$

$$Y_i = r_i\big(h_{i+1}R_{i+1} - h_{i-1}R_{i-1}\big).$$

Then P_i checks the following one-shot equation:

$$r_i S_i \cdot G \stackrel{?}{=} Y_i + X_i. \tag{1}$$

If verified sequentially, using $s_j = sk_j\, h_j + r_j \pmod q$ for $j \in \{i-1, i+1\}$, we obtain

$$r_i s_{i+1} \cdot G \;=\; r_i\, pub_{i+1}\, h_{i+1} \;+\; r_i R_{i+1}, \tag{2}$$

$$r_i s_{i-1} \cdot G \;=\; r_i\, pub_{i-1}\, h_{i-1} \;+\; r_i R_{i-1}. \tag{3}$$

Subtracting Eq. (3) from Eq. (2) yields

$$r_i(s_{i+1} - s_{i-1}) \cdot G = r_i\big(pub_{i+1}h_{i+1} - pub_{i-1}h_{i-1}\big) + r_i\big(R_{i+1} - R_{i-1}\big),$$

which is exactly Eq. (1). Hence, the one-shot verification is equivalent to the pair of sequential verifications. □

Definition 1. First-round message acceptability. If an honest party P_i, after checking Equation (1), proceeds to the second round without requesting retransmission, then P_i is said to *accept* its neighbors' first-round messages.

Theorem 1. EUF-CMA authenticity of first-round messages via reduction to *single* Schnorr. If a PPT adversary $\mathcal{A}$ forges a valid first-round message for some honest P_i with non-negligible probability, then there exists a reduction $\mathcal{F}$ that breaks the EUF-CMA security of the *single-signature* Schnorr scheme with non-negligible advantage.

Proof. Game 0. Real execution. We consider the real protocol execution, where $\mathcal{A}$ may intercept, modify, or inject first-round messages $m_j = (ID_j, R_j, s_j)$.

Game 1. Oracle setup and ROM programming. The reduction $\mathcal{F}$ samples a target identity P_{j^*} uniformly at random. All Schnorr signing queries for P_{j^*} are answered via the *single-signature* Schnorr signing oracle; all other parties are simulated honestly. Hash queries are answered using a ROM table.

Game 2. From adjacent acceptance to a fresh single-signature forgery. By Lemma 1, if $\mathcal{A}$ makes an honest neighbor P_{j+1} accept in the first round, then the two underlying *single* Schnorr verifications for the adjacent parties succeed. If *all* first-round signatures attributed to the target P_{j^*} had been obtained from the signing oracle, then $\mathcal{A}$ would not have produced any *new* valid signature for P_{j^*}. Therefore, successful acceptance implies the existence of at least one *fresh* component

$$m_{j^*}^* = (ID_{j^*}^*, R_{j^*}^*, s_{j^*}^*)$$

for P_{j^*} that was never returned by the oracle. Hence, $\mathcal{F}$ outputs $(ID_{j^*}^*, R_{j^*}^*, s_{j^*}^*)$ as a valid *single* Schnorr EUF-CMA forgery.

By a standard index-guessing argument and ROM programming bounds, we obtain

$$Adv_{\text{1st-auth}}^{\text{EUF-CMA}}(\mathcal{A}) \;\le\; n \cdot Adv_{\text{Schnorr}}^{\text{EUF-CMA}}(\mathcal{F}) + \frac{q_H + 1}{|G|} + negl(\lambda),$$

where n is the number of participants, q_H is the number of hash queries, and $|G|$ is the group order. Since the *single-signature* Schnorr scheme is EUF-CMA secure in the ROM, the success probability of $\mathcal{A}$ is negligible. □

Lemma 2. Equivalence of aggregated one-shot verification and sequential verification.

Proof. After receiving the second-round messages, participant P_i computes

$$T_{-i} = \sum_{j \neq i} T_j, \quad R_{-i} = \sum_{j \neq i} R_j, \quad W_{-i} = \sum_{j \neq i} H_j \cdot pub_j.$$

Then P_i checks

$$T_{-i} \cdot pub_i \stackrel{?}{=} sk_i \cdot (R_{-i} + W_{-i}). \tag{4}$$

Since $pub_i = sk_i \cdot G$ and $sk_i \in \mathbb{Z}_q^*$, Eq. (4) is equivalent to

$$T_{-i} \cdot G \stackrel{?}{=} R_{-i} + W_{-i}, \tag{5}$$

which matches the sum of per-user Schnorr verifications $t_j \cdot G \stackrel{?}{=} R_j + H_j \cdot pub_j$ for all $j \neq i$. Hence, the aggregated one-shot check holds if and only if all individual Schnorr checks hold. □

Theorem 2. EUF-CMA authenticity of second-round messages. If a PPT adversary $\mathcal{A}$ makes an honest P_i accept the second-round aggregate verification with non-negligible probability, then there exists a reduction $\mathcal{F}$ that breaks the EUF-CMA security of the *single-signature* Schnorr scheme with non-negligible advantage.

Proof. Game 0 Real execution. $\mathcal{A}$ interacts with honest parties and may intercept, modify, and inject second-round messages M_j.

Game 1. Oracle setup and ROM programming. The reduction $\mathcal{F}$ picks a target identity P_{j^*} uniformly at random. All Schnorr signing queries for P_{j^*} are answered via the *single-signature* Schnorr signing oracle; other parties are simulated honestly. Hash queries are answered using a ROM table.

Game 2. From aggregate acceptance to a fresh single-signature forgery. Suppose $\mathcal{A}$ succeeds in making some honest P_i accept the aggregated check Eq. (5). By Lemma 2, this implies that, for every included index $j \neq i$, the single-signature check $t_j \cdot G \stackrel{?}{=} R_j + H_j \cdot pub_j$ holds. If all such signatures for the target P_{j^*} had been obtained from the signing oracle, then $\mathcal{A}$ would not have produced any *new* valid signature for P_{j^*}. Therefore, successful aggregate acceptance implies the existence of at least one *fresh* component

$$(ID_{j^*}^*, X_{j^*}^*, R_{j^*}^*, t_{j^*}^*)$$

for P_{j^*} that was never returned by the signing oracle. Thus, $\mathcal{F}$ outputs $(ID_{j^*}^*, X_{j^*}^*, R_{j^*}^*, t_{j^*}^*)$ as a valid *single* Schnorr EUF-CMA forgery.

By a standard index-guessing argument and ROM programming bounds, we obtain

$$Adv_{\text{2nd-auth}}^{\text{EUF-CMA}}(\mathcal{A}) \;\leq\; n \cdot Adv_{\text{Schnorr}}^{\text{EUF-CMA}}(\mathcal{F}) + \frac{q_H + 1}{|G|} + negl(\lambda),$$

where n is the number of participants, q_H is the number of hash queries, and $|G|$ is the group order. Since the single-signature Schnorr scheme is EUF-CMA secure in the ROM, the success probability of $\mathcal{A}$ is negligible. □

2.2. Confidentiality Proof

Theorem 3. Session-key confidentiality under the CDH assumption in the ROR-CPA model. Under the Computational Diffie–Hellman (CDH) assumption, the proposed protocol achieves indistinguishability of the session key in the ROR-CPA model. Formally, for any PPT adversary $\mathcal{A}$,

$$\left|\Pr[\mathcal{A}^{\mathsf{Real}} = 1] - \Pr[\mathcal{A}^{\mathsf{Random}} = 1]\right| \leq \mathsf{negl}(\lambda).$$

CDH Assumption. Let G be an elliptic-curve group of prime order q with generator G. Given $(G, a \cdot G, b \cdot G)$ for unknown $a, b \in \mathbb{Z}_q^*$, no PPT adversary can compute $ab \cdot G$ with non-negligible probability. That is,

$$Adv_{\mathcal{A}}^{\mathsf{CDH}}(\lambda) = \Pr\big[\mathcal{A}(G, a \cdot G, b \cdot G) = ab \cdot G\big] \leq \mathsf{negl}(\lambda).$$

Proof. We prove by a sequence of games $G0$–$G3$.

Game $G0$. Real execution. All parties $\{P_1, P_2, \ldots, P_n\}$ execute the protocol honestly and derive the real group session key K. In the challenge phase, $\mathcal{A}$ is given either the real K or a random key. Let $p_0 = \Pr[b' = 1|S_0]$ denote the probability that $\mathcal{A}$ guesses correctly in $G0$.

Game $G1$. Replacing two ephemeral public keys. Replace the first-round messages R_{i-1}, R_i of parties P_{i-1} and P_i by two random group elements $R'_{i-1}, R'_i \in G$. Implicitly there exist unique $r'_{i-1}, r'_i \in \mathbb{Z}_q^*$ such that $R'_{i-1} = r'_{i-1} \cdot G$ and $R'_i = r'_i \cdot G$. The distribution of (R_{i-1}, R_i) and (R'_{i-1}, R'_i) is identical, so the session key K remains unchanged. Thus, $|p_0 - p_1| = 0$, where $p_1 = \Pr[b' = 1|S_1]$.

Game $G2$. Replacing the CDH term. In the shared secret K, the contribution of P_{i-1} and P_i involves a CDH term of the form $ab \cdot G$ (where $R'_{i-1} = b \cdot G$, $R'_i = a \cdot G$). The session key can be written as

$$K_i = nab \cdot G + \Theta,$$

$$\Theta = (n+1) \cdot X_i + (n+2) \cdot X_{i+1} + \ldots + X_{i-2},$$

where Θ collects all remaining terms independent of $ab \cdot G$. Since $ab \cdot G$ cannot be computed under the CDH assumption, we replace $nab \cdot G$ with a random group element $T \in G$, yielding

$$K'_i = T + \Theta.$$

If $\mathcal{A}$ can distinguish between K_i and K'_i, then the challenger can extract $ab \cdot G = n^{-1}T$, contradicting the CDH assumption. Hence, $|p_1 - p_2| \leq \mathsf{negl}(\lambda)$, where $p_2 = \Pr[b' = 1|S_2]$.

Game $G3$. All ephemeral keys replaced. Replace all parties' ephemeral public keys R_j by independent random elements $R'_j \in G$, without storing the corresponding secrets r_j. The group session key now becomes a uniformly random string K_{rand}. Since (R'_j) are identically distributed as $(r_j \cdot G)$ and the key is already replaced by random, we have $|p_2 - p_3| = 0$, where $p_3 = \Pr[b' = 1|S_3] = 1/2$.

Conclusion. By the triangle inequality,

$$|p_0 - 1/2| \leq |p_0 - p_1| + |p_1 - p_2| + |p_2 - p_3| \leq \mathsf{negl}(\lambda).$$

Therefore, under the CDH assumption, the proposed protocol achieves session-key confidentiality in the ROR-CPA. □

3. INFORMAL SECURITY ANALYSIS

3.1. Replay Resistance

Each round message embeds a timestamp as a core input, enforcing a strict validity window. If the timestamp deviates from the verifier's clock, verification fails. Moreover, because timestamps are bound to Schnorr signatures, only entities holding valid private keys can generate updated valid signatures with fresh timestamps. Since adversaries cannot alter signatures or generate new valid ones, replay attempts are effectively prevented.

3.2. Forward and Backward Secrecy

Session keys incorporate fresh randomness generated inside each party's TEE at every invocation. Due to the hardware isolation of TEEs, adversaries cannot extract or predict these random values. Even if one session key is compromised, past keys remain unrecoverable (forward secrecy) and future keys remain unpredictable (backward secrecy), ensuring lifecycle security of key agreement.

3.3. key consistency

In the proposed protocol, each participant derives the final session key locally using broadcast parameters and its own private randomness, eliminating the need for centralized coordination or a trusted third party. Signature verification in both rounds guarantees that all parties maintain a consistent view of R_i and X_i, while any forged or inconsistent intermediate values are detected and discarded during the verification phase. Using the following key computation formula, it is straightforward to prove that the protocol ensures correctness. Consequently, under honest execution, all honest parties compute the same session key, which guarantees both correctness and key consistency.

$$\begin{aligned}
K_i &= nr_i \cdot R_{i-1} + (n-1) \cdot X_i + (n-2) \cdot X_{i+1} \\
&\quad + \cdots + X_{i-2} \\
&= r_i \cdot R_{i-1} + (r_i \cdot R_{i-1} + X_i) \\
&\quad + (r_i \cdot R_{i-1} + X_i + X_{i+1}) + \cdots \\
&\quad + (r_i \cdot R_{i-1} + X_i + \cdots + X_{i-2}) \\
&= r_i \cdot R_{i-1} + (r_i \cdot R_{i-1} + r_i \cdot R_{i+1} \\
&\quad - r_i \cdot R_{i-1}) \\
&\quad + (r_i \cdot R_{i-1} + r_i \cdot R_{i+1} - r_i \cdot R_{i-1} \\
&\quad + r_{i+1} \cdot R_{i+2} - r_{i+1} \cdot R_i) + \cdots \\
&\quad + (r_i \cdot R_{i-1} + r_i \cdot R_{i+1} - r_i \cdot R_{i-1} \\
&\quad + r_{i+1} \cdot R_{i+2} - r_{i+1} \cdot R_i + \cdots \\
&\quad + r_{i-2} \cdot R_{i-1} - r_{i-2} \cdot R_{i-3}) \\
&= r_i \cdot R_{i-1} + r_i \cdot R_{i+1} + r_{i+1} \cdot R_{i+2} \\
&\quad + \cdots + r_{i-2} \cdot R_{i-1} \\
&= r_i \cdot R_{i+1} + r_{i+1} \cdot R_{i+2} + \cdots \\
&\quad + r_{i-2} \cdot R_{i-1} + r_{i-1} \cdot R_i
\end{aligned}$$

where $X_i = r_i \cdot (R_{i+1} - R_{i-1})$. The right-hand side is a cyclically invariant symmetric sum, and therefore it follows that $K_i = K_j$ for any i, j. By the deterministic nature of the key derivation function, it follows that $Key_i = Key_j$, that is, all honest parties derive the same session key.