Efficient and Secure Design of ID-3PAKA Protocol Using ECC

Daya Sagar Gupta
Department of Computer Science and Engineering
Rajiv Gandhi Institute of Petroleum Technology, Jais
Amethi - 229305 Utter Pradesh, India
Email: dayasagar.ism@gmail.com

Krittibas Parai
Department of Computer Science and Engineering
Siliguri Institute of Technology, West Bengal 734009, India
Department of Computer Science and Engineering
Indian Institute of Information Technology Kalyani
West Bengal 741235, India
krittibas.sit@gmail.com

Mohammad S. Obaidat
Fellow of IEEE and Fellow of SCS
Dean and Professor, College of Computing and Informatics
University of Sharjah, Sharjah, 27272, UAE
KAIST, University of Jordan, Amman 11942, Jordan
University of Science and Technology, Beijing 100083, China
Amity University, Noida, UP 201313, India
msobaidat@gmail.com

SK Hafizul Islam
Department of Computer Science and Engineering
Indian Institute of Information Technology Kalyani
West Bengal 741235, India
hafi786@gmail.com

2021 International Conference on Computer, Information and Telecommunication Systems (CITS) | 978-1-6654-4913-7/21/$31.00 ©2021 IEEE | DOI: 10.1109/CITS52676.2021.9618445

***Abstract*—In this modern era, the need for key agreement protocols equipped with high-speed Internet has become necessary in information/network security. An authenticated key agreement (AKA) protocol is a technique that negotiates a secret session key among various users in a network and is used in many scenarios like mutual authentication, data integrity, and many more. The current state-of-the-art include many AKA protocols that are either vulnerable to different attacks or incur massive communication and computational costs. This paper proposes an identity-based three-party authenticated key agreement (ID-3PAKA) protocol using elliptic curve cryptography (ECC). The proposed ID-3PAKA protocol generates a shared secret session key among three users. The security of the proposed ID-3PAKA protocol relies on the hardness assumption of the ECDLP and the CDH problems. The proposed ID-3PAKA protocol is safe against active and passive attacks, which is proved through the simulation of the OFMC and Cl-AtS back-ends of the AVISPA tool.**

***Index Terms*—Authenticated Key Agreement, Bilinear pairing, Elliptic curve, Identity-based cryptography.**

I. Introduction

The importance of key agreement protocols in network communication for security and privacy paradigms has become crucial nowadays [1], [2]. These key exchange protocols are used to securely transmit a confidential message by encryption using a symmetric-key encryption technique, which in turn uses a private key for both the cryptographic operations, encryption, and decryption. Because a symmetric-key encryption algorithm needs a common private key for their communication at both ends. A key exchange protocol mainly transfers a secret session key among a group of authentic parties. This negotiated session key works as the private key in symmetric key cryptography. Diffie and Hellman [3] firstly proposed the idea of a two-party key exchange protocol in the year 1976. Their protocol helps to compute a shared secret session key anonymously among two parties, using the hardness assumption of discrete logarithm problem (DLP). After that, many key exchange protocols are proposed [4], [5] in the literature. We studied these protocols and found that most of them are vulnerable to active and passive attacks.

A. Literature Review

In 1976, Steiner et al. [6] extended the Diffie-Hellman protocol for group communication. Barua et al. [7] designed an unauthenticated group key exchange protocol and an authenticated group key exchange protocol. Their protocols use the concept of Identity-based cryptography (IBC) [8]. The current literature is also equipped with various key exchange protocols using the idea of IBC. In 2021, Islam and Biswas [9] proposed an efficient and secure group key agreement protocol with the hardness assumption of ECDL problem. Tan [5] proposed an ID-3PAKA protocol which is secured under the Bilinear Diffie-Hellman (BDH) problem and DLP. In 2014, Han and Zhu [10] put forwarded an ID-based key exchange protocol for a multi-server environment. They analyzed its security in the random oracle model (ROM). Xiong et al. [11] put forwarded an ID-3PAKA protocol for message communication over an insecure network. The security of the protocol is based on the computational Diffie-Hellman (CDH) problem.

978-1-6654-4913-7/21/$31.00 ©2021 IEEE

They also validated the security claim through the simulation AVISPA (Automated Validation of Internet Security Protocols and Applications) tool. Gupta and Biswas [12] proposed a 2PAKA protocol and a 3PAKA protocol. Liu et al. [13] put forwarded an ID-2PAKA protocol for the distributed computing environments. They used the AVISPA tool for the security analysis. Wang et al. [14] designed an ID-3PAKA protocol based on threshold cryptography, and they proved its security based on the hardness assumption of the CDH problem.

B. Contributions

In this paper, we have presented a secure ID-3PAKA protocol based on the hardness assumption of the ECDL and, CDH problem. This protocol is based on elliptic curve cryptography (ECC). The proposed protocol also uses the properties of bilinear mapping. The proposed ID-3PAKA exhibits efficient communication and computational cost, and it may be used in any resource-constrained environment. We have simulated the proposed protocol in the OFMC (On-the-fly Model-Checker) and CL-based Model-Checker (CL-AtSe) back-ends of the AVISPA tool [15]. The simulation result shows that the proposed protocol is secure against active and passive attacks.

C. Paper Structure

Section II describes related mathematics and definitions which are useful for developing the proposed protocol. Section III describes the proposed ID-3PAKA protocol. Section IV discusses the correctness and security of the proposal. Section V analyzes the performance of the proposed ID-3PAKA protocol. Lastly, Section VI concludes the paper.

II. Backgrounds

Here, we present the related mathematics to be helpful to understand the proposed ID-3PAKA protocol.

A. Elliptic Curve Cryptography

Let an elliptic curve $E(\mathbb{Z}_q^*)$ over a field $\mathbb{Z}_q^*$ of prime order q. The equation of elliptic curve can be described as follows [16]:

$$y^2 \bmod q \equiv x^3 + ax + b \bmod q \tag{1}$$

In Eq. (1), the constants $a, b \in \mathbb{Z}_q^*$ and the condition $(4a^3 + 27b^2) \not\equiv 0 \bmod q$ holds true. The set $E(\mathbb{Z}_q^*$ of elliptic curve points together with an element $\mathcal{O}$, called *point at infinity*, forms an *additive cyclic group*. In this elliptic curve group, the point $\mathcal{O}$ serves as the *additive identity element*. Let two points on E are *M* and *N*. The addition *"M+N"* is also a point on E and can be calculated by the following rules:

- If *M* and *N* are two different point on the curve (1) with different *x* and *y* co-ordinates, then *"M+N"* will be an image of a point *P* through x-axis, i.e. ,-*P* and this point *P* is another point which is join of the lines *M* and *N*.
- If $M = N$, i.e., they overlap, then "$M + M$" will be the addition and is computed by taking the tangent at Mt on the curve (1) . The tangent drawn must join the curve at a point, say *P*. Thus the addition "$M + M$" will be the image of *P* through the x-axis.
- If M is a point on (1) , then the inverse of a point *M* will be $-M$. The addition of these points, i.e., $M + (-M)$, is a point at infinity because the line joining these two points will meet at infinity called *Point of Infinity* and denoted by $\mathcal{O}$. Thus, $M + (-M) = \mathcal{O}$.
- The scalar multiplication, i.e., *tM* where M is an elliptic curve point and t is a scalar, is computed as $tM = M + M + \cdots + M$ (t *times*.

B. Bilinear Pairing

Let G and G_1 be additive and multiplicative groups of prime order q, respectively. We define the bilinear paring $\hat{e} : G \times G \longrightarrow G_1$ on two groups G and G_1, with the following properties:

- **Bilinear:** If $L, M \in G$ and $l, m \in \mathbb{Z}_q^*$, the bilinear property holds as $\hat{e}(lL, mM) = \hat{e}(L, M)^{lm}$.
- **Non-degeneracy:** If $L \in G$, then $\hat{e}(L, L) \neq 1$ exists.
- **Computability:** If $L, M \in G$, then $\hat{e}(L, M)$ must be computed efficiently.

C. Identity-based Encryption

An Identity-based Encryption (IBE) scheme [17], [18] consists of the four polynomial time-bounded algorithms:

- **Setup**: The PKG executes this algorithm. The input of this algorithm is a security parameter *t*, and it outputs *param* and PKG's master secret key.
- **Extract**: The PKG executes this algorithm. The inputs of this algorithm are the identity of the user and the private key of the PKG; it outputs the identity-based private key of the user.
- **Encrypt**: This algorithm is executed by the sender. The input of this algorithm is a message *m*, and the receiver's identity. It outputs the ciphertext c.
- **Decrypt**: The receiver executes this algorithm to decrypt c. The inputs of this algorithm are c and the receiver's private key, and the output is *m*.

D. Computational Problem

- *Elliptic curve discrete logarithm (ECDL) problem:* To obtain an integer $r \in \mathbb{Z}_q^*$ from $Q = rP$ is hard, when $(P, Q) \in G$ is given. The probability of solving this problem is defined as $\mathbf{Adv}_{\mathcal{A}}^{ECDL} = \mathbf{Pr}[\mathcal{A}(P, Q = rP) = r : P, Q \in G; r \in \mathbb{Z}_q^*]$ where $\mathcal{A}$ denote an adversary. The ECDL assumption is $\mathbf{Adv}_{\mathcal{A}}^{ECDL} < \epsilon(k)$ [19].
- *Computational Diffie-Hellman (CDH) problem:* To find lmL is hard when (P, lP, mP) is given, and $l, m \in \mathbb{Z}_q^*$ are unknown. The probability of solving this problem is defined as $\mathbf{Adv}_{\mathcal{A}}^{CDH} = \mathbf{Pr}[\mathcal{A}(P, lP, mP) = lmP : P \in G; l, m \in \mathbb{Z}_q^*]$. The CDH assumption is $\mathbf{Adv}_{\mathcal{A}}^{CDH} < \epsilon(k)$.

III. Proposed ID-3PAKA Protocol

Here, we design an ID-3PAKA protocol to exchange a shared secret session key among three users. The developed protocol is based on the IBC and uses the properties of bilinear pairing and ECC. Let the three users say U_1, U_2, and U_3, wish to exchange a common session key among them. Besides U_1,

U_2, and U_3, we consider another trusted entity, called PKG, to generate the identity-based private key of the users. Our ID-3PAKA protocol includes the following algorithms.

A. Setup

This algorithm takes 1^k, where k denotes the security parameter. It outputs a list of global parameters as follows:

- PKG generates a large prime number q of k-bit, and an elliptic curve group G of order q, and another multiplicative group G_1 of same order q.
- PKG chooses a generator $P \in G$ of order n, and a bilinear map $\hat{e} : G \times G \longrightarrow G_1$.
- PKG selects two cryptographic hash functions $H_1 : \{0,1\}^* \longrightarrow G_2$ and $H_2 : G_1 \times G_2 \longrightarrow \mathbb{Z}_q^*$
- PKG chooses a master private key $s \in z_q^*$ uniformly at random, and calculates the master public key as $P_0{=}s{\cdot}P$.
- PKG outputs the global parameters $\Delta = \{q, G, G_1, P, P_0, \hat{e}, H_1(\cdot), H_2(\cdot)\}$ publicly.

B. Extract

The input of this algorithm is $\langle \Delta, ID_i \rangle$, where ID_i signifies the identity of the user U_i,$i = 1, 2, 3$. The user U_i submits its ID_i to PKG over a secure channel to compute the identity-based private key. The PKG computed the identity-based private key d_i of U_i as follows:

$$d_i = \Big(\frac{s}{s+q_i}\Big)P, \quad \text{where } q_i = H_1(ID_i)$$

C. Authenticated Key Agreement

This sub-section describes the key agreement phase of the proposed ID-3PAKA protocol. To compute a shared session key, all the users U_1, U_2 and U_3 execute the following steps:

- U_1 chooses a number $r_1 \in \mathbb{Z}_q^*$ uniformly at random, and calculates ψ_1 as:
  $$\psi_1 = r_1 P$$
  Similarly, U_2 selects a number $r_2 \in \mathbb{Z}_q^*$ uniformly at random, and calculates ψ_2 as:
  $$\psi_2 = r_2 P$$
  Similarly, Similarly, U_3 selects a number $r_3 \in \mathbb{Z}_q^*$ uniformly at random, and calculates ψ_2 as:
  $$\psi_3 = r_3 \cdot P \in G^*$$
- U_1 computes a signature σ_1 as:
  $$\sigma_1 = r_1 d_1 = r_1\Big(\tfrac{s}{s+q_1}\Big)P$$
  and, sends $\langle \psi_1, \sigma_1 \rangle$ to U_2 and U_3.
  In the similar manner, U_2 calculates the signature σ_2 as:
  $$\sigma_2 = r_2 d_2 = r_2\Big(\tfrac{s}{s+q_1}\Big)P$$
  and, $\langle \psi_2, \sigma_2 \rangle$ to U_1 and U_3.
  Similarly, U_3 calculates a signature σ_3 as:
  $$\sigma_3 = r_3 d_3 = r_3\Big(\tfrac{s}{s+q_3}\Big)P$$
  and, sends $\langle \psi_3, \sigma_3 \rangle$ to U_1 and U_2.

- U_1 verifies the received messages $\langle \psi_2, \sigma_2 \rangle$ and $\langle \psi_3, \sigma_3 \rangle$ of U_2 and U_3, respectively as follows:
  $$\hat{e}(\sigma_2, Q_2) = \hat{e}(\psi_2, P_0)$$
  $$\hat{e}(\sigma_3, Q_3) = \hat{e}(\psi_3, P_0)$$
  where $Q_i = q_i P + P_0$, $i = 1, 2, 3$. If the above verifications are successful, U_1 calculates $X_1 = r_1(\psi_2 - \psi_3)$, and sends it to U_2 and U_3.

  Similarly, U_2 verifies the received messages $\langle \psi_1, \sigma_1 \rangle$ and $\langle \psi_3, \sigma_3 \rangle$ of U_1 and U_3, respectively as follows:
  $$\hat{e}(\sigma_1, Q_1) = \hat{e}(\psi_1, P_0)$$
  $$\hat{e}(\sigma_3, Q_3) = \hat{e}(\psi_3, P_0)$$
  If the above verifications are successful, U_2 calculates $X_2 = r_2(\psi_3 - \psi_1)$, and sends it to U_1 and U_3.

  Similarly, U_3 verifies the received messages $\langle \psi_1, \sigma_1 \rangle$ and $\langle \psi_2, \sigma_2 \rangle$ of U_1 and U_2, respectively as follows:
  $$\hat{e}(\sigma_1, Q_1) = \hat{e}(\psi_1, P_0)$$
  $$\hat{e}(\sigma_2, Q_2) = \hat{e}(\psi_2, P_0)$$
  If the above verifications are successful, U_3 calculates $X_3 = r_3(\psi_1 - \psi_2)$, and sends it to U_1 and U_2.

- U_1 receives X_2 from U_2 and X_3 from U_3. Now, U_1 computes the secret session key as:
  $$SK_1 = 3r_1\psi_3 + 2X_1 + X_2 = (r_1r_2 + r_2r_3 + r_3r_1)P$$
  Similarly, U_2 receives X_1 from U_1 and X_3 from U_3. Now, U_3 calculates the secret session key as:
  $$SK_2 = 3r_2\psi_1 + 2X_2 + X_3 = (r_1r_2 + r_2r_3 + r_3r_1)P$$
  Similarly, U_3 receives X_1 from U_1 and X_2 from U_2. Now, U_2 calculates the secret session key as:
  $$SK_3 = 3r_3\psi_2 + 2X_3 + X_1 = (r_1r_2 + r_2r_3 + r_3r_1)P$$

It can be easily seen that

$$SK = SK_1 = SK_2 = SK_3 = (r_1r_2 + r_2r_3 + r_3r_1)P$$

IV. Security Analysis

Here, we have shown the correctness of the proposed ID-3PAKA protocol. Further, we also analyzed and found that the proposed protocol resists various known attacks.

A. Correctness

Lemma 1: If all users $U_i; i = 1, 2, 3$ correctly execute the proposed protocol then,

$$\hat{e}(\sigma_i, Q_i) = \hat{e}(\psi_i, P_0)$$

Proof 1: Since, we have

$$\begin{aligned}
\hat{e}(\sigma_i, Q_i) &= \hat{e}\Big(r_i\Big(\frac{s}{s+q_i}\Big)P, Q_i\Big) \ \ [\sigma_i = r_i\Big(\frac{s}{s+q_i}\Big)P] \\
&= \hat{e}\Big(r_i\Big(\frac{s}{s+q_i}\Big)P, q_iP + P_0\Big) \ \ [Q_i = q_iP + P_0] \\
&= \hat{e}\Big(r_i\Big(\frac{s}{s+q_i}\Big)P, q_iP + sP\Big) \ \ [P_0 = sP] \\
&= \hat{e}\Big(r_i\Big(\frac{s}{s+q_i}\Big)P, (q_i+s)P\Big) \\
&= \hat{e}(r_isP, P) \\
&= \hat{e}(r_iP, sP) \\
&= \hat{e}(\psi_i, P_0)
\end{aligned}$$

B. Security Analysis

This subsection states and proves the following theorem to analyze the security of the proposed ID-3PAKA protocol.

Theorem 1: The proposed ID-3PAKA protocol is secure against a randomized polynomial time-bounded adversary $\mathcal{A}$.

Proof 2: This theorem assumes that a randomized polynomial time-bounded adversary $\mathcal{A}$ wishes to attack the proposed ID-3PAKA protocol. This security proof believes that the view against the ID-3PAKA protocol is a computational perspective. The three users U_1, U_2 and U_3 wish to negotiate a shared secret key. The view of the proposed protocol may be expressed as follows:

$$\begin{aligned}
&\langle U_1(r_1), U_2(r_2), U_3(r_3)\rangle(\Delta, ID_i : i = 1, 2, 3) \\
&\quad = (SK_1, SK_2, DK_3)
\end{aligned}$$

where $\langle U_1(r_1), U_2(r_2), U_3(r_3)\rangle(\Delta)$ denotes an interactive execution of the proposed ID-3PAKA protocol among U_1, U_2 and U_3. In the proposed protocol, the input of U_1, U_2, and U_3 are r_1, r_2 and r_3, respectively. However, they exhibit the outputs as SK_1, SK_2 and SK_3, respectively. Here, $\Delta = \{q, G, G_1, P, P_0, \hat{e}, H_1(\cdot), H_2(\cdot)\}$ indicates the global parameters generated during the initialization phase.

$\mathcal{A}$ can obtain the public information (x, y), which is made available during the execution of the setup phase of the proposed proposal. Thus, the *view* of $\mathcal{A}$ is described as follows:

$$\begin{aligned}
&\textbf{View}\langle U_1(r_1), U_2(r_2), U_3(r_3)\rangle(\Delta, ID_i; i = 1, 2, 3) \\
&\quad = (x, y)
\end{aligned}$$

The security of proposed ID-3PAKA protocol is assumed in the following aspects:

(1) Assume that $\mathcal{A}$ wants to attack the exchanged session key SK. The probability in which $\mathcal{A}$ can compute the original session key SK is given as :

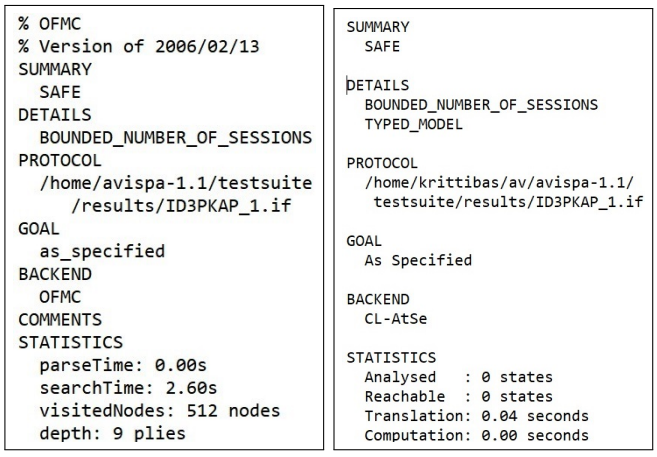
$$\begin{aligned}
&\textbf{Pr}[\mathcal{A}(\Delta, ID_i, \textbf{View}\langle U_1(r_1), U_2(r_2), U_2(r_2)\rangle(\Delta, ID_i) : \\
&\qquad\qquad i = 1, 2, 3) = SK] \\
=\ &\textbf{Pr}[\mathcal{A}(\Delta, ID_i, (x, y) : i = 1, 2, 3) = SK] \\
=\ &\textbf{Pr}(\Delta, ID_i, (\psi_i, X_i) : i = 1, 2, 3) = SK] \\
=\ &\textbf{Pr}(\Delta, ID_i, (r_iP, r_i\psi_i) : i = 1, 2, 3) = SK] \\
<\ &\epsilon(k) \qquad [\because \textbf{Adv}_{\mathcal{A}}^{ECDL} < \epsilon(k), \textbf{Adv}_{\mathcal{A}}^{CDH} < \epsilon(k)]
\end{aligned}$$

(2) Now, assume that $\mathcal{A}$ tries to guess the values $r_i, i = 1, 2, 3$. The success probability of $\mathcal{A}$ to guess $r_i, i = 1, 2, 3$ are given as follows:

$$\begin{aligned}
&\textbf{Pr}[\mathcal{A}(\Delta, ID_i, \textbf{View}\langle U_1(r_1), U_2(r_2), U_3(r_3)\rangle(\Delta, ID_i) : \\
&\qquad\qquad i = 1, 2, 3) = (r_1, r_2, r_3)] \\
=\ &\textbf{Pr}[\mathcal{A}(\Delta, ID_i, (x, y) : i = 1, 2, 3) = (r_1, r_2, r_3)] \\
=\ &\textbf{Pr}[\mathcal{A}(\Delta, ID_i, (r_1P, r_2P, r_3P) : i = 1, 2, 3) = (r_1, r_2, r_3)] \\
<\ &\epsilon(k) \qquad [\because \textbf{Adv}_{\mathcal{A}}^{ECDL} < \epsilon(k)]
\end{aligned}$$

C. Simulation of ID-3PAKA Protocol in AVISPA Tool

AVISPA is a widely accepted simulation software tool to verify the protocol's security against active and passive attacks, replay attacks, and man-in-the-middle attacks. The simulation result shows that the proposed scheme is SAFE under OFMC and CL-AtSe back-ends.



(a) OFMC back-end (b) Cl-AtSe back-end

Fig. 1: Simulation of the proposed protocol under AVISPA tool.

V. Performance Analysis

Here, we have analyze the proposed ID-3PAKA protocol in terms of computation and communication costs. Next, the implementation of the proposed proposed protocol has been done using pairing-based cryptography (PBC) library [20].

TABLE I: Computation cost of various operations

Cryptographic operation	Notation	Computation cost
Modular Inversion	t_{in}	$12t_{ml}$
Modular Exponentiation	t_{ex}	$21t_{ml}$
Bilinear Pairing	t_{bp}	$87t_{ml}$
Point Addition	t_{ad}	$0.12t_{ml}$
Point Multiplication	t_{pt}	$29t_{ml}$
Cryptographic Hash	t_h	t_{ml}
MTP Hash	t_{mtp}	$29t_{ml}$

The cost of various cryptographic operations are included in Table I in terms of t_{ml} where t_{ml} is the cost of computing the modular multiplication operation [19]. The execution cost

of the proposed protocol during the key agreement phase is evaluated as shown follows.

- The parties U_1, U_2 and U_3 compute $\psi_i = r_i P$ and $\sigma_i = r_i d_i$, $i = 1, 2, 3$, respectively. The execution cost for each U_i is $2t_{pt}$.
- Next, U_1, U_2 and U_3 verify the signatures received from other users by checking whether $\hat{e}(\sigma_i, Q_i) = \hat{e}(\psi_i, P_0)$ holds, where $Q_i = (H_1(ID_i)P + P_0)$, $i = 1, 2, 3$, respectively. The execution cost incurred for each U_i is $2t_{bp}$.
- The computation cost of X_i for each U_i is t_{pt}.

Thus, each U_i incurs a total computation cost of $3t_{pt} + 2t_{bp} \approx 261t_{ml}$.

In case of communication overheads, every user U_i sends $\langle \psi_i, \sigma_i \rangle$. Hence, the communication cost of each U_i is $2|G_1|$.

The simulation of the proposed ID-3PAKA protocol has been carried out using the PBC library [20]. We have used Dell Latitude 3400 laptop with Intel(R) Core(TM) i7-8565U CPU @ 1.80GHz processor and 16 GB of RAM for the simulation results. The Ubuntu-20.04.3 LTS with 64-bit has been used as the operating system. All the operations have been executed 100 times, and a mean of these executions has been considered. Table II shows the execution results of various cryptographic operations. The proposed ID-3PAKA protocol has been implemented using a singular Type-A elliptic curve. The bilinear pairing has been taken as symmetric pairing. After the simulation, we have found that the *Setup* phase of the proposed ID-3PAKA protocol takes $2.011\ msec$ for calculating the public key $P_0 = sP$ of PKG. The *Key generation* phase of the proposed ID-3PAKA protocol takes $2.017\ msec$ for computing the secret key d_i of U_i. Thus, the *Initialization* phase of the proposed ID-3PAKA protocol takes a total of $4.028\ msec$. For the *Key agreement* phase of the proposed ID-3PAKA protocol, every user U_i computes $\{\sigma_i, \psi_i\}$ with cost $3.429\ msec$. In addition, the running time to compute X_i and $\hat{e}(\sigma_i, Q_i) = \hat{e}(\psi_i, P_0)$ are is $2.011\ msec$ and $8.011\ msec$, respectively. Hence, the running time of *Key agreement* phase is $13.451\ msec$.

TABLE II: Execution time of various operations

Operation	Execution time
t_{in}	$0.0065\ msec$
t_{ex}	$0.5321\ msec$
t_{bp}	$4.134\ msec$
t_{ad}	$0.0003\ msec$
t_{pt}	$2.011\ msec$
t_{mtp}	$2.0022\ msec$

VI. Conclusion

In this paper, we have devised a secure ID-3PAKA protocol based on the properties of the elliptic curve and bilinear pairing. We have discussed the correctness and security of the protocol. The performance analysis indicates the computation and communication costs of the protocol are reasonable. The protocol is secured based on the hardness assumption of the ECDL and CDH problems. The simulation results on the OFMC and CL-AtSe back-ends of the AVISPA tool proved that the protocol is secure against active and passive attacks. Due to the strong security and low computational cost, the proposed protocol may be helpful in many applications with limited communication and computation costs.

References

[1] M. S. Obaidat and N. Boudriga, *Security of e-Systems and Computer Networks*. Cambridge University Press, September 2007.

[2] M. S. Obaidat, I. Traore, and I. Woungang, Eds., *Biometric-based Physical and Cybersecurity Systems*. Springer International Publishing, 2019.

[3] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, 1976.

[4] C.-M. Chen, K.-H. Wang, T.-Y. Wu, and E. K. Wang, "On the security of a three-party authenticated key agreement protocol based on chaotic maps," *Data Science and Pattern Recognition*, vol. 1, no. 2, pp. 1–10, 2017.

[5] Z. Tan, "An efficient identity-based tripartite authenticated key agreement protocol," *Electronic Commerce Research*, vol. 12, no. 4, pp. 505–518, 2012.

[6] M. Steiner, G. Tsudik, and M. Waidner, "Diffie-hellman key distribution extended to group communication," in *Proceedings of the 3rd ACM Conference on Computer and Communications Security*, 1996, pp. 31–37.

[7] R. Barua, R. Dutta, and P. Sarkar, "Extending joux's protocol to multi party key agreement," in *International Conference on Cryptology in India*. Springer, 2003, pp. 205–217.

[8] D. S. Gupta, G. Biswas, and R. Nandan, "Security weakness of a lattice-based key exchange protocol," in *2018 4th International Conference on Recent Advances in Information Technology (RAIT)*. IEEE, 2018, pp. 1–5.

[9] S. H. Islam and G. P. Biswas, "A pairing-free identity-based authenticated group key agreement protocol for imbalanced mobile network," *Annals of Telecommunications*, vol. 67, no. 11-12, pp. 547–558, 2012.

[10] W. Han and Z. Zhu, "An id-based mutual authentication with key agreement protocol for multiserver environment on elliptic curve cryptosystem," *International Journal of Communication Systems*, vol. 27, no. 8, pp. 1173–1185, 2014.

[11] H. Xiong, Z. Chen, and F. Li, "New identity-based three-party authenticated key agreement protocol with provable security," *Journal of Network and Computer Applications*, vol. 36, no. 2, pp. 927–932, 2013.

[12] D. S. Gupta and G. P. Biswas, "On securing bi-and tri-partite session key agreement protocol using ibe framework," *Wireless Personal Communications*, vol. 96, no. 3, pp. 4505–4524, 2017.

[13] C.-L. Liu, W.-J. Tsai, T.-Y. Chang, and T.-M. Liu, "Ephemeral-secret-leakage secure id-based three-party authenticated key agreement protocol for mobile distributed computing environments," *Symmetry*, vol. 10, no. 4, p. 84, 2018.

[14] Z. Wang, Z. Ma, S. Luo, and H. Gao, "Key escrow protocol based on a tripartite authenticated key agreement and threshold cryptography," *IEEE Access*, vol. 7, pp. 149 080–149 096, 2019.

[15] S. H. Islam, R. Amin, G. P. Biswas, M. S. Faras, X. Li, and S. Kumari, "An improved three party authenticated key exchange protocol using hash function and elliptic curve cryptography for mobile-commerce environments," *ournal of King Saud University - Computer and Information Sciences*, vol. 29, no. 3, pp. 311–324, 2017.

[16] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of computation*, vol. 48, no. 177, pp. 203–209, 1987.

[17] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Workshop on the theory and application of cryptographic techniques*. Springer, 1984, pp. 47–53.

[18] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," *SIAM journal on computing*, vol. 32, no. 3, pp. 586–615, 2003.

[19] D. S. Gupta, S. H. Islam, M. S. Obaidat, P. Vijayakumar, N. Kumar, and Y. Park, "A provably secure and lightweight identity-based two-party authenticated key agreement protocol for iiot environments," *IEEE Systems Journal*, vol. 15, no. 2, pp. 1732–1741, 2020.

[20] B. Lynn, "Pbc library-the pairing-based cryptography library (2013)," 2013. [Online]. Available: https://crypto.stanford.edu/pbc/