IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 10 May 2023, accepted 1 June 2023, date of publication 9 June 2023, date of current version 20 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3284832

RESEARCH ARTICLE

An Efficient and Reliable User Access Protocol for Internet of Drones

SAJID HUSSAIN[1], MUHAMMAD FAROOQ[2], BANDER A. ALZAHRANI[3], AIIAD ALBESHRI[3], KHALID ALSUBHI[3], AND SHEHZAD ASHRAF CHAUDHRY[4], (Member, IEEE)

[1]Department of Cyber Security, Air University, Islamabad 44000, Pakistan
[2]Department of Mathematics and Statistics, College of Arts and Sciences, Abu Dhabi University, Abu Dhabi, United Arab Emirates
[3]Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia
[4]Department of Computer Science and Information Technology, College of Engineering, Abu Dhabi University, Abu Dhabi, United Arab Emirates

Corresponding author: Shehzad Ashraf Chaudhry (ashraf.shehzad.ch@gmail.com)

The Deanship of Scientific Research (DSR) at King Abdulaziz University (KAU), Jeddah, Saudi Arabia has funded this project, under grant no. (RG-5-611-43). The work of Shehzad Ashraf Chaudhry was supported by the Abu Dhabi University's Office of Research and Sponsored Programs under Grant 19300810.

ABSTRACT The Internet of Drones (IoD) has blown up the interest of academia and industry due to their deployment in military and civil fields. In the IoD environment, drones collect and transmit real-time data to the users through an insecure wireless medium. A secure and efficient authentication protocol is required to ensure that only authorized drones communicate and send data to authentic users. For this purpose, Zhang et al. introduced an authentication protocol for the IoD environment and asserted that it could hold out against many known attacks. However, after cautious scrutiny, this paper manifests that many shortcomings still exist in their protocol design. This article introduces an enhanced, lightweight, and secure protocol for the IoD environment to mitigate the drawbacks of Zhang et al.'s protocol. The performance comparison depicts that the proposed scheme provides enhanced security with minimal computation cost. This rise in communication cost is legitimate as the proposed protocol renders better protection than the preceding protocols. The formal automated analysis is carried out via ProVerif to prove that the proposed scheme withstands numerous attacks.

INDEX TERMS Key agreement, authentication, security, Internet of Drones, proverif.

I. INTRODUCTION

The curiosity of the industry and academia about Unmanned Aerial Vehicles (UAVs) and intelligent aviation technologies is increasing daily owing to several applications of UAVs in daily life, including surveillance, rescue, agriculture, delivery, and so on. Unmanned Ariel Vehicles (UAVs) play a crucial role in updating airspace navigation services with the help of the internet of drones (IOD), using the artificial intelligence, which is multilevel network control architecture (MVCA). [1]. Unmanned Aerial Vehicles (UAVs) have various uses, just as rescue systems, target tracking and detection, observation of the environment, healthcare systems, data gathering, goods dispersal, disaster handling, smart city traffic monitoring system, and security surveillance system [2]. Hence, the controller can collect real-time data using drones while staying remote [3].

Drones are a novel configuration of moving IoT objects, enabling them to be used widely as sensing devices in IoT environments [4]. Moreover, the basic architecture is presented in Figure 1. The combination of the IoT domain and smart drones is named the Internet of Drones (IoD). IoD is MVCA explicitly formulated to manage the airspace by placing drone technology and systematizing the drone coordination [2]. A drone is a vital part of the IoD environment. Drones use various sensor combinations to collect information from the desired area. Multi-drones can be used to gather information in a distributed way, reducing battery consumption and the cost required to deploy the infrastructure [5].

IoD is a novel pattern in wireless communication that uses IoT technologies and artificial intelligence to achieve its different vital tasks. With some recent advancements,

The associate editor coordinating the review of this manuscript and approving it for publication was Emrecan Demirors.

the security and privacy of drone is still the critical requirement. IoD networks have limited resources as drones have fewer power resources, storage, and computational power. Therefore, a lightweight and secure authentication and key agreement protocol ensure a safe and efficient mechanism for communicating the IoD network. Due to limited computational power, drones can process only simple operations, not complex ones. Hence, the operations performed on the drone side should be designed to consume significantly less computational power and battery. With all these limitations, the most crucial factor is to achieve authentication between the user and drones before exchanging the gathered information simultaneously by satisfying the confidentiality requirements.

This paper is arranged as heed: The adversarial model adopted is explained in Section I-A, Section III shortly investigates the existing literature. A brief revisit of the scheme of Zhang et al. is provided in Section IV followed by its weakness analysis in Section V. Our protocol is outlined in Section VI and its security analysis is done in Section VII. The comparative analysis is conducted in Section VIII. lastly, the article wind up in Section IX.

A. ADVERSARIAL MODEL

Common adversarial model [6], [7], [8], [9] has been considered in this article with $\mathcal{A}$ as an active adversary having following capabilities:

1) $\mathcal{A}$ vehemently possesses control over the communication channel, which is public/open in nature [10], [11].
2) $\mathcal{A}$ can re-transmit and modify the old message and is also able to send a fake/forged message.
3) $\mathcal{A}$ can also extract the data/parameters stored in a smartcard using power analysis techniques [12], [13].
4) $\mathcal{A}$ may be an outsider or could be a privileged insider user of the system, and $\mathcal{A}$ may try to break the security in addition to the privacy of the user, and system [14], [15].
5) Private-key of the *RC* cannot be compromised.

II. MOTIVATIONS AND CONTRIBUTIONS

The main contributions of this work are briefly presented in this section:

1) To safeguard user-drone communication, we proposed an authentication scheme based on a symmetric key.
2) Utilizing the automated security software ProVerif, the security of the introduced scheme was examined.
3) Based on the analysis, the proposed system could survive known threats and provide an outstanding balance between security and effectiveness.
4) Because drones are resource-constrained and have limited processing capacity, the computation cost of the introduced protocol is lower than that of the protocols over the drone.
5) The suggested protocol avoids timestamp-based two-way authentication protocols' clock synchronization problem.
6) The suggested protocol is superior in terms of security and performance analysis compared to existing protocols.

III. RELATED WORK

Authentication by key agreement (AKA) protocols allow entities to authenticate mutually and share a session key to the insecure channel. In this section, numerous AKA-related protocols are reviewed. Ferrag et al. [16] presented a detailed review of previously published surveys and authentication protocols related to the Internet of Things (IoT). They have also discussed various formal security verification techniques for the IoT. Ferrag et al. also discussed several threat models associated with IoT. Yaacoub et al. [17] surveyed and presented a comprehensive review of the different aspects of drones related to their security and limitations. They have shown a details overview of drone architecture, communication types, and their classification. In addition, they have also discussed the drone uses and their applications and security vulnerabilities and threats faced by the drones. The current limitations and recommendations for future research directions are also discussed. Challa et al. [18] introduced a protocol based on ECC authentication to secure Wireless Healthcare Sensor Network (WHSN). In their protocol, the communication between Trusted Authority (TA) and the User (U) is secured by employing ECC. In contrast, the Hash function and bitwise XOR operation are used on low-powered sensors. However, Ali et al. [19] identified that Chall et al.'s protocol is prone to numerous attacks, essentially reply attacks, denial-of-service (DoS) attacks, forgery attacks, lacks mutual authentication, and suffers from design faults. To protect cloud-based Industrial IoT (IIoT), Das et al. [20] proposed a biometric authentication scheme. They claimed that the proposed protocol renders anonymity and untraceability and is secure for well-known attacks. However, Hussain and Chaudhry [21] found Das et al.'s protocol vulnerable to stolen verifiers, smart device theft, and traceability attacks.

Wazid et al. [22] also introduced a lightweight authentication and key agreement protocol for IoD based on hash functions and bitwise XOR operations. In their protocol, users can access real-time data directly from drones after authentication rather than from the server. In their protocol, the drones are divided into various flying zones. Ali et al. [23] also devised an authentication scheme for IoD. The method [23] uses the same concept of temporal credentials, first presented in [24]. Chaudhry et al. [25] argued that the scheme [24] presented by Srinivas et al. also lacks mutual authentication to the problem of user traceability. Moreover, they [25] also nullified the arguments of Wazid et al. [22] regarding the invincibility of their proposed scheme and showed that the system of Wazid et al. has insecurities against a stolen verifier, session key exposure, user, drone and server forgery attacks.

Recently, in 2022, an authentication scheme using pre-stored authentication parameters and symmetric hash functions was proposed by Yu et al. [26]. Owing to the

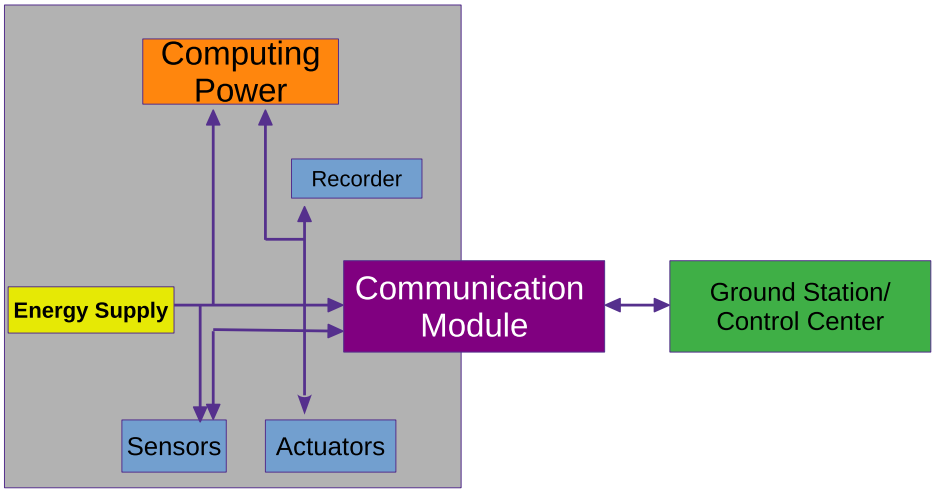


FIGURE 1. Typical architecture of drone.

usage of only lightweight parameters, the scheme of Yu et al. provides efficiency; however, the method can only accommodate one mobile user and needs to be more scalable. In the same year, Tanveer et al. also proposed an IoD authentication scheme. Tanveer et al. [27] used Elliptic Curve Cryptography (ECC), AEAD, and symmetric hash and related lightweight operations in their design. Citing Pu et al., the scheme of Tanveer et al. does not provide drone anonymity. Moreover, the Usage of ECC has an efficiency disadvantage over symmetric key operations. In their authentication scheme, Chaudhry et al. [28] also generated ECC-based certificates for providing user access to drones. Subsequently, Das et al. [29] proved that the certificates generated in the scheme of Chaudhry et al. are insecure, and the scheme is not practical. Zhang et al. [30] also proved that the session key in the scheme presented by Hussain et al. [31] could be easily exposed to an attacker. Another ECC-based scheme was proposed by Nikooghadam et al. [32]; however, as per the criticism explained in [26], the scheme [32] is insecure against replay and impersonation attack in addition to several other flaws.

Zhang et al. [33] also presented a lightweight AKA protocol of IoD, and they only employed a one-way hash function and bitwise XOR operation. In the proposed protocol, they asserted that it provides mutual authentication and can withstand various known attacks. However, in this article, we have proved that Zhang et al.'s protocol is prone to secret parameters leakage, privileged insider, user impersonation, control server impersonation, drone impersonation, parallel session-key, reply attack, lacks mutual authentication, and local user authentication. This paper introduces an enhanced protocol to overcome the drawbacks mentioned above.

IV. A BRIEF REVIEW OF THE SCHEME OF ZHANG et al

A brief review of the scheme of Zhang et al. [33] is presented in this section. Table 1 depicts various symbols used in the preceding sections of this paper. The scheme of Zhang et al. consists of four phases, namely i) setup phase, ii) user registration phase, iii) drone registration, and iv) authentication phase. All these phases are briefed as follows:

TABLE 1. Notations Guide.

Symbols	Representations
U_i, PID_i, PWD_i, BIO_i	i^{th} user, its personal identity, password and biometric
MD_i	i^{th} users mobile device
CS, ID_{CS}, MSK, k	Control server, its identity, 1024 bits private master-key, and 160 bits mask-key
n	160 bits public parameter selected by CS
DR_j, ID_{DR_j}	j^{th} drone and its identity
ρ_s, ρ_i, ρ_j	The pseudonym of CS, U_i, and DR_j
R_{and_m}	m^{th} random number of 160 bits
T_{CS}, T_i, T_{DR_j}	Current timestamps of CS, U_i, and DR_j
$Gen(.), Rep(.)$	Fuzzy biometric generator and reproduction functions
$\delta T, TC$	Maximum allowable transmission delay and present time
$i \stackrel{?}{=} j$	Checks if i equals to j
$h(.)$	Cryptographic one way hash function
$\oplus, \|\|$	Bitwise XOR and concatenation operators
$\mathcal{A}, \mathcal{I}, U_{\mathcal{A}}$	An adversary, intruder and privileged insider

A. SETUP PHASE

The control-serve *CS* phase picks a 160 bits random key *MSK* and marks it as its own private master key. The *CS* then picks a mask key (k) of size 160 bits and a public key n with size 160 bits. *CS* selects a hash function h and makes the parameters $\{h, n, \rho_s\}$ public where $\rho_s = h(ID_{DR_j}||k)$, while $\{MSK, k\}$ are kept private.

B. USER REGISTRATION PHASE

U_i initiates this step and subsequent steps are communicated among U_i and *CS* over a secure channel:

1) In this phase, user (U_i) picks his/her personal identity PID_i and password PWD_i, and transmits it to *CS*.
2) On receiving PID_i and PWD_i from U_i, in reply *CS* sends $\{\alpha_i, \rho_i, \rho_j\}$ where $\rho_i = h(ID_i||k), \alpha_i = h(PID_i||MSK)$. *CS* also stores $\{PID_i, \alpha_i, \rho_i\}$ in list L_s.
3) U_i receives $\{\alpha_i, \rho_i, \rho_j\}$ and computes $\alpha_i^m = h(PID_i||PWD_i) \oplus \alpha_i, \rho_i^m = h(PID_i||PWD_i) \oplus \rho_i$. Finally, U_i saves the parameters $\{\alpha_i^m, \rho_i^m, \rho_j\}$ securely.

C. DRONE REGISTRATION PHASE

The drone DR_j initiates this step and subsequent steps are communicated among DR_j and CS over a secure channel: Following are the steps performed to register the drone with the system:

1) DR_j picks an identity ID_{DR_j} and transmits it to CS.
2) On receiving ID_{DR_j} from DR_j, the CS computes $\rho_j = h(ID_{DR_j}||k), \alpha_j = h(ID_{DR_j}||MSK)$. CS stores $\{ID_{DR_j}, \alpha_j, \rho_j\}$ in list L_s, and transmits $\{\alpha_j, \rho_j\}$ to DR_j via private channel.
3) DR_j stores $\{\alpha_j, \rho_j\}$ in its memory.

D. AUTHENTICATION PHASE

The following steps are performed among U_i, CS, and DR_j to complete the authentication phase:

1) U_i provides his/her PID_i and PWD_i, and mobile device MD_i calculates $\rho_i = h(PID_i||PWD_i) \oplus \rho_i^m, \alpha_i = h(PID_i||PWD_i) \oplus \alpha_i^m$. Then MD_i picks an arbitrary number $R_{and_1} \in \mathcal{Z}_n^*$ and present timestamp T_i. Finally, MD_i transmits the authentication message $MSG_1 = \langle EXP_1, EXP_2, EXP_3, EXP_4\rangle$ to CS via insecure channel where:
$EXP_1 = h(\rho_s||T_i) \oplus \rho_i$
$EXP_2 = h(\rho_i||\rho_s||\alpha_i) \oplus R_{and_1}$
$EXP_3 = h(\rho_i||\rho_s||\alpha_i||R_{and_1}) \oplus \rho_j$
$EXP_4 = h(\rho_i||\rho_j||\rho_s||\alpha_i||R_{and_1})$
2) Upon receiving the authentication message $MSG_1 = \langle EXP_1, EXP_2, EXP_3, EXP_4\rangle$ from U_i, CS first checks the condition $|TC - T_i \leq \delta T|$. If true, CS retrieves α_i and computes:
$\rho_i' = h(\rho_s||T_i) \oplus EXP_1$
$R'_{and_1} = h(\rho_i||\rho_s||\alpha_i) \oplus EXP_2$
$\rho_j' = h(\rho_i||\rho_s||\alpha_i||R_{and_1}) \oplus EXP_3$
$EXP'_4 = h(\rho_i'||\rho_j'||\rho_s||\alpha_i||R'_{and_1})$
3) CS checks $EXP'_4 \stackrel{?}{=} EXP_4$. If true, CS retrieves α_j from L_s corresponding to ρ_j'. Next CS computes:
$EXP_5 = h(\rho_j'||\alpha_j) \oplus R'_{and_1}$
$EXP_6 = h(\rho_j'||\rho_s||\alpha_j'||R'_{and_1}) \oplus \rho_i'$
$EXP_7 = h(\rho_i'||\rho_j'||\rho_s||\alpha_j'||R'_{and_1})$
Finally, CS transmits the message $MSG_2 = \langle EXP_5, EXP_6, EXP_7\rangle$ via insecure channel.
4) Upon receiving the MSG_2 from CS, DR_j computes:
$R''_{and_1} = EXP_5 \oplus h(\rho_j||\alpha_j)$
$\rho_i'' = EXP_6 \oplus h(\rho_j||\rho_s||\alpha_j||R''_{and_1})$
$EXP'_7 = h(\rho_i''||\rho_j||\rho_s||\alpha_j||R''_{and_1})$
DR_j checks the condition $EXP'_7 \stackrel{?}{=} EXP_7$. If false, the session terminates; else, the next step is executed.
5) DR_j picks an arbitrary number $R_{and_2} \in \mathcal{Z}_n^*$ and computes:
$EXP_8 = h(\rho_j||\rho_i''||R''_{and_1}) \oplus R_{and_2}$
$EXP_9 = h(R''_{and_1}||R_{and_2})$
$SK_{ji} = h(\rho_i''||\rho_j||\rho_s||EXP_9)$
$EXP_{10} = h(\rho_i''||\rho_j||\rho_s||R''_{and_1}||R_{and_2}||EXP_9)$
Finally, DR_j transmits the message $MSG_3 = \langle EXP_8, EXP_{10}\rangle$ to U_i via insecure channel.
6) Upon receiving the MSG_3 from DR_j, U_i computes:
$R'_{and_2} = EXP_8 \oplus h(\rho_j||\rho_i||R_{and_1})$
$EXP'_9 = h(R_{and_1}||R'_{and_2})$
$EXP'_{10} = h(\rho_i||\rho_j||\rho_s||R_{and_1}||R'_{and_2})$
$SK_{ij} = h(\rho_i||\rho_j||\rho_s||EXP'_9)$
Finally, U_i checks whether $EXP'_{10} \stackrel{?}{=} EXP_{10}$, if true then $SK_{ij}(= SK_{ji})$ is accepted.

V. WEAKNESSES OF THE SCHEME OF ZHANG et al

This section explores the weaknesses of the scheme of Zhang et al. [33]. The analysis conducted in this section shows some critical insecurities of the scheme of Zhang et al. against some serious threats, as explained in the following subsections:

A. PRIVILEGED INSIDER ATTACK

As depicted in Subsections IV-C and IV-B, various parameters related to U_i and DR_j are stored in the CS database. Any dishonest individual having enough privilege can access these parameters and can launch various attacks based on stored parameters.

B. LEAKAGE OF SECRET PARAMETERS

In Zhang et al.'s scheme, once U_i is authenticated, it transmits a message containing $MSG_1 = \langle EXP_1, EXP_2, EXP_3, EXP_4\rangle$ to CS via the insecure channel. An $\mathcal{A}$ can intercept this request and can access these parameters. Subsequently, $\mathcal{A}$ initiates an attack as follows:

1) First $\mathcal{A}$ will intercept the message:

$$MSG_1 = \langle EXP_1, EXP_2, EXP_3, EXP_4\rangle \quad (1)$$

2) The parameter ρ_s is publicly available, T_i is transmitting openly, and $\mathcal{A}$ being a privileged insider can access α_i corresponding to ρ_i. $\mathcal{A}$ will compute:

$$\mathcal{X}_{\mathcal{A}} = h(\rho_s||T_i) \quad (2)$$
$$\mathcal{Y}_{\mathcal{A}} = h(\rho_i||\rho_s||\alpha_i) \quad (3)$$
$$\mathcal{Z}_{\mathcal{A}} = h(\rho_i||\rho_s||\alpha_i||R_{and_1}) \quad (4)$$

3) EXP_1, EXP_2, EXP_3, and EXP_4 are:

$$EXP_1 = h(\rho_s||T_i) \oplus \rho_i \quad (5)$$
$$EXP_2 = h(\rho_i||\rho_s||\alpha_i) \oplus R_{and_1} \quad (6)$$
$$EXP_3 = h(\rho_i||\rho_s||\alpha_i||R_{and_1}) \oplus \rho_j \quad (7)$$
$$EXP_4 = h(\rho_i||\rho_j||\rho_s||\alpha_i||R_{and_1}) \quad (8)$$

4) By employing Equations 2, 3, and 4, $\mathcal{A}$ can compute:

$$\rho_i = \mathcal{X}_{\mathcal{A}} \oplus EXP_1 \quad (9)$$
$$R_{and_1} = \mathcal{Y}_{\mathcal{A}} \oplus EXP_2 \quad (10)$$
$$\rho_j = \mathcal{Z}_{\mathcal{A}} \oplus EXP_3 \quad (11)$$

Hence, the secret parameters $\{\rho_i, R_{and_1}, \rho_j\}$ have been compromised.

C. U_i IMPERSONATION ATTACK

In order to impersonate on behalf of a legitimate user, the $\mathcal{A}$ adopts the procedure explained as follows:

1) To forge a legal message MSG_1, $\mathcal{A}$ requires the knowledge of $\{\rho_s, \rho_i, \rho_j, \alpha_i\}$. As a privileged user, $\mathcal{A}$ can access the parameters $\{\rho_j, \alpha_i\}$ from server as described in Section V-A, whereas, ρ_s is a public parameter, ρ_i has been compromised as described in Section V-B.
2) Next $\mathcal{A}$ will pick an arbitrary number $R^{\mathcal{A}}_{and_1} \in \mathcal{Z}^*_n$, present timestamp $T^{\mathcal{A}}_i$ and will compute:

$$EXP^{\mathcal{A}}_1 = h(\rho_s||T^{\mathcal{A}}_i) \oplus \rho_i \tag{12}$$
$$EXP^{\mathcal{A}}_2 = h(\rho_i||\rho_s||\alpha_i) \oplus R^{\mathcal{A}}_{and_1} \tag{13}$$
$$EXP^{\mathcal{A}}_3 = h(\rho_i||\rho_s||\alpha_i||R^{\mathcal{A}}_{and_1}) \oplus \rho_j \tag{14}$$
$$EXP^{\mathcal{A}}_4 = h(\rho_i||\rho_j||\rho_s||\alpha_i||R^{\mathcal{A}}_{and_1}) \tag{15}$$

Finally, $\mathcal{A}$ will transmit the message containing $MSG^{\mathcal{A}}_1 = \langle EXP^{\mathcal{A}}_1, EXP^{\mathcal{A}}_2, EXP^{\mathcal{A}}_3, EXP^{\mathcal{A}}_4, T^{\mathcal{A}}_i\rangle$ to CS via open channel.
3) The CS on reception of $MSG^{\mathcal{A}}_1$, checks the validity of $|TC - T^{\mathcal{A}}_i \leq \delta T|$. The validity holds as $T^{\mathcal{A}}_i$ is freshly generated. The CS now computes:

$$\rho'_i = h(\rho_s||T^{\mathcal{A}}_i) \oplus EXP^{\mathcal{A}}_1 \tag{16}$$
$$R'^{\mathcal{A}}_{and_1} = h(\rho_i||\rho_s||\alpha_i) \oplus EXP^{\mathcal{A}}_2 \tag{17}$$
$$\rho'_j = h(\rho_i||\rho_s||\alpha_i||R'^{\mathcal{A}}_{and_1}) \oplus EXP^{\mathcal{A}}_3 \tag{18}$$
$$EXP'^{\mathcal{A}}_4 = h(\rho'_i||\rho'_j||\rho_s||\alpha_i||R'^{\mathcal{A}}_{and_1}) \tag{19}$$

CS will finally check the condition $EXP'^{\mathcal{A}}_4 \stackrel{?}{=} EXP^{\mathcal{A}}_4$. This condition will be verified successfully as all the parameters used by the $\mathcal{A}$ are genuine, as described above. Hence, $\mathcal{A}$ has been successful in impersonating a legal user U_i.

D. CONTROL SERVER IMPERSONATION ATTACK

As a privileged insider $\mathcal{A}$ has all the necessary parameters as described in Sections V-A, V-B, and V-C to impersonate as a CS. The $\mathcal{A}$ adopts following procedure:

1) $\mathcal{A}$ will compute:

$$EXP^{\mathcal{A}}_5 = h(\rho'_j||\alpha_j) \oplus R'^{\mathcal{A}}_{and_1} \tag{20}$$
$$EXP^{\mathcal{A}}_6 = h(\rho'_j||\rho_s||\alpha'_j||R'^{\mathcal{A}}_{and_1}) \oplus \rho'_i \tag{21}$$
$$EXP^{\mathcal{A}}_7 = h(\rho'_i||\rho'_j||\rho_s||\alpha'_j||R'^{\mathcal{A}}_{and_1}) \tag{22}$$

where $R'^{\mathcal{A}}_{and_1}$ is an arbitrary number chosen by the $\mathcal{A}$. Finally, $\mathcal{A}$ will transmit the message containing $MSG^{\mathcal{A}}_2 = \langle EXP^{\mathcal{A}}_5, EXP^{\mathcal{A}}_6, EXP^{\mathcal{A}}_7\rangle$ to DR_j via insecure channel.
2) Upon receiving the $MSG^{\mathcal{A}}_2$ from CS, DR_j will compute:

$$R''^{\mathcal{A}}_{and_1} = EXP_5\mathcal{A} \oplus h(\rho_j||\alpha_j) \tag{23}$$
$$\rho''_i = EXP_6\mathcal{A} \oplus h(\rho_j||\rho_s||\alpha_j||R''^{\mathcal{A}}_{and_1}) \tag{24}$$
$$EXP'_7 = h(\rho''_i||\rho_j||\rho_s||\alpha_j||R''^{\mathcal{A}}_{and_1}) \tag{25}$$

3) DR_j will finally check the condition $EXP'_7 \stackrel{?}{=} EXP^{\mathcal{A}}_7$. This condition will be verified successfully as all the parameters are genuinely and directly accessed by the $\mathcal{A}$ from CS.
Therefore, the scheme is prone to control server impersonation attacks.

E. DRONE IMPERSONATION ATTACK

A privileged insider can also impersonate a DR_j towards U_i. Subsequent are the steps performed by the $\mathcal{A}$ to impersonate as a DR_j:

1) Firstly, $\mathcal{A}$ will choose an arbitrary number $R^{\mathcal{A}}_{and_2}$, whereas $\mathcal{A}$ can extract R_{and_1} from MSG_1 as described in Section V-B and will compute:

$$EXP^{\mathcal{A}}_8 = h(\rho_j||\rho''_i||R_{and_1}) \oplus R^{\mathcal{A}}_{and_2} \tag{26}$$
$$EXP^{\mathcal{A}}_9 = h(R_{and_1}||R^{\mathcal{A}}_{and_2}) \tag{27}$$
$$SK^{\mathcal{A}}_{ji} = h(\rho''_i||\rho_j||\rho_s||EXP^{\mathcal{A}}_9) \tag{28}$$
$$EXP^{\mathcal{A}}_{10} = h(\rho''_i||\rho_j||\rho_s||R_{and_1}||R^{\mathcal{A}}_{and_2}||EXP^{\mathcal{A}}_9) \tag{29}$$

Finally, $\mathcal{A}$ will transmit the message containing $MSG^{\mathcal{A}}_3 = \langle EXP^{\mathcal{A}}_8, EXP^{\mathcal{A}}_{10}\rangle$ to U_i via insecure channel.
2) Upon receiving the message $MSG^{\mathcal{A}}_3$, U_i will compute:

$$R'^{\mathcal{A}}_{and_2} = EXP^{\mathcal{A}}_8 \oplus h(\rho_j||\rho_i||R_{and_1}) \tag{30}$$
$$EXP'_9 = h(R_{and_1}||R'^{\mathcal{A}}_{and_2}) \tag{31}$$
$$EXP'_{10} = h(\rho_i||\rho_j||\rho_s||R_{and_1}||R'_{and_2}||EXP'_9) \tag{32}$$

Finally, U_i will check the condition $EXP'_{10} \stackrel{?}{=} EXP^{\mathcal{A}}_{10}$. The $\mathcal{A}$ will pass this test as well because all the parameters are extracted from transmitted messages and from CS. Hence, the scheme is prone to drone impersonation attacks.

F. LACK OF MUTUAL AUTHENTICATION

As described in Sections V-C, V-D, and V-E that an $\mathcal{A}$ can impersonate as a user, control server, and drone; hence, the scheme doesn't render mutual authentication.

G. PARALLEL SESSION-KEY ATTACK

Consider a privileged insider $\mathcal{A}$ who has successfully intercepted the authentication messages $\{MSG_1, MSG_2, MSG_3\}$ and has the knowledge of parameters $\{R_{and_1}, \rho_i, \rho_s, \rho_j\}$ as described in Section V-B.

1) In order to launch the parallel session-key attack $\mathcal{A}$ needs the knowledge of R_{and_2}, which can be acquired by adopting the subsequent procedure:

$$R_{and_2} = h(\rho_j||\rho_i||R_{and_1}) \oplus EXP_8 \tag{33}$$
$$EXP^{\mathcal{A}}_9 = h(R_{and_1}||R_{and_2}) \tag{34}$$

2) $\mathcal{A}$ will compute the session-key by computing:

$$SK_{ji} = h(\rho_i||\rho_j||\rho_s||EXP^{\mathcal{A}}_9) \tag{35}$$

Consequently, $\mathcal{A}$ can severally and independently compute the session-key $SK_{ij}(= SK_{ji})$ making the

scheme of Zhang et al. prone to the parallel session attack.

H. REPLY AND DENIAL-OF-SERVICE (DoS) ATTACK

In Zhang et al.'s scheme after the authentication of U_i, CS transmits a message containing the $MSG_2 = \langle EXP_5, EXP_6, EXP_7 \rangle$ to DR_j. Upon receiving the message from CS, DR_j extracts the parameter R_{and_1} from the message and checks the condition $EXP'_7 \stackrel{?}{=} EXP_7$. Now if $\mathcal{A}$ captures the message as it is transmitting over the insecure channel and re-transmits it to DR_j, then there is no way to verify the freshness of the message. DR_j will simply examine the condition $EXP'_7 \stackrel{?}{=} EXP_7$, which will be true as $\mathcal{A}$ is only transmitting the captured message and DR_j will do the computation and send response to U_i. Hence the scheme suffers from the reply and DoS attacks.

I. INEFFICIENT AUTHENTICATION PHASE

In Zhang et al.'s scheme, the U_i enters his/her identity and password and sends the authentication message to CS. CS received this message and performed the authentication. Now, if a U_i enters an invalid identity and password, MD_i will still compile message MSG_1 and will transmit it to CS without first checking the authenticity of the U_i.

As the number of users increases so, more requests to authenticate the user will be sent to CS. This results in bad resource utilization, as the user can be first authenticated at the local/user side, and then a message should be transferred to CS.

VI. PROPOSED PROTOCOL

The proposed protocol comprises five phases: i) setup phase, ii) user registration phase, iii) drone registration phase, iv) authentication and key agreement phase, and v) Password update phase. Each of these phases is explained in the following subsections:

A. SETUP PHASE

In this phase, the (CS) selects a 1024 bits random number MSK and marks it as its own master key. CS chooses $h(.)$, which is a one-way hash function and publicizes $\{h(.)\}$, whereas, keeps $\{MSK\}$ private.

B. USER REGISTRATION PHASE

To enter the system and to utilize its resources over the private channel, the user U_i is required to register with the CS first. Moreover, the proposed user registration phase is explained in Figure 2. Subsequent are the steps performed by the U_i to register with the CS:

1) To initiate registration, the U_i picks his/her personal identity PID_i and password PWD_i, computes $HID_i = h(PID_i)$, and transmits it to the CS over a secure channel.
2) Upon receiving message from U_i, in reply CS sends $\{\rho_i\}$ where $\rho_i = h(HID_i||MSK)$.

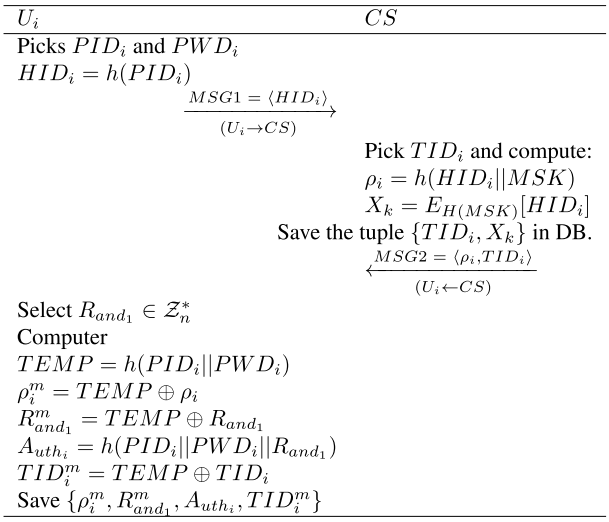


FIGURE 2. Proposed user registration phase.

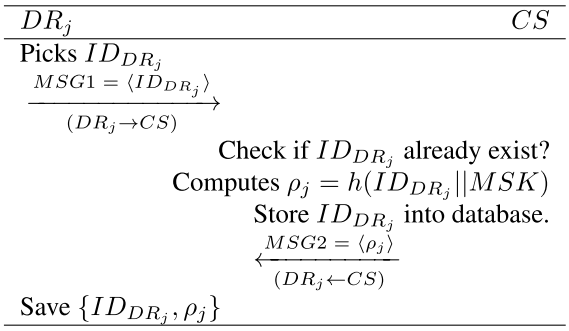


FIGURE 3. Proposed drone registration phase.

3) U_i receives $\{\rho_i\}$, selects an arbitrary number $R_{and_1} \in \mathcal{Z}_n^*$, and computes $TEMP = h(PID_i||PWD_i)$, $\rho_i^m = TEMP \oplus \rho_i$, $R_{and_1}^m = TEMP \oplus R_{and_1}$, and $A_{uth_i} = h(PID_i||PWD_i||R_{and_1})$. Finally, U_i saves the parameters $\{\rho_i^m, R_{and_1}^m, A_{uth_i}\}$ securely.

C. DRONE REGISTRATION PHASE

Drone registration phase is illustrated in Figure 3 and is explained in following steps:

1) DR_j picks an identity ID_{DR_j} and broadcast it to CS over the private channel.
2) Upon receiving the request of registration from DR_j, CS first checks if ID_{DR_j} already exists in the database, if true CS rejects the registration request and DR_j has to select unique identity. Else, CS computes $\rho_j = h(ID_{DR_j}||MSK)$. CS transmits $\{\rho_j\}$ to DR_j via private channel.
3) DR_j stores $\{ID_{DR_j}, \rho_j\}$ in its memory.

D. AUTHENTICATION AND KEY ESTABLISHMENT PHASE

The U_i initiates this step to establish a session key with DR_j with the help of CS. The steps in this phase as depicted in Figure 4 are explained as follows:

1) U_i provides his/her PID_i and PWD_i, and mobile device MD_i calculates $TEMP' = h(PID_i||PWD_i)$, $\rho_i = TEMP' \oplus \rho_i^m$, $R_{and_1} = TEMP' \oplus R_{and_1}^m$, $A'_{uth_i} = h(PID_i||PWD_i||R_{and_1})$. If $A_{uth_i} \stackrel{?}{=} A'_{uth_i}$ is terminate session terminates, else continues.
2) MD_i selects an arbitrary number $R_{and_2} \in \mathcal{Z}_n^*$ and present timestamp T_i. Finally, MD_i transmits the authentication message $MSG_1 = \langle EXP_1, EXP_2, EXP_3, HID_i, T_i\rangle$ to CS via insecure channel where:
$HID_i = h(PID_i)$
$EXP_1 = h(\rho_i||T_i||R_{and_2})$
$EXP_2 = \rho_i \oplus R_{and_2}$
$EXP_3 = ID_{DR_j} \oplus \rho_i$
3) Upon receiving the authentication message $MSG_1 = \langle EXP_1, EXP_2, EXP_3, HID_i, T_i\rangle$ from U_i, CS first checks $|TC - T_i \leq \delta T|$. If true, CS computes $ID_{DR_j} = \rho_i \oplus EXP_3$ and checks the existence of ID_{DR_j} in the CS's database. If it exists, the process continues as follows:
$\rho_i' = h(HID_i||MSK)$
$R_{and_2} = \rho_i' \oplus EXP_2$
4) CS checks $EXP_1 \stackrel{?}{=} h(\rho_i'||T_i||R_{and_2})$. If false, session terminates else CS picks present timestamp T_{cs} and computes:
$\rho_j' = h(ID_{DR_j}||MSK)$
$EXP_4 = \rho_j' \oplus R_{and_2}$
$EXP_5 = h(\rho_j'||T_{cs}||R_{and_2})$
$EXP_6 = h(R_{and_2}||\rho_i') \oplus \rho_j'$
Finally, CS transmits the message $MSG_2 = \langle EXP_4, EXP_5, EXP_6, T_{cs}\rangle$ via insecure channel.
5) Upon receiving the MSG_2 from CS, DR_j checks $|TC - T_{cs} \leq \delta T|$. If true, DR_j computes $R_{and_2} = EXP_4 \oplus \rho_j$ and checks the condition $EXP_5 \stackrel{?}{=} h(\rho_j||T_{CS}||R_{and_2})$. If false, the session terminates; else, the next step is executed.
6) DR_j picks an arbitrary number $R_{and_3} \in \mathcal{Z}_n^*$, present timestamp T_{DR_j}, and computes:
$SK_{ji} = h(R_{and_2}||R_{and_3}||(EXP_6 \oplus \rho_j))$
$EXP_7 = h(SK_{ji}||T_{DR_j}||(EXP_6 \oplus \rho_j))$
$EXP_8 = (EXP_6 \oplus \rho_j) \oplus R_{and_3}$
Finally, DR_j transmits the message $MSG_3 = \langle EXP_7, EXP_8, T_{DR_j}\rangle$ to U_i via insecure channel.
7) Upon receiving the MSG_3 from DR_j, U_i first checks the condition $|TC - T_{DR_j} \leq \delta T|$. If true U_i computes:
$TEMP'' = h(R_{and_2}||\rho_i)$
$R_{and_3} = EXP_8 \oplus TEMP''$
$SK_{ij} = h(R_{and_2}||R_{and_3}||TEMP'')$
$EXP_7' \stackrel{?}{=} h(SK_{ij}||T_{DR_j}||TEMP'')$
8) Finally, U_i checks whether $EXP_7' \stackrel{?}{=} EXP_7$, if true then session-key $SK_{ij}(= SK_{ji})$ is saved and will be used to communicate securely.

E. PASSWORD UPDATE PHASE

The following procedure is adopted by the U_i for its urge to update existing password:

1) First U_i will need to get authenticated successfully by adopting the procedure as described in the "Login and authentication phase".
2) Next U_i will provide new password PWD_i^{new}, and MD_i will computes $TEMP' = h(PID_i||PWD_i^{new})$, $TEMP = h(PID_i||PWD_i^{old})$, $\rho_i^{m'} = \rho_i^m \oplus TEMP \oplus TEMP'$, $R_{and_1}^{m'} = TEMP \oplus R_{and_1}^m \oplus TEMP'$, and $A'_{uth_i} = h(PID_i||PWD_i^{new}||R_{and_1})$.
3) Finally, MD_i saves the parameters $\{\rho_i^{m'}, R_{and_1}^{m'}, A'_{uth_i}\}$ by replacing $\{\rho_i^m, R_{and_1}^m, A_{uth_i}\}$ securely.

VII. SECURITY ANALYSIS

Automated formal security analysis and informal security analysis of the introduced protocol have been presented in this section.

A. AUTOMATED ANALYSIS USING ProVerif

The simulation-based results of the proposed protocol are presented in this section. The results are conducted using the Proverif tool. Its declaration and events are explained in Figure 5 (a), 5 (b), 5 (c), 5 (d), and 5 (e), respectively. ProVerif is primarily designed to check the robustness of authentication protocol in the existence of active attacks [34]. To simulate the protocol, three events have been defined, including (i) user, (ii) CS, and (iii) drone. Though, the starting events and ending events related to user $\mathcal{U}_i$, Control Server $\mathcal{CS}$ and drone $\mathcal{DR}_j$ is described as follows:

```
begin_Ui(bitstring) and end_Ui(bitstring),

begin_CS(bitstring) and end_CS(bitstring)

begin_DRj(bitstring) and end_DRj(bitstring)
```

Simulation results as presented in Figure 5 (f) illustrate that all the processes initiated and halted successfully. At the same time, the result shown in Figure depicts that the attacker query is not capable of computing and getting the session key as computed by the processes during the authentication procedure.

B. INFORMAL ANALYSIS

In this section, various well-known attacks, how adversaries can launch them, and what measures are taken to protect the scheme against them are discussed as adversaries can exploit the protocol by using the information transferred over a public channel, shared in the previous sessions, or information stored over the server and on the user's device.

1) REPLAY ATTACK

In the introduced protocol, timestamps $\{T_i, T_{CS}, T_{DR_j}\}$ and random numbers $\{R_{and1}, R_{and2}\}$ are employed to ensure that introduced protocol is fresh and secure from replay attack. In case a message is replayed, the timestamp will not pass the freshness test, or the random number may become inconsistent. Hence, the introduced protocol is resilient against replay attacks.

User (U_i)	Control Server (CS)	Drone (DR_j)
Enter PID_i and PWD_i		
Compute		
$TEMP^{'} = h(PID_i \| \| PWD_i)$		
$\rho_i = TEMP^{'} \oplus \rho_i^m$		
$R_{and_1} = TEMP^{'} \oplus R_{and_1}^m$		
$A_{uth_i}^{'} = h(PID_i \| \| PWD_i \| \| R_{and_1})$		
$A_{uth_i} \stackrel{?}{=} A_{uth_i}^{'}$		
Select $R_{and_2} \in \mathcal{Z}_n^*$ and T_i		
$TID_i = TEMP^{'} \oplus TID_i^m$		
$EXP_1 = h(\rho_i \| \| T_i \| \| R_{and_2})$		
$EXP_2 = \rho_i \oplus R_{and_2}$		
$EXP_3 = ID_{DR_j} \oplus \rho_i$		
$\xrightarrow[(U_i \rightarrow CS)]{MSG_1 = \langle EXP_1, EXP_2, EXP_3, TID_i, T_i \rangle}$		
	$\|TC - T_i \leq \delta T\|$	
	Check if TID_i exists in the DB?	
	and fetch corresponding X_k	
	Compute:	
	$HID_i = D_{H(MSK)}[X_k]$	
	$\rho_i^{'} = h(HID_i \| \| MSK)$	
	$ID_{DR_j} = \rho_i^{'} \oplus EXP_3$	
	Check if ID_{DR_j} exists in the DB?	
	$R_{and_2} = \rho_i^{'} \oplus EXP_2$	
	$EXP_1 \stackrel{?}{=} h(\rho_i^{'} \| \| T_i \| \| R_{and_2})$	
	Select T_{cs} and TID_i^{new}	
	Replace TID_i with TID_i^{new} compute:	
	$\rho_j^{'} = h(ID_{DR_j} \| \| MSK)$	
	$EXP_4 = \rho_j^{'} \oplus R_{and_2}$	
	$EXP_5 = h(\rho_j^{'} \| \| T_{cs} \| \| R_{and_2})$	
	$EXP_6 = h(R_{and_2} \| \| \rho_i^{'}) \oplus \rho_j^{'}$	
	$EXP_7 = TID_i^{new} \oplus h(HID_i \| \| \rho_i^{'} \| \| T_i)$	
	$\xrightarrow[(CS \rightarrow DR_j)]{MSG_2 = \langle EXP_4, EXP_5, EXP_6, EXP_7, T_{CS} \rangle}$	
		$\|TC - T_{cs} \leq \delta T\|$
		$R_{and_2} = EXP_4 \oplus \rho_j$
		$EXP_5 \stackrel{?}{=} h(\rho_j \| \| T_{CS} \| \| R_{and_2})$
		Pick $R_{and_3} \in \mathcal{Z}_n^*$ and T_{DR_j}
		$SK_{ji} = h(R_{and_2} \| \| R_{and_3} \| \| (EXP_6 \oplus \rho_j))$
		$EXP_8 = h(SK_{ji} \| \| T_{DR_j} \| \| (EXP_6 \oplus \rho_j))$
		$EXP_9 = (EXP_6 \oplus \rho_j) \oplus R_{and_3}$
	$\xleftarrow[(U_i \leftarrow DR_j)]{MSG_3 = \langle EXP_7, EXP_8, EXP_9, T_{DR_j} \rangle}$	
$\|TC - T_{DR_j} \leq \delta T\|$		
$TEMP^{''} = h(R_{and_2} \| \| \rho_i)$		
$R_{and_3} = EXP_9 \oplus TEMP^{''}$		
$SK_{ij} = h(R_{and_2} \| \| R_{and_3} \| \| TEMP^{''})$		
$EXP_8^{'} = h(SK_{ij} \| \| T_{DR_j} \| \| TEMP^{''})$		
$EXP_8^{'} \stackrel{?}{=} EXP_8$, if true save SK_{ij}		
Compute:		
$TID_i^{new} = EXP_7 \oplus h(h(PID_i) \| \| \rho_i \| \| T_i)$		
$TID_i^{new^m} = TID_i^{new} \oplus TEMP^{'}$		
Replace TID_i^m with $TID_i^{new^m}$		
	$\xleftrightarrow[(U_i \ \& \ DR_j \text{ both save the same session-key})]{SK_{ij} (= SK_{ji})}$	

FIGURE 4. Proposed authentication and key agreement phase.

2) KEY FRESHNESS

Although the session key is constructed by some distinct identities, random numbers, and timestamps, If $\mathcal{A}$ gets the session key secrets $\{R_{rand1}, R_{rand3}, EXP_6\}$ through session key guessing attack, still $\mathcal{A}$ can't get session key without knowledge of long term parameter p_j. Hence, the key freshness is ensured by generating a session-specific novel key.

```
(* --------- Channels ----------*)
free ChSec:channel [private]. (*secure channel between Ui,
Sm and CS*)
free ChPub:channel.(*public channel between Ui, Sm and CS*)
(*---------- Constants and Variables ---------*)
free PWDi:bitstring[private].free PIDi:bitstring [private].
free IDdrj:bitstring. free pi:bitstring.
free MSK:bitstring[private].free SKij:bitstring [private].
free IDUi :bitstring. free IDCS :bitstring.
free pj:bitstring. free IDDRj :bitstring.
(*-------Queries------*)
query id:bitstring; inj-event(end_Ui(IDUi)) ==>
inj-event(start_Ui(IDUi)).
query id:bitstring; inj-event(end_CS(IDCS)) ==>
inj-event(start_CS(IDCS)).
query id:bitstring; inj-event(end_DRj(IDDRj)) ==>
inj-event(start_DRj(IDDRj)).
query attacker(SKij).
```

(a) Channels, Constants & Variables, and Queries

```
(*=====*Events*=====*)
event start_Ui(bitstring).
event end_Ui(bitstring).
event start_CS(bitstring).
event end_CS(bitstring).
event start_DRj(bitstring).
event end_DRj(bitstring).

(*========Constructors=======*)
fun h(bitstring):bitstring.
fun Inverse(bitstring):bitstring.
fun Concat(bitstring,bitstring):bitstring.
fun XOR(bitstring,bitstring):bitstring.
fun Mul(bitstring,bitstring):bitstring.

(*======Equations=======*)
equation forall a:bitstring; Inverse(Inverse(a))=a.
equation forall a:bitstring, b:bitstring;XOR(XOR(a,b),b)=a.
```

(b) Events, Constructors, and Equations

```
(*-------------Ui--------------*)
let pUi=
event start_Ui(IDUi);
let TEMP'=h(Concat(PIDi,PWDi)) in
let xpi=XOR(TEMP',pi) in new Rrand1:bitstring;
let Rrand=XOR(TEMP',Rrand1) in
let Auth1=h(Concat(PIDi,(PWDi,Rrand1))) in
new Ti:bitstring; let HIDi=h(PIDi) in
new Rrand2:bitstring;
let EXP1=h(Concat(pi,(Ti,Rrand2))) in
let EXP2=XOR(pi, Rrand2) in let EXP3=XOR(IDdrj,pi) in
out(ChPub,(EXP1,EXP2,EXP3,HIDi,Ti));
in(ChPub,(EXP7:bitstring,EXP8:bitstring,Tdrj:bitstring));
let TEMP''=h(Concat(Rrand2,pi)) in
let Rrand3=XOR(EXP8,TEMP'') in
let xSKij=h(Concat(Rrand2,(Rrand3,TEMP''))) in
if EXP7=h(Concat(SKij,(Tdrj,TEMP''))) then
event end_Ui(IDUi) else 0.
```

(c) User Event

```
(*-------------CS--------------*)
let pCS=
event start_CS(IDCS);
in(ChPub,(EXP1:bitstring,EXP2:bitstring,EXP3:bitstring,
HIDi:bitstring,Ti:bitstring));
let xIDdrj=XOR(pi,EXP3) in
let pi'=h(Concat(HIDi,MSK)) in
let Rrand2=XOR(pi,EXP3) in
if EXP1=h(Concat(pi,(Ti,Rrand2))) then
new Tcs:bitstring;
let p'j=h(Concat(IDdrj,MSK)) in
let EXP4=XOR(p'j,Rrand2) in
let EXP5=h(Concat(p'j,(Tcs,Rrand2))) in
let EXP6=XOR((Concat(Rrand2,pi)),p'j) in
out(ChPub,(EXP4,EXP5,EXP6,Tcs));
event end_CS(IDCS)
else
0.
```

(d) Control Center Event

```
(*-------------DRj--------------*)
let pDRj=
event start_DRj(IDDRj);
in(ChPub,(EXP4:bitstring,EXP5:bitstring,EXP6:bitstring,Tcs:
bitstring));
let Rrand2=XOR(EXP4,pj) in
if EXP5=h(Concat(pj,(Tcs,Rrand2))) then
new Tdrj:bitstring;
new Rrand3:bitstring;
let xSKij=h(Concat(Rrand2,(EXP6,pj))) in
let EXP7=h(Concat(SKij,(Tdrj,(XOR(EXP6,pj))))) in
let EXP8=XOR(Rrand3,(XOR(EXP6,pj))) in
out(ChPub,(EXP7,EXP8,Tdrj));
event end_DRj(IDDRj)
else
0.
process (( !pDRj) | (!pCS) | (!pUi))
```

(e) Drone Event

```
RESULT inj-event(end_DRj(IDDRj[])) ==> inj-event(start_DRj
(IDDRj[])) is true.
-- Query not attacker(SKij[]) in process 1
Translating the process into Horn clauses...
Completing...
Starting query not attacker(SKij[])
RESULT not attacker(SKij[]) is true.

Verification summary:
Query inj-event(end_Ui(IDUi[]))==>inj-event(start_Ui(IDUi[]
)) is true.
Query inj-event(end_CS(IDCS[]))==>inj-event(start_CS(IDCS[]
)) is true.
Query inj-event(end_DRj(IDDRj[]))==>inj-event(start_DRj(
IDDRj[])) is true.

Query not attacker(SKij[]) is true.
```

(f) Proverif Results

FIGURE 5. ProVerif code.

3) USER ANONYMITY AND UNTRACEABILITY

Suppose an adversary $\mathcal{A}$ monitors and intrudes during the exchange of login and authentication messages. Not at all the intruded values $\{EXP_1, EXP_2, EXP_3, HID_i, T_i, EXP_4, EXP_5, EXP_6, T_{CS}, EXP_7, EXP_8, T_{DR_j}\}$ does not include any un-encrypted values helpful in recognizing the user U_i or Drone DR_j. Hence the introduced protocol employs anonymity. Additionally, all the values are made-up of random nonces, which remain unique during the various sessions. Hence the introduced protocol is resilient against untraceability and anonymity.

4) SMART CARD STOLEN ATTACK

Let $\mathcal{A}$ being an active attacker, steal the smart card of an authorized user, or the card be misplaced and found by $\mathcal{A}$. As an active attacker, $\mathcal{A}$ can obtain any sensitive information from the smart card. An attacker can took all the details related to $\{\rho_i^m, R_{and_1}^m, A_{uth_i}\}$. In order to obtain any sensitive parameters, $\mathcal{A}$ needs to know the values of PID_i and PWD as these values are kept as a secret to $\mathcal{A}$, which clearly implies that stolen smart card attack is not possible on the proposed scheme.

5) IMPERSONATION ATTACK

For completing a successful impersonation attack, the $\mathcal{A}$ needs to generate a valid and legitimate request message. The $\mathcal{A}$ ties to forge a valid request $MSG_1 = \langle EXP_1, EXP_2, EXP_3, HID_i, T_i \rangle$. However, an attacker can not produce a legitimate request because it requires the knowledge of $\{P_i, Rand_2\}$. Hence, the introduced protocol can be resilient against impersonation attacks.

6) MAN IN THE MIDDLE (MIM) ATTACK

To launch the MIM attack, $\mathcal{A}$ tries to detain and alter the login request from U_i to CS. However, the request message can only be forged or altered with the information of secret testimonials. Hence, the introduced protocol is resilient against the above-mentioned attack.

7) PERFECT FORWARD SECRECY

The session key shared among the three participating entities is generated using random nonces added by both the U_i and CS. Therefore, if the MSK-private key or any of the session keys, the $\mathcal{A}$ cannot benefit from computing future or past session keys. Hence, the introduced protocol incorporates perfect forward secrecy.

8) DOS ATTACK

Our approach depends heavily on secret secrets to successfully complete internal authentication, and in order to do so, U_i must determine if the particular condition $A_{uthi} = A_{uthi}$ is satisfied. If the requirement manages to hold, local authentication will be accomplished. U_i transmits the AKE query to CS following authentication. If not, U_i stops the AKE process from running and restricts itself from making several AKE queries to CS and DR_j. As a result, our system is immune to DoS attacks.

9) DE-SYNCHRONIZATION ATTACK

$\mathcal{A}$ may build a de-synchronization malicious activity by releasing the intercepted communication unless the network participants are managing pseudonyms while the AKE procedure is running. Both U_i and CS maintain updated TID_i in memory to protect themselves from the de-synchronization exploit. U_i may switch to using old TID_i old for the AKE procedure if $\mathcal{A}$ stops it by discarding the authentication messages. As a result, de-synchronization attacks cannot affect our proposed AKE scheme.

10) SECURE MUTUAL AUTHENTICATION

Fully secured mutual authentication is achieved by all entities in our proposed scheme. CS checks to see if $EXP_1 \stackrel{?}{=} h(\rho_i'||T_i||R_{and_2})$ after accepting the login response message "$M1 = \langle EXP_1, EXP_2, EXP_3, TID_i, T_i\rangle$" from U_i. DR_j determines either $EXP_5 \stackrel{?}{=} h(\rho_j||T_{CS}||R_{and_2})$ after receiving the authentication demand messages $\langle EXP_4, EXP_5, EXP_6, EXP_7, T_{CS}\rangle$ from CS. DR_j verifies $CS's$ identity if it is legitimate. When U_i receives the $EXP_7, EXP_8, EXP_9, T_{DR_j}$ authentication verification communications from DR_j, it checks to see if either $EXP_8' \stackrel{?}{=} EXP_8$. When the criterion is met, secures mutual authentication across U_i, CS, and DR_j is achieved.

VIII. COMPARATIVE ANALYSIS

The introduced protocol is compared with existing benchmarks such as Wazid et al. [22], Singh et al. [35], Zhang et al. [33], Yu et al. [26], Tanveer et al. [27], and Nikooghadam et al. [32] in this section.

A. FUNCTIONALITY COMPARISON

In this subsection, we compare the functionality feature provision of the proposed protocol with the protocols presented by Wazid et al. [22], Singh et al. [35], Zhang et al. [33], Yu et al. [26], Tanveer et al. [27], and Nikooghadam et al. [32]. TheTable 2 depicts the comparisons,

TABLE 2. Comparison of functionality features.

FNR	[22]	[35]	[33]	[26]	[27]	[32]	Our
SSA	×	–	×	✓	✓	✓	✓
ANO	✓	×	✓	✓	✓	✓	✓
MUA	✓	×	×	✓	✓	✓	✓
KSA	✓	✓	×	✓	✓	✓	✓
SKA	×	×	×	✓	✓	✓	✓
MOA	×	×	×	✓	✓	✓	✓
DCA	×	×	×	✓	✓	✓	✓
REP	×	✓	×	✓	✓	×	✓
MIA	✓	×	×	✓	✓	✓	✓
IMA	✓	×	×	✓	✓	×	✓
UNT	✓	×	✓	✓	✓	✓	✓
SDA	✓	–	✓	✓	✓	✓	✓
SCA	✓	✓	✓	×	×	✓	✓

Note:- FNR: Functional Requirements; SPA: Server spoofing attack; ANM: Anonymity; MUA: Mutual authentication; KSA: Known session-key attack; SKA: Session key agreement; MOA: Modification attack; DCA: Drone Capture attack; REP: Replay attack; MIA: Man-in-the-middle attack; IMA: Impersonation attack; UNT: Untraceability; SDA: Stolen smart device attack; SCA: Scalability/In-Correctness✓: Secure or provides; ×:Insecure against or does not provide

TABLE 3. Experimental Results.

↓Operation/ Device→	User	Control Server	Drone
T_b: Bilinear-Pairing	17.36	4.038	12.52
T_{ecm}: ECC Point-Multiplication	5.116	0.926	4.107
T_{eca}: ECC Point-Addition	0.013	0.006	0.018
T_h: One-way Hash	0.009	0.004	0.006
T_{sym}: Symmetric Encryption	0.017	0.008	0.013

where ✓ verifies that mentioned protocol provides the functionality or resists the attack. Whereas × confirms that the mentioned protocol lacks the required feature, and – implies that the feature is not applicable or cannot be determined. The Table 2 provides concrete evidence that the proposed protocol provides all required features. In contrast, other protocols have weaknesses against one or more features.

B. COMPUTATION ANALYSIS

Detailed comparisons concerning the computation cost of different protocols are presented in this section. Furthermore, as per the experimental setup mentioned in [31], performed using MIRACL library over three corresponding devices: Xiaomi-Redme Note-8 equipped with OctaCore Max-2.01GHz processor, 4-GB RAM Andriod V-9 and MIUI V-11.0.7. The smartphone simulates a user/mobile device. To Control-center (CS), we experimented HP-EliteBook 8460P-Intel(R), Core(TM)-i7-2620M,2.7GHz Processor, 4-GB RAM, and Ubuntu 16.0 LTS OS; whereas we use Pi3-B+ with 1.4GHz processor (Cortex-A53(ARMv8)) and 1-GB RAM to simulate a drone. The running times of the used computation operations are depicted in Table 3.

The Table 4 depicts that the introduced scheme has less computation cost than the schemes presented in [22], [26], [27], [32], [33], and [35]. The cryptographic operations performed in the proposed are reduced, and an efficient scheme has been introduced as shown in the Table 4.

TABLE 4. Running time comparisons.

Protocol	User	Control Server	Drone	Total Cost (ms)
[22]	$16T_h + 1T_{fe}$	$9T_h$	$7T_h$	5.33
[35]	$4T_{ecm}$	-	$5T_{ecm}$	40.9
[33]	$10T_h$	$7T_h$	$7T_h$	0.16
[26]	$1T_{fe} + 12T_h$	$9T_h$	$1T_{fe} + 8T_h$	9.415
[27]	$3T_{ecm} + 4T_{sym} + 9T_h$	$1T_{ecm} + 3T_{sym} + 4T_h$	$2T_{ecm} + 2T_{sym} + 7T_h$	25.233
[32]	$2T_{ecm} + 6T_h$	$8T_h$	$2T_{ecm} + 5T_h$	18.562
Our	$7T_h$	$6T_h + 1T_{sym}$	$3T_h$	0.11

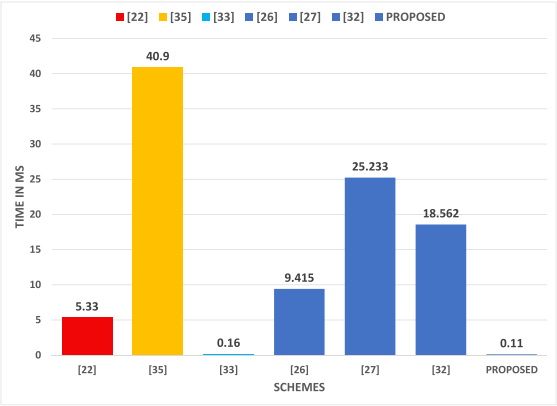


FIGURE 6. Comparison based on computation cost.

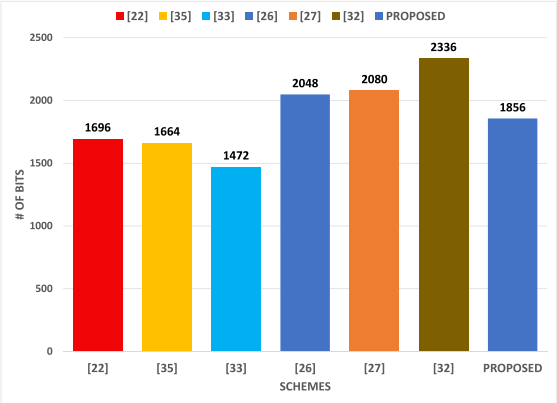


FIGURE 7. Comparison of communication cost.

Therefore, it could be concluded that our scheme provides all security requirements, and compared to the rest of the schemes [22], [33], [35], our method is best suited in practical drone scenarios due to its' security provisions and lightweight properties.

C. COMMUNICATION ANALYSIS

The Table 5 depicts the communication overhead of the proposed scheme and related schemes. To comprehend the comparisons, the subsequent assumption as per the sizes of several parameters are considered as follows: random number and symmetric encryption blocks are regarded as 128 bits of length, and the identities, time stamps, and ECC parameters

TABLE 5. Communication cost comparison.

Protocol	Message Exchanged	Bits Exchanged
Wazid et al. [22]	3	1696
Singh et al. [35]	2	1664
Zhang et al. [33]	3	1472
Yu et al. [26]	4	2048
Tanveer et al. [27]	3	2080
Nikooghadam et al. [32]	3	2336
Our	3	1856

are fixed at the size of 160 bits. Due to the duality of the points (P_x, P_y), the size of an ECC point is $160 + 160 = 320$ bits.

The communication cost of the various protocols is shown in the Table 5 and also in Figure 7.

As per the assumed values, the communication cost of the authentication phase of the proposed scheme is 1856 bits.

The Table 5 shows that the proposed scheme introduced some extra communication cost in comparison to the related schemes [22], [33], [35], Yu et al. [26], Tanveer et al. [27], and Nikooghadam et al. [32]. Nevertheless, the proposed scheme has better security and offers the least communication cost.

IX. CONCLUSION

The privacy and security issues related to IoD are expanding as their adaption surges. To subdue the privacy and security issues related to drones, Zhang et al. introduced an authentication protocol for drones. We have reviewed and cryptanalyzed Zhang et al.'s protocol and found many vulnerabilities. Further, we have initiated the protocol to overcome the vulnerabilities found in Zhang et al.'s protocol while preserving its benefits simultaneously. The security and comparative analysis of the introduced protocol has been performed to show that the introduced protocol is secure and provides better security features with low communication and computation overheads.

REFERENCES

[1] S. Banerjee, V. Odelu, A. K. Das, J. Srinivas, N. Kumar, S. Chattopadhyay, and K. R. Choo, "A provably secure and lightweight anonymous user authenticated session key exchange scheme for Internet of Things deployment," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8739–8752, Oct. 2019.

[2] M. Gharibi, R. Boutaba, and S. L. Waslander, "Internet of Drones," *IEEE Access*, vol. 4, pp. 1148–1162, 2016.

[3] R. Hassan, F. Qamar, M. K. Hasan, A. H. M. Aman, and A. S. Ahmed, "Internet of Things and its applications: A comprehensive survey," *Symmetry*, vol. 12, no. 10, p. 1674, Oct. 2020.

[4] A. H. M. Aman, A.-H.-A. Hashim, H. A. M. Ramli, and S. Islam, "Packet loss and packet delivery evaluation using network simulator for multicast enabled network mobility management," *Int. J. Future Gener. Commun. Netw.*, vol. 10, no. 4, pp. 41–50, Apr. 2017.

[5] M. Bae and H. Kim, "Authentication and delegation for operating a multi-drone system," *Sensors*, vol. 19, no. 9, p. 2066, May 2019.

[6] D. Dolev and A. C. Yao, "On the security of public key protocols," *IEEE Trans. Inf. Theory*, vol. IT-29, no. 2, pp. 198–208, Mar. 1983.

[7] W.-H. Yang and S.-P. Shieh, "Password authentication schemes with smart cards," *Comput. Secur.*, vol. 18, no. 8, pp. 727–733, Jan. 1999.

[8] S. A. Chaudhry, A. Irshad, J. Nebhen, A. K. Bashir, N. Moustafa, Y. D. Al-Otaibi, and Y. B. Zikria, "An anonymous device to device access control based on secure certificate for Internet of Medical Things systems," *Sustain. Cities Soc.*, vol. 75, Dec. 2021, Art. no. 103322.

[9] A. Irshad, S. A. Chaudhry, O. A. Alomari, K. Yahya, and N. Kumar, "A novel pairing-free lightweight authentication protocol for mobile cloud computing framework," *IEEE Syst. J.*, vol. 15, no. 3, pp. 3664–3672, Sep. 2021.

[10] M. Tanveer, G. Abbas, Z. H. Abbas, M. Bilal, A. Mukherjee, and K. S. Kwak, "LAKE-6SH: Lightweight user authenticated key exchange for 6LoWPAN-based smart homes," *IEEE Internet Things J.*, vol. 9, no. 4, pp. 2578–2591, Feb. 2022.

[11] M. A. Saleem, S. H. Islam, S. Ahmed, K. Mahmood, and M. Hussain, "Provably secure biometric-based client–server secure communication over unreliable networks," *J. Inf. Secur. Appl.*, vol. 58, May 2021, Art. no. 102769.

[12] S. H. Islam, M. S. Obaidat, and R. Amin, "An anonymous and provably secure authentication scheme for mobile user," *Int. J. Commun. Syst.*, vol. 29, no. 9, pp. 1529–1544, Jun. 2016.

[13] M. F. Ayub, S. Shamshad, K. Mahmood, S. H. Islam, R. M. Parizi, and K. R. Choo, "A provably secure two-factor authentication scheme for USB storage devices," *IEEE Trans. Consum. Electron.*, vol. 66, no. 4, pp. 396–405, Nov. 2020.

[14] M. Tanveer, A. H. Zahid, M. Ahmad, A. Baz, and H. Alhakami, "LAKE-IoD: Lightweight authenticated key exchange protocol for the Internet of Drone environment," *IEEE Access*, vol. 8, pp. 155645–155659, 2020.

[15] S. H. Islam, A. K. Das, and M. K. Khan, "A novel biometric-based password authentication scheme for client-server environment using ECC and fuzzy extractor," *Int. J. Ad Hoc Ubiquitous Comput.*, vol. 27, no. 2, pp. 138–155, 2018.

[16] M. A. Ferrag, L. A. Maglaras, H. Janicke, J. Jiang, and L. Shu, "Authentication protocols for Internet of Things: A comprehensive survey," *Secur. Commun. Netw.*, vol. 2017, pp. 1–41, 2017.

[17] J.-P. Yaacoub, H. Noura, O. Salman, and A. Chehab, "Security analysis of drones systems: Attacks, limitations, and recommendations," *Internet Things*, vol. 11, Sep. 2020, Art. no. 100218.

[18] S. Challa, A. K. Das, V. Odelu, N. Kumar, S. Kumari, M. K. Khan, and A. V. Vasilakos, "An efficient ECC-based provably secure three-factor user authentication and key agreement protocol for wireless healthcare sensor networks," *Comput. Electr. Eng.*, vol. 69, pp. 534–554, Jul. 2018.

[19] Z. Ali, A. Ghani, I. Khan, S. A. Chaudhry, S. H. Islam, and D. Giri, "A robust authentication and access control protocol for securing wireless healthcare sensor networks," *J. Inf. Secur. Appl.*, vol. 52, Jun. 2020, Art. no. 102502.

[20] A. K. Das, M. Wazid, N. Kumar, A. V. Vasilakos, and J. J. P. C. Rodrigues, "Biometrics-based privacy-preserving user authentication scheme for cloud-based industrial Internet of Things deployment," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4900–4913, Dec. 2018.

[21] S. Hussain and S. A. Chaudhry, "Comments on 'biometrics-based privacy-preserving user authentication scheme for cloud-based industrial Internet of Things deployment,'" *IEEE Internet Things J.*, vol. 6, pp. 10936–10940, 2019.

[22] M. Wazid, A. K. Das, N. Kumar, A. V. Vasilakos, and J. J. P. C. Rodrigues, "Design and analysis of secure lightweight remote user authentication and key agreement scheme in Internet of Drones deployment," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3572–3584, Apr. 2019.

[23] Z. Ali, B. A. Alzahrani, A. Barnawi, A. Al-Barakati, P. Vijayakumar, and S. A. Chaudhry, "TC-PSLAP: Temporal credential-based provably secure and lightweight authentication protocol for IoT-enabled drone environments," *Secur. Commun. Netw.*, vol. 2021, pp. 1–10, Dec. 2021.

[24] J. Srinivas, A. K. Das, N. Kumar, and J. J. P. C. Rodrigues, "TCALAS: Temporal credential-based anonymous lightweight authentication scheme for Internet of Drones environment," *IEEE Trans. Veh. Technol.*, vol. 68, no. 7, pp. 6903–6916, Jul. 2019.

[25] S. A. Chaudhry, J. Nebhen, A. Irshad, A. K. Bashir, R. Kharel, K. Yu, and Y. B. Zikria, "A physical capture resistant authentication scheme for the Internet of Drones," *IEEE Commun. Standards Mag.*, vol. 5, no. 4, pp. 62–67, Dec. 2021.

[26] S. Yu, A. K. Das, Y. Park, and P. Lorenz, "SLAP-IoD: Secure and lightweight authentication protocol using physical unclonable functions for Internet of Drones in smart city environments," *IEEE Trans. Veh. Technol.*, vol. 71, no. 10, pp. 10374–10388, Oct. 2022.

[27] M. Tanveer, A. U. Khan, N. Kumar, and M. M. Hassan, "RAMP-IoD: A robust authenticated key management protocol for the Internet of Drones," *IEEE Internet Things J.*, vol. 9, no. 2, pp. 1339–1353, Jan. 2022.

[28] S. A. Chaudhry, K. Yahya, M. Karuppiah, R. Kharel, A. K. Bashir, and Y. B. Zikria, "GCACS-IoD: A certificate based generic access control scheme for Internet of Drones," *Comput. Netw.*, vol. 191, May 2021, Art. no. 107999.

[29] A. K. Das, B. Bera, M. Wazid, S. S. Jamal, and Y. Park, "IGCACS-IoD: An improved certificate-enabled generic access control scheme for Internet of Drones deployment," *IEEE Access*, vol. 9, pp. 87024–87048, 2021.

[30] M. Zhang, C. Xu, S. Li, and C. Jiang, "On the security of an ECC-based authentication scheme for Internet of Drones," *IEEE Syst. J.*, vol. 16, no. 4, pp. 6425–6428, Dec. 2022.

[31] S. Hussain, S. A. Chaudhry, O. A. Alomari, M. H. Alsharif, M. K. Khan, and N. Kumar, "Amassing the security: An ECC-based authentication scheme for Internet of Drones," *IEEE Syst. J.*, vol. 15, no. 3, pp. 4431–4438, Sep. 2021.

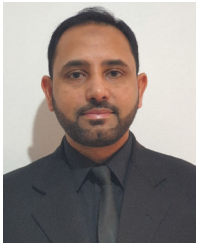
[32] M. Nikooghadam, H. Amintoosi, S. H. Islam, and M. F. Moghadam, "A provably secure and lightweight authentication scheme for Internet of Drones for smart city surveillance," *J. Syst. Archit.*, vol. 115, May 2021, Art. no. 101955.

[33] Y. Zhang, D. He, L. Li, and B. Chen, "A lightweight authentication and key agreement scheme for Internet of Drones," *Comput. Commun.*, vol. 154, pp. 455–464, Mar. 2020.

[34] V. Cheval and B. Blanchet, "Proving more observational equivalences with ProVerif," in *Proc. Int. Conf. Princ. Secur. Trust*. Cham, Switzerland: Springer, 2013, pp. 226–246.

[35] J. Singh, A. Gimekar, and S. Venkatesan, "An efficient lightweight authentication scheme for human-centered industrial Internet of Things," *Int. J. Commun. Syst.*, p. e4189, Nov. 2019.

SAJID HUSSAIN received the M.S. degree from International Islamic University, Islamabad, in 2018. Currently, he is a Lecturer with Air University Islamabad, Pakistan. He has published several articles at top venues, including IEEE Internet of Things Journal, IEEE Systems Journal, and *Computer Standards & Interfaces* (Elsevier). His research interests include computer networking, network security, network communication, information security, cryptography, elliptic/hyper elliptic curve cryptography, encryption, and authentication.



MUHAMMAD FAROOQ is currently an Instructor in mathematics and statistics with the College of Arts and Sciences, Abu Dhabi University, Abu Dhabi, United Arab Emirates. Before this, he was with Mohammad Ali Jinnah University, Karachi, Pakistan, for a period of six years, then he joined the Al Zahra College for Women, Muscat, Oman, in 2010. His current research interests include cryptography and information security. He was also a Coordinator in mathematics and statistics courses and a member of the Curriculum Review and Retention and Engagement Committees.

BANDER A. ALZAHRANI received the M.Sc. degree in computer security and the Ph.D. degree in computer science from Essex University, U.K., in 2010 and 2015, respectively. He is currently an Associate Professor with King Abdulaziz University, Saudi Arabia. He has published more than 70 research papers in international journals and conferences. His research interests include wireless sensor networks, information centric networks, bloom filter data structure and its applications, secure content routing, and authentication protocols in IoT.

AIIAD ALBESHRI received the M.S. and Ph.D. degrees in information technology from the Queensland University of Technology, Brisbane, Australia, in 2007 and 2013, respectively. He has been an Associate Professor with the Computer Science Department, King Abdulaziz University, Jeddah, Saudi Arabia, since 2018. His current research interests include security and trust in cloud computing and big data.

KHALID ALSUBHI received the M.S. and Ph.D. degrees in computer science from Waterloo University, Waterloo, Canada, in 2009 and 2016, respectively. His research interests include intrusion detection systems, privacy of healthcare systems, big data, and resource management in distributed systems.

SHEHZAD ASHRAF CHAUDHRY (Member, IEEE) received the master's and Ph.D. degrees (Hons.) from International Islamic University, Pakistan. Currently, he is an Associate Professor in cybersecurity engineering with the Department of Computer Science and Information Technology, College of Engineering, Abu Dhabi University, Abu Dhabi, United Arab Emirates. Before this, he was with Istanbul Gelisim University, Turkey, University of Sialkot, Pakistan, and International Islamic University, Islamabad, Pakistan. He has supervised more than 40 graduate students in their research. Working in the field of information and communication security, he has published extensively in prestigious venues, such as *IEEE Communications Standards Magazine*, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE INTERNET OF THINGS JOURNAL, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE TRANSACTIONS ON INDUSTRY APPLICATIONS, IEEE TRANSACTIONS ON RELIABILITY, *ACM Transactions on Internet Technology*, Sustainable Cities and Society (Elsevier), *FGCS*, *IJEPES*, *Computer Networks*, *Computer Communications*, and *Digital Communications and Networks*. He has over 170 publications and with an H-index of 42, I-10 index of 99, and accumulate impact factor of more than 400. He has also published more than 135 SCI/E indexed manuscripts and has been cited more than 4700 times. His current research interests include lightweight cryptography, elliptic/hyper elliptic curve cryptography, multimedia security, E-payment systems, MANETs, SIP authentication, smart grid security, IP multimedia subsystems, and next generation networks. He was awarded a gold medal for achieving maximum distinction of 4/4 CGPA in his maters. In 2018, considering his research, Pakistan Council for Science and Technology granted him the Prestigious Research Productivity Award, while affirming him among Top Productive Computer Scientist in Pakistan. For the consecutive three years (i.e., 2020, 2021, and 2022), he is being listed among top 2% computer scientists across the world in Stanford University's report.