BAKP-IoDA: Blockchain Driven Authentication and Key Agreement Protocol for Internet of Drones Based Applications

Amarjit Sripesh
Department of Computer Science and Engineering, Graphic Era Deemed to be University,
Dehradun 248 002, India
sripesh.amarjit.2017458@gmail.com

Mohammad Wazid
Department of Computer Science and Engineering, Graphic Era Deemed to be University,
Dehradun 248 002, India
wazidkec2005@gmail.com

D. P. Singh
Department of Computer Science and Engineering, Graphic Era Deemed to be University,
Dehradun 248 002, India
devesh.geu@gmail.com

Ashok Kumar Das
Center for Security, Theory and Algorithmic Research,
International Institute of Information Technology, Hyderabad 500 032, India
iitkgp.akdas@gmail.com,ashok.das@iiit.ac.in

Bharat Verma
Centric Consulting, Gurugram,
Haryana 122015, India
erbharat33@gmail.com

ABSTRACT

Recently, Internet of Drones (IoD) has emerged as an important topic to research in academy and industry due to its generic services for various drone applications. In an IoD environment, the deployed drones and the ground station server (GSS) communicate over an open network which leads to security and privacy breaches. To mitigate these issues, a blockchain driven authentication and key establishment protocol for secure communication in IoD has been designed (in short, it is called as BAKP-IoDA). The security analysis of BAKP-IoDA shows its robustness against different attacks. During the performance comparison, it has been observed that BAKP-IoDA maintains superior security along with extra functionality features with the competing schemes. Additionally, the blockchain based practical demonstration of BAKP-IoDA shows its impact on various important network performance parameters, like computation time and transactions per second.

CCS CONCEPTS

• **Networks** → **Security protocols**; • **Security and privacy** → **Authentication**.

KEYWORDS

Internet of Drones (IoD), authentication, key agreement, blockchain, security.

ACM Reference Format:
Amarjit Sripesh, Mohammad Wazid, D. P. Singh, Ashok Kumar Das, and Bharat Verma. 2022. BAKP-IoDA: Blockchain Driven Authentication and Key Agreement Protocol for Internet of Drones Based Applications. In *5th ACM Workshop on Drone-Assisted Wireless Communications for 5G and Beyond (DroneCom '22), October 17, 2022, Sydney, NSW, Australia.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3555661.3560859

DroneCom '22, October 17, 2022, Sydney, NSW, Australia

ACM ISBN 978-1-4503-9514-4/22/10…$15.00
https://doi.org/10.1145/3555661.3560859

1 INTRODUCTION

The "Internet of Drones (IoD)" is considered as a network architecture that is specifically designed to facilitate communications between autonomous flying vehicles (unmanned aerial vehicles (UAVs)) and a number of network entities located on the ground having the ground station servers (GSSs) and users [3]. Due to its excellent flexibility to a wide range of difficult circumstances, IoD has recently acquired space. UAVs can, in fact, be successfully used for a variety of purposes, including "agriculture, search and rescue operations, security and surveillance systems and mission-critical services" [1]. Despite the fact that IoD has a lot of potential applications, it has some information security related issues. As in IoD, most of the communication happens through the public channel, the potential attackers get some chance to compromise the security of IoD. The exchanged data can be leaked to the unauthorized parties, or may be altered in unauthorized way. Therefore, we need some security mechanisms, like authentication, access control and key management to secure the communication in IoD. The security of these schemes can be further improved with the inclusion of blockchain technology as blockchain is a tamper proof technology and also various advantages (blockchain is immutable, anonymous, distributed and transparent). In this paper, we focus on designing of a blockchain driven authentication and key establishment protocol for secure communication in IoD.

1.1 Research contributions

The research contributions of the paper are given below.

- A blockchain driven authentication and key establishment protocol for secure communication of IoD is proposed (in short BAKP-IoDA) that provides secure mutual authentication and key establishment between the drones and the GSS.

- The network and threat models of BAKP-IoDA are provided for the better understanding of the proposed security mechanism.
- The security analysis of BAKP-IoDA proves its robustness against various possible potential attacks required in IoD deployment. BAKP-IoDA also provides superior security and extra functionality features while these are compared with the existing competing approaches.
- Practical demonstration of BAKP-IoDA using blockchain simulation shows its impact on the important performance parameters.

2 RELATED WORK

Cho *et al.* [6] proposed an authentication scheme for UAVs in an IoD environment. It provides the mitigation of various attacks, which can happen via malicious drones deployment in IoD. Ever [10] proposed an authentication scheme for an IoD environment with the use of "Elliptic Curve Cryptography (ECC)" techniques. In their scheme, several sensors are deployed in different areas which are then formed into the clusters. The sensor nodes are linked with a cluster head. The cluster heads communicate among each other as well as with the deployed UAVs.

Fang *et al.* [9] presented another authentication scheme for an "Internet of Things (IoT)" environment. In their scheme, the deployment of heterogeneous-type IoT smart devices is done on the basis of a trust model. Bera *et al.* [2] proposed a "blockchain driven access control scheme" for an IoD communication. In their scheme, the access control was provided among the drones and GSS.

Very recently, Chaudhry *et al.* [5] proposed another access control protocol for an IoD communication. However, their protocol was shown to be insecure against different potential attacks and it also fails to achieve the anonymity property [7].

In summary, the existing schemes proposed for IoD are vulnerable to various attacks and also lack more functionality features. Therefore, it is necessary to propose a new authentication and key establishment scheme for the secure communication in IoD based applications.

3 SYSTEM MODELS

In this section, we provide the details of the system models of the proposed BAKP-IoDA.

3.1 Network model

The network model of the proposed blockchain driven authentication protocol for IoD based applications is given Fig. 1. In the given model, we have flying drones (DR), deployed GSSs and cloud servers (CSs), where the cloud servers form a peer-to-peer cloud servers network (P2PCS). Drones fly as per the assigned flying zone, collect the data of their surroundings and then send it to the respective GSS. Next, the GSS creates encrypted transactions from the received data and creates partial blocks from them. Each partial block has information, like owner GSS_j's identity (i.e., ID_{GSS_j}), owner's public key (i.e., Q_{GSS_j}) and the encrypted transactions (i.e., $E_{Q_{GSS_j}}(T_X)$), where GSS_j, Q_{GSS_j}, and T_X represent the j^{th} GSS and its ECC-based public key, and a transaction, respectively. After calculating a partial block $PBLK$, GSS_j sends it to the cloud server CS_k of P2PCS network for its consensus and addition in the blockchain BC. After receiving $PBLK$ from GSS_j, CS_k makes a full block BLK from it by adding other desirable information, such as a) hash of previous block, b) hash of this block, c) timestamp value, d) Merkle tree root value, e) random nonce value, f) owner's identity, g) owner's public key, h) encrypted transactions, and i) signature on this block. After that CS_k initiates the consensus of BLK. In case of successful consensus of BLK is added in the blockchain BC for its further use. For the execution of consensus process algorithms, like "Practical Byzantine Fault Tolerance (pBFT)" or "Proof of Work (PoW)" can be used. This communication environment has security related issues as it may be vulnerable to various information security related attacks. Therefore, the different communications, such as drone-to-drone, drone-to-ground station server, ground station server-to-cloud server and cloud server-to-cloud server should happen securely. For such purpose, security protocols, like, mutual authentication and key establishment can be utilized. The trusted registration authority (TA) is the trusted entity of the network, and it performs the registration of the other legitimate entities.

3.2 Threat model

In this section, we discuss about the important threats of this communication environment. The threats are covered under two important models: 1) "Dolev-Yao (DY) model" and 2) "Canetti and Krawczyk's adversary model (CK-adversary model)".

- Under the DY model, we assume that everyone has access to communication medium to the malicious actor (attacker). Due to that an attacker $\mathcal{A}$ has potential to change, alter, delay or drop the exchanged messages [8].
- The "CK-adversary model" is other essential model, which is utilized in the designing of BAKP-IoDA [4]. It states that $\mathcal{A}$ is capable like the DY model part from that he/she has the ability to obtain the secret data (for example, "session keys" of the established sessions). $\mathcal{A}$ can hijack the sessions and their associated data.
- $\mathcal{A}$ can do the physical stealing of "deployed/ flying drones". So, $\mathcal{A}$ then attempts to obtain secret data, like secret credentials and keys from their memory through the execution of "power analysis attacks" [11].
- It is important to highlight that GSSs are kept under protection to prevent their physical stealing. Due to that $\mathcal{A}$ is not able miss-use the stored secret values, like secret identities and keys to launch other potential attacks on BAKP-IoDA (for example, "impersonating, MiTM, and unauthorised session key computation attacks" [12]).
- The TA is the trusted entity of the network and performs the registration of the other legitimate entities.

4 THE PROPOSED PROTOCOL: BAKP-IODA

In this section, we provide the details of the BAKP-IoDA. It includes four important phases: a) "registration phase", b) "mutual authentication and key establishment phase", c) "dynamic device addition phase" and d) "blockchain implementation phase". The details of these phases are given below.

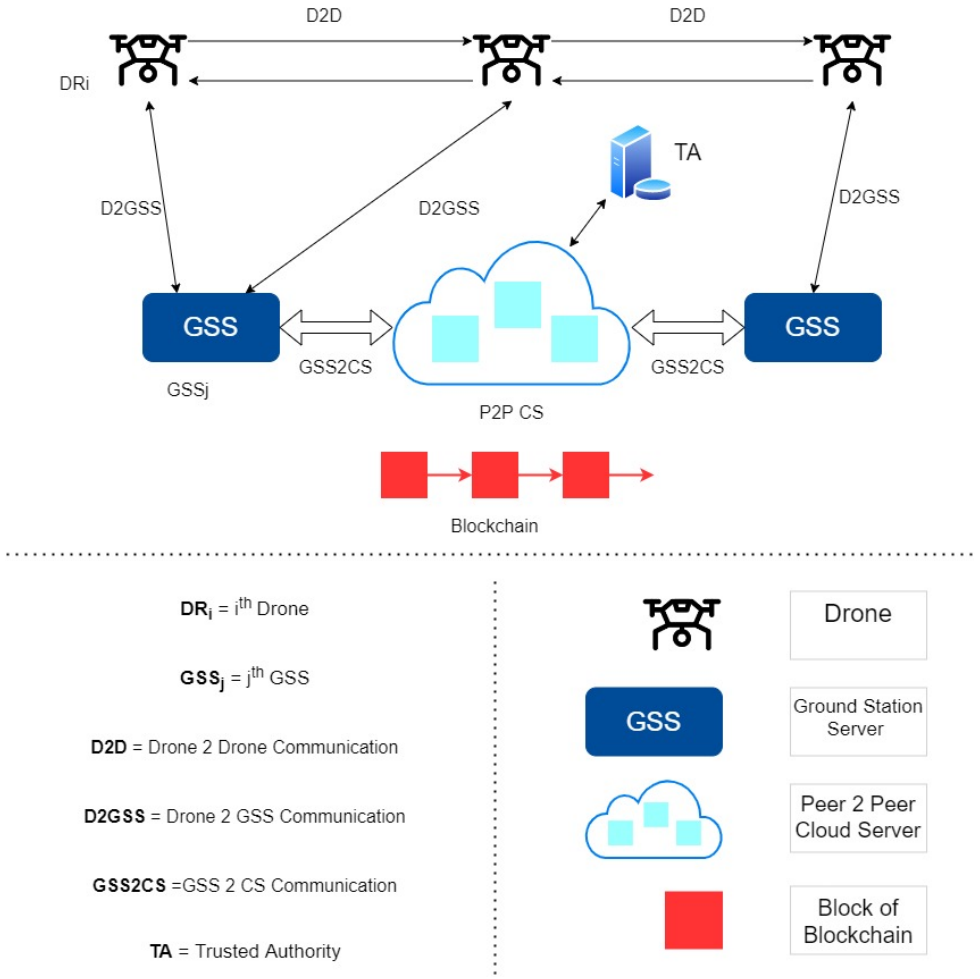

Figure 1: Network model of blockchain driven authentication protocol for IoD based applications

4.1 Registration phase

In this phase, the TA does the registration of other entities (i.e., drones DR, ground station server GSS, cloud server CS) of the network.

4.1.1 Registration of drones. Before the deployment of drones, the TA is required to register all the drones. The process is executed using the following steps:

- **RDR1:** TA generates an identity of drone DR_i as ID_{DR_i} and computes its pseudo identity as $RID_{DR_i} = h(ID_{DR_i} || k_{TA})$, where k_{TA} is the secret key of TA. TA again generates temporary identity of DR_i as TID_{DR_i}. Then TA generates a shared secret key of DR_i and GSS_j as $k_{DR_i-GSS_j}$.
- **RDR2:** Finally, TA stores $\{TID_{DR_i}, RID_{DR_i}, h(k_{DR_i-GSS_j} || RID_{DR_i}), h(\cdot)\}$ in the memory of DR_i and then DR_i is deployed in the respective flying zone as per the requirement.

4.1.2 Registration of ground station server. Before the deployment of a ground station server GSS_j, the TA is required to register all the GSS. The registration of GSS_j is conducted using the following steps:

- **RGSS1:** TA generates an identity of GSS_j as ID_{GSS_j} and computes its pseudo identity as $RID_{GSS_j} = h(ID_{GSS_j} || k_{TA})$, where k_{TA} is the secret key of TA.
- **RGSS2:** Finally, TA stores $\{\{(TID_{DR_i}, RID_{DR_i}, k_{DR_i-GSS_j})| i = 1, 2, \cdots, num_{DR}\}, RID_{GSS_j}, h(\cdot)\}$ in the memory (database) of GSS_j, where num_{DR} are the total number of drones under GSS_j.

In a similar way, the registration of each cloud server CS_k can be conducted.

4.2 Mutual authentication and key establishment phase

In this phase, DR_i verifies the authenticity of corresponding GSS_j and GSS_j verifies the authenticity of corresponding DR_i. After the successful completion of mutual authentication, a session key $SK_{DR_i-GSS_j} = SK_{GSS_k-DR_i}$ is established between a drone DR_i and a GSS, GSS_j. The process is executed using the following steps:

- **DGAK1:** DR_i initiates the process by generating a random number/ nonce r_1 and current timestamp T_1. Then, DR_i computes $M_1 = r_1 \oplus h(k_{DR_i-GSS_j} || RID_{DR_i})$ and $M_2 = h(r_1 || T_1 || h(k_{DR_i-GSS_j} || RID_{DR_i}))$. DR_i constructs a message $MSG_1 = \{TID_{DR_i}, M_1, M_2, T_1\}$ and sends it to GSS_j through an open channel.
- **DGAK2:** After receiving the message MSG_1 from DR_i, GSS_j checks the timeliness of timestamp value T_1 using the verifying condition: "$|T_1 - T_1^*| \leq \Delta T$, where ΔT represents the maximum transmission delay and T_1^* represents MSG_1's receiving time". If the condition holds, GSS_j executes Step DGAK3; otherwise, MSG_1 is rejected.
- **DGAK3:** GSS_j fetches the tuple $\{TID_{DR_i}, RID_{DR_i}, (k_{DR_i-GSS_j})\}$ corresponding to the received TID_{DR_i}. After that, GSS_j computes $r_1 = M_1 \oplus h(RID_{DR_i} || h(k_{DR_i-GSS_j} || RID_{DR_i})$ and $M_2' = h(r_1 || T_1 || h(k_{DR_i-GSS_j} || RID_{DR_i}))$ to check the condition $M_2 = M_2'$. If the condition holds, DR_i is "authenticated with GSS_j"; otherwise, GSS_j aborts the session with DR_i.
- **DGAK4:** GSS_j generates a random number/nonce r_2 and current timestamp T_2. Then GSS_j computes $M_3 = h(r_2 || RID_{GSS_j}) \oplus h(k_{DR_i-GSS_j} || RID_{DR_i})$ and the session key as $SK_{GSS_j-DR_i} = h(r_i || h(r_2 || RID_{GSS_j}) T_1 || T_2 || h(k_{DR_i-GSS_j} || RID_{DR_i}))$. GSS_j also computes $M_4 = h(SK_{GSS_j-DR_i} || RID_{DR_i} || T_2)$. GSS_j generates new temporary identity $TID_{DR_i}^{new}$ for DR_i to compute $M_5 = TID_{DR_i}^{new} \oplus h(r_1 || h(r_2 || RID_{GSS_j}) || RID_{DR_i} || T_2)$. GSS_j then constructs the message $MSG_2 = \{M_3, M_4\ M_5\ T_2\}$ and sends it to DR_i through an open channel.
- **DGAK5:** After receiving the message MSG_2 from GSS_j, DR_i checks the timeliness of timestamp value T_2 using the condition: "$|T_2 - T_2^*| \leq \Delta T$, where T_2^* represents MSG_2's receiving time". If the condition holds, GSS_j executes Step DGAK6; otherwise, MSG_2 is rejected.
- **DGAK6:** DR_i computes $h(r_2 || RID_{GSS_j}) = M_3 \oplus h(RID_{DR_i} || h(k_{DR_i-GSS_j} || RID_{DR_i}))$ and then the session key as $SK_{DR_i-GSS_j} = h(r_1 || h(r_2 || RID_{GSS_j}) || T_1 || T_2 || h(k_{DR_i-GSS_j} || RID_{DR_i}))$. DR_i calculates $M_4' = h(SK_{DR_i-GSS_j} || RID_{DR_i} || T_2)$ and checks whether $M_4' = M_4$? If it is so, "GSS_j is successfully authenticated with DR_i" and the "computed session key is correct". DR_i then computes the new temporary identity, $TID_{DR_i}^{new} = M_5 \oplus h(r_1 || h(r_2 || RID_{GSS_j}) || RID_{DR_i} || T_2)$ and replaces TID_{DR_i} with $TID_{DR_i}^{new}$ in its memory for its use in the near future session. DR_i generates fresh timestamp value T_3 and calculates the session key verifier as $M_{SKV} = h(SK_{DR_i-GSS_j} || T_3)$. Finally, DR_i constructs a message, MSG_3 as $\{M_{SKV}, T_3\}$ and sends it to GSS_j via a public channel.
- **DGAK7:** After receiving the message MSG_3 from DR_i, GSS_j checks the timeliness of timestamp value T_3 using the condition: "$|T_3 - T_3^*| \leq \Delta T$, where T_3^* represents MSG_3's receiving time". If the condition does not hold, MSG_3 is rejected. GSS_j computes $M_{SKV}' = h(SK_{GSS_j-DR_i} || T_3)$ and checks the condition: $M_{SKV}' = M_{SKV}$? If this condition holds, GSS_j assumes that "DR_i has computed the correct session key".

At the end of this phase, both GSS_j and DR_i establish the common session key $SK_{DR_i-GSS_j} = SK_{GSS_j-DR_i}$ for their secure communication.

4.3 Dynamic device addition phase

It is always required to provide the facility like dynamic addition of new devices (i.e., drones) in the network, beacuse some of the devices can be malfunctioned or physically compromised by the attacker. This phase is executed with the help of the following steps:

- **RDR1:** TA generates an identity of the new drone DR_i^{new} as $ID_{DR_i}^{new}$ and computes its pseudo identity as $RID_{DR_i}^{new} = h(ID_{DR_i}^{new} || k_{TA})$, where k_{TA} is the secret key of TA. TA again generates temporary identity of DR_i^{new} as $TID_{DR_i}^{new}$. Then TA generates a shared secret key of DR_i^{new} and GSS_j as $k_{DR_i^{new}-GSS_j}$.
- **RDR2:** TA stores $\{TID_{DR_i}^{new}, RID_{DR_i}^{new}, h(k_{DR_i^{new}-GSS_j} || RID_{DR_i}^{new}), h(\cdot)\}$ in the memory of DR_i^{new} and then DR_i^{new} is deployed in the respective flying zone as per the requirement. TA also sends the registration information of DR_i^{new} to the corresponding GSS in the secure way.

4.4 Blockchain implementation phase

In this phase, a blockchain BC is implemented, which stores the data of the drones in the form of encrypted transactions. The details are given using the following steps:

- DR_i flies as per the assigned flying zone and collects the data of its surroundings and then sends it to GSS_j in the secure way through established $SK_{DR_i-GSS_j}$. Then GSS_j creates encrypted transactions from the received data and creates partial block $PBLK$ from it. Each partial block has information, like owner's identity (i.e., ID_{GSS_j}), owner's public key (i.e., Q_{GSS_j}) and the encrypted transactions (i.e., $E_{Q_{GSS_j}}(T_X)$). After calculating the partial block $PBLK$, GSS_j sends it to the cloud server CS_k of P2PCS network for its consensus and addition in the blockchain BC. The communication between GSS_j and CS_k happen in the secure way through the computed and established session key $SK_{GSS_j-CS_k}$.
- After receiving $PBLK$ from GSS_j, CS_k makes full block BLK from it by adding other desirable information (i.e., hash of previous block, hash of this block, timestamp value, Merkle tree root value, random nonce value, owner's identity, owner's public key, encrypted transactions, signature on this block). Then, CS_k initiates the consensus of the BLK. In case of successful consensus of BLK, it is added in the blockchain BC for further use. For the execution of consensus process, pBFT algorithm can be used. The data (transactions) stored in the blocks can be further utilized for various purposes (i.e., weather forecasting, smart farming, security and surveillance).

5 SECURITY ANALYSIS

5.1 Replay attack

In BAKP-IoDA, we have used different timestamp values i.e., T_i, where $i = 1, 2, 3$, which are also verified at the recipient's end using

the condition $|T_i - T_i^*| \leq \Delta T$, where ΔT represents the maximum transmission delay and T_i^* represents message's receiving time. If this condition holds then message is treated as the fresh one and accepted by the recipient. In this way proposed scheme provide security against the replay attack.

5.2 Man-in-the-middle (MiTM) and impersonation attacks

In this BAKP-IoDA, we have different secret values i.e., k_{TA}, $k_{DR_i-GSS_j}$, which are unknown $\mathcal{A}$. Without knowing these secret values $\mathcal{A}$ can not compute or modify the values of exchanged messages MSG_1, MSG_2, MSG_3. In these circumstances, $\mathcal{A}$ does not have ability launch MiTM and impersonation attacks on BAKP-IoDA.

5.3 Privileged insider attack

In BAKP-IoDA, there is provision for the deleting of secret values i.e., $k_{DR_i-GSS_j}$ when an entity has been registered. Therefore, these secret values are not available to the privileged insider user of TA. Hence privileged insider user of TA with malicious intention can not launch other potential attacks (i.e., "MiTM, impersonation, credentials guessing") on BAKP-IoDA. Hence BAKP-IoDA is secured against the privileged insider attack

5.4 Physical drone capture attack

In BAKP-IoDA, each drone has some registration information, which are unique for each drone. If a drone is physically compromised by $\mathcal{A}$. Then $\mathcal{A}$ may use the steps of sophisticated power analysis attack to extract the secret information from its memory. However, such kind of malicious tasks are not that much helpful to $\mathcal{A}$ as each drone has distinct credentials and different computed and established session key. The physical compromising of a specific drone does not impact the security of the entire network and remaining part of the communication is still safe and secure. Hence BAKP-IoDA is secured against physical drone capture attack.

5.5 Stolen verifier attack

In BAKP-IoDA, we store different values in the secured region of the database of cloud servers. Moreover cloud servers do not store the secret credentials of drones. Apart from that GSSs are under the physical security. Under these circumstances, the secret credentials of various entities are not available to $\mathcal{A}$. Without knowing the secret values, $\mathcal{A}$ can not launch stolen verifier attack and other associated attacks on proposed scheme. Hence BAKP-IoDA is secured against stolen verifier attack.

5.6 Ephemeral secret leakage (ESL) attack

In BAKP-IoDA, session key computed using the long term secret values (like, secret keys and identities i.e., $k_{DR_i-GSS_j}$, RID_{DR_i}, RID_{GSS_j}) and short term secret values (like, random nonce and timestamp values i.e., T_1, r_1). Due to the use these values, we get separate session keys for different entities in different sessions. Therefore, $\mathcal{A}$ does not get any chance to compromise the security of the session key. Hence BAKP-IoDA provides security to session key under the ephemeral secret leakage (ESL) attack.

5.7 Anonymity and untraceability

In proposed scheme, we do not exchange any identity in the plain text format. Moreover, each exchanged message is computed through the freshly generated timestamp value and random nonce values. There is also the provision of updating of temporary identity of drone in each session. Due to this mechanism, it is very different for $\mathcal{A}$ to trace the exchanged messages. Therefore, protocol supports anonymity and untraceability properties.

5.8 Other potential attacks

In BAKP-IoDA, we have used the mechanism of blockchain. The entire data of the communication system is maintained in the form of blockchain over the P2PCS network. Due to this mechanism, it is very difficult for $\mathcal{A}$ to reveal or update the data stored in the form of blockchain. $\mathcal{A}$ can not launch database related attacks i.e., SQL injection, cross-site scripting (XSS) on BAKP-IoDA.

6 PRACTICAL DEMONETISATION

In this section, we provide the practical implementation of the BAKP-IoDA.

The experiments were conducted on windows 11-21H2, 64 bit operating system having 8 GB of random access memory (RAM) and Intel (R) core (TM) i5-8250 U with 160 GHZ-180 GHZ processor. It had java SE development kit version 13.0.2. The programming was done in JavaScript. For the blockchain consensus, we have taken proof of work (PoW) consensus mechanism. In which we set level of difficulty as 3. We have considered three cases, the number of drones were considered as 10, 20, and 30 in case-1, case-2 and case-3, respectively. The number of mined blocks were 20, 40 and 80 in case-1, case-2 and case-3, respectively. In each block, we have 12 encrypted transactions. The results and outcomes of practical implementation are given below.

6.1 Estimation of computation time

The computation time (sec) is the time, which is required for the implemented blockchain for the completion of all required steps. The values of computation time (sec) are 1.04 sec, 1.80 sec, and 3.26 sec under case-1, case-2 and case-3, respectively. Here it important to mention that the values of computation time (sec) increase from case-1 to case-2 and case-2 to case-3 because in case-2 and case-3, we have to mine more number of blocks, which requires some extra time. The same results are reported in Fig. 2.

6.2 Estimation of transactions per second

The transactions per second (TPS) is another important performance parameter of blockchain's implementation. In each block, we have certain number transactions. TPS is value of number of transactions executed per second of time. The values of TPS are 231 sec, 265 sec, and 295 sec under case-1, case-2 and case-3, respectively. Here it important to mention that the values of TPS increase from case-1 to case-2 and case-2 to case-3 because in case-2 and case-3, we have to mine more number of blocks with more number transactions. Hence TPS values increase accordingly. The same results are reported in Fig. 3.

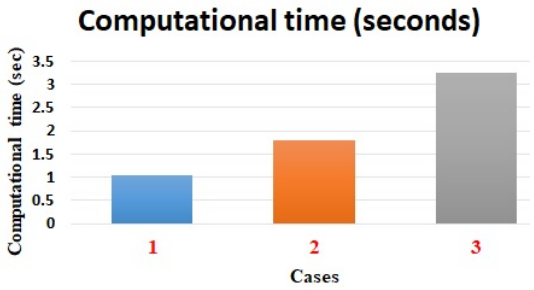

Figure 2: computation time values under different cases

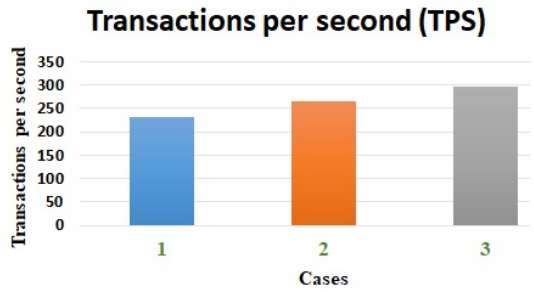

Figure 3: Transactions per second values under different cases

7 COMPARATIVE STUDY

In this section, we compare the security and functionality features of BAKP-IoDA with the other similar protocols of Internet of drones (IoD) i.e., the protocols of Bera *et al.* [2] and Chaudhry *et al.* [5].

The comparison is given in Table 1. Various features used in this table are as follows: *Note:* $S\phi_1$: "supports anonymity property"; $S\phi_2$: "supports untraceability property"; $S\phi_3$: "protection against replay attack"; $S\phi_4$: "protection against man-in-the-middle attack"; $S\phi_5$: "provides full mutual authentication"; $S\phi_6$: "provides key agreement"; $S\phi_7$:"prevents device/drone impersonation attack"; $S\phi_8$: "prevents GSS/ server impersonation attack"; $S\phi_9$:"prevents malicious device deployment attack"; $S\phi_{10}$: "prevents drone/ device physical capture attack"; $S\phi_{11}$: "provides session key security under CK-adversary model"; $S\phi_{12}$: "provides dynamic device/drone addition phase"; $S\phi_{13}$: "provides blockchain implementation"; Yes: "a protocol is secure or supports a particular functionality/security feature"; No: "the protocol is not secure or does not support a particular functionality/security feature".

The existing protocols lack in the essential functionality features and also vulnerable to various attacks. However, BAKP-IoDA provides extra functionality features with better security features. Therefore, BAKP-IoDA provides better security along with extra functionality features than other existing protocols.

8 CONCLUSION

A blockchain driven authentication and key establishment protocol for secure communication in IoD was presented, called BAKP-IoDA. The security analysis of BAKP-IoDA was conducted to proves its security against the various potential attacks. In the conducted comparative study, it was identified that BAKP-IoDA provided superior security along with extra functionality features as compared to those for other competing schemes. In the end, a practical implementation of BAKP-IoDA was given provided to see its effect on various network performance parameters through blockchain simulation.

Table 1: Comparison of security and functionality attributes

Feature	Bera *et al.* [2]	Chaudhry *et al.* [5]	BAKP-IoDA
$S\phi_1$	No	No	Yes
$S\phi_2$	No	No	Yes
$S\phi_3$	Yes	Yes	Yes
$S\phi_4$	Yes	Yes	Yes
$S\phi_5$	Yes	No	Yes
$S\phi_6$	Yes	Yes	Yes
$S\phi_7$	Yes	No	Yes
$S\phi_8$	Yes	No	Yes
$S\phi_9$	Yes	No	Yes
$S\phi_{10}$	Yes	No	Yes
$S\phi_{11}$	Yes	No	Yes
$S\phi_{12}$	Yes	Yes	Yes
$S\phi_{13}$	No	No	Yes

REFERENCES

[1] Abdelzahir Abdelmaboud. 2021. The Internet of Drones: Requirements, Taxonomy, Recent Advances, and Challenges of Research Trends. *Sensors* 21, 17 (2021).
[2] Basudeb Bera, Durbadal Chattaraj, and Ashok Kumar Das. 2020. Designing secure blockchain-based access control scheme in IoT-enabled Internet of Drones deployment. *Computer Communications* 153 (2020), 229–249.
[3] Pietro Boccadoro, Domenico Striccoli, and Luigi Alfredo Grieco. 2021. An extensive survey on the Internet of Drones. *Ad Hoc Networks* 122 (2021), 102600.
[4] R. Canetti and H. Krawczyk. 2001. Analysis of key-exchange protocols and their use for building secure channels. In *International Conference on the Theory and Applications of Cryptographic Techniques– Advances in Cryptology (EUROCRYPT 2001)*. Springer, Innsbruck (Tyrol), Austria, 453–474.
[5] Shehzad Ashraf Chaudhry, Khalid Yahya, Marimuthu Karuppiah, Rupak Kharel, Ali Kashif Bashir, and Yousaf Bin Zikria. 2021. GCACS-IoD: A certificate based generic access control scheme for Internet of drones. *Computer Networks* 191 (2021), 107999.
[6] Geumhwan Cho, Junsung Cho, Sangwon Hyun, and Hyoungshick Kim. 2020. SENTINEL: A Secure and Efficient Authentication Framework for Unmanned Aerial Vehicles. *Applied Sciences* 10, 9 (2020).
[7] Ashok Kumar Das, Basudeb Bera, Mohammad Wazid, Sajjad Shaukat Jamal, and Youngho Park. 2021. iGCACS-IoD: An Improved Certificate-Enabled Generic Access Control Scheme for Internet of Drones Deployment. *IEEE Access* 9 (2021), 87024–87048.
[8] D. Dolev and A. Yao. 1983. On the security of public key protocols. *IEEE Transactions on Information Theory* 29, 2 (1983), 198–208.
[9] Dongfeng Fang, Yi Qian, and Rose Qingyang Hu. 2020. A Flexible and Efficient Authentication and Secure Data Transmission Scheme for IoT Applications. *IEEE Internet of Things Journal* 7, 4 (2020), 3474–3484.
[10] Yoney Kirsal Ever. 2020. A secure authentication scheme framework for mobile-sinks used in the Internet of Drones applications. *Computer Communications* 155 (2020), 143–149.
[11] T. S. Messerges, E. A. Dabbish, and R. H. Sloan. 2002. Examining smart-card security under the threat of power analysis attacks. *IEEE Trans. Comput.* 51, 5 (2002), 541–552.
[12] Mohammad Wazid, Ashok Kumar Das, Vanga Odelu, Neeraj Kumar, and Willy Susilo. 2020. Secure Remote User Authenticated Key Establishment Protocol for Smart Home Environment. *IEEE Transactions on Dependable and Secure Computing* 17, 2 (2020), 391–406.