IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 28 June 2024, accepted 22 July 2024, date of publication 30 July 2024, date of current version 15 August 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3435688

RESEARCH ARTICLE

Mechanism for Device Authentication and Session Key Generation in Industrial Internet of Things Networks

AKSHAY KUMAR[1], USHA JAIN[2], (Member, IEEE), MUZZAMMIL HUSSAIN[1], MOHAMMAD KHALID IMAM RAHMANI[3], (Senior Member, IEEE), AND ABDULBASID S. BANGA[3]

[1]Department of Computer Science and Engineering, Central University of Rajasthan, Ajmer, Rajasthan 305817, India
[2]Department of Computer Science and Engineering, Manipal University Jaipur, Jaipur, Rajasthan 303007, India
[3]College of Computing and Informatics, Saudi Electronic University, Riyadh 11673, Saudi Arabia

Corresponding authors: Usha Jain (ushajain.230806@gamil.com), Abdulbasid S. Banga (a.banga@seu.edu.sa), and Mohammad Khalid Imam Rahmani (m.rahmani@seu.edu.sa)

ABSTRACT Industrial Internet of Things (IIoT), also known as Industry 4.0 is a major revolution in the industry and it is designed for enhancement of productivity and reliability. Security is a prime concern in the implementation of IIoT technology along with other concerns like patch management. An inefficient or insecure security mechanism would permit rogue (illegitimate) devices to access crucial data and resources in the network. In order to reduce the involvement of malicious devices in IIoT networks and ensure the security of crucial data, we suggest a device authentication and session key generation scheme for devices in IIoT networks. The proposed mechanism is based on the bitwise XOR operation, one-way hash function, and concatenation operation. In this scheme, the server and node generate the session key independently, and it cannot be generated by any third entity. The proposed mechanism is scrutinized for its security using AVISPA and ProVerif tools (automatically) and BAN logic (mathematically) and it has been determined from the generated results that the proposed scheme is safe and immune to any security threats. Performance of the proposed scheme has been verified through experimental analysis and from the result generated. It has been proved that it consumes less resources with maximum throughput. Hence it is efficient and secure.

INDEX TERMS Cryptography, device authentication, Industry 4.0, IIoT device, Industrial Internet of Things (IIoT) networks, security.

I. INTRODUCTION

Evolution of electronics, communications and computing technologies has led to the development of state-of-the-art technologies like smart systems, Internet of Things (IoT), smart cities etc. Industrial IoT (IIoT) is one among these that has evolved over a period of time for improving the efficiency, safety, quality, sustainability and traceability in the industry. IIoT enables collection of data from machines and storing on a cloud server. Previously, the machines were dumb, and the data generated was rarely used (as shown in Fig. 1). This technology has improved the efficiency, productivity, safety, quality, sustainability, and traceability leading to a revolution in the industry. IIoT enables collection of commercial data from the machines and it's storage in a cloud server. IIoT made the industrial machines dynamic and enabled them to communicate for efficiency and quality in the production and delivery systems. IIoT technology is vastly used in industries like automobile, agriculture, oil, and gas [1], [2], [3]. Industrial robots have revolutionised the automobile industry by reducing the manpower, expenses, and time of production apart from increasing efficiency and reliability. Sensors collect data about soil, nutrients and moisture, weather conditions and help the agriculture sector in selecting the crop and improving the yield. A fleet of autonomous aircrafts are deployed to maintain oil and gas

The associate editor coordinating the review of this manuscript and approving it for publication was Chakchai So-In.

© 2024 The Authors. This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

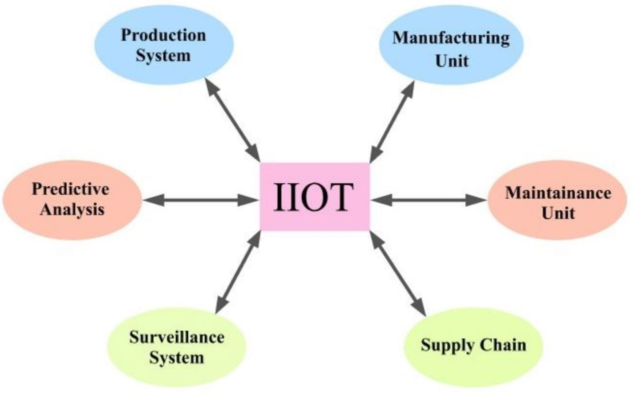


FIGURE 1. Industrial IoT environment.

industry by detecting potential problem in pipelines [4], [5]. Even though IIoT systems offer a wide range of services to the end users, they have their own challenges like interoperability of devices, heterogeneous devices, compatibility, security etc [6], [7]. Communications in IIoT applications are mostly dependent on public communication channel, where an intruder can easily intercept the communications and capture crucial data or disturb an ongoing communication or access resources illegitimately [8]. IIoT networks need decentralized techniques with high interoperability, lightweight and scalable for providing security services.Secrecy (C), integrity (I), authentication (A), authorization and access control (A), and availability (A) are among the CIA+ model's established IIoT security requirements [9]. Security concerns have played a significant role in preventing organisations from migrating to IIoT. A rogue device at an industry site can hinder performance by preventing reputable devices from exchanging reliable and authentic data. In traditional sectors, it's common to believe that all smart gadgets are trustworthy and cooperative. In actuality, rogue devices frequently engage in nefarious activity with IoT devices. Therefore, it is important and difficult to distinguish between a benign and a malicious IoT device [10]. Designing secure data transmission techniques is pivotal in IIoT, but IIoT is designed to support industrial systems, which rely on time sensitive data like operating commands and real-time sensor data. Thus, any mechanism providing security in IIoT need to ensure timeliness of communications [11]. Authentication, privacy, information security, intrusion detection are few of the crucial security requirements of an IIoT network [12], [13]. An efficient authentication and key generation scheme may ensure privacy, data security and intrusion detection. Hence, in this paper we propose a mechanism for device authentication and session key generation in IIoT networks.

In this paper, we have designed a device authentication mechanism that eliminates illegitimate devices from accessing resources in an IIoT network and communicates with legitimate devices. We also proposed a key agreement scheme between the devices in an IIoT network. The proposed scheme employs one-way hash function, bitwise XOR and concatenation operations, Here the IIoT devices are logically divided into three groups as i) devices, ii) intermediate devices, and iii) server in hierarchy. Initially, the identity of the device and an authentication parameter are stored in the devices during deployment. Later, a device requests for session key to the server through an intermediate device by sending its hashed id, some random number, and a timestamp. The intermediate node forwards the request to server by adding its own identity onto it. The server verifies the received values and computes the session key by using the parameters for authentication, secret value picked, identifiers of the device & intermediate device, and random number of a device. The device recomputes the session key using received values, its random number and secret value picked. The proposed scheme is safe as the server and the device generate the session keys using the values exchanged and randomly generated or picked values, neither the device nor the server shares its secret value with any other entity, due to which no other entity except server and the device can compute the session key (as shown Fig. 2). The proposed scheme is verified for its security using ProVerif tool (automatically) and BAN logic (manually) and it has been found that the proposed scheme is safe and immune to any security threats. The performance analysis done through experimentation has proved that it consumes less resources like power, processing resources and storage. Hence, it is both secure and efficient. The proposed scheme is both secure, safe from the known attacks and is lightweight, as it doesn't involve complex mathematical or cryptographic operations. On the other hand, the previously designed mechanisms either involve complex mathematical and cryptographic operations or prone to one or more security attacks and moreover overload the system with overheads like, computational, communicational, energy and storage. The proposed mechanism has been evaluated for security using various approaches both mathematically and automated. The proposed mechanism is evaluated using BAN logic mathematically and using AVISPA and ProVerif automatedly. The proposed mechanism is also evaluated experimentally by running it on a sensor using HC-05 TTL and USB to TTL UART modules. The mechanism is coded in Python to run on the experimental setup.

A. CONTRIBUTIONS

Major contributions of our paper are as follows:

- Review the existing works and identify their limitations.
- Proposed a lightweight, dynamic authentication and session-key generation protocol for secure communication among IIoT devices in Industrial IoT (IIoT).
- Theoretical security analysis of the proposed protocol is carried out which represents protection from eavesdropping, device capture, replay, impersonation, MITM, desynchronization, stolen database of hub node, session-key guessing and impersonation attacks and ensures mutual authentication, forward and backward

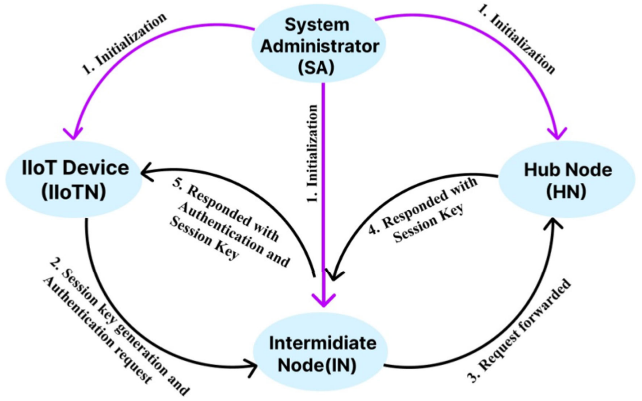

1. Initialization
System Administrator (SA)
1. Initialization
1. Initialization
IIoT Device (IIoTN)
Hub Node (HN)
5. Responded with Authentication and Session Key
4. Responded with Session Key
2. Session key generation and Authentication request
Intermidiate Node(IN)
3. Request forwarded

FIGURE 2. IIoT device authentication and session key generation scheme.

secrecy, sessions with anonymity and unlinkability of the proposed protocol.

- Conventional verification using Burrow-Abadi-Needham (BAN) logic validates the security of the proposed protocol.
- The automated protocol verification tools like AVISPA and ProVerif verify the security of the proposed authentication protocol using Dolev-Yao model as claimed.
- The computation, communication and memory overheads of our proposed protocol are calculated and analysed with other existing protocols.
- The energy consumption and throughput of the proposed protocol are computed using HC-05 Bluetooth module and transistor-to-transistor logic to analyse the energy efficiency of the proposed protocol for IIoT devices.

B. ROAD MAP

The rest of the paper is organized as follows: Section II defines the existing IIoT device authentication mechanism for IIoT networks. The proposed mechanism for IIoT device authentication is shown in Section III. Section IV analyses the proposed mechanism theoretically, and Section V discusses the evaluation of the proposed protocol using BAN logic (for formal analysis), AVISPA (Automated Validation of Internet Security Protocols and Applications) and ProVerif to analyse the proposed mechanism's security automatically. The overhead evaluation and comparison of the proposed mechanism with recent mechanisms and discussion about the experimental analysis of the proposed protocol is discussed in the same section. Finally, we arrive to the conclusions in section VI.

II. REVIEW OF LITERATURE

Security in Industrial Internet of Things (IIoT) has been a point of concern and has attracted many researchers to work on possible feasible solutions towards enhancing the security in applications of IIoT. Authentication of devices and key management are two pivotal security requirements in IIoT applications [14], [15], [16], [17], [18], [19]. Authentication ensures that only legitimate devices are part of any communication and creates a secure channel between the legitimate devices, whereas key management ensures confidentiality and privacy. In this section, we give a brief report of the related work done by different authors towards security in IIoT, authentication and key agreement/management.

Mick et al. [20] proposed lightweight authentication and secure routing for Named Data Networking (NDN) IoT in smart cities (LASeR) in 2018. In this, the authors designed a pre-shared key extensible authentication protocol (EAP-PSK) which works in three phases. In phase one, a sensor node discovers a path and the legitimacy of the network is verified. A sensor node to join the network sends a route discovery request, forwards it to its neighbour node authenticated with the Island manager using two long keys namely Authentication Key (AKSN) and key-derivation key (KDKSN). In second phase, the Island manager establishes a trust with sensor node through authenticating using Authentication key of previously exchanged nonces. Here, the sensor node authenticates itself with Island manager. In third phase, all the nodes between sensor node and anchor node are updated with next hop fields through route advertisement. The proposed protocol is simulated in NS-3 and its performance is evaluated. However, this scheme is prone to Distributed Denial-of-Service (DDoS) attack.

A simple authentication and key agreement approach for smart metering in Smart Energy Networks (SEN) was discussed by Kumar et al. [21]. In the designed mechanism, a smart meter (SM), domestically establishes a secure communication with the utility server through neighbourhood area network (NAN) gateway through authentication and key agreement. Every smart meter acquires an authentication token from NAN gateway during registration phase. In the next phase, it authenticates itself with the NAN gateway and establishes a session key to secure communications between SM and NAN gateway. The session key is produced using smart meter ID, NAN gateway ID and other computed values during authentication process. At each stage, the validity of the exchanged messages is verified through timestamp to avoid replay and DDoS attacks. The work is analysed for security using AVISPA tool and its performance analysis is done. The finding is that the proposed protocol requires smart meter to perform 5 hash and 2 MAC operations which is computationally infeasible.

Li et al. [22] designed a mutual authentication protocol for IoT devices considering the resource constrain nature of the devices. The scheme is based on novel public key encryption technique. The protocol runs for multiple passes depending on the level of security requirements. The protocol comprises four algorithms namely system setup (Setup), key generation (KeyGen), initialization (Init) and mutual authentication (Auth). Initially, system parameters like message size, number of passes, nonces etc. are determined in setup using security parameters as input. Later, pairs of public and private keys are generated by KeyGen with system parameter generated during setup as inputs. The devices exchange their identities and public keys during Init and finally the devices mutually authenticate each other during Auth. This mechanism is emulated in Cooja simulator and evaluated using CC2538 evaluation modules. The mechanism is light in weight as it avoids complex computations like modulo exponentiation. However, this scheme is prone to DDoS attacks.

Li et al. [23] designed a key agreement and mutual authentication scheme which is based on hash function and XOR operation. This scheme has three levels of communications: primary node, second level node and hub node. The system administrator registers sensor nodes. Sensor nodes request the hub node for authentication through intermediary node and set up a session key in a secure manner. The nodes exchange their parameters using hash function. AVISPA tool is used to analyse this scheme. The mechanism is lightweight and safe against various potential threats but it has high expense of computation overhead, communication cost, storage requirement, and energy consumption.

Jaya et al. [24] designed a lightweight and efficient scheme for human centred IIoTs. The designed scheme works in three phases namely: Preliminary phase, Registration phase, authentication phase and key exchange phase. During preliminary phase, initial RSA settings are done by a trusted third party (TTP) also known as registration centre (RC). In registration phase, RC generates an EID for a registered node using one-way hash function and ID of the node and sends it to registered new node. In authentication phase, devices/nodes mutually authenticate using the values received from RC and current timestamp values. Finally, in key exchange phase, two nodes generate a common session key individually using the parameter exchanged/generated during previous phases. This scheme is also prone to Man-in-the-Middle and DDoS attacks.

Abdi Nasib Far [25] designed a robust and energy efficient authentication protocol for IIoT. The designed protocol is a three-factor authentication scheme. It is based on biometrics and works in four phases namely, initiation, registration, authentication and key management and password change. In initialisation phase, the gateway wireless node (GWN) generates its public and private keys, assigns identity of sensor nodes, and calculates the shared secret key. The public key, ID of node and shared secret key are stored onto the memory of sensor node and the node is deployed in the target field. In registration phase, a user registers with the GWN through his/her mobile device and biometrics. Whenever a user intends to access the data of any node, he/she has to authenticate himself/herself using its identity and password. Upon successful verification a session key is generated by GWN using identity of user, ID of sensor node and the parameters and is shared among GWN, user and sensor node. A user can update or renew the password in password change phase without communication GWN. This protocol achieves optimal performance but is complex in nature due to frequent usage of hash function and it is prone to DDoS attack.

Aman and Sikdar [26] designed a hybrid protocol for attestation with authentication in IIoT. The designed protocol performs attestation of devices in IIoT networks and performs mutual authentication between the devices. The attestation is done through SWATT, a software-based attestation technique. In this, it verifies the memory contents and verifies the presence of any malicious tampering of the memory. The attestation is done based on the time taken to validate the memory, a malicious device/software takes more time than a benign device/software. This protocol also performs mutual authentication between the devices intrinsically. An IoT device here undergoes attestation only if the other device requests to do so and the requesting device has already extended the protocol with the IoT device and vice-versa. This protocol is verified using Mao-Boyd logic formally and the analysis is done using ProVerif. However, this protocol is vulnerable to Man-in-the-Middle and DoS attacks.

Shen et al. [27] designed a cross-domain IIoT device authentication mechanism. In cross-domain networks, the devices are under control of different server, and this makes it hard to perform either authentication or key exchange between them. Here, the authors have designed a mechanism that performs authentication using a Blockchain, which is a block-based, distributed global ledger with many transactions. The information is shared by corresponding key generation centre (KGC) and can be used for cross-domain authentication. The blockchain is maintained by a group comprising KGC of each domain. An identity-based signature algorithm is used for authentication where each device has a global public key in its identity and the signature of a device is generated using its signing private key, the signature of a device is verified using its public key. This algorithm is known as blockchain assisted authentication Elliptic Curve based Diffie Hellman Key Exchange (ECBDE) which is used to share a private key between the devices. Many entities like KGC, Authentication Agent Server (AAS), Blockchain Agent Server (BAS) and storage server interact with each other to execute the protocol and mutual authentication in cross domain and key exchange. This mechanism ensures secure authentication among devices in cross-domain IIoT and key exchange, but it is prone to Sybil attack and forward and backward secrecies.

Esfahani et al. [28] proposed a mechanism for authentication of machine-to-machine (M2M) communications in IIoT. In this, mechanism works in two phases, namely registration and authentication. During registration phase, a sensor node performs registration with Authentication server (AS) by sending its identity, using this identity, and applying hash and XOR operations. The sensor node's memory is where the AS stores the parameters it generates. Each sensor node must be able to authenticate itself with the router following the registration step. Here, the sensor node requests for authentication of router using its unrealistic identity. The router performs few computations using hash function, XOR operation, pre-shared secret key (PSA) to accomplish the authentication. Also, a random number generated by sensor node and router is employed to compute a session key. The mechanism is light in weight due to low computational and storage overheads. It is also resistant to replay, MITM, impersonation and modification attacks. But it is vulnerable to DDoS attacks as an adversary may attack a target noderouter using multiple fake identities.

Shahzad et al. [29] proposed a lightweight authentication mechanism for M2M communication in IIoT networks. The protocol uses exclusive-OR and hash functions and is based on timestamp and pre-shared key. The protocol functions in four phases namely initialization, registration, authentication, and update phase. This protocol facilitates authentication between a sensor (S) and the Controller (C) with support of an authentication server (A). In initialisation phase, the authentication server generates a timestamp (TS) and shares with both server (S) and controller (C), upon the request of S. In registration phase, S generates a message encrypted with PSK comprising hash and XOR of its ID, PSK, nonce and timestamp and sends to C; C validates the received message by comparing the timestamps, ID of S and nonce then it sends back as encrypted message with PSK, comprising of hash & XOR of its ID, PSK nonce and timestamp. In authentication phase, the sensor verifies the received message from C and sends an encrypted message comprising hash of nonce and timestamp. C compares the timestamp and verifies the received values, it generates a key k as hash of nonce of C, nonce of S and ID of S and sends an encrypted message comprising hash of nonce of S, nonce of C, new timestamp to S. In update phase, S verifies the received values and send an OK message to C and also generates its own key as hash of nonce of S, nonce of C and ID of S. The protocol is verified for security using AVISPA tool and BAN logic. But it may compromise the confidentiality as adversary may generate the key K as hash of S, hash of C and timestamp.

A multifactor authenticated key agreement technique for IIoT was addressed by Vinoth et al [30]. In this scheme, a user can access the data from sensors, deployed in industry through gateway nodes. A user authenticates himself/herself through user ID, password, and biometrics card. The protocol works in 06 phases namely, registration phase, login phase, authenticated key agreement phase, biometrics and password update phase, device joining phase and device revocation phase. In registration phase, a user registers with GWN using its identity, password and biometrics. A biometric key is computed from biometrics of the user, in login phase, a user logs on to the GWN using its ID, password and biometrics through card reader. In the authenticated key agreement phase, the GWN verifies the newness of login request, by checking the timestamp values and computes a series of values using XOR and hash functions, exchanging and validating the values. The user and the sensing devices establish a secure session key. An authorized user updates his/her biometrics and password through biometrics and password update phase. A new sensing device joins the network through devices joining phase and devices leave the network through revocation phase to ensure forward/backward secrecy. The proposed scheme is secure from many attacks but at the expense of high computational, storage and communicational overheads.

Tanveer et al. [31] have proposed an authentication protocol for IIoT which is resource, efficient. The devised approach makes use of an associative data (AEAD) primitive AEGIS long with hash function and a lightweight cryptographic (LWC)-based authentication encryption. The resource-efficient authentication protocol (REAP) creates a session key between users and deployed sensing devices and permits primary-preserving user authentication. REAP-IIoT functions in six phases namely sensor device registration (SDR) phase, user registration (UR) phase, authenticated key exchanged (AKE), biometric password change (BPC) phase, revocation phase (RP), dynamic sensor device in deployment (DSDX) phase. Before an SD is deployed, a trusted agent registers it during the SDR phase, computes its identification (ID), and secret parameter (SP), and saves it in the memory of the SD. During user registration (UR) phase, TA assigns secret parameters (SP) to users and each user is assigned a list of SD it is authorised to access. UR phase is performed offline, considering the security issues. In AKE phase, user first performs local authentication by getting its secret parameters validated. During BPC phase, a user needs to frequently change its password keeping the biometrics information unaltered. In RP, a new sensitive/secret data is provided to legitimate user, upon losing it and in DSDR phase a new sensor node is deployed by assigning it a unique ID and secret parameters. The mechanism is secure and immune to numerous security attacks but is prone to DDoS attack.

Sohail et al. [32] presented a secure collaborative data sharing platform in this paper that makes use of consortium block chain technology. In order to protect user anonymity and data integrity, authorsÂ suggested an authentication mechanism that used HMACÂ and ECCÂ to regulate access to data. Cui et al. [33] created a blockchain-based method for edge computing-based IIoT device authentication. Without involving the Certificate Authority (CA), authors generated and disseminated anonymous identities for the smart devices via the edge servers. The authors [34] demonstrated how key recovery attacks, which allow anyone to retrieve a user's whole set of private keys, defeated Wang et al.'s scheme [35]. After that, they suggested a brand-new secure CLS technique and demonstrated its unforgeability in the face of adversaries of types I and II while adhering to the Diffie-Hellman problem's hardness assumption. An improved multi-factor safe authentication and key agreement technique for the IIoT was proposed by Han et al [36]. after they investigated the causes of insecurity. This protocol, which exclusively makes use of symmetric cryptography, hash functions, and XOR operations, increased protocol's security. It was shown through formal security research and informal security talks that the protocol could withstand a wide range of known assaults.

Most of the previously designed mechanisms either involve complex mathematical and cryptographic operations or prone to one or more security attacks and moreover overload the system with overheads like, computational, communicational, energy and storage. The security analysis

TABLE 1. Comparative analysis of proposed mechanism with other mechanisms.

Scheme	Cryptographic Methods	Implementation	Possible Attacks
[20]	AE	NS-3	DDoS
[21]	Hash + MAC + SE + AE	AVISPA	Backward secrecy.
[22]	AE	Cooja Simulator and Micro-controller CC2538	DDoS, Eavesdropping, and device capture attacks
[23]	Hash + XOR + SE + AE	AVISPA and NS-3	MITM, Eavesdropping, Backward and Forward Secrecy
[24]	ECDSA	Not Implemented	Sybil, MITM, DDoS and Secrecy
[25]	Hash	Not Implemented	DDoS, Device Capture Attacks
[26]	AE and PUF	Mao-Boyd Logic and ProVerif	Impersonation, MITM, Replay and DDoS attacks
[27]	Blockchain + ECDSA	Not Implemented	Sybil, Backward and Forward Secrecy
[28]	Hash + XOR + SE	Not Implemented	Device Capture and Eavesdropping, DDoS attacks
[29]	Hash + XOR + SE	BAN Logic and AVISPA	Sybil, Impersonation, MITM, Device Capture attacks
[30]	Hash + XOR + SE	Not Implemented	Eavesdropping, Spoofing, Backward and Forward Secrecy
[31]	Hash	Not Implemented	Eavesdropping, Impersonation, DDoS attacks
[33]	SE + AE	Random Oracle Model, ProVerif	DDoS, Desynchronization, MITM attacks
[32]	AE + HMAC	Real-or-Random model, Scyther, AVISPA	MITM, Device capture, Spoofing attacks
[34]	Hash + SE	Not Implemented	DDoS, MITM, Impersonation attacks
[36]	SE + Hash + XOR	Real-or-Random Model	DDoS, Replay, MITM, Backward Secrecy
Ours	XOR + Hash	BAN logic, AVISPA, ProVerif & HC-05 Bluetooth module	Secure against various well-known attacks.

AE = Asymmetric Encryption, SE = Symmetric Encryption, ECDSA = Ellptic Curve Digital Signature Algorithm, PUF = Physically Unclonable Function.

is either incomplete or the simulation/experimental analysis is not done.

Privacy and security are primary concerns of any application area of IoT, similarly in Industry 4.0 aka IIoT security and privacy of the data and communications is a matter of concern. From the survey of related work as shown in Table 1, we could conclude that even though some work has been done towards security of IIoT networks but most of them are either complex in nature or insecure. Hence, with an intention to enhance the security of communications in IIoT networks, in this paper we proposed a mechanism for mutual authentication of devices in an IIoT network to ensure privacy and also to establish a key for secure transmission of data between the devices. The proposed mechanism is not only secure but also lightweight as it consumes less resources and making it suitable for applicable IIoT networks.

TABLE 2. Notations used in the proposed mechanism.

Notation	Description
id_n	Unique identity of IIoT device
id_{in}	Unique identity of intermediate node
id_{hn}	Unique identity of hub node
TID_n	Temporary identity of IIoT device
a_n	Authentication parameter used at IIoT device side
b_n	Authentication parameter used at hub node side
I_n	Intermediate parameter used at hub node side
r_n	Random number of IIoT device
t_n	Time-stamp generated by IIoT device
J_n	Secret key value of hub node (choosen by hub node)
K_n	Secret key value of IIoT device (choosen by hub node)
K_s	Secret session key
p, q, d	Parameter used for authentication
$\oplus$	Bitwise OR
(a, b)	Concatenation operation
$h(\bullet)$	One-way hash function

III. PROPOSED SCHEME

The proposed scheme is thoroughly explored in this section. Notations used in the proposed mechanism are mentioned in Table 2:

Here, we are considering three communicating entities: IIoT device (*IIoTN*), intermediate node (*Inter_N*), and hub node (*Hub_N*). The proposed mechanism consists of two phases: initialisation phase, and authentication phase with session key generation. Considering *IIoTN* is a second level node, it communicates with *Hub_N* via a *Inter_N*, a first-level node. If *IIoTN* is a first-level node, then the *Inter_N* can be simply removed from the system to enable direct communication with *Hub_N*.

In the proposed mechanism, the system administrator (SA) is a key component during initialisation phase that manages the initialization phase, which involves assigning identities to IIoT devices, intermediate nodes, and hub nodes. The Industrial Internet of Things (IIoT) environment's communication system integrity and secrecy are guaranteed by the SA, which is in charge of creating authentication parameters and safely storing the relevant information.

In initialisation phase, system administrator (SA) assigns identities to all IIoT devices (*IIoTN*.), intermediate node (*Inter_N*), and hub node (*Hub_N*). SA selects a secret value J_n at hub node and computes authentication parameter $a_n = (id_n \oplus id_{hn}, K_n)$. At the end of initialisation phase, system administrator stores $< id_n, id_{in}, id_{hn}, a_n >$, $< id_{in}, id_n, id_{hn} >$; and $< id_{hn}, id_n, id_{in}, J_n >$ in the storage of *IIoTN*, intermediate node and hub node, respectively. Notably, neither at the IIoT device nor at the hub node is any storage of K_n necessary. It is solely employed to produce a_n. The permanent real identity for IIoT device *IIoTN* is represented by the identity id_n.

The next phase is authentication phase with session key generation. In this phase, IIoT device (*IIoTN*) selects random number (r_n) and timestamp (t_n). Later IIoT device computes the following parameters with temporary identity:

$$p = (id_n \oplus r_n)$$
$$q = (h(a_n, id_{hn}) \oplus r_n)$$
$$TID_n = h(p, t_n)$$

Now IIoT device sends a message $M1 :< TID_n, a_n, t_n, q >$ to intermediate node. As intermediate node receives a message M1, it adds its own identity to the message M1 and forwards the updated message $M2 :< id_{in}, TID_n, a_n, t_n, q >$ to hub node. Hub node receives the message M2 and starts the verification of id_{in}, a_n and t_n. Later hub node performs the following steps:

$$\begin{aligned} I_n &= h(a_n, id_{hn}) \\ {r_n}^* &= (I_n \oplus q) \\ p^* &= (id_n \oplus {r_n}^*) \\ {TID_n}^* &= h(p^*, t_n) \\ {TID_n}^*? &= TID_n \end{aligned}$$

On the successful verification of TID_n, hub node computes the following parameters:

$$\begin{aligned} d &= p^* \oplus J_n \\ {a_n}^+ &= h(id_n \oplus id_{hn}, J_n) \\ b_n &= (r_n \oplus J_n, id_n) \oplus {a_n}^+ \\ K_s &= (h(id_n \oplus p^*), r_n, J_n) \end{aligned}$$

Now hub node stores the session key (K_s) in its database and forwards a message $M3 \quad :< \quad id_{in}, TID_n, d, b_n \quad >$ to intermediate node. Intermediate node verifies its own identity id_{in} from the received message and forwards the message $M4 \quad :< \quad TID_n, d, b_n \quad >$ to IIoT device. As IIoT device receives the message M4, it performs the verification of TID_n, b_n and performs the following computation to calculate session key for future communication:

$$\begin{aligned} J_n^* &= p \oplus d \\ {a_n}^{++} &= h(id_n \oplus id_{hn}, J_n^*) \\ {b_n}^* &= (r_n \oplus J_n, id_n) \oplus a_n^{++} \\ {b_n}^*? &= b_n \\ K_s &= (h(id_n \oplus p), r_n, {J_n}^*) \end{aligned}$$

Now IIoT device stores the session key to communicate with other devices in future. In this way, IIoT device authenticates itself and gets a session key from hub node for future communication (as shown in Fig. 3). In Fig. 4, Flow chart of the proposed protocol shows the process of secure communication and authentication of IIoT devices via Hub node.

IV. THEORETICAL EVALUATION

The proposed scheme is theoretically evaluated for security and its resistance towards various security attacks in this section.

A. EAVESDROPPING

In the proposed mechanism, an attacker can access all the parameters of the communication. Identity of the IIoTN (id_n) has been securely transmitted over channel using temporary identity TID_n. It is secured using one-way hash function $h(\bullet)$. The parameter r_n is secured using another parameter p. The updated authentication parameter ${a_n}^+$ is transmitted using b_n. Parameter d helps in securing J_n using $h(\bullet)$ function. In this way, an attacker is unable to access and compute any parameter over the channel. Therefore the proposed mechanism secures against any eavesdropping attack.

B. SESSIONS WITH ANONYMITY AND UNLIKABILITY

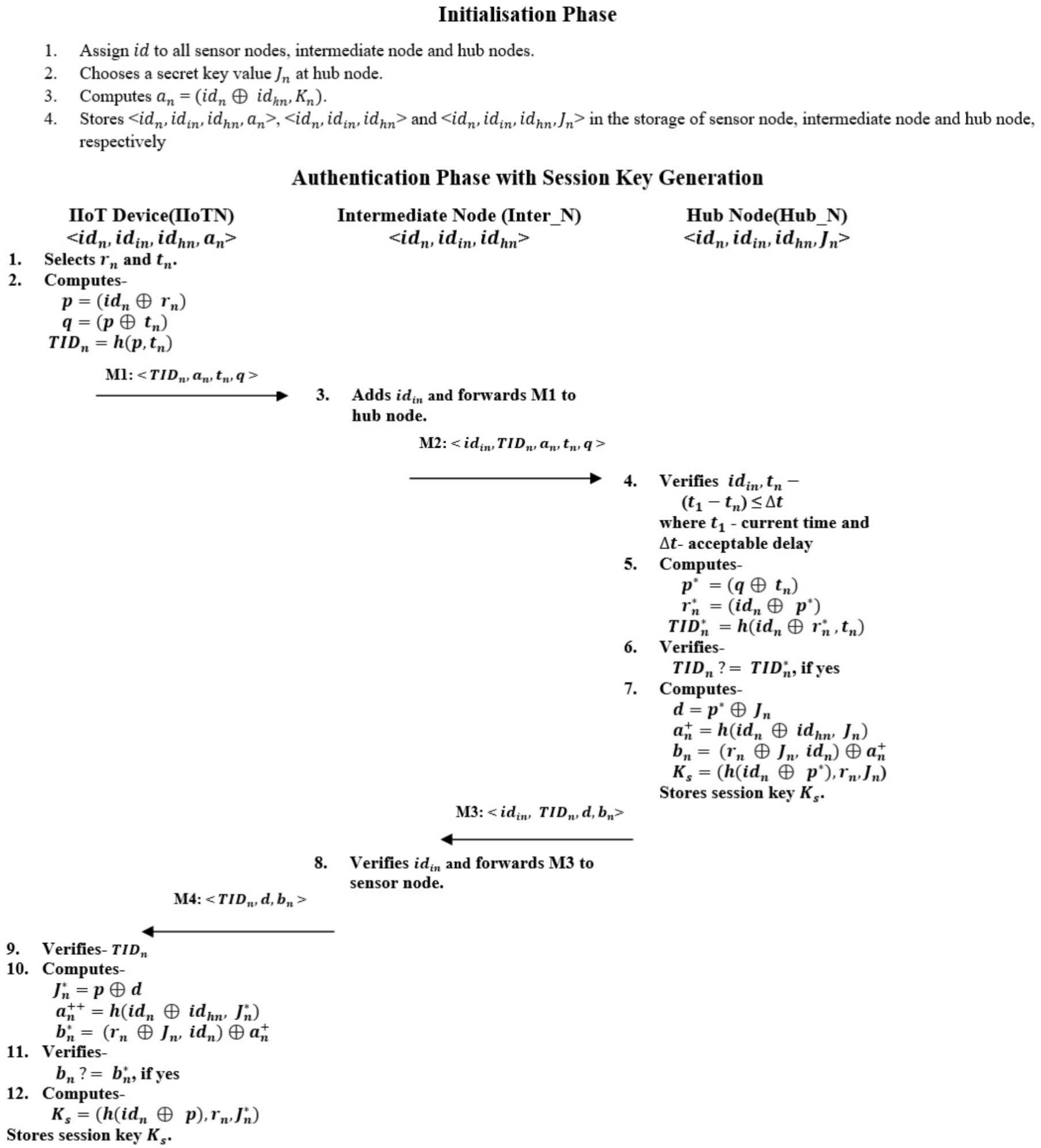
The objective of this service is to make it impossible for an attacker to determine a communicating device's IIoTN identity from the communication parameters intercepted and to link one session to another of the same device of same IIoTN. The temporary identity of the device TID_n is calculated using random number r_n which is fresh, random, and independent. An attacker is unable to identify two unique TID_n for same IIoTN. Authentication parameter of hub side b_n is also based on random number r_n. Our designed protocol makes sure that an attacker cannot determine these randomly chosen parameters in order to acquire some fixed parameter. Therefore, each performed session's communication parameters are separate, arbitrary, and fresh. Our technique maintains the devices' anonymity and the sessions' unlikability since an illegitimate entity cannot join two or more sessions to the same device in IIoTN.

C. IMPERSONATION ATTACK

If an impersonation attempt is successful, the attacker has the power to produce a reliable tuple $< TID_a, a_a, t_a, q >$. Here, we have two circumstances: An attacker records a sent tuple $< \quad TID_a, a_a, t_a, q \quad >$ by listening on the channel, but he/she is unable to identify the accompanying id_a; An attacker is aware of identity (id_a) and a_a. In the second scenario, an attacker conquers the device IIoTN, and this is a different kind of onslaught. As $TID_a \quad = h(p, t_n)$ is linked to the timestamp in the first scenario, the captured tuple prevents an attacker from impersonating a device without the matching id_a because the attacker cannot generate a legitimate temporary identity without the corresponding id_a.

D. SPOOFING ATTACK ON HUB NODE (HN)

An attacker needs to produce a valid tuple of the form $< TID_n, d, b_n >$, to pose as the HN to an IIoT device IIoTN. In this message, $TID_a = h(p, t_n)$ is computed using p where p is computed using r_n. The random number r_n is inaccessible to the attacker. The parameter d helps to compute ${J_n}^*$. In the tuple, the parameter $b_n \quad = \quad (r_n \oplus {J_n}^*, id_n) \oplus {a_n}^+$, where ${J_n}^* = p \oplus d$. The parameters p and d are totally unknown to an attacker. As, an attacker cannot generate a valid parameter from these parameters, our proposed mechanism defends against HN spoofing attacks.



Initialisation Phase

1. Assign id to all sensor nodes, intermediate node and hub nodes.
2. Chooses a secret key value J_n at hub node.
3. Computes $a_n = (id_n \oplus id_{hn}, K_n)$.
4. Stores $<id_n, id_{in}, id_{hn}, a_n>$, $<id_n, id_{in}, id_{hn}>$ and $<id_n, id_{in}, id_{hn}, J_n>$ in the storage of sensor node, intermediate node and hub node, respectively

Authentication Phase with Session Key Generation

IIoT Device(IIoTN) $<id_n, id_{in}, id_{hn}, a_n>$

Intermediate Node (Inter_N) $<id_n, id_{in}, id_{hn}>$

Hub Node(Hub_N) $<id_n, id_{in}, id_{hn}, J_n>$

1. Selects r_n and t_n.

2. Computes-

$p = (id_n \oplus r_n)$

$q = (p \oplus t_n)$

$TID_n = h(p, t_n)$

M1: $< TID_n, a_n, t_n, q >$

3. Adds id_{in} and forwards M1 to hub node.

M2: $< id_{in}, TID_n, a_n, t_n, q >$

4. Verifies $id_{in}, t_n -$ $(t_1 - t_n) \le \Delta t$ where t_1 - current time and Δt- acceptable delay

5. Computes-

$p^* = (q \oplus t_n)$

$r_n^* = (id_n \oplus p^*)$

$TID_n^* = h(id_n \oplus r_n^*, t_n)$

6. Verifies-

$TID_n ?= TID_n^*$, if yes

7. Computes-

$d = p^* \oplus J_n$

$a_n^+ = h(id_n \oplus id_{hn}, J_n)$

$b_n = (r_n \oplus J_n, id_n) \oplus a_n^+$

$K_s = (h(id_n \oplus p^*), r_n, J_n)$

Stores session key K_s.

M3: $< id_{in}, TID_n, d, b_n>$

8. Verifies id_{in} and forwards M3 to sensor node.

M4: $< TID_n, d, b_n >$

9. Verifies- TID_n

10. Computes-

$J_n^* = p \oplus d$

$a_n^{++} = h(id_n \oplus id_{hn}, J_n^*)$

$b_n^* = (r_n \oplus J_n, id_n) \oplus a_n^+$

11. Verifies-

$b_n ?= b_n^*$, if yes

12. Computes-

$K_s = (h(id_n \oplus p), r_n, J_n^*)$

Stores session key K_s.

FIGURE 3. **Proposed protocol for device authentication and session key generation.**

E. DDOS ATTACK

It concerns a malicious attempt to obstruct regular traffic in order to prevent devices from functioning as intended. To prevent this attack, we have included the timestamp t_n with every message. Before reaching its threshold value $(\triangle t)$, each message is valid. This threshold will cause the message to be dropped.

F. REPLAY ATTACK

In the proposed mechanism, timestamps are used to safeguard the mechanism from any possible replay attacks. *IIoTN* inserts the timestamp t_n in such a way that it prevents an attacker from either wipe or change it. Moreover, every message being transmitted is encrypted disabling an arbitrary in understanding

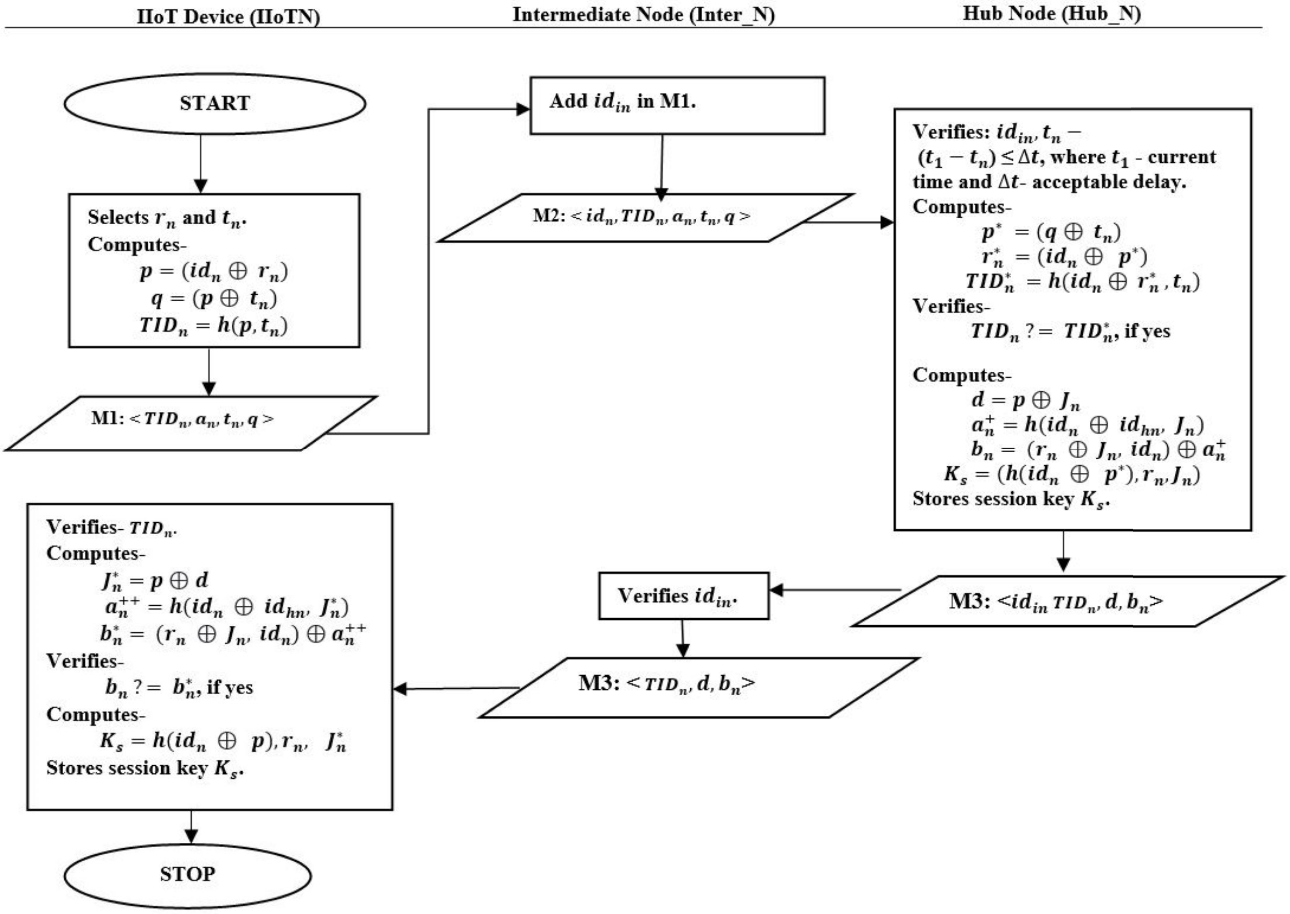

IIoT Device (IIoTN)
Intermediate Node (Inter_N)
Hub Node (Hub_N)
START
Selects r_n and t_n.
Computes-
$p = (id_n \oplus r_n)$
$q = (p \oplus t_n)$
$TID_n = h(p, t_n)$
M1: $< TID_n, a_n, t_n, q >$
Add id_{in} in M1.
M2: $< id_n, TID_n, a_n, t_n, q >$
Verifies: $id_{in}, t_n - (t_1 - t_n) \leq \Delta t$, where t_1 - current time and Δt- acceptable delay.
Computes-
$p^* = (q \oplus t_n)$
$r_n^* = (id_n \oplus p^*)$
$TID_n^* = h(id_n \oplus r_n^*, t_n)$
Verifies-
$TID_n \mathrel{?=} TID_n^*$, if yes
Computes-
$d = p \oplus J_n$
$a_n^+ = h(id_n \oplus id_{hn}, J_n)$
$b_n = (r_n \oplus J_n, id_n) \oplus a_n^+$
$K_s = (h(id_n \oplus p^*), r_n, J_n)$
Stores session key K_s.
M3: $<id_{in} TID_n, d, b_n>$
Verifies id_{in}.
M3: $< TID_n, d, b_n>$
Verifies- TID_n.
Computes-
$J_n^* = p \oplus d$
$a_n^{++} = h(id_n \oplus id_{hn}, J_n^*)$
$b_n^* = (r_n \oplus J_n, id_n) \oplus a_n^{++}$
Verifies-
$b_n \mathrel{?=} b_n^*$, if yes
Computes-
$K_s = h(id_n \oplus p), r_n, J_n^*$
Stores session key K_s.
STOP

FIGURE 4. Flowchart of the proposed protocol for device authentication and session key generation.

the message, which reduces the possibility of replay attack.

G. SESSION KEY GUESSING ATTACK

In our proposed mechanism, the session key $K_s = (h(id_n \oplus p^*), r_n, J_n)$ is computed at hub node. During the authentication process, This key is not a function input in any way. At the conclusion of the protocol, it is computed. The safety of the main secret key J_n is not jeopardized by the leak of K_s.

H. STOLEN DATABASE OF HUB NODE

The secret key of Hub_N is J_n. Except J_n, Hub_N does not keep any additional data in its database. Other identities of IIoT devices and intermediate nodes that are not confidential are stored in the database of hub node. These identities are verified after receiving the messages. Hence, this is the merit of our proposed mechanism.

I. DEVICE COMPROMISE ATTACK

The attacker is skilled at compromising as many IIoT devices as possible and is familiar with the tuples $< id_n, id_{in}, id_{hn}, a_n >$. The parameter $a_n = (id_n \oplus id_{hn}, K_n)$ helps in keeping secret the the IIoT device's main secret key K_n. It ensures that the exposed tuples do not disclose any details regarding the IIoT device's main secret key K_n. Therefore, our proposed mechanism performs secure communication.

J. MAN-IN-THE-MIDDLE ATTACK

Protection against replay attacks, spoofing attacks on hub node, and IIoT device impersonation attacks lead to defence against man-in-the-middle attacks.

K. DESYNCHRONIZATION ATTACK

The authentication method is prone to a desynchronization attack if the two entities are forced to synchronise the updating of their status. In this instance, the attacker corrupts the link after an individual entity maintains its status, preventing the other side from updating its state. This attack can be used to compromise any authentication protocol that calls for the server that houses the validation tables. The hub node lacks the capacity to maintain any validation tables in our proposed scheme. Therefore, desynchronization is not possible.

L. FORWARD AND BACKWARD SECRECY

The objective of this security measure is to ensure that any session keys that are disclosed do not compromise the confidentiality of any previous or upcoming sessions. It is obvious that the proposed scheme provides this security. The step used to get session key is $K_s = (h(id_n \oplus p^*), r_n, J_n)$. The one-way hash function $h(\bullet)$ protects all important parameters, while r_n and J_n all undergo dynamic change over sessions. This results in forward/backward security for our proposed scheme.

V. EVALUATION

For the proposed mechanism, this section covers the automatic security evaluation using AVISPA and the ProVerif tools, as well as the formal security evaluation using BAN logic. Afterwards, various communication, computing, and storage costs related to the proposed mechanism are computed and contrasted with those of other mechanisms that are currently proposed. The energy consumption of the proposed mechanism has been calculated by experimental analysis, and it has been compared to other existing mechanisms.

A. FORMAL SECURITY EVALUATION

The formal security evaluation is carried out utilising BAN (Burrows-Abadi-Needham) logic in this section. The theoretical analysis of security attacks was done in the previous section.

BAN logic is an analysis tool which helps in verifying the security of the proposed mechanism using well defined rules and assumptions [37], [38], [39]. Here we are employing BAN logic to demonstrate that our proposed scheme offers proper key agreement, mutually authenticating IIoT device *IIOTN* and the hub node *Hub_N*. The BAN logic symbols and principles are discussed in [40].

1) GOALS

In order to demonstrate the key agreement and mutual authentication, the sub-goals (SG) and goals (G) are as follows:

$$SG1: Hub\_N| \equiv IIoTN| \equiv (IIoTN \xleftrightarrow{p} Hub\_N)$$

$$SG2: Hub\_N| \equiv (IIoTN \xleftrightarrow{a_n} Hub\_N)$$

$$G1: IIoTN| \equiv Hub\_N| \equiv (IIoTN \xleftrightarrow{K_s} Hub\_N)$$

$$G2: IIoTN| \equiv (IIoTN \xleftrightarrow{K_s} Hub\_N)$$

2) IDEALIZATIONS

The transmitted messages between IIoT device IIoTN and the hub node Hub_N are idealized as follows and the communications with intermediate node (Inter_N) are avoided because of their least contribution in the mutual authentication of IIoT device IIoTN and the hub node Hub_N:

$$M1: IIoT \rightarrow Hub\_N :< IIoTN \xleftrightarrow{p} Hub\_N, r_n, t_n, IIoTN \xleftrightarrow{a_n} Hub\_N >_{IIoTN \xleftrightarrow{id_n} Hub\_N}$$

$$M2: Hub\_N \rightarrow IIoTN :< IIoTN \xleftrightarrow{p} Hub\_N, r_n, J_n, t_n, IIoTN \xleftrightarrow{K_s} Hub\_N >_{IIoTN \xleftrightarrow{id_n} Hub\_N}$$

D1 Using M1, A1 and message meaning rule, we derive,

$$\frac{Hub\_N|\equiv(IIoTN \xleftrightarrow{id_n} Hub\_N), Hub\_N \triangleleft < IIoTN \xleftrightarrow{p} Hub\_N, r_n, t_n, IIoTN \xleftrightarrow{a_n} Hub\_N >_{IIoTN \xleftrightarrow{id_n} Hub\_N}}{Hub\_N|\equiv IIoTN|\sim< IIoTN \xleftrightarrow{p} Hub_N, r_n, t_n, a_n >}$$

D2 Using A2, and the freshness rule, we derive,

$$\frac{Hub\_N|\equiv\#(t_n)}{Hub\_N|\equiv IIoTN|\sim< IIoTN \xleftrightarrow{p} Hub\_N, r_n, t_n, a_n >}$$

D3 Using A3, D1, D2, and the nonce-verification rule, we derive,

$$\frac{IIoTN|\equiv\#(r_n), Hub\_N|\equiv IIoTN|\sim< IIoTN \xleftrightarrow{p} Hub\_N, r_n, t_n, IIoTN \xleftrightarrow{a_n} Hub\_N >}{Hub\_N|\equiv IIoTN|\sim< IIoTN \xleftrightarrow{p} Hub\_N, r_n, t_n, a_n >}$$

D4 Using D3, and the belief rule, we derive,

$$\frac{Hub\_N|\equiv IIoTN|\sim< IIoTN \xleftrightarrow{p} Hub\_N, r_n, t_n, IIoTN \xleftrightarrow{a_n} Hub\_N >}{Hub\_N|\equiv IIoTN|\sim IIoTN \xleftrightarrow{p} Hub\_N}$$

(Sub-goal SG1)

D5 Using A4, A5, A6, A7, D4, and the jurisdiction rule, we derive,

$$\frac{Hub\_N|\equiv IIoTN|\Rightarrow< IIoTN \xleftrightarrow{p} Hub\_N, r_n, t_n, IIoTN \xleftrightarrow{a_n} Hub\_N >}{Hub\_N|\equiv(IIoTN \xleftrightarrow{a_n} Hub\_N}$$

(Sub-goal SG2)

FIGURE 5. Derived messages through message 1 and obtaining sub-goal 1 AND sub-goal 2.

3) ASSUMPTIONS

The assumptions to evaluate the security of our proposed mechanism are as follows:

$$A1: Hub\_N| \equiv (IIoTN \xleftrightarrow{id_n} Hub\_N)$$

$$A2: Hub\_N| \equiv \#(t_n)$$

$$A3: IIoTN| \equiv \#(r_n)$$

$$A4: Hub\_N| \equiv IIoTN| \equiv (IIoTN \xleftrightarrow{a_n})Hub\_N)$$

$$A5: Hub\_N| \sim (K_n)$$

$$A6: IIoTN| \equiv (IIoTN \xleftrightarrow{id_n} Hub\_N)$$

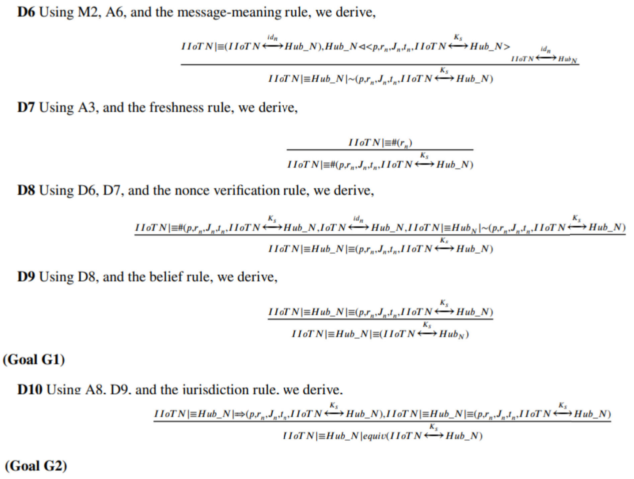
$$A7: IIoTN| \equiv Hub\_N| \sim (IIoTN \xleftrightarrow{id_{hn}} Hub\_N)$$

$$A8: IIoTN| \equiv Hub\_N| \equiv (IIoTN \xleftrightarrow{K_s} Hub\_N)$$

4) ANALYSIS

Here we are proving the security of our proposed scheme and deriving the above defined sub-goals and goals using rules, assumptions, and deducted messages as shown in Fig. 5 and Fig. 6.

This concludes that all the defined goals and sub-goals to prove the security of the proposed scheme offer mutual authentication with key agreement between IIoT device IIoTN and the hub node Hub_N. Hence, the proposed scheme is secure and no attack can be detected.



D6 Using M2, A6, and the message-meaning rule, we derive,

$$\frac{IIoTN|\equiv(IIoTN\xleftrightarrow{id_n}Hub\_N),Hub\_N\triangleleft<p,r_n,J_n,t_n,IIoTN\xleftrightarrow{K_s}Hub\_N>_{IIoTN\xleftrightarrow{id_n}Hub_N}}{IIoTN|\equiv Hub\_N|\sim(p,r_n,J_n,t_n,IIoTN\xleftrightarrow{K_s}Hub\_N)}$$

D7 Using A3, and the freshness rule, we derive,

$$\frac{IIoTN|\equiv\#(r_n)}{IIoTN|\equiv\#(p,r_n,J_n,t_n,IIoTN\xleftrightarrow{K_s}Hub\_N)}$$

D8 Using D6, D7, and the nonce verification rule, we derive,

$$\frac{IIoTN|\equiv\#(p,r_n,J_n,t_n,IIoTN\xleftrightarrow{K_s}Hub\_N,IoTN\xleftrightarrow{id_n}Hub\_N,IIoTN|\equiv Hub_N|\sim(p,r_n,J_n,t_n,IIoTN\xleftrightarrow{K_s}Hub\_N)}{IIoTN|\equiv Hub\_N|\equiv(p,r_n,J_n,t_n,IIoTN\xleftrightarrow{K_s}Hub\_N)}$$

D9 Using D8, and the belief rule, we derive,

$$\frac{IIoTN|\equiv Hub\_N|\equiv(p,r_n,J_n,t_n,IIoTN\xleftrightarrow{K_s}Hub\_N)}{IIoTN|\equiv Hub\_N|\equiv(IIoTN\xleftrightarrow{K_s}Hub_N)}$$

(Goal G1)

D10 Using A8. D9. and the iurisdiction rule. we derive.

$$\frac{IIoTN|\equiv Hub\_N|\Rightarrow(p,r_n,J_n,t_n,IIoTN\xleftrightarrow{K_s}Hub\_N),IIoTN|\equiv Hub\_N|\equiv(p,r_n,J_n,t_n,IIoTN\xleftrightarrow{K_s}Hub\_N)}{IIoTN|\equiv Hub\_N|equiv(IIoTN\xleftrightarrow{K_s}Hub\_N)}$$

(Goal G2)

FIGURE 6. Derived messages through message 2 and obtaining goal 1 AND goal 2.

B. AUTOMATED SECURITY EVALUATION

AVISPA supports a wide range of analyses, including reachability analysis, secrecy analysis, authentication analysis, and message trace analysis. Whereas ProVerif is specifically designed for symbolic analysis, and it focuses on proving properties such as secrecy and authentication. Using the AVISPA and ProVerif tools, we have performed the security analysis of the proposed scheme in this section.

1) ANALYSIS THROUGH AVISPA TOOL

A security evaluation tool for verification of protocols and applications that are essential to Internet security is AVISPA (Automated Validation of Internet Security Protocols and Applications) [41], [42]. Since AVISPA is a popular safety-analysis method among researchers, we employ it to check the safety objectives of the proposed scheme. The program is written in High-Level Protocol Specification Language, often known as HLPSL, to determine if a security protocol is SAFE or UNSAFE in co-relation to predetermined goals. Additionally, AVISPA is supported by the HLPSL's four integrated backends (Tree Automata based on Automatic Approximations for the Analysis of Security Protocols (TA4SP), SAT-based model checker, Constraint-Logic-based Attack Searcher (Cl-AtSe), and On-the-Fly Model-Checker (OFMC)), and abstraction-based techniques [43]. The HLPSL standard is translated into a language with a lower level than HLPSL. The so-called HLPSL2IF interpreter, which is opaque to the user, is used in the intermediate form. The AVISPA tool's back-ends read intermediate form (IF) directly. Each participant in the protocol simulation takes on a separate role in AVISPA, a language designed for roles. Each agent operates independently of the others, receives some starting data through parameters, and interacts with the other agents via channels. The Dolev-Yao [44] model is used to create the attacker, with the added potential for the attacker to play an appropriate role during a protocol run. The role system includes descriptions of the number of principals, sessions, and roles.

TABLE 3. Tabular representation of AVISPA output.

Backends	Output
OFMC	SAFE and Bounded number of Sessions
Cl-Atse	SAFE and Bounded number of Sessions

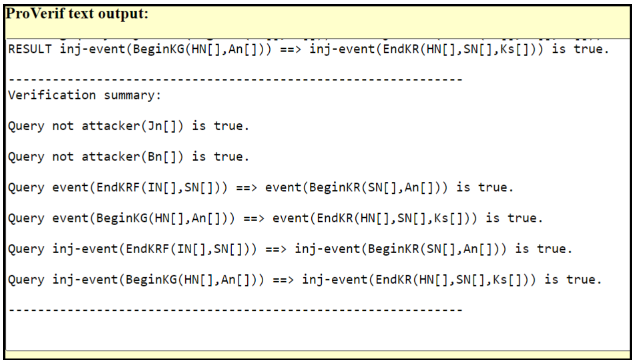
As we know that an intruder knows all the public constraints and transmitted messages, any kind of vulnerability, attack or disclosure of key can be identified using this tool. The output of AVISPA represents that the proposed scheme is secured against various known threats for Dolev-Yao model. Finally, the request statement ($HN, N, hubn\_node, Bn$) is used to authenticate the IIoT device N and the hub node HN through B_n. The authentication parameter 'B_n' incorporates secret value 'J_n' and random number 'r_n'. This secret value 'J_n' is accepted by IIoTN and the random number 'r_n' is also accepted by IIoTN. IIoTN believes in the presence of Hub_N and relies on parameter 'J_n' and 'r_n'. The authentication parameter 'B_n' is fresh and never replayed and the existence of Hub_N is bounded to protocol_id hubn_node. Our protocol claims the authentication based on a parameter (B_n), and it is found safe and secure at the OFMC, and the Cl-AtSe model checkers as shown in Table 3.

2) ANALYSIS THROUGH PROVERIF TOOL

ProVerif tool is used to analyse the security of the proposed protocol automatedly using Horn clauses. This tool handles unbounded number of sessions with different cryptographic operations like XOR and hash functions. Secrecy, authentication, and equivalence are among the processes that can be proved through ProVerif tool [45], [46], [47].

In our proposed protocol, the three processes such as IIoT devices (processSN), intermediate node (processIN) and hub node (processHN) are defined with begin and end events like registration, authentication, and session key generation. To ensure the security, different queries are imposed over the security parameters like secret key 'J_n' (Jn) and authentication parameter 'b_n' (Bn). Queries are applied on events to ensure the accomplishment of communication between processes. The queries are as follows: query attacker(Jn), attacker(Bn), query event(EndKRF(IN, SN))==>event (BeginKR(SN, An)), query event(BeginKG(HN, An))==>event(EndKR(HN, SN, Ks)), query inj-event(EndKRF(IN, SN))==>inj-event (BeginKR(SN, An)), and query inj-event(BeginKG(HN, An))==>inj-event(EndKR(HN, SN, Ks))

The findings of the proposed protocol are shown in Fig. 7. The proposed protocol is found safe and secure, and all queries are true. Here, we can conclude that our proposed protocol is invulnerable to various potential attacks as mentioned in Section IV.



```
ProVerif text output:
RESULT inj-event(BeginKG(HN[],An[])) ==> inj-event(EndKR(HN[],SN[],Ks[])) is true.

--------------------------------------------------------------
Verification summary:

Query not attacker(Jn[]) is true.

Query not attacker(Bn[]) is true.

Query event(EndKRF(IN[],SN[])) ==> event(BeginKR(SN[],An[])) is true.

Query event(BeginKG(HN[],An[])) ==> event(EndKR(HN[],SN[],Ks[])) is true.

Query inj-event(EndKRF(IN[],SN[])) ==> inj-event(BeginKR(SN[],An[])) is true.

Query inj-event(BeginKG(HN[],An[])) ==> inj-event(EndKR(HN[],SN[],Ks[])) is true.

--------------------------------------------------------------
```

FIGURE 7. ProVerif tool summary report.

TABLE 4. Calculation of communication cost.

Schemes	Communicating Nodes	Overheads	Total Cost
Li et al. [23]	Node and GWN	2688 bits	336 bytes
Cui et al. [33]	ES and SD	4904 bits	613 bytes
Proposed Scheme	$IIoTN \rightarrow Inter\_N$	512bits	252 bytes
	$Inter\_N \rightarrow Hub\_N$	528 bits	
	$Hub\_N \rightarrow Inter\_N$	496 bits	
	$Inter\_N \rightarrow IIoTN$	480 bits	

C. OVERHEAD EVALUATION AND COMPARISON

Overhead evaluation helps in proving the efficiency of our proposed protocol in terms of the communication cost, computation overhead, storage requirements, energy consumption and throughput achieved.

1) COMMUNICATION COST

The communication cost depends upon the messages transmitted between communicating entities like IIoT device, intermediate node, and hub node. Assume that the size of identity is 16 bits, the size of the timestamp is 32 bits, and the size of other parameters are 160 bits. In the transmission ($IIoTN \rightarrow Inter\_N$), IIoTN transmits a message $M1 :< TID_n, a_n, t_n, q >$. The size of this message is $3(160) + 32 = 512$ bits. In communication ($Inter\_N \rightarrow Hub\_N$), Inter_N transmits a message $M2 :< id_{in}, TID_n, a_n, t_n, q >$ and the size of this message is 528 bits. The size of messages $M3 :< id_{in}, TID_n, d, b_n >$ and $M4 :< TID_n, d, b_n >$ are transmitted from ($Hub\_N \rightarrow Inter\_N$) and ($Inter\_N \rightarrow IIoTN$), respectively. The size of the message M3 is 496 bits and the size of message M4 is 480 bits. The proposed protocol has a total communication cost of $512 + 528 + 496 + 480 = 2016$ bits. In Table 3 and Fig. 10, the proposed scheme is contrasted with the existing mechanism developed by Li et al. [23] and Cui et al. [33]. It is clear that our proposed scheme incurs the least communication cost as shown in Table 4 and Fig. 8.

2) COMPUTATION OVERHEAD/COST

Our proposed scheme employs two cryptographic operations: XOR operation and hash function. The computation cost for XOR operation (t_{XOR}) is considered negligible and the computation cost of hash function (t_{Hash}) is considerable.

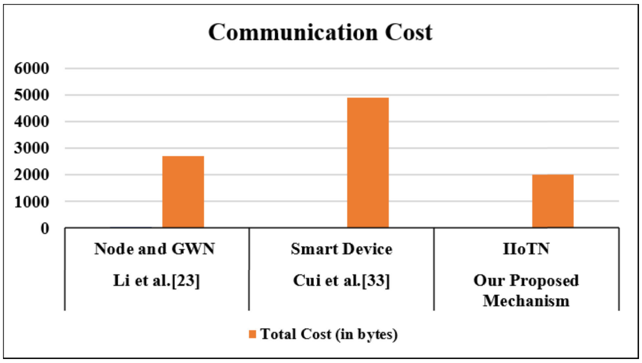


FIGURE 8. Evaluation of communication cost.

TABLE 5. Calculation of computation cost.

Schemes	Node/device	Operations	Overall Computation Cost
Li et al. [23]	Node	$4t_{Hash} + 2t_{SYM}$	$4 * 0.06 + 2 * 0.06 = 0.36ms$
	Node and GWN	$19t_{Hash} + 8t_{SYM} + 3t_{MM}$	$19 * 0.06 + 8 * 0.06 + 3 * 0.15 = 1.97ms$
Cui et al. [33]	SD	$3T_{mm} + 3T_{Hash}$	$3 * 0.15 + 3 * 0.06 = 0.63ms$
	ES and SD	$8T_{mm} + 3T_{ma} + 7T_{Hash}$	$8 * 0.15 + 3 * 0.15 + 7 * 0.06 = 2.07ms$
Proposed Scheme	IIoTN	$4t_{Hash} + 6t_{XOR} \approx 4t_{Hash}$	$4 * 0.06 = 0.24ms$
	IIoTN and Hub_N	$8t_{Hash} + 14t_{XOR} \approx 8t_{Hash}$	$8 * 0.06 = 0.48ms$

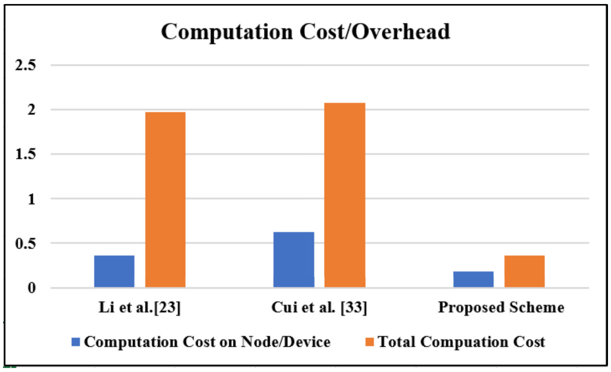


FIGURE 9. Evaluation of computation cost.

The computation cost of t_{Hash} is 0.06 millisec (ms) [48]. In our proposed scheme, the *IIoTN* device employs 6 XOR operations and 4 hash functions. The total operations at *IIoTN* device are 6 t_{XOR} + 4 t_{Hash}. The *Hub_N* node, on the other hand, carries out 8 XOR operations and 4 hash functions. The operations at *Hub_N* node are 8 t_{XOR} + 4 t_{Hash}. The overall operations performed in our proposed scheme are 8 $t_{Hash} + 14\ t_{XOR} \approx 8\ t_{Hash}$. Whereas the computation overhead of Li et al. [23] is 19 t_{Hash} + 8 t_{sym} + 3 t_{mm} where the cost of symmetric encryption (t_{sym}), addition operations (t_{ma}) and modular multiplication (t_{mm}) operations are taken from [48]. The computation overhead of edge server (ES) and smart device (SD) from Cui et al. [33] is $8T_{mm} + 3T_{ma} + 7T_{Hash}$ and the overhead of the smart device is $3T_{mm} + 3T_{Hash}$. The computation overhead of our proposed scheme and existing mechanisms proposed by Li et al. [23] and Cui et al. [33] is shown in Table 5 and Fig. 9.

The result from Table 4 and Fig. 11 shows that our proposed mechanism offers the least computational overhead which supports maximum battery life.

TABLE 6. Calculation of storage cost.

Schemes	Node	Storage Cost (bits)	Total Storage Cost (bits)
Li et al. [23]	Node	288	720
	GWN	432	
Cui et al. [33]	SD	840	840
Proposed Scheme	IIoTN	208	576
	Inter_N	48	
	Hub_N	320	

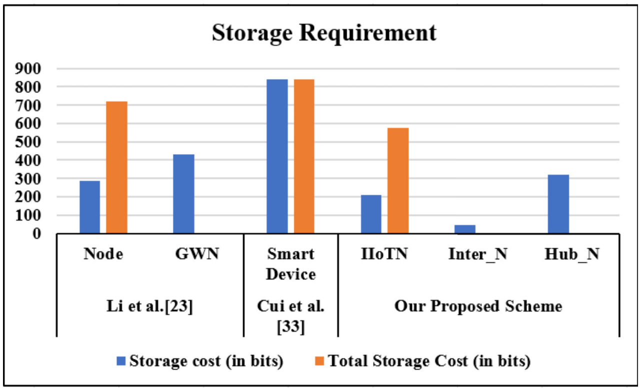


FIGURE 10. Evaluation of storage requirement.

3) STORAGE REQUIREMENT

In our proposed mechanism, IIoT device ($IIoTN$) stores tuple $< id_n, id_{in}, id_{hn}, a_n >$, intermediate node (Inter_N) tuple $< id_{in}, id_n, id_{hn} >$; and hub node ($Hub\_N$) stores tuple $< id_{hn}, id_n, id_{in}, J_n >$. Assume the hub node stores the identity of '8' registered IIoT devices, '1' intermediate node and its own identity from the network. The hub node stores identity of 10 nodes at a time. Our proposed protocol uses SHA-1 mechanism as hash function. As discussed earlier the size of different parameters, the storage requirement of $IIoTN$ is $3(16) + 160 = 208$ bits; the storage requirement of $Inter\_N$ is $3(16) = 48$ $bits$; and the storage requirement of Hub_N is $(16 * 10) + 160 = 320$ bits. In [23], the gateway node (GWN) stores its own identity, other nodes' identities and the pair of public-private keys; and the node stores its own identity, identity of gateway node and secret key in its storage. Assume the sizes of public and private keys are 128 bits each. In [33], the storage overhead of Cui et al. is mentioned as 840 bits. Table 6 and Fig. 10 compares the storage requirements of our proposed mechanism and that of Li et al. [23] and Cui et al. [33].

D. EXPERIMENTAL ANALYSIS

The proposed protocol was implemented using Bluetooth modules connected to a personal computer through Universal Serial Bus (USB) to transistor-to-transistor logic (TTL) connectors. Two Bluetooth modules were connected, one as Main and another as Secondary to a single computer for time synchronization. The payload of the proposed protocol was generated through Python program and transmitted from Main module to Secondary module through Bluetooth.

TABLE 7. Details of experimental setup.

IoT Device	HC-05 Bluetooth TTL modules
Connectors	USB to TTL connectors
Communication Technology	IEEE 802.15.1
Communication range	2m
Frequency	2.4GHz
Spectrum	ISM Spectrum (2400 to 2483.5MHz)

TABLE 8. Calculation of energy consumption.

Schemes	Execution time	Total Energy Consumption
Li et al. [23]	33.20 sec.	3286.8 Joules
Cui et al. [33]	58.10 sec.	5751.9 Joules
Proposed Scheme	19.20 sec.	1900.8 Joules

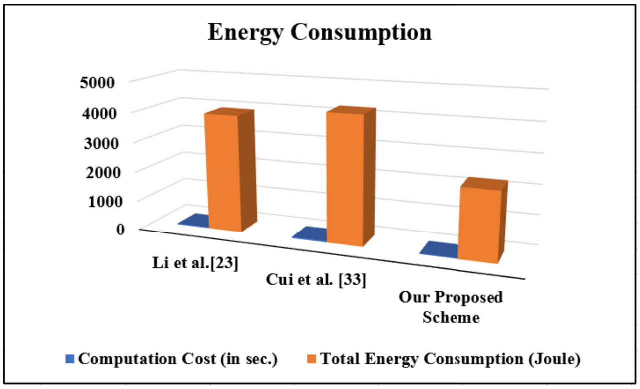


FIGURE 11. Evaluation of energy consumption.

TABLE 9. Calculation of throughput.

Schemes	Execution time	Throughput
Li et al. [23]	33.20 sec.	80.96 bps
Cui et al. [33]	58.10 sec.	84.40 bps
Proposed Scheme	19.20 sec.	105 bps

The time taken for transmission was calculated at Main module and time taken for receiving was calculated at Secondary module. The experimental setup is discussed in Table 7:

HC-05 Bluetooth module works on 3.3 Voltage (V) and operating current is 30 milli-Ampere (mA). The consumed energy is calculated using voltage(V), current(I) and payload transmission time (T).

Energy consumed by the proposed protocol $= V * I * T = 3.3\ V * 30\ mA * 19.20\ seconds = 1900.8$ Joules

Energy consumed by Li et al. [23] $= V * I * T = 3.3\ V * 30\ mA * 33.20\ seconds = 3286.8$ Joules

Energy consumed by Cui et al. [33] $= V * I * T = 3.3\ V * 30\ mA * 58.10\ seconds = 5751.9$ Joules

It is clear from Fig. 11 and Table 8 that the proposed protocol consumes less energy than Li et al. [23] and Cui et al. [33] for providing security, hence it is more suitable for IIoT networks.

TABLE 10. Comparison analysis of proposed mechanism with other mechanisms.

Schemes	Overheads/Costs Communication	 Computation	 Storage	Energy Consumption	Throughput	Possible Attacks	Implementation
Li et al. [23]	336 Bytes	0.36 ms	90 bytes	3286.8 J	80.96 bps	MITM, Eavesdropping, Backward and Forward Secrecy	AVISPA, NS3
Cui et al. [33]	613 Bytes	0.63 ms	105 Bytes	4266.9 J	84.40 bps	DDoS, Desynchronization, MITM attacks	Random Oracle Model, ProVerif
Ours	252 Bytes	0.24 ms	72 Bytes	1900.8 J	105 bps	Secure against various well-known mentioned attacks	BAN logic, AVISPA, ProVerif & HC-05 Bluetooth module

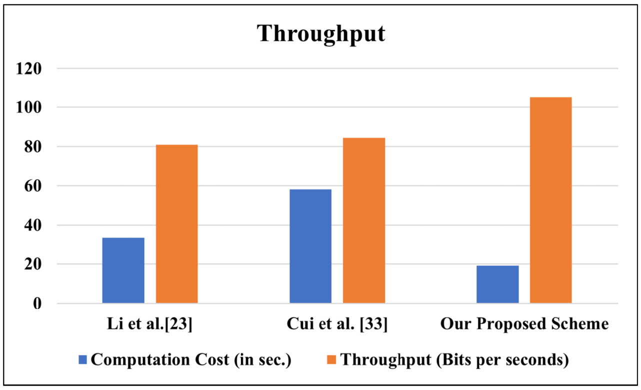


FIGURE 12. Evaluation of throughput.

1) THROUGHPUT

A valid measure of network performance, throughput is the actual pace at which data is successfully delivered over a network connection [49]. The throughput of our proposed protocol is 105 bps (bits per second), which is obtained on basis of the experimental setup. The throughput of Li et al. is calculated as 80.96 bps. The throughput of the mechanism proposed by Cui et al. is calculated as 84.40 bps. It is shown in Table 9 and Fig. 12 that the proposed mechanism offers the maximum throughput.

Performance of the proposed scheme has been evaluated through parameters like storage overhead, communication cost, computation overhead, throughput, and energy consumed and also compared with existing schemes. The energy consumed is extracted experimentally based on the execution time. The comparison of our proposed mechanism with other mechanisms [23], [33] is shown in tables 10.

VI. CONCLUSION

Industrial Internet of Things (IIoT) also known as Industry 4.0 is an emerging paradigm that aims to revolutionize the industry while improving productivity and reliability. Analogous to other IoT applications, security is a prime concern in the implementation of IIoT technology. In this research, we developed an IIoT device authentication and session key generation scheme for devices in IIoT networks. The proposed scheme avoids the participation of rogue devices and enhances the security of network and crucial data. The proposed scheme employs simple cryptographic operations including concatenation, bitwise XOR and one-way hash function. The proposed scheme is checked for security using AVISPA and ProVerif tools and mathematically using BAN logic. It has been proved that the proposed scheme is safe and avoids almost all identified vulnerabilities. The performance analysis of the proposed scheme found that it uses less power, needs less storage and computational resources with maximum throughput. It can be easily adopted in IIoT applications to secure the network and enhance the productivity and reliability.

DATA AVAILABILITY

The ProVerif and AVISPA code of this article will be shared on request by the corresponding author.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest to report regarding the present study.

REFERENCES

[1] A. B. Kathole, K. N. Vhatkar, and S. D. Patil, "IoT-enabled pest identification and classification with new meta-heuristic-based deep learning framework," *Cybern. Syst.*, vol. 55, no. 2, pp. 380–408, Feb. 2024.

[2] S. Kumbhare, S. A. Ubale, G. Dharmale, N. Mhala, and N. Gandhewar, "IoT-enabled agricultural waste management for sustainable energy generation," *Int. J. Intell. Syst. Appl. Eng.*, vol. 12, no. 13s, pp. 477–482, 2024.

[3] A. B. Kathole, K. N. Vhatkar, S. Kumbhare, J. Katti, and V. V. Kimbahune, "IoT-based smart agriculture for onion plant disease management: A comprehensive approach," *Int. J. Intell. Syst. Appl. Eng.*, vol. 12, no. 13s, pp. 472–476, 2024.

[4] H. Boyes, B. Hallaq, J. Cunningham, and T. Watson, "The Industrial Internet of Things (IIoT): An analysis framework," *Comput. Ind.*, vol. 101, pp. 1–12, Oct. 2018.

[5] A. Hazra, M. Adhikari, T. Amgoth, and S. N. Srirama, "A comprehensive survey on interoperability for IIoT: Taxonomy, standards, and future directions," *ACM Comput. Surv.*, vol. 55, no. 1, pp. 1–35, Jan. 2023.

[6] S. Schneider, "The Industrial Internet of Things (IIoT) applications and taxonomy," in *Internet of Things and Data Analytics Handbook*. Hoboken, NJ, USA: Wiley, 2017, pp. 41–81.

[7] S. Vitturi, C. Zunino, and T. Sauter, "Industrial communication systems and their future challenges: Next-generation Ethernet, IIoT, and 5G," *Proc. IEEE*, vol. 107, no. 6, pp. 944–961, Jun. 2019.

[8] M. Tanveer, A. A. A. El-Latif, A. U. Khan, M. Ahmad, and A. A. Ateya, "LEAF-IIoT: Lightweight and efficient authentication framework for the Industrial Internet of Things," *IEEE Access*, vol. 12, pp. 31771–31787, 2024.

[9] S. F. Tan and A. Samsudin, "Recent technologies, security countermeasure and ongoing challenges of Industrial Internet of Things (IIoT): A survey," *Sensors*, vol. 21, no. 19, p. 6647, Oct. 2021.

[10] G. Rathee, F. Ahmad, N. Jaglan, and C. Konstantinou, "A secure and trusted mechanism for industrial IoT network using blockchain," *IEEE Trans. Ind. Informat.*, vol. 19, no. 2, pp. 1894–1902, Feb. 2023.

[11] T. Gebremichael, L. P. I. Ledwaba, M. H. Eldefrawy, G. P. Hancke, N. Pereira, M. Gidlund, and J. Akerberg, "Security and privacy in the Industrial Internet of Things: Current standards and future challenges," *IEEE Access*, vol. 8, pp. 152351–152366, 2020.

[12] H. Vargas, C. Lozano-Garzon, G. A. Montoya, and Y. Donoso, "Detection of security attacks in industrial IoT networks: A blockchain and machine learning approach," *Electronics*, vol. 10, no. 21, p. 2662, Oct. 2021.

[13] M. Wang, Y. Sun, H. Sun, and B. Zhang, "Security issues on Industrial Internet of Things: Overview and challenges," *Computers*, vol. 12, no. 12, p. 256, Dec. 2023.

[14] X. Yu and H. Guo, "A survey on IIoT security," in *Proc. IEEE VTS Asia–Pacific Wireless Commun. Symp. (APWCS)*, Aug. 2019, pp. 1–5.

[15] M. Agrawal, J. Zhou, and D. Chang, "A survey on lightweight authenticated encryption and challenges for securing industrial IoT," in *Security and Privacy Trends in the Industrial Internet of Things*. Cham, Switzerland: Springer, 2019, pp. 71–94.

[16] M. Serror, S. Hack, M. Henze, M. Schuba, and K. Wehrle, "Challenges and opportunities in securing the Industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 17, no. 5, pp. 2985–2996, May 2021.

[17] C. Patel, A. K. Bashir, A. A. AlZubi, and R. Jhaveri, "EBAKE-SE: A novel ECC-based authenticated key exchange between industrial IoT devices using secure element," *Digit. Commun. Netw.*, vol. 9, no. 2, pp. 358–366, Apr. 2023.

[18] E. Lara, L. Aguilar, M. A. Sanchez, and J. A. García, "Lightweight authentication protocol for M2M communications of resource-constrained devices in Industrial Internet of Things," *Sensors*, vol. 20, no. 2, p. 501, Jan. 2020.

[19] C. Lupascu, A. Lupascu, and I. Bica, "DLT based authentication framework for industrial IoT devices," *Sensors*, vol. 20, no. 9, p. 2621, May 2020.

[20] T. Mick, R. Tourani, and S. Misra, "LASeR: Lightweight authentication and secured routing for NDN IoT in smart cities," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 755–764, Apr. 2018.

[21] P. Kumar, A. Gurtov, M. Sain, A. Martin, and P. H. Ha, "Lightweight authentication and key agreement for smart metering in smart energy networks," *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 4349–4359, Jul. 2019.

[22] N. Li, D. Liu, and S. Nepal, "Lightweight mutual authentication for IoT and its applications," *IEEE Trans. Sustain. Comput.*, vol. 2, no. 4, pp. 359–370, Oct. 2017.

[23] X. Li, J. Peng, J. Niu, F. Wu, J. Liao, and K. R. Choo, "A robust and energy efficient authentication protocol for Industrial Internet of Things," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1606–1615, Jun. 2018.

[24] J. Singh, A. Gimekar, and S. Venkatesan, "An efficient lightweight authentication scheme for human-centered Industrial Internet of Things," *Int. J. Commun. Syst.*, vol. 36, no. 12, pp. 1–13, Nov. 2019.

[25] H. A. N. Far, M. Bayat, A. K. Das, M. Fotouhi, S. M. Pournaghi, and M. A. Doostari, "LAPTAS: Lightweight anonymous privacy-preserving three-factor authentication scheme for WSN-based IIoT," *Wireless Netw.*, vol. 27, no. 2, pp. 1389–1412, Feb. 2021.

[26] M. N. Aman and B. Sikdar, "ATT-Auth: A hybrid protocol for industrial IoT attestation with authentication," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 5119–5131, Dec. 2018.

[27] M. Shen, H. Liu, L. Zhu, K. Xu, H. Yu, X. Du, and M. Guizani, "Blockchain-assisted secure device authentication for cross-domain industrial IoT," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 5, pp. 942–954, May 2020.

[28] A. Esfahani, G. Mantas, R. Matischek, F. B. Saghezchi, J. Rodriguez, A. Bicaku, S. Maksuti, M. G. Tauber, C. Schmittner, and J. Bastos, "A lightweight authentication mechanism for M2M communications in industrial IoT environment," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 288–296, Feb. 2019.

[29] K. Shahzad, M. Alam, N. Javaid, A. Waheed, S. A. Chaudhry, N. Mansoor, and M. Zareei, "SF-LAP: Secure M2M communication in IIoT with a single-factor lightweight authentication protocol," *J. Sensors*, vol. 2022, pp. 1–16, Nov. 2022.

[30] R. Vinoth, L. J. Deborah, P. Vijayakumar, and N. Kumar, "Secure multifactor authenticated key agreement scheme for industrial IoT," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3801–3811, Mar. 2021.

[31] M. Tanveer, A. Alkhayyat, A. U. Khan, N. Kumar, and A. G. Alharbi, "REAP-IIoT: Resource-efficient authentication protocol for the Industrial Internet of Things," *IEEE Internet Things J.*, vol. 9, no. 23, pp. 24453–24465, Dec. 2022.

[32] M. N. Sohail, A. Anjum, I. A. Saeed, M. H. Syed, A. Jantsch, and S. Rehman, "Optimizing industrial IoT data security through blockchain-enabled incentive-driven game theoretic approach for data sharing," *IEEE Access*, vol. 12, pp. 51176–51192, 2024.

[33] J. Cui, Y. Zhu, H. Zhong, Q. Zhang, C. Gu, and D. He, "Efficient blockchain-based mutual authentication and session key agreement for cross-domain IIoT," *IEEE Internet Things J.*, vol. 11, no. 9, pp. 16325–16338, Sep. 2024.

[34] K.-A. Shim, "A secure certificateless signature scheme for cloud-assisted industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 20, no. 4, pp. 6834–6843, Apr. 2024.

[35] W. Wang, H. Xu, M. Alazab, T. R. Gadekallu, Z. Han, and C. Su, "Blockchain-based reliable and efficient certificateless signature for IIoT devices," *IEEE Trans. Ind. Informat.*, vol. 18, no. 10, pp. 7059–7067, Oct. 2022.

[36] Y. Han, H. Guo, J. Liu, B. B. Ehui, Y. Wu, and S. Li, "An enhanced multifactor authentication and key agreement protocol in Industrial Internet of Things," *IEEE Internet Things J.*, vol. 11, no. 9, pp. 16243–16254, May 2024.

[37] J. M. Sierra, J. C. Hernández, A. Alcaide, and J. Torres, "Validating the use of ban logic," in *Proc. Int. Conf. Comput. Sci. Appl. (ICCSA)*, Assisi, Italy. Cham, Switzerland: Springer, May 2004, pp. 851–858.

[38] J. Wen, M. Zhang, and X. Li, "The study on the application of BAN logic in formal analysis of authentication protocols," in *Proc. 7th Int. Conf. Electron. Commerce (ICEC)*, 2005, pp. 744–747.

[39] K. Fan, H. Li, and Y. Wang, "Security analysis of the kerberos protocol using BAN logic," in *Proc. 5th Int. Conf. Inf. Assurance Secur.*, vol. 2, Aug. 2009, pp. 467–470.

[40] U. Jain, S. Pirasteh, and M. Hussain, "Lightweight, secure, efficient, and dynamic scheme for mutual authentication of devices in Internet-of-Things-Fog environment," *Concurrency Comput., Pract. Exper.*, vol. 35, no. 1, p. e7428, Jan. 2023.

[41] A. Armando, D. Basin, Y. Boichut, Y. Chevalier, L. Compagna, J. Cuellar, P. H. Drielsma, P. C. Heám, O. Kouchnarenko, J. Mantovani, S. Mödersheim, D. von Oheimb, M. Rusinowitch, J. Santiago, M. Turuani, L. Viganò, and L. Vigneron, "The AVISPA tool for the automated validation of Internet security protocols and applications," in *Proc. 17th Int. Conf. Comput.-Aided Verification*, Scotland, U.K. Cham, Switzerland: Springer, Jul. 2005, pp. 281–285.

[42] L. Viganò, "Automated security protocol analysis with the AVISPA tool," *Electron. Notes Theor. Comput. Sci.*, vol. 155, pp. 61–86, May 2006.

[43] J. A. Hurtado Alegría, M. C. Bastarrica, and A. Bergel, "AVISPA: A tool for analyzing software process models," *J. Softw., Evol. Process*, vol. 26, no. 4, pp. 434–450, Apr. 2014.

[44] I. Cervesato, "The Dolev–Yao intruder is the most powerful attacker," in *Proc. 16th Annu. Symp. Log. Comput. Sci. (LICS*, vol. 1, 2001, pp. 1–2.

[45] H. M. N. Al Hamadi, C. Y. Y. M. J. Zemerly, M. A. Al-Qutayri, and A. Gawanmeh, "Verifying mutual authentication for the DLK protocol using ProVerif tool," *Int. J. Inf. Secur. Res.*, vol. 3, no. 1, pp. 256–265, Mar. 2013.

[46] B. Blanchet, B. Smyth, V. Cheval, and M. Sylvestre, "ProVerif 2.00: Automatic cryptographic protocol verifier, user manual and tutorial," INRIA, Ecole Normale Superieure, Paris, France, Max-Planck Institut Informatik, Saarbrücken, Germany, 2018, pp. 5–16. [Online]. Available: https://bblanche.gitlabpages.inria.fr/proverif/manual.pdf

[47] B. Blanchet, V. Cheval, and V. Cortier, "ProVerif with lemmas, induction, fast subsumption, and much more," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2022, pp. 69–86.

[48] X. Li, M. H. Ibrahim, S. Kumari, A. K. Sangaiah, V. Gupta, and K.-K.-R. Choo, "Anonymous mutual authentication and key agreement scheme for wearable sensors in wireless body area networks," *Comput. Netw.*, vol. 129, pp. 429–443, Dec. 2017.

[49] E. K. K. Edris, M. Aiash, M. A. Khoshkholghi, R. Naha, A. Chowdhury, and J. Loo, "Performance and cryptographic evaluation of security protocols in distributed networks using applied pi calculus and Markov chain," *Internet Things*, vol. 24, Dec. 2023, Art. no. 100913.

AKSHAY KUMAR received the M.Tech. degree in computer science and engineering from the Central University of Punjab. He is currently pursuing the Ph.D. degree in computer science and engineering with the Central University of Rajasthan. He has published three papers at international conferences of repute. His research interests include the Industrial Internet of Things (IIoT) and security in IoT.

USHA JAIN (Member, IEEE) received the M.Tech. degree in systems engineering from Devi Ahilya Vishwavidhyalaya, Indore, India, in 2014, and the Ph.D. degree from the Central University of Rajasthan, Ajmer, India, in 2021. She is currently an Assistant Professor in computer science engineering with Manipal University Jaipur, Rajasthan, India. She has published research articles in reputed conferences and journals. Her research interests include information security, wireless sensor networks, and the IoT.

MUZZAMMIL HUSSAIN received the M.Tech. degree in computer science and engineering from Jawaharlal Nehru Technological University, Hyderabad, India, in 2009, and the Ph.D. degree in computer science from Kakatiya University, Warangal, India, in 2012. In 2012, he joined the Christu Jyothi Institute of Technology and Sciences, Warangal, as a Professor and a Principal. In 2013, he joined the Central University of Rajasthan, Ajmer, India, as an Assistant Professor in computer science and engineering. He has more than 50 publications in reputed conferences and journals and edited one book. His research interests include information security, wireless sensor networks, cognitive radio networks, mobile ad hoc networks, and the IoT security. He is a reviewer of many reputed journals.

MOHAMMAD KHALID IMAM RAHMANI (Senior Member, IEEE) was born in Patharghatti, Kishanganj, India, in 1975. He received the B.Sc. (Engg.) degree in computer engineering from Aligarh Muslim University, India, in 1998, the M.Tech. degree in computer engineering from Maharshi Dayanand University, Rohtak, in 2010, and the Ph.D. degree in computer science engineering from Mewar University, India, in 2015. From 1999 to 2006, he was a Lecturer with the Maulana Azad College of Engineering and Technology, Patna. From 2006 to 2008, he was a Lecturer and a Senior Lecturer with the Galgotias College of Engineering and Technology, Greater Noida. From 2010 to 2011, he was an Assistant Professor with MVN, Palwal. Currently, he is an Associate Professor with the Department of Computer Science, College of Computing and Informatics, Saudi Electronic University, Riyadh, Saudi Arabia. He has published more than 80 research papers in journals (47 SCI) and conferences of international repute and three book chapters. He holds one USA patent and another Australian patent of innovation. His research interests include algorithms, the IoT, cryptography, image retrieval, pattern recognition, machine learning, and deep learning.

ABDULBASID S. BANGA has been an Assistant Professor with the College of Computing and Informatics, Saudi Electronic University, Riyadh, Saudi Arabia, since 2014. He has more than 18 years of academic experience teaching in various universities in India. Previously, he was an Assistant Professor with the GLS Institute of Computer Technology (MCA Course), Ahmedabad, India. He has vast experience teaching at national and international levels. He is associated with various technical societies of national and international reputation. He has published various research articles in reputable journals. He is a Life Member of the Computer Society of India (CSI), profoundly engrossed in software estimation models, software engineering, data science, machine learning, artificial intelligence, blockchain technology, the Internet of Things, and cloud computing.