Post Quantum Lattice-Based Secure Framework using Aggregate Signature for Ambient Intelligence Assisted Blockchain-Based IoT Applications

Prithwi Bagchi, Basudeb Bera, Ashok Kumar Das, Sachin Shetty, Pandi Vijayakumar, and Marimuthu Karuppiah

Abstract

Many classical cryptographic techniques are breakable due to the quantum computing security threats, and it leads to design public key cryptography based on post-quantum cryptography primitives and security protocols. In recent years, Lattice-Based Cryptography (LBC) becomes a prominent post-quantum cryptographic primitive that can be applied in both traditional and emerging security domains, including encryption, key agreement, digital signature and homomorphic encryption. In this article, we first provide a LBC-based security framework using aggregate signature that can be applied in ambient intelligence-assisted blockchain-based Internet of Things (IoT) applications, called LAS-AIBIoT. In LAS-AIBIoT, the wearable/medical devices deployed in the patients' body securely send the sensing secret data (encrypted messages) with their respective lattice-based signatures to their nearby controller nodes (CN), where the CNs forward these secret messages to the attached aggregator node (Aggr). Each Aggr verifies the individual signature of the devices and constructs the aggregate signature on the received secret messages and signatures, and sends the aggregated secret messages with their aggregate signature to the cloud server(s) for block construction in the blockchain center. Through the consensus protocol, the block is then mined and added into the blokchain. We show the robustness of LAS-AIBIoT against various potential attacks including quantum computing security threats through the threat model discussed in this article. Finally, through the blockchain-based simulation study we show that LAS-AIBIoT can be applied for real-time ambient intelligence-assisted IoT applications.

Introduction

Advancement of Information and Communications Technology (ICT) has led to conceive the concept of Internet of Things (IoT). In IoT, billion of physical devices (also called IoT smart devices) around the world are connected to the Internet. The IoT smart devices are responsible to collect several information including military, healthcare, agricultural and wildlife data. Since the communication among the entities in the IoT network takes place via open medium, there are different threats where an adversary can intentionally launch various attacks, such as replay, impersonation, man-in-the-middle, denial of service and privileged-insider attacks. Furthermore, sometimes it becomes more serious when some IoT devices are physically captured and the adversary launches other attacks using the extracted credentials stored in the IoT devices.

An Ambient Assisted Living (AAL) is a healthcare system that offers an ecosystem of various medical sensor devices (wearable and implantable devices), wireless networks, computers and software applications for monitoring the health-related information. AAL applications also provides user-dependent services for the patients who are specifically elder and disabled people, where in problematic scenario the monitoring of the vital signs or controlling a patient's movement tracking can be done via the analysis of sensed heterogeneous information. In general, an ambient intelligence system needs to handle with a wide variety of smart devices, which range from the smart phones and tablets with medium capability, to consumer electronics, sensors, actuators and wearable devices having severe resource constraints [1].

Prithwi Bagchi, Baudeb Bera, and Ashok Kumar Das are with the International Institute of Information Technology, India.

Sachin Shetty is with Old Dominion University, USA.

Pandi Vijayakumar is with the University College of Engineering Tindivanam, India.

Marimuthu Karuppiah is with the Presidency University, Bengaluru, India.

Digital Object Identifier: 10.1109/IOTM.001.2100215

To handle AAL efficiently, Cubo *et al.* [1] developed a platform in order to handle the integration and behavior-aware composition of smart devices whereas the various services can be stored and also accessed via the cloud in the AAL-based applications. He and Zeadally [2] suggested an authentication scheme for the AAL system, where mutual authentication between controller and the user is performed with the help of the AAL server. In addition, successful mutual authentication allows the data transmitted to the controller and the attached users is encrypted using the established session key among them.

In recent years, Lattice-Based Cryptography (LBC) demonstrates as a promising cryptographic primitive that is treated as a quantum-safe alternative to the existing public-key cryptosystems. Moreover, the LBC implementations are also notable primarily due to their inherent linear algebra-based matrix/vector operations on integers [3]. As a result, it makes them a viable option to be considered for the resource constrained devices including the IoT devices. For instance, Chaudhary *et al.* [4] designed a lattice-based secure scheme for smart healthcare in a future smart city environment. They used both key exchange and authentication mechanisms which are lightweight in nature.

Blockchain technology makes unique opportunities for complexity reduction, enabling trust-less collaboration, and also secure and immutable data creation. In healthcare, the blockchain technology has the potential to place the patients at the center of a healthcare ecosystem, and at the same time to increase security, privacy, and interoperability of the healthcare related data. Thus, the blockchain technology can offer a new computing tool for health-related information exchange by utilizing the electronic medical records more efficiently and securely. Nehra *et al.* [5] proposed a blockchain-based ambient assisted living application which is more applicable for the critically ill patients. The use of blockchain in their implementation makes much stronger security in the application.

Research Challenges

Quantum computing is a big threat on the security of various traditional public key based cryptographic schemes, because the computational hard problems like Integer Factorization Problem (IFP) in RSA-based public key cryptosystem and Discrete Logarithm Problem (DLP) in public-key based ElGamal cryptosystem can be solved in polynomial time. In addition, digital signature also plays an important role in verifying whether a received message is authentic or not by a verifier. At present, Lattic-Based Cryptography (LBC) is a prominent post-quantum cryptographic primitive that can be applied in both traditional and emerging security domains. The post-quantum lattice-based cryptographic signature schemes are specifically based on the hardness of some few well-known (average case) problems, like "Learning with Errors (LWE)" and "Shortest Integer Solution (SIS)" [6]. The LWE problem is a computational problem to find a vector s from a given matrix A and a given vector $b = As + e$, e represents a unknown small error vector. A signature scheme relies on LWE is called the standard lattice-based signature (LBS) scheme. To reduce memory requirement in standard LBS, which uses matrix-vector multiplications, a ring lattice-based signature scheme has been suggested. In a ring lattice-based signature scheme, the entire matrix is replaced by a single row vector, which reduces the memory. The hardness of a ring lattice-based scheme depends on mostly ring-LWE and ring-SIS problems. Aggregate signature is a variant of the digital signature schemes, where a group of users being the signers can generate the signatures and send the signatures to an aggregator. To prove the authenticity of the signers, the aggregator verifies the individual signatures of the signers and then generates a single compact signature by combining the received signatures. The verifier verifies the aggregate signature using the public keys of the signers and the aggregator on the received aggregate messages. One of the advantage of using aggregate signature is that it reduce the size of the generated aggregate signature. The aggregate messages along with the aggregate signature are then included in the blocks which are verified and mined by the cloud server(s) in a blockchain center (BC). The cloud servers in the BC form a Peer-to-Peer (P2P) CS network. Since the data in an ambient intelligence-assisted IoT application is strictly confidential and private, we put the encrypted data (encrypted transactions) in the blocks that are encrypted by the lattice-based public key of the aggregator node.

It is recommended that the session key should be established using both the short-term and long-term secrets so that the effect of session keys compromise will be minimal under the CK-adversary model.

Outline

The organization of this article is as follows. The (network and threat) models associated with the proposed LAS-AIBIoT are discussed in the next section. After that, we discuss a generalized lattice-based encryption and aggregate signature scheme for healthcare system and show how such a scheme can be applied in constructing blockchain-based framework for ambient intelligence in IoT. A brief security analysis based on the proposed threat model has been provided in the next section. Finally, blockchain implementation, related work and concluding remarks are provided in this article.

System Models

This section elaborates the network as well as threat models of the proposed LAS-AIBIoT for a private blockchain-based ambient intelligence in IoT.

Network Model

We consider several wearable/medical devices (WDs) as IoT smart devices that are deployed in a patient's body, the controller node (CNs) considered as laptop-type devices, and the edge servers (also termed as aggregators *Aggr*), as the network entities. An edge server being a trusted authority (here, a hospital authority) *TA* in a hospital acts as a registration authority for registering the entities, like *WD*s, *CN*s and *Aggr* in a hospital. Apart from this, we consider the cloud servers which form a distributed P2P network for blockchain technology. In this article, *WD*s are considered as signers, whereas the edge servers are considered as aggregators (*Aggr*s) as they are responsible for secure data collection from the deployed *WD*s. During the registration process, *WD*s, *CN*s, and *Aggr*s are provided with the lattice-based secret and public keys pairs by the *TA*. The *TA* the makes the public keys of all the entities as public, which can be later used for lattice-based encryption and signature generation processes on the sensing medical information of the registered patients. The *WD*s send the encrypted medical data using the $Aggr_i$'s public key and an attached signature generated with its own secret key on the encrypted data to the associated *CN* via public channel. Next, the *CN* forwards the messages to the associated aggregator $Aggr_i$ and $Aggr_i$ can perform verification on the received messages from *CN*s. Next, on the received messages $Aggr_i$ generates a lattice-based aggregate signature on the received signatures and then constructs a new message containing these received encrypted messages and the computed aggregate signature in order to send it to the P2P cloud servers network as transaction formats. The in-charge P2P cloud server constructs blocks with the received transactions and preforms the blockchain execution for adding the received transactions into blockchain using the voting-based consensus mechanism.

Threat Model

In the considered ambient intelligence-assisted IoT application (LAS-AIBIoT), the generated data is confidential and it needs to maintain the privacy. In our network model, the wearable devices *WD*s communicate with the connected controller node *CN* via public channel, where the patient related private information needs to be sent by the *CN* and also to be exchanged with the associated aggregators *Aggr*, whereas *Aggr* communicates with the cloud server(s) in the blockchain center. Since all the communications in this network are performed over the insecure (public) channel, the data privacy becomes a serious concern. We consider a widely-recognized security threat model, called the "Dolev-Yao (DY) threat model" [7] in which an adversary, say $\mathcal{A}$ can interrupt the communication by performing various tasks, like modification and deletion of the legitimate messages and injection of the unauthorized massages over the communication channel. In addition, a recent *de facto* adversary model, called the "Canetti and Krawczyk's model (CK-adversary model)" [8] has been adopted, where the $\mathcal{A}$ has the ability more than that for the DY threat model, and $\mathcal{A}$ can additionally compromise the session states by hijacking a session. This means that through the session hijacking attacks, the adversary $\mathcal{A}$ will have short term secrets, such as random nonces, generated during the session key establishment and the long-term secrets if they are stored in insecure memory of the devices (entities) along with the intercepted messages exchanged among the communicating parties in the network. As a result, it is recommended that the session key should be established using both the short-term and long-term secrets so that the effect of session keys compromise will be minimal under the CK-adversary model. Finally, we assume that the secret credentials loaded into the physically captured wearable/medical device(s) can be extracted by $\mathcal{A}$ using the "power analysis attacks" [9] as in case of the sensor nodes capture in wireless sensor networks [10].

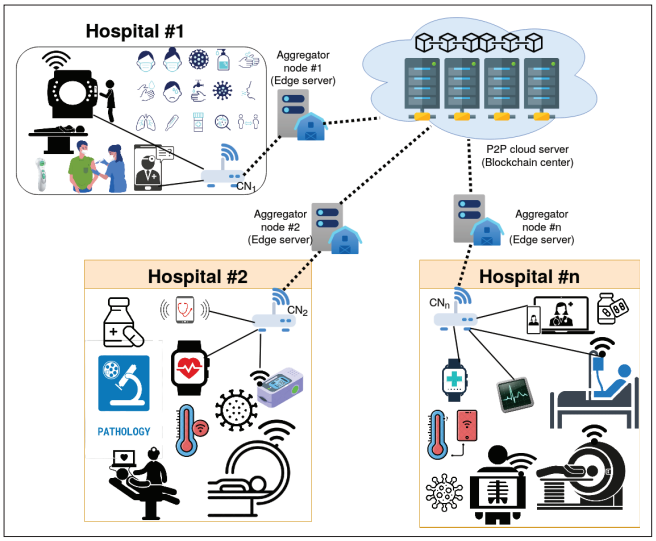

FIGURE 1. Network model for an ambient intelligence-assisted IoT application.

Generalized Lattice-Based Encryption and Aggregate Signature

We now discuss a generalized lattice-based encryption and aggregate signature scheme for healthcare system, which consist a set of algorithms:

1. Key generation (*KeyGen*)
2. Lattice-based encryption (*LEnc*)
3. Lattice-based decryption (*LDec*)
4. Lattice-based single signature generation (*LSign*) where a message is signed by an entity
5. Lattice-based single signature verification (*LSVer*) where the signature is verified by a verifier
6. Lattice-based aggregate signature generation (LASign) where an aggregator signs the combined messages and their signatures to form a compact short aggregate signature
7. Lattice-based aggregate signature verification (*LAVer*) where a verifier verifies the aggregate signature. The details of these algorithms are provided below.

- *KeyGen*: The trusted authority *TA* of each hospital runs this algorithm in offline mode to generate the secret (private) and public key pairs for each of the deployed wearable/medical devices *WD*s which act as signers, controller nodes (*CN*s), and the edge server (here, it is the *TA* itself) which is also considered as an aggregator *Aggr*. The generated key pairs are then sent securely to the corresponding entities via secure channel, and the public keys are made publicly available.
- *LEnc*: This algorithm is executed by each signer (*WD*) in the network with the receiver *Aggr*'s public key. In this algorithm, the utilized data is considered as the healthcare-related private and sensitive data.
- *LDec*: This algorithm is performed by the receiver, who is the aggregator *Aggr*. Once the encrypted message is received from the *WD* via *CN*, *Aggr* can decrypt the encrypted message with its own secret key and can also view the original data, if needed.
- *LSign*: Each signer, being a wearable device *WD*, in the network runs this lattice-based signature generation algorithm in order to generate the signature, where *WD* uses its own secret key. For the verification purpose, a verifier will use the corresponding public key using the *LVer* algorithm.
- *LVer*: The authenticity of the message can be verified by performing this algorithm, where the *Aggr* will run this algorithm for verifying the messages from the *WD*s via the *CN*s.
- *LASign*: The aggregator *Aggr* runs this algorithm on the received signatures on the corresponding messages from the control node *CN*. Here *CN* acts like a gateway between *WD*s and *Aggr*. The *Aggr* produces an aggregate signature (or a compact short signature) on the received messages along with their signatures, and then sends the combined message with the aggregate signature to the associated cloud server in the BC.
- *LAVer*: Once a cloud Server CS receives the aggregate signed messages from *Aggr*, CS executes this algorithm for verifying the received aggregate signature with the help of all the public keys of the signers and the *Aggr*, and all the available public information.

Proposed Blockchain-Based Framework for Ambient Intelligence in IoT using Lattice-Based Aggregate Signature

In this section, we describe the proposed scheme (LAS-AIBIoT), where it incorporates the above discussed lattice-based aggregate signature scheme and also the blockchain technology to provide stronger security as compared to the schemes proposed under the traditional public-key based cryptosystems.

An aggregate signature scheme permits each signer to sign a different message with the help of its own private key. Next, all the generated signatures are aggregated into a single compact (short) signature, called an aggregate signature. On the other side, a multisignature scheme permits the multisigners to sign only one message jointly. The proposed LAS-AIBIoT is more inclined towards the application of aggregate signature because each wearable device *WD* as a signer can sign an encrypted data using its own private key and the aggregator node having all the individual signatures gathered from the wearable devices (signers) can produce a compact aggregate signature on those signatures using its own private key. Later, the aggregate signature can be verified by any verifier using the public keys of the signers.

Various phases connected to the proposed LAS-AIBIoT are described below.

Registration Phase

Each trusted authority *TA* in respective hospitals is responsible for registering all the network entities prior to their deployment. To register the network entities, the *TA* runs in offline the following key generation algorithm.

$KeyGen(1^k) \rightarrow (pk_i, sk_i)$: The *TA* runs this algorithm with a security parameter k as an input, and produces lattice-based secret (private) and public key pair, say (sk_i, pk_i) for each signer WD_i, where i = 1, 2, ⋯, n and also the lattice-based secret and public key pair (sk_a, pk_a) for the aggregator *Aggr*, where n is the number of *WD*s to be deployed in the network. For this purpose, the *TA* chooses a fixed polynomial a of degree at most m-1 to generate the public key, whereas the secret key sk_i is another pair of polynomials having the degree at most $m - 1$. The public key is of the form: $pk_i = (a, 1) \cdot sk_i$ which is also a polynomial of degree at most m-1. All the coefficients of each polynomial in sk_i are integers that are picked from the set [−1, 1], whereas the coefficients of each pk_i and the fixed polynomial a are chosen from a finite field $\mathbb{Z}_p$, where p is considered as a large prime. Finally, the *TA* securely sends the corresponding secret and public keys pairs to each signer and the *Aggr*. The generated public keys are then declared as public by the *TA*.

Data Encryption/Decryption Phase

Once the registration process is over for the wearable/medical devices, the devices are now ready for deployment. Next, they need to communicate with the associated controller node *CN* by sending sensed data securely. To do so, the devices *WD* will execute the following lattice-based algorithms.

- $LEnc(pk_a, data_i, TS_i) \rightarrow LEData_i$: Each signer *WD* executes this algorithm to produce the lattice-based encrypted data, say $LEData_i$ on the sensed healthcare-related private data $data_i$ with the generated current timestamp TS_i by the public key

pk_a of the *Aggr*. After that, *WD* generates a lattice-based signature on the encrypted data ($LEData_i$) with its own secret key sk_i using the *LSign* algorithm, and then constructs a message $Msg_i = \{LEData_i, LSign(sk_d, LEData_i), TS_i\}$, which is sent to the connected *CN* via public channel.

- $LDec(sk_a, LEData_i) \rightarrow (pk_a, data_i, TS_i)$: Once the *Aggr* receives the Msg_i from the *CN*, *Aggr* first verifies the timeliness of received timestamp TS_i, and if it is valid, the *Aggr* can proceed to decrypt $LEData_i$ using its corresponding secret key sk_a. The decryption algorithm then produces the outputs as $(pk_a, data_i, TS_i)$.

Single Signature Generation/Verification Phase

In this phase, the lattice-based signatures are generated by each signer on the respective encrypted data $LEData_i$, and the signature is then verified by the *Aggr*.

- $LSign(sk_i, LEData_i) \rightarrow \sigma_i$: Each signer *WD* runs this algorithm with its own secret key sk_i to produce a lattice-based single signature σ_i on $LEData_i$. Here, each signature for the respective *WD* is at most $m - 1$ degree polynomial with integer coefficients that lie in the interval$[-(k - 32), (k - 32)]$, where k is a given positive integer. It is worth noting that the parameter k is responsible for controlling a better trade-off between the security and the run-time in a single signature generator/verification phase. Thus, we keep the value of k to be fixed. Once this process is completed, each signer *WD* sends the message $Msg_i = \{LEData_i, LSign(sk_d, LEData_i), TS_i\}$ to the *CN* via public channel.
- $LVer(pk_i, \sigma_i) \rightarrow 1$ or $(\perp)$: After receiving the message Msg_i from the *CN*, the *Aggr* can verify the messages for their legitimacy by running this algorithm, which takes public key (pk_i) of *WD* and the respective signature σ_i as inputs, and produces 1 for a successful verification of signature; otherwise, an invalid entity *perp* is generated to indicate that it is an invalid signature.

Aggregate Signature Generation/Verfication Phase

This phase allows the aggregate signature generation and verification processes that are discussed below.

- $LASign((pk_i, LEData_i, \sigma_i) \mid i = 1, 2, \cdots, n) \rightarrow \sigma$: The *Aggr* will perform this lattice-based aggregate signature based on all the received signatures σ_i, public keys pk_i and encrypted data $LEData_i$, where $i = 1, 2, \cdots, n$. This algorithm produces a compact short signature σ and the *Aggr* will construct a message containing this aggregate signature, a list of encrypted data, say *LEData*, where $LEData = \langle LEData_1, LEData_2, \cdots, LEData_n \rangle$, and the current timestamp TS_a. Finally, *Aggr* will send the message $\langle$LEData, *LASign* $(=\sigma)$, $TS_a\rangle$ to the associated cloud server, CS.
- $LAVer((pk_i, LEData, \sigma) \mid i = 1, 2, \cdots, n) \rightarrow 1$ or $\perp$: After receiving $\langle LEData, LASign\ (=\sigma), TS_a\rangle$, CS can verify it by performing this lattice-based aggregate signature verification algorithm. If the verification is valid, which means that the signature verification algorithm produces the value 1 (for invalid signature, it is $\perp$), the CS will consider the encrypted data $LEData_i$ as the transactions Tx_i that will contribute to a transactions pool.

Block Generation, Verification and Addition Phase

Blockchain formation and addition will be executed by the cloud server(s)in the P2P cloud server network. A transactions pool is created by the cloud servers with the received messages which contain the encrypted data and their aggregate signatures from the aggregator nodes *Aggr*. These messages are considered as transactions and stores into the transactions pool. Once the pool reaches to a certain threshold value (say, t), a leader is elected in the round-robin fashion from the cloud server network or using some leader selection algorithm, and constructs a new block with these transactions.

Next, a consensus algorithm will be executed for the block verification and addition into blockchain. At first, an elected leader has the responsibility to add a new block into the blockchain by performing a consensus algorithm (here, we consider a voting-based "Practical Byzantine Fault Tolerance (PBFT)" consensus algorithm [11]). The constructed new block contains the following information:

- Ablock version
- Hash of the previous block
- Timestamp (block generation time)
- Merkle tree root
- A list of lattice-based encrypted transactions
- Lattice-based aggregate signature
- Hash of the current block. A complete block structure is shown in Fig. 2.

Block Header	
Block Version	BVer
Previous Block Hash	PBH
Merkle Tree Root	MTR
Timestamp	TS
Creator of Block	Identity of CS_i
List of Encrypted Transactions	$\{LEnc(Tx_i) \mid i = 1, 2, ..., t\}$
Aggregate Signature	LASign
Current Block Hash	CBHash

FIGURE 2. Structure of a block formed by distributed cloud servers.

Once a block is created by the leader in the P2P CS network, the created block will be broadcasted to the other peer nodes in the network for block verification. The verification and addition processes are executed by the voting-based PBFT algorithm. Once the block is verified by the other peer nodes, they send a verification status to the leader. When the number of the valid replied messages reaches to a pre-defined threshold value for block addition, which is $2n_f + 1$ (n_f being the number of faulty nodes in the P2P network), a decision will be taken to add the mined block into the blockchain.

The overall process of the proposed blockchain-based framework for ambient intelligence in IoT using lattice-based aggregate signature approach is explained in Fig. 3.

Security Analysis

In this section, we show the robustness of the proposed scheme against the following attacks under the discussed threat model, which uses both the DY and CK-adversary models.

Replay Attack

A replay attack occurs when an adversary tries to fraud another legal entity in the network by means of reusing the exchanged information during transmission. In the proposed LAS-AIBIoT, the messages exchanged between the *WD*s and the respective *CN*, and also between the *CN* and its associated *Aggr* contain the fresh timestamps. On the receiving sides, the verification of the timestamps will determine whether the messages are fresh or not. In this way, LAS-AIBIoT has ability to defend the replay attack.

Impersonation Attacks

An impersonation attack permits an adversary to falsify a illegal message in order to defraud receiving entities in a network, where the adversary acts on behalf of a sending party. In the exchanges messages between the *WD*s and the respective *CN*,

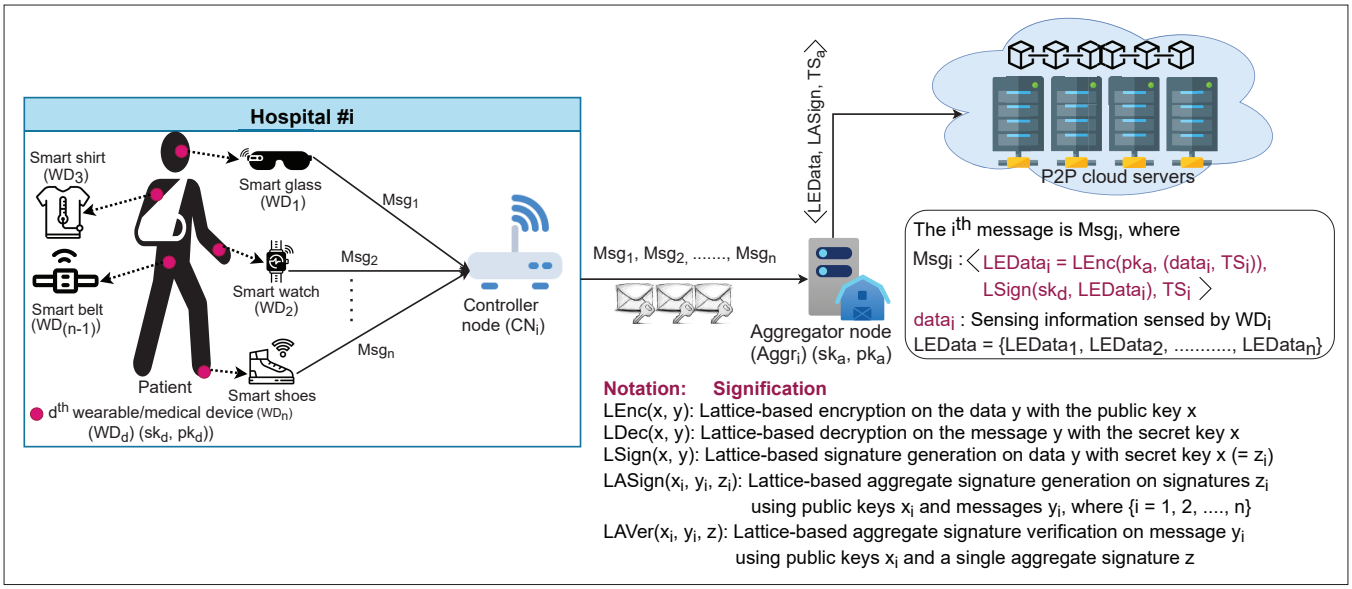

FIGURE 3. Overall architecture of the proposed blockchain-based framework.

and also between the *CN* and its associated *Aggr*, the presence of lattice-based single and aggregate signatures resists the adversary to create valid messages on behalf of an entity (e.g., *WD*, *CN* or *Aggr*), because the signatures are created by the secret (private) keys of the singers. Thus, the use of signatures makes the proposed LAS-AIBIoT to resist against *WD*/*CN*/*Aggr* impersonation attacks.

Man-in-the-Middle (MiTM) Attack

MiTM attack is possible when an adversary attempts to modify/delete/update the messages' contents using the intercepted messages so that the modified messages will be treated as valid ones to the recipients in the network. However, the utilization of the lattice-based signatures and encrypted data make an adversary not to modify the messages on the fly even the intercepted messages are available to him/her. As a result, the adversary will not be successful to launch MiTM attack in the proposed LAS-AIBIoT.

Privileged-Insider Attack

This attack permits a trusted user within an organization (in this case, a hospital), known as a privileged-insider attacker, can misuse the secret credentials in order to mount other attacks, like MiTM and impersonation attacks. However, in the proposed LAS-AIBIoT, each *TA* being the trusted edge server of a hospital is only responsible for generating private-public keys pair and sensing to the information secretly to the concerned entities in the network. Hence, the chance of a privileged-insider attack is eliminated in LAS-AIBIoT.

Physical WD Capture Attack

According to the threat model, the IoT-based wearable devices are not physically protected as they can not monitored in 24 × 7. Thus, if an *WD* is physically compromised, all the credentials pre-loaded in its memory are also compromised using the power analysis attacks. However, all the generated private-public keys pairs for the wearable devices are distinct and unique. This means that the secure communication among the non-compromised devices is not affected by the compromised of wearable devices in the proposed LAS-AIBIoT. In other words, LAS-AIBIoT is resilient against physical WD capture attack.

Other Quantum Attacks

The hybrid lattice-reduction refers to an attack where an adversary will break the shortest vector problem (SVP). The SVP can be explained as follows: given a basis of vectors of a lattice where the vectors are treated as the fixed-length tuples of integers, to determine a non-zero vector whose length will be the length of the shortest vector. When the hybrid lattice-reduction and meet-in-the middle attack are combined together, they form a hybrid attack. The hybrid attack is an important attack to be considered to evaluate the security of several lattice-based cryptographic schemes, such as NTRU (NTRUEncrypt and NTRUSign). A generic attack refers to an attack where a secret key is attempted to be recovered by an adversary based on generation of the decryption errors. It is a kind of chosen-ciphertext attack (CCA). This is useful for the schemes that are proposed for the CCA security. In case of quantum MiTM attack, an adversary blocks all the calibration signals and then transmits the faked calibration signals in order to disturb the activation timing calibration of the detectors. In the proposed LAS-AIBIoT, we have used the lattice-based post-quantum keys along with the blockchain technology. This helps LAS-AIBIoT to achieve security against various attacks from both classical as well as quantum computers including the "hybrid lattice-reduction," "generic" and quantum MiTM attacks.

Blockchain Implementation

In this section, we execute the blockchain over a decentralized P2P distributed system, where the number of distributed servers is taken as 7. Here, the assumed peer nodes (also called servers) are considered with the configuration setting: "Ubuntu 18.04.3 LTS, Intel® Core™ i5-8400 CPU @ 2.80GHz × 6, Memory 7.6 GiB, OS type 64-bit, disk 152.6 GB" and the programming language was written in node.js with VS CODE 2019. The blocks addition into the blockchain requires a distributed consensus algorithm, and for this case, we adopted widely-recognized voting based PBFT consensus algorithm [11]. The simulation is performed under two cases:

1. Case 1: In this case, we fixed the number of transactions for each block into the blockchain
2. Case 2: Under this case, we varied the number of transactions with a fixed blockchain size as 41. The details of these cases are discussed below.

- **Case 1:** The number of transactions in the varied number of blocks into the blockchain is taken as 33. The simulation results provided in Fig. 4a demonstrates that the computational time (in seconds) increases linearly (as the graph of the results is linear type which is represented as a state line) with an increased number of blocks.
- **Case 2:** Fig. 4b provides the simulation results, where the size of the blockchain is fixed at 41 and the number of transactions is varied. The computational time is also represented in Fig. 4b,

which signifies that the computational time increases linearly for constructing the blockchain when a varied number of transactions are present in the blocks.

Related Work

Jing [12] proposed a lattice-based homomorphic aggregate signature (HAS) scheme for multi users over a binary field, which is based on the "linearly homomorphic signature (LHS)" for a single user. In their scheme, there is a trusted authority, called the private key generator (PKG), which has the responsibility to provide private and public key pairs to all users by executing the *setup* algorithm. Next, a user runs the *sign* algorithm for signing a message, whereas the verifier runs the *verify* algorithm for verifying the signed message. The security of their proposed HAS scheme is based on the hardness of the computational SIS problem.

Zhang *et al.* [6] designed a post-quantum lattice-based blind signature scheme and the security of their scheme is based on the average-case of the SIS problem. Their scheme is based on the rejection sampling theory. The expected number of times required for producing the result of the blind signature in their scheme is at most e^2, where e is a random challenge drawn by the verifier.

Ma and Jiang [13] proposed another lattice-based aggregate signature protocol using the blockchain technology, where their scheme produces smaller signatures sizes than the existing schemes. Their scheme also supports the public key aggregation and the security of their scheme lies on the hardness of the "ring-based short integer solution (ring-SIS) assumption."

Jiao and Xiang [14] designed a lattice-based ring signature protocol for a vehicular ad-hoc network (VANET). In their scheme, the vehicles make a ring with neighbor vehicles with the help of the road-side units. Their scheme provides the "unconditional anonymity" for the ring members that are considered as vehicles, and also guarantees the signature unforgeability. Their scheme supports anti-quantum security and the security of their scheme relies on the hardness of the SIS problem. Moreover, they utilized the "bimodal Gaussian distribution" for constructing a large block in their ring signature scheme.

Cai *et al.* [15] also proposed a blind aggregate signature scheme based on the lattice for the multi-party transactions in an industrial blockchain, which has ability to provide the anti-quantum security. Their scheme uses the quantum key distribution which generates the quantum keys for the trading parties in the industrial blockchain network. Moreover, their scheme resists against "quantum intercept-resend (QIR)" and "quantum man-in-the-middle (QMITM)" attacks.

In Table 1, we compare the performance of the proposed security framework (LAS-AIBIoT) with the relevant lattice-based security schemes that use blockchain or non-blockchain framework, such as the schemes of Zhang *et al.* [6], Ma and Jiang [13], Jiao and Xiang [14], and Cai *et al.* [15]. It is observed that the schemes [6, 14] are centralized ones because they are not based on the blockchain technology, whereas the other schemes [13, 15] and LAS-AIBIoT are decentralized in nature. Moreover, in terms of efficiency, the scheme [15] is better than other schemes.

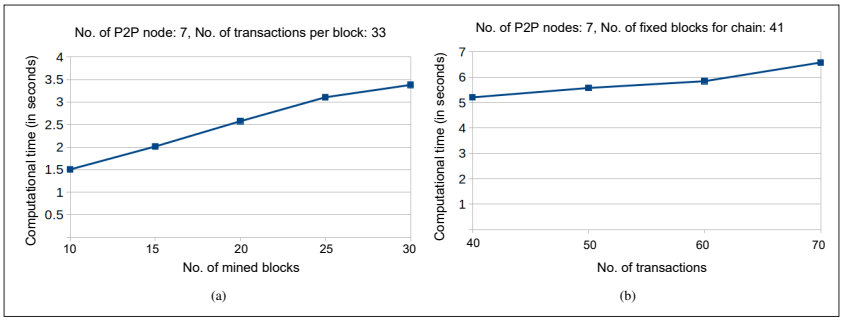

FIGURE 4. Blockchain simulation results: a) case 1 b) case 2.

The proposed security framework (LAS-AIBIoT) is based on the lattice-based aggregate signature which also applies the blockchain technology to store the private healthcare data of the patients in a private blockchain in encrypted forms in the blocks of the blockchain. In addition, the aggregate signature is created by an aggregator node *Aggr* of a hospital for a set of transactions in a block. As a result, the proposed LAS-AIBIoT not only resists traditional attacks like replay, impersonation, MiTM, privileged-insider and physical *WD* capture attacks, but also other quantum attacks like "hybrid lattice-reduction," "generic" and quantum MiTM attacks. Moreover, the proposed LAS-AIBIoT is efficient in computation due to lattice-based vector and matrix operations. Thus, LAS-AIBIoT is suitable for resource-constrained wearable devices too. In addition, in terms of privacy protection, the proposed LAS-AIBIoT is better than the other compared schemes.

Conclusion

In this article, we designed a lattice-based security framework using aggregate signature concept in an ambient intelligence-assisted private blockchain-based IoT environment. In the proposed LAS-AIBIoT, the single signatures are generated by the wearable devices where the aggregator nodes can verify those signatures. On the other side, the aggregate signature, which is created by an aggregator on the received encrypted transactions and single signatures, can be further verified by a cloud server before building a block that needs to be mined into the blockchain. Through the blockchain simulation study and security analysis on the proposed threat model, it was shown that LAS-AIBIoT is practical, efficient and robust against various traditional and quantum attacks.

Acknowledgments

The authors thank the anonymous reviewers and the associate editor for their valuable feedback on the article, which helped us to improve its quality and presentation.

References

[1] J. Cubo, A. Nieto, and E. Pimentel, "A Cloud-Based Internet of Things Platform for Ambient Assisted Living," *Sensors*, vol. 14, no. 8, 2014, pp. 14,070–14,105.
[2] D. He and S. Zeadally, "Authentication Protocol for an Ambient Assisted Living System," *IEEE Commun. Mag.*, vol. 53, no. 1, 2015, pp. 71–77.
[3] V. Chamola *et al.*, "Information Security in the Post Quantum Era for 5G and Beyond Networks: Threats to Existing Cryptography, and Post-Quantum Cryptography," *Computer Commun.* vol. 176, 2021, pp. 99–118.
[4] R. Chaudhary *et al.*, "LSCSH: Lattice-Based Secure Cryptosystem for Smart Healthcare in Smart Cities Environment," *IEEE Commun. Mag.*, vol. 56, no. 4, 2018, pp. 24–32.

Scheme	Architecture	Security	Efficiency	Privacy Protection
Zhang *et al.* [6]	Centralized	Moderate	Moderate	Low
Ma and Jiang [13]	Decentralized	High	Moderate	Low
Jiao and Xiang [14]	Centralized	Moderate	High	Moderate
Cai *et al.* [15]	Decentralized	High	Low	Moderate
Proposed (LAS-AIBIoT)	Decentralized	High	Moderate	High

TABLE 1. Performance comparison with related schemes.

[5] V. Nehra, U. Rungta, and B. Choudhury, "Blockchain Enabled Ambient Assisted living for Critically ill Patients," *11th Int'l. Conf. Cloud Computing, Data Science Engineering (Confluence)*, Noida, India, 2021, pp. 575–81.
[6] P. Zhang *et al.*, "A New Post-Quantum Blind Signature From Lattice Assumptions," *IEEE Access*, vol. 6, 2018, pp. 27,251–58.
[7] D. Dolev and A. Yao, "On the security of public key protocols," IEEE Transactions on Information Theory, vol. 29, no. 2, pp. 198–208, 1983.
[8] R. Canetti and H. Krawczyk, "Universally Composable Notions of Key Exchange and Secure Channels," *Int'l. Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT'02)*, Amsterdam, The Netherlands, 2002, pp. 337–51.
[9] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, "Examining Smart-Card Security Under the Threat of Power Analysis Attacks," *IEEE Trans. Computers*, vol. 51, no. 5, 2002, pp. 541–52.
[10] A. K. Das, "A Random Key Establishment Scheme for Multi-Phase Deployment in Large-Scale Distributed Sensor Networks," *Int'l. J. Information Security*, vol. 11, no. 3, 2012, pp. 189–211.
[11] M. Castro and B. Liskov, "Practical Byzantine Fault Tolerance," *3rd Symp. Operating Systems Design and Implementation (OSDI'99)*, New Orleans, USA, 1999.
[12] Z. Jing, "An Efficient Homomorphic Aggregate Signature Scheme Based on Lattice," *Mathematical Problems in Engineering*, vol. 2014, 2014, p. 536527.
[13] C. Ma and M. Jiang, "Practical Lattice-Based Multisignature Schemes for Blockchains," *IEEE Access*, vol. 7, 2019, pp. 179,765–78.
[14] C. Jiao and X. Xiang, "Anti-Quantum Lattice-Based Ring Signature Scheme and Applications in VANETs," *Entropy*, vol. 23, no. 10, p. 1364, 2021.
[15] Z. Cai *et al.*, "A Quantum Blind Multi-Signature Method for the Industrial Blockchain," *Entropy*, vol. 23, no. 11, pp. 1–17, 2021.

Biographies

Prithwi Bagchi (prithwi.bagchi@research.iiit.ac.in) received his M.Sc. degree in mathematics from Presidency University, West Bengal, India. He is currently pursuing his Ph.D. degree in computer science and engineering from IIIT Hyderabad, India. His research interests are cryptography, network security and blockchain technology.

Basudeb Bera (basudeb.bera@research.iiit.ac.in) received his M.Sc. degree in mathematics and computing in 2014 from IIT (ISM) Dhanbad, India, and M.Tech. degree in computer science and data processing in 2017 from IIT Kharagpur, India. He is currently pursuing his Ph.D. degree in computer science and engineering from IIIT Hyderabad, India. His research interests are cryptography, network security and blockchain technology. He has published over 30 papers in international journals and conferences in his research areas.

Ashok Kumar Das [SM] (iitkgp.akdas@gmail.com) received a Ph.D. degree in computer science and engineering, an M.Tech. degree in computer science and data processing, and an M.Sc. degree in mathematics from IIT Kharagpur, India. He is currently an Associate Professor with the Center for Security, Theory and Algorithmic Research, IIIT, Hyderabad, India. He was also a visiting faculty with the Virginia Modeling, Analysis and Simulation Center, Old Dominion University, Suffolk, VA 23435, USA. His current research interests include cryptography, system and network security, blockchain and AI/ML security. He has authored over 335 papers in international journals and conferences in the above areas, including over 290 reputed journal papers. He is on the editorial board of IEEE Systems Journal, Journal of Network and Computer Applications (Elsevier), Computer Communications (Elsevier), Journal of Cloud Computing (Springer), Cyber Security and Applications (Elsevier), IET Communications and KSII Transactions on Internet and Information Systems.

Sachin Shetty [SM] (sshetty@odu.edu) received the Ph.D. degree in modeling and simulation from Old Dominion University in 2007. He was an Associate Professor with the Electrical and Computer Engineering Department, Tennessee State University, USA. He is currently a Professor with the Virginia Modeling, Analysis and Simulation Center, Old Dominion University. He holds a joint appointment with the Department of Modeling, Simulation and Visualization Engineering and the Center for Cybersecurity Education and Research. He has authored and co-authored over 175 research articles in journals and conference proceedings and two books. His research interests lie at the intersection of computer networking, network security, and machine learning. He was a recipient of the DHS Scientific Leadership Award. He has served on the Technical Program Committee of ACM CCS, IEEE INFOCOM, IEEE ICDCN, and IEEE ICCCN.

Pandi Vijaykumar (vijibond2000@gmail.com) received the B.E. degree in computer science and engineering from Madurai Kamaraj University, Madurai, India, in 2002, the M.E. degree in computer science and engineering from the Karunya Institute of Technology, Coimbatore, India, in 2005, and the Ph.D. degree in computer science and engineering from Anna University, Chennai, India, in 2013. His current research interests include key management in network security, VANET security, and multicasting in computer networks. He has published various quality papers in reputed journals like IEEE Transactions/Journals, ACM transaction, Elsevier, Springer, etc.

Marimuthu Karuppiah (marimuthume@gmail.com) received the B.E. degree in computer science and engineering from Madurai Kamaraj University, Madurai, India, in 2003, the M.E. degree in computer science and engineering from Anna University, Chennai, India, in 2005, and the Ph.D. degree in computer science and engineering from VIT University, Vellore, India, in 2015. He is currently a Professor with the Department of Computing Science and Engineering, Presidency University, Bengaluru, India. His current research interests include cryptography and wireless network security, in particular, authentication and encryption schemes.