IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 14 May 2024, accepted 10 July 2024, date of publication 16 July 2024, date of current version 26 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3429137

RESEARCH ARTICLE

LA-IMDCN: A Lightweight Authentication Scheme With Smart Contract in Implantable Medical Device Communication Networks

JAYAPRAKASH KAR[1], (Senior Member, IEEE), XIAOGUANG LIU[2,3,4], AND FAGEN LI[4], (Member, IEEE)

[1]Centre for Cryptology, Cyber Security and Digital Forensics, Department of Computer Science and Engineering, The LNM Institute of Information Technology, Jaipur, Rajasthan 302031, India
[2]Guangxi Key Laboratory of Cryptography and Information Security, Guilin University of Electronic Technology, Guilin, Guangxi 541004, China
[3]School of Mathematics, Southwest Minzu University, Chengdu 610041, China
[4]School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 610056, China

Corresponding author: Jayaprakash Kar (jayaprakashkar@lnmiit.ac.in)

This work is supported in part by Science and Engineering Research Board, Department of Science and Technology, Govt of India under the scheme of State University Research Excellence with Project ID-SUR/2022/004771, Data Security Council of India, NASSCOM and in part by the Foreign Experts Program of Ministry of Science and Technology of China under Grant DL2022186001L.

ABSTRACT Implantable medical devices (IMDs) in medical sciences have provided a quantum leap in network transformation. The communication network with IMDs typically has a wireless radio frequency (RF) telemetry or wired connection. IMDs, being devices, have more computing, communication capabilities and decision-making. Furthermore, these devices are being used to improve patients' quality of life by medicating various chronic diseases. The captured data is stored in a medical server through a controller node. Our work focuses on wireless communication, so sensitive patient data over a public channel might be tampered with or eavesdropped by unauthorised access. Furthermore, the leakage of health data and malfunctioning of IMDs are vital in constructing cryptographic protocols, particularly in the design of remote user authentication. In this paper, we proposed a novel secure remote user authentication scheme using a lightweight consortium blockchain for the communication network with IMDs.

INDEX TERMS Authentication, blockchain, key-establishment, controller node, implementable medical device.

I. INTRODUCTION

In the past two years, blockchain technology has attracted much interest. It can be understood that the blockchain functions as a circular database of users. The users of this database can process data about specific nodes connected to the network. In the conventional data-sharing approach, users maintain data through centralized permissions. This process is decentralised by blockchain, which allows users to trade with one another without the involvement of a third party. This is the main advantage of the blockchain process. For example, have User C represent a so-called third party, such as a government or health regulator. 'Traditionally, when User A and User B execute a transaction, User C intervenes to validate both users' identities. However, the blockchain configuration does not require user C to be involved. The blockchain environment has paved the way for new possibilities for transactions. Blockchain technology allows users to digitise, encode, and insert virtually any information transaction in an immutable, distributed, and secure manner.

IMDs are electronic devices implanted in the human body for diagnostic, monitoring, and therapeutic purposes. IMD continues to grow in popularity, with more than 25 million US citizens now relying on IMD for health-related functions.IMD is still in its very early phases and still has a lot of regulatory obstacles to overcome. IMD products that support wireless charging are still in the lengthy trial stage and will not be available to consumers. Therefore, one of the major concerns for IMD design is to reduce power

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

consumption. Typically, IMD batteries should last 5-10 years. This significantly limits the complexity of the safety mechanism. For example, complex cryptographic calculations and long-distance radio transmission are considered prohibited.

A medical device postmarket surveillance (PMS) system now requires an infinitely greater data value to be analysed. This is due to Health Authorities' increasingly complex regulatory and demanding requirements to better understand the evaluation of the safety of medical devices. One of the primary goals of the new regulations is to ensure the timely, dependable, and efficient exchange of PMS data in order to identify medical device safety issues and take appropriate action. As regulatory agencies increase device security reviews, there is an increasing need for a proactive approach to the PMS process. This has motivated several individuals in the safety assessment of medical devices to look at potential remedies to problems brought on by the evolving regulatory environment [6]. Furthermore, they recognise the importance of addressing some of the process-related bottlenecks. As in other areas of the medical device industry, stakeholders are beginning to work on solutions based on technologies like artificial intelligence (AI) that may help modify the reactive system of PMS in use for medical devices. Machine learning, automation of robotic processes, the Internet of things, and blockchain are some of the explored solutions in medical devices thus far.

The ability of IMDs to communicate and network wirelessly is a key source of security issues. Adjacent eavesdroppers can catch all transmitted packets due to the openness of the wireless channels. This can disclose patient privacy, such as the IMD's existence and its model, and other common wireless attacks, such as message forging, tampering and responding. Furthermore, suppose the IMD allows remote access to the hospital or doctor. In that case, cyber assaults on the hospital's network/server might lead to patient data or credentials theft. It is, therefore, extremely desirable to develop a straightforward but effective access control method for IMDs.

Any cryptographic system faces challenges in terms of effective and secure key management [4]. If an attacker discovers the keys using any method, such as brute force, side-channel attacks, physical access to the system, weak encryption, replay attacks, and so on, the intruder has complete control over the targeted system [2]. Therefore, key management is one of the most crucial components of the cryptographic system. The security of the infrastructure depends on the security of the keys that control it. Blockchain infrastructure uses PKI to validate IoT devices; in this scenario, the security of IMDs and the infrastructure depends on the reliability of third parties.

II. RELATED WORKS

In recent years, many researchers have been working on realistic authentication solutions for IoT devices like IMDs. The primitives of existing IoT authentication systems can be used to classify them. To solve performance and security challenges, including key management and storage costs in the smart grid context, Wang et al. [13] developed an anonymous recording and gateway-based authentication system. The signals provided by smart meters are authenticated and aggregated using homomorphic encryption and HMAC, which significantly reduces the amount of data transferred in this approach. Edge computing IoT network security vulnerabilities have been well investigated in [14], highlighting the integration's security and privacy threats. As already mentioned, edge computing-based IoT applications require efficient authentication schemes. Halperin et al. [3] presented the vulnerabilities of commercial implantable cardioverter-defibrillators (ICD). Equipped with oscilloscopes and software radios, they could reverse engineer the ICD's communication protocol and obtain personal information about the patient and the ICD. Heterogeneous cryptosystems (symmetric and non-symmetric) are used to provide various levels of security. Their protocol consists of two stages: global authentication and local authentication. It did not provide anonymity property. Because of its decentralized, stable, secure, and immutable nature, Mingxiao et al. [8] surveyed the usefulness of blockchain. It was discovered that the consensus algorithm was crucial to the success of the project blockchain. They went through a unique consensus algorithm's fundamentals, characteristics, performance analysis, and application scenarios. Zhang et al. [15] Studied smart grid concerns such as secured communication, dependable mutual authentication and privacy credentials, key management and centred on key management among smart metres. They introduced a consortium blockchain-based decentralized keyless signature technique that is computationally cheap, time efficient, scalable, reliable, and efficient. Pal et al. [10] discussed the importance of key management and the need for Blockchain technology to eliminate the need for third-party providers to validate transactions over the network. This article covers an overview of blockchain, examining existing Blockchain PKI and key management for Blockchain wallets.

A. NETWORK ARCHITECTURE AND COMPONENTS

The IMD communication network model consists of three components, namely Controller Node (CN), users (e.g. Doctors, Nurses, Healthcare providers, etc.) (U) and blockchain (BC). The communication network is depicted in figure 1.

- **CN**: It works as a registration authority or trusted medical server, which plays the role of enrolling system setup and registration of all users and healthcare providers. The server all the data from all IMDs using wireless communications such as Bluetooth, ZigBee, etc. It's assumed that the servers are honest and curious and have high processing power and storage capacity. Additionally, it distributes the key materials and deploys blockchain to keep records of the users and control the node's key materials through smart

TABLE 1. Notations.

Notation	Description
ID_i	ID of ith user
s	Master secret key
sk_i	Private key of ith user
ψ	Hash of Id, s,sk_i
pk_i	Public Key of the ith user
Q	Hash of ID_i
T_i	Timestamp of transmitting the parameters

contracts for identification verification, key updates, and revocation.

- **U**: User U may be a doctor or healthcare provider who, after successful authentication, wants to access information kept in the CN.
- **BC**: The role of BC is that all public key materials should be recorded in the smart contract. Our proposed scheme stores the record of key issuing, updating of key materials and revocation in trusted ways.

B. DEPLOYMENT OF SMART CONTRACT

The public key's information is managed by deploying smart contracts. The system functions used in the smart contract system initialization are the authentication of users, creation of a new user in the ledger and revocation of access. It also manages access rights administration by introducing a simplified device removal process through smart contracts, reducing administrative overhead and enhancing system usability.

C. TECHNICAL ADVANTAGE

The main advantages of deploying our smart contract in the proposed scheme LA-IMDCS are as follows

1) Granular Access Control: By matching the public key of the requester with the initially assigned public key stored on the blockchain, the smart contract ensures granular access control. Only users with the correct public key can update or modify IMD.
2) Limited Time Access: The contract introduces a time-bound access mechanism wherein users are granted access for a limited time period. This temporal restriction adds an additional layer of security by reducing the window of vulnerability for unauthorized access.
3) Efficient Device Removal: Removing a device or user from the blockchain node is streamlined through the use of the delete function. If the user's public key matches the initially assigned public key, the deletion process can be completed swiftly, ensuring efficient access rights management.
4) Improved Security: Public key authentication provides heightened security in contrast to conventional access control methods. Using cryptographic methods, the system effectively minimizes the potential for unauthorized access, manipulation, or data integrity breaches.

III. PROPOSED PROTOCOL

The protocol comprises four phases: (i) System initialization, (ii) Registration, (iii) Authentication and (iv) Update and revocation.

A. SYSTEM INITIALIZATION

This phase is carried out by the CN gateway, which is assumed to be a trusted third-party authority (TA). Additionally, It is expected to carry out offline tasks. This includes (1) assigning a unique identity to all users, (2) security parameters, and tracing log records, among other things. The set-up process is performed by the CN-gateway by choosing an elliptic curve E defined over a prime field F_p, which is given by $E(F_p)$. In addition, it chooses a point $P \in E(F_p)$ with order n. The CN gateway performs the following computations.

1) Sets additive cyclic groups G_1 with generator P and a multiplicative group G_2. Both the groups are of the same prime order q.
2) Chooses two cryptographic hash functions $H_1 : \{0, 1\}^* \rightarrow \mathbb{Z}_q^*, H_2 : \{0, 1\}^* \times G_1 \rightarrow \mathbb{Z}_q^*$.
3) Picks a number s at random, which is considered as the master secret key, and computes master public key $P_{pub} = sP$.

Finally, CN-gateway publishes
$param = \{G_1, G_2, \hat{e}, q, P_{pub}, H, H_1, H_2\}$ and keeps secret the master secret key s.

B. REGISTRATION

It invokes the algorithm 1 The registration phase is the CN and the user U in an interactive manner. Assume that the communication channel is secure and private.

1) CN generates a unique identities $\{ID_i : i = 1, 2 \ldots n$ for each user U_i participating in the protocol and computes $Q_i = H(ID_i)$. This phase is performed for the registration of all the users including patients, doctors and healthcare providers. The process is carried out by the CN. It uses ID_i and performs the following steps
   a) $Q_i = H(ID_i)$, user's public key $pk_i = (Q_i + s)P$ and private key $sk_i = \frac{1}{Q_i+s} \cdot P$. These key pairs (pk_i, sk_i) is computed for i^{th} user.
   b) $\psi_i = H_1(ID_i \| s \| sk_i)$.

   The function `struct IMD` is created for managing the public key and invokes the initialization i.e. algorithm 1. Upload the parameters pk_i, ψ, Q_i and T_i to the smart contract.
2) Then invokes the algorithm 5 i.e. `updateIMDT` $(oldpk_i, pk_i, Q_i, \psi_i, T_1)$ checks whether the user has previously registered, if the user is already registered, then re-validates the key materials (pk_i, Q_i, ψ_i, T_1) stored in the `struct IMD` and updates the `struct IMD` if a new user is registered.
3) Prior to the authentication, the algorithm 2 i.e. `queryIMDT`(pk_i) is invoked by the CN to retrieve the parameters pk_i, ψ_i, Q_i and T_i; for the user U_i.

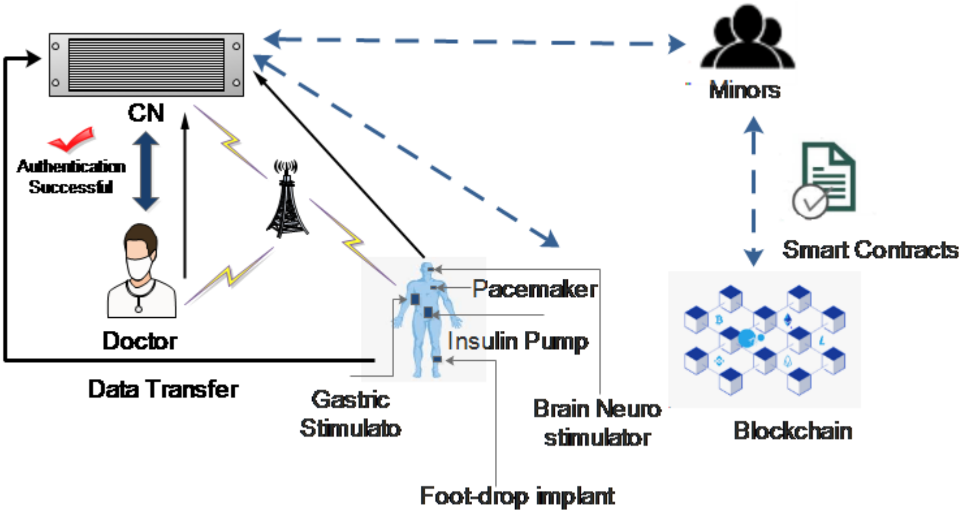


FIGURE 1. Network model.

```
Algorithm 1 LA_IMDCN System Initialization
  begin
      contract LA_IMDCN {
      address owner;
      struct IMD{
          byte32 $\psi$;
          uint256 $pk_i$;
          uint256 $Q$;
          DataTime $T_1$;
      }
       IMD[] public IMDT;
      constructor LA_IMDCN() {
          $owner = msg.sender$;
          $len = 0$;
          Return 1;
      }
  end
```

```
Algorithm 2 LA_IMDCN Query
  begin
      function queryIMDT ($pk_i$){
      if Exist ($IMD[i].pk_i == pk_i$) then
          Return IMD;
      end if
      else
          Return 0;
  end
```

C. AUTHENTICATION

The authentication process is initiated by any arbitrary user, say U_i. It performs the following actions:

1) It Chooses $\gamma_i \in \mathbb{Z}_q^*$ at arbitarily and calculates $V_i = \gamma_i P$.

2) Computes $W_i = sk_i \gamma_i$ and $\beta = H_2(ID_i \| V_i \| W_i \| T_1)$. Where T_1 is the current time stamp while transmitting the parameters.

3) Computes $E_{\psi_i}(ID_i, T_1) = C$.

4) Computes $\alpha_1 = MAC_\beta[ID_i, T_1, V_i, \psi_i]$.

and obtains the parameters $\{C, W_i, \alpha_1, T_1\}$. U_i sends the parameters $\{C, W_i, \alpha_1, T_1\}$ to CN by using the standard TLS or SSL protocol.

On receiving $\{W_i, \alpha_1, T_1\}$, CN checks the validity of T_1 and compares with the current time stamp T_2. If $T_2 - T_1 = \Delta T$

For authentication at CN, on receiving the parameters from user U_i; queryIMDT(pk_i) is invoked for validating the user's identity. If the function returns "0", then the user is invalid. Otherwise, returns struct IMD validating the parameters (pk_i, Q_i, ψ_i, T_1) and then executing the algorithm 3.

```
Algorithm 3
  begin
      $D_{\psi_i}(C) = (ID_i, T_1)$
      $Q_i = H(ID_i)$
      $sk_i = \frac{1}{s+Q_i} \cdot P$.
      $\psi_i^* = H_1(ID_i \| s \| sk_i)$.
      if $T_1^* = T_1, \psi_i^* = \psi_i$ then
          $V_i' = (s + Q_i') \cdot W_i$.
      end if
      $\beta' = H_2(ID_i \| V_i' \| W_i \| T_1)$
      $\alpha_1' = MAC_{\beta'}[ID_i, T_1^*, V_i', \psi_i]$
      if $\alpha_1' = \alpha_1$ then
           Chooses $x \in \mathbb{Z}_q^*$ at random, $R = xP$.
          $S = x\psi_i \cdot V_i$.
          $\alpha_2 = MAC_{\beta'}[V_i, T_2, R, S]$.
          $k = H_3(ID_i \| \psi_i \| V_i \| R \| S)$.
      end if
       Return : $\{R, S, \alpha_2, T_2\}$
  end
```

Finally CN sends $\{R, S, \alpha_2, T_2\}$ to the user U_i. On receiving it, the user U_i check the validity of T_2 by verifying $T_3 - T_2 \leq \Delta T$. Then, it executes the algorithm 4.

Algorithm 4

```
begin
    $S^* = \gamma_i \psi_i \cdot R$.
    $\alpha_2' = MAC_\beta[V_i, T_2^*, R, S^*]$
    if $\alpha_2' = \alpha_2$ then
        $k = H_3(ID_i \| \psi_i \| V_i \| R \| S)$.
    end if
end
```

D. REVOCATION AND UPDATE

- **Update**: Once the initialisation and registration is completed and the parameters are generated, then the updateIMDT($oldpk_i, pk_i, Q_i, \psi_i, T_1$) is called by the CN for access control before the authentication phase. The function checks the address of the contract's owner by equating its address with the user's address initiating the transaction; if the user's public key already exists in the blockchain, the given parameters are updated in the structure. If the public key is not present in the blockchain, a new tuple for the parameter is created in the IMD struct.
- **Revoke** - Revocation works in two cases: (1) If CN discovers suspicious behaviour. (2) If the user wants to leave the system, CN invokes the revokeIMDT(PK) which will revoke the user's access to the public key, deletes the entry (pk_i, Q, ψ, T_1) and CN will restrict the communication of user from authentication phase.

Algorithm 5 Update LA_IMDCN

```
begin
    function0 updateIMDT($oldpk_i, pk_i, Q, \psi, T_1$){
    if owner != msg.sender then Return 0;
    end if
    else
        if Exist($IMD[i].pk_i = oldpk_i$) then
            $IMD[i].pk_i = pk_i$;
            $IMD[i].Q = Q$;
            $IMD[i].\psi = \psi$;
            $IMD[i].T_1 = T_1$;
            Return 1;
        end if
        else
            $len + +$
            $IMD[len].pk_i = pk_i$;
            $IMD[len].Q = Q$;
            $IMD[len].\psi = \psi$;
            $IMD[len].T_1 = T_1$;
            Return 1;
end
```

Algorithm 6 Revoke LA_IMDCN

```
begin
    function revokeIMDT ($pk_i$)
    if owner != msg.sender then Return 0;
    end if
    else
        if Exist($IMD[i].pk_i == pk_i$) then
            $Release(IMD[i])$;
            for $i = 0$ to $len$ do
                $IMD[i] = IMD[i + 1]$;
            end for
            $Len - -$;
            Return 1;
        end if
        else
            Return 0;
end
```

IV. PERFORMANCE EVALUATION

The section that follows assesses the performance of the proposed scheme. The execution was carried out on the Ethereum test network. This computes the gas cost for the operations performed in the smart contract. Computational cost regarding computation time and communication overhead is evaluated in relation to the cryptographic operations performed at each step of the above-proposed scheme. In the following section, we also contrast the cost with such relevant protocols and schemes.

A. IMPLEMENTATION ON ETHEREUM

In order to simulate the Ethereum blockchain, Remix is more efficient because it uses open-source solidity. Additionally Remix also supports testing, debugging and deploying of smart contracts and much more. The implementation details are as follows:

1) We have set up two accounts that represent Controller node CN and the user U for the test. Solidity Compiler, version 0.8.7+commit.e28d00a7, is used for our simulation. The address of the deployed smart contract is 0× 6f0D54d283a7a2413Bb849d64D16Db16D5f81209. The hash of the transaction is given by 0xa4cf158799b4aee8f34e3cea5d1a7e31653fd5879559806fcb8412df25382194.
The figure: 2 demonstrates the use of smart contracts by Remix. This accomplishes every goal the algorithm set out to accomplish 1. Algorithms 5 and 2 include functions for updating, querying, and revoking. We can see from the results that the implementation of smart contracts has the highest cost, coming in at about USD 15.11065. In the system, the deployment process is only performed once. Update, inquiry, and revocation expenses are around USD 10.0469, 9.9983, and 8.8054, respectively, but other activities could be called frequently. In other words, one implantable medical

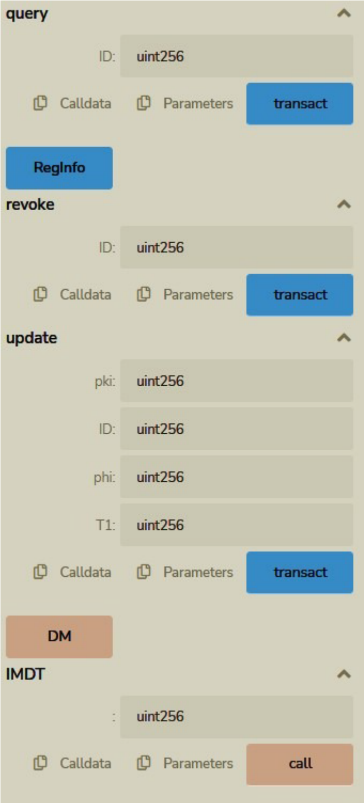
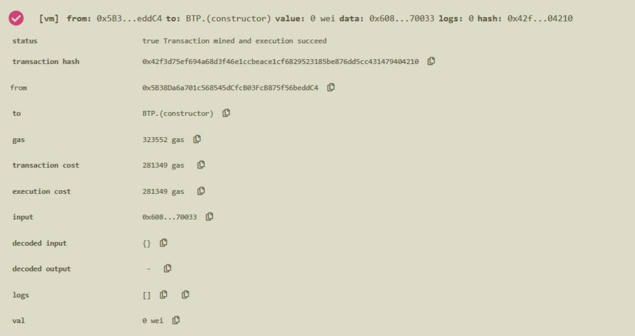


FIGURE 2. Deployment of smart contract.



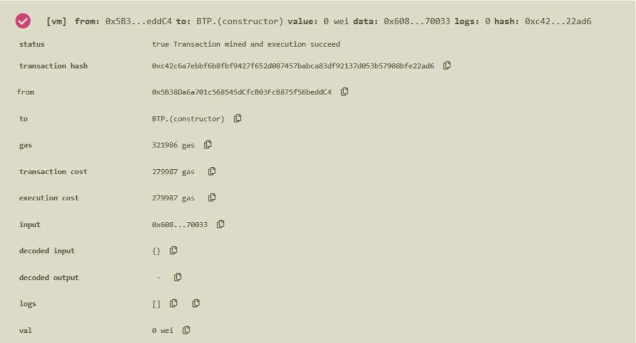
```
[vm]  from: 0x5B3...eddC4 to: BTP.(constructor) value: 0 wei data: 0x608...70033 logs: 0 hash: 0x42f...04210
status              true Transaction mined and execution succeed
transaction hash    0x42f3d75ef694a68d3f46e1ccbeace1cf6829523185be876dd5cc431479404210
from                0x5B38Da6a701c568545dCfcB03FcB875f56beddC4
to                  BTP.(constructor)
gas                 323552 gas
transaction cost    281349 gas
execution cost      281349 gas
input               0x608...70033
decoded input       {}
decoded output       -
logs                []
val                 0 wei
```

FIGURE 3. Update function.

device spends roughly USD 10.0469 to authenticate with the Controller Node.

2) The update transaction for each device is called only once for new users or if the information for any implantable medical device is updated during that time. The Query transaction is carried out every time the CN establish the connection with the device, which includes the accompanying transaction cost. The Revoke transaction is carried out once for each device when the access is revoked.

(1 ether = 1, 606.83 USD) The simulation of the Ethereum blockchain on the proposed network architecture 1 is illustrated in this section. We have set up a server for the controller node. Assume that it has high computational capability. The communication is established between the user, which might be through a web browser or any mobile device. The authentication procedure is started by the user and performs the operations briefed in section III-C. A single desktop computer with a processor of Intel Core i5 7600 clocked at 3.5 GHz and 16GB RAM serves as the server. It runs on the Ubuntu 16.04 LTS operating system. In our experiment, the mobile device is the user which interacts with the controller device CD. The configuration of the mobile device is Samsung Galaxy S5 with Quad-core 2.45G processor, 2G bytes memory and Google Android 4.4.2 operating system. We use the Pairing-based Cryptosystems Library 6 and libgmp via the gmpy2 Python Module 7 in the implementation. In order to evaluate the communication overhead and computational time, we choose the additive cyclic group G_1 with generator P of order as the prime q defined on the elliptic curve $E : y^2 = x^3 + x \bmod p$. The bilinear pairing is defined as $\hat{e} : G_1 \times G_1 \to G_2$, where G_1 is a cyclic additive group generated by the element P which lies on the underline elliptic curve E. The two primes p and q that we have chosen are of size 256 and 160 bits, respectively. The group element in G_1 and G_2 are 512 bits. We assume that the sizes of the private and public keys are 512 bits and 256 bits, respectively. A generic hash function is supposed to have a length of 160 bits, whereas an identity, timestamp, random number, and random number are each assumed to have lengths of 32 bits [5].



```
[vm]  from: 0x5B3...eddC4 to: BTP.(constructor) value: 0 wei data: 0x608...70033 logs: 0 hash: 0xc42...22ad6
status              true Transaction mined and execution succeed
transaction hash    0xc42c6a7ebbf6b8fbf9427f652d087457babca83df92137d053b57908bfe22ad6
from                0x5B38Da6a701c568545dCfcB03FcB875f56beddC4
to                  BTP.(constructor)
gas                 321986 gas
transaction cost    279987 gas
execution cost      279987 gas
input               0x608...70033
decoded input       {}
decoded output       -
logs                []
val                 0 wei
```

FIGURE 4. Query function.

TABLE 2. Gas cost of smart contract.

Operation	Gas used	Actual cost(ether)	USD
Deploy	486492	0.009368372	15.1065
Update	323552	0.006230636	10.0469
Query	321986	0.006200496	9.9983
Revoke	283572	0.005460713	8.8054

B. COMPUTATIONAL COST

The computation cost in the authentication process indicates the processing delays at various user ends and the controller node as a result of various cryptographic activities. The communication overhead is measured based on the duration

TABLE 3. Execution time.

Device/Server	T_{sm}	T_{bp}	T_h	T_{exp}	T_{en}
CN	3.823	4.728	0.004	3.772	0.01
Device/user	67.349	103.464	0.006	66.548	0.082

of the messages sent between the user and controller node. It should be noted that since we do not account for transmission and communication delays, it is possible to omit the execution time incurred during the process of calling the smart contract's query function and obtaining transaction data from the blockchain. We list the most time-consuming activities carried out in existing authentication schemes that are important and compare them to the suggested technique since the computational overhead caused by various cryptographic operations has an immediate influence on system performance. While comparing the cost, the evaluation is done on the server, and the device side of similar types of authentication schemes proposed by Wang et al. [12], Ni et al. [9], Kumari et al. [7] and Shen et al. [11].

We counted bilinear pairing, elliptic curve scalar multiplication, hashing/MAC, exponentiation, and AES-128 bit encryption and decryption among the cryptographic processes. consider the time of execution of these cryptographic operations performed in the proposed schemes are denoted by as T_{sm}, T_{bp}, T_h, T_{exp} and T_{en} respectively. The computational time for the operations, including integer addition and multiplication, has not been considered in our experimental analysis as these are low-cost operations. Table 3 shows how long each of the aforementioned processes took to complete.

We take into account the group size, the length of an identity, a hash,a random number, and a timestamp, which are indicated as $|ID|$, $|G|$, $|R|$, $|H|$, and $|T|$, respectively, in order to calculate the communication overhead.

C. EXPERIMENTAL SET-UP

In the experiment, the desktop and mobile devices perform each operation 1200 times to determine the final average execution time. We analyse the transmission costs of several pertinent authentication techniques using the experiment's findings, including Wang et al. [12], Ni et al. [9], Kumari et al. [7] and Shen et al. [11] and the results shows that the LA-IMDCN scheme has low-cost. The comparison is illustrated in 4.

V. SECURITY ANALYSIS

To demonstrate the robustness of the proposed blockchain-based lightweight authentication scheme LA-IMDCN, we adopted the security and adversary model [1] with provable security. The following subsection briefs the model where the adversary $\mathscr{A}$ communicates with the users $IMD - CN^k$, the k^{th} instance of the entity user U and the CN gateway. The user may be a doctor, healthcare provider, etc., and CN is the controller node that serves as a storage device or cloud storage summarized in the network model 1. The following subsection discusses the adversary and security model.

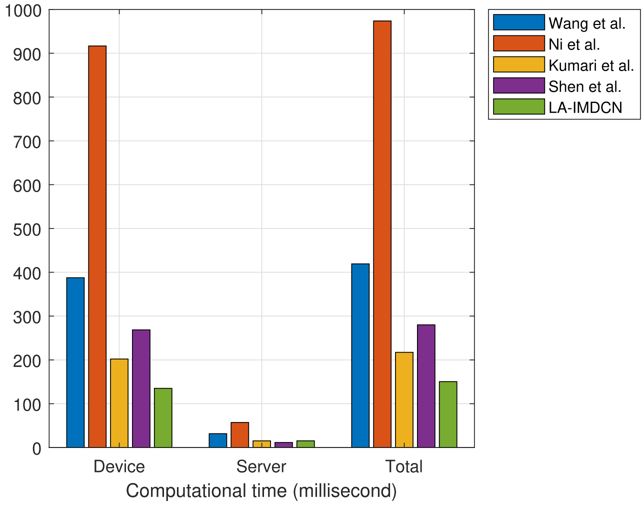


FIGURE 5. Computational time(ms).

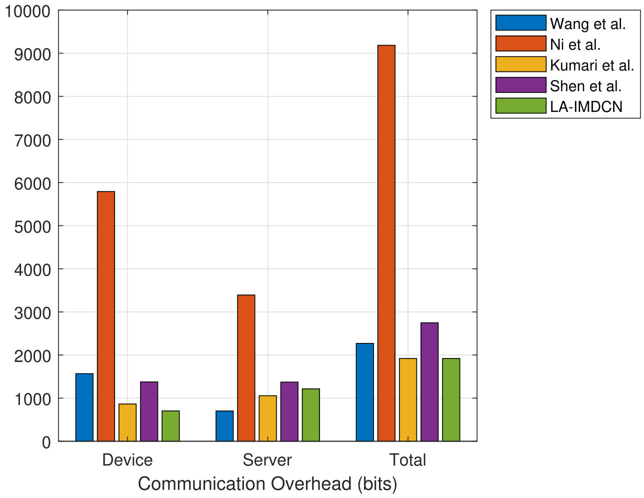


FIGURE 6. Communication overhead (bits).

A. ADVERSARY AND SECURITY MODEL

The model is simulated as a game between the challenger $\mathscr{C}$ and the adversary $\mathscr{A}$. $\mathscr{A}$ submits the queries in adaptive manner to $\mathscr{C}$. $\mathscr{C}$ answers these queries. These are performed by the following functions.

- `Setup`: $\mathscr{C}$ sends the system parameters for $\mathscr{A}$ as a response to the submitted queries.
- $\mathscr{C}$ chooses h_i randomly, where $H(x_i) = h_i$. Add $\{x_i, h_i\}$ in the list L_H and sends h_i to $\mathscr{A}$ as answer to this query.
- `Send`$(IMD - CN^k, s_k)$: $\mathscr{C}$ answers according to the rules of the proposed LA-IMDCN on receiving of s_k which is sent by $\mathscr{A}$.
- `Corrupt`(U): When the following query is submitted by using the identity ID_i of the user U_i, $\mathscr{C}$ responds with the private key sk_i of the user U_i to $\mathscr{A}$.

TABLE 4. Computation cost and communication overhead.

	Computation Cost (ms)		Communication cost (bits)		
	Device	**Server**	**Device**	**Server**	**No. of rounds**
Wang et al.'s Scheme [12]	$5T_{sm}+T_h+T_{en}$	$7T_{sm}+2T_h+T_{en}$	$2\|G\|+\|r\|+\|T\|$	$\|G\|+\|H\|+\|T\|$	**2**
Ni et al.'s Scheme [9]	$6T_{exp}+5T_{bp}$	$3T_{exp}+T_{bp}$	$8\|G\|+3\|r\|+\|H\|$	$4\|G\|+2\|r\|+2\|H\|$	**4**
Kumari et al.'s Scheme [7]	$3T_{sm}$	$3T_{sm}$	$\|G\|+2\|H\|+\|ID\|$	$2\|G\|+\|T\|$	**3**
Shen et al.'s Scheme [11]	$3T_{sm}+T_{exp}$	$3T_{sm}+T_{exp}$	$2\|r\|+2\|H\|+\|ID\|$	$2\|r\|+2\|H\|+\|ID\|$	**2**
Proposed LA-IMDCN	$2T_{sm}+4T_h+T_{en}$	$4T_{sm}+6T_h+T_{en}$	$\|G\|+\|H\|+\|T\|$	$2\|G\|+\|H\|+\|T\|$	**2**

- `Reveal`$(\mathcal{P}^j)$: The session key is obtained using this query. $\mathscr{A}$ computes the session key for the instance $IMD-CN^k$ of an user U and another instance $IMD-CN^l$ of CN gateway during the j^{th} execution of the protocol $\mathcal{P}$.
- `Test`$(IMD-CN^k)$: $\mathscr{C}$ flips a coin c and responds the queries asking the session key by $\mathscr{A}$.

We adopt the security model used to prove the proposed protocol's security. The adversary $\mathscr{A}$ communicates with CN-gateway. Let CN^k denotes the k^{th} instances of CN. $\mathscr{A}$ submits several queries to the challenger $\mathscr{C}$ and $\mathscr{C}$ responses and send the answers accordingly to $\mathscr{A}$. The ways of communication, both the answer and queries, are briefed in subsection V-A.

Guess: The challenger $\mathscr{C}$ toss a coin and obtain α. In order to breach the security of LA-IMDCN, $\mathscr{A}$ attempts to assume the value α' and win the game if the guessed value $\delta' = \delta$.Let $\mathcal{E}_1$ represent the event where $\mathscr{A}$ guesses the correct value δ So the advantage of $\mathscr{A}$ is defined as

$$Adv(\mathscr{A}) = |2Pr[\mathcal{E}_1] - 1|$$

where $Pr[\delta' = \delta]$ denote the probability that $\alpha' = \alpha$.

Definition 1: The Proposed protocol LA-IMDCN *is secured if $Adv(\mathscr{A})$ is negligible.*

Theorem 1: The Proposed protocol LA-IMDCN *preserves mutual authentication.*

Proof: The adversary $\mathscr{A}$ executes `Send`(U, m_1) and in case the challenger $\mathscr{C}$ is able to obtain $\psi^* = \psi$ and $\alpha 1' = \alpha$, them m_1 is legitimate, where $m_1 = \{W_i, \alpha_1, T_1\}$. $\mathscr{C}$ checks the session and the validity of m_1 by using the private keys s and $sk_i = \frac{1}{Q_i+s}$ of CN and U respectively. $\mathscr{C}$ searches the list L_H obtains a record with probability $\frac{1}{q_h}$ and another record for m_l with probability $\frac{1}{q_m}$. Hence $\mathscr{A}$ can produce a forge message m_1. The probability of the event $\mathcal{E}_2$ for $\mathscr{A}$ to forge message m_1 is $Pr[\mathcal{E}_2] = \frac{1}{q_h q_m}$. Similarly $\mathscr{A}$ attempts to forge m_2 by executing `Send`(CN, m_2), where $m_2 = (R, S, \alpha_2, T_2)$. Here if $\mathscr{C}$ successfully able to verify $\alpha_2 = MAC_{\beta'}[V_i, T_2^*, R, S^*]$. Then $\mathscr{C}$ obtains a records from the list L_m with probability $\frac{1}{q_m}$. In case, if it returns two legitimate messages $\{R, S, \alpha_2, T_2\}$ and $\{\bar{R}, \bar{S}, \bar{\alpha_2}, \bar{T_2}\}$, then $\mathscr{C}$ computes $(\gamma_i - \bar{\gamma_i}) \cdot P$ and probability of the occurring of the event $\mathcal{E}_3 = \frac{1}{p \cdot q_h q_m^2}$. Hence it concludes that $Adv(\mathscr{A})$ is negligible. □

Theorem 2: The Proposed protocol LA-IMDCN *is semantically secure if the Elliptic Curve Discrete Logarithm Problem (ECDLP) is hard. authentication.*

Proof: The $\mathscr{C}$ has negligible advantage δ on execution of tt Test query for computing the correct session key $k = H_3(ID_i\|\psi_i\|V_i\|R\|S)$. Let $\mathcal{E}_k$ denote the correctly computes the session key k. During the execution of the `Test` query, $\mathcal{A}$ guesses the outcomes of δ with probability $\geq \frac{1}{2}$. Therefore

$$Pr[\mathcal{E}_k] \geq \frac{\epsilon}{2}. \tag{1}$$

Let $\mathcal{E}^{CN}_{Test}$ and $\mathcal{E}^{U}_{Test}$ denote the event that CN and U are queried by `Test` respectively. Hence we get $\frac{\epsilon}{2} \leq Pr[\mathcal{E}_k] = Pr[\mathcal{E}_k \wedge \mathcal{E}^{U}_{\texttt{Test}}] + Pr[\mathcal{E}_k \wedge \mathcal{E}^{CN}_{\texttt{Test}} \wedge \mathcal{E}_2] + Pr[\mathcal{E}_k \wedge \mathcal{E}^{CN}_{\texttt{Test}} \wedge \neg\mathcal{E}_2.$

$$Pr[\mathcal{E}_k \wedge \mathcal{E}^{CN}_{\texttt{Test}} \wedge \mathcal{E}_2] + Pr[\mathcal{E}_k \wedge \mathcal{E}^{CN}_{\texttt{Test}} \wedge \neg\mathcal{E}_2 \leq \frac{\epsilon}{2} - Pr[\mathcal{E}_2] \tag{2}$$

Since $Pr[\mathcal{E}^{CN}_{\texttt{Test}} \wedge \neq \mathcal{E}_2 = \mathcal{E}^{U}_{\texttt{Test}}$, therefore

$$Pr[k = H_3(ID_i\|\psi_i\|V_i\|R\|S)] \geq \frac{\epsilon}{4} - Pr[\mathcal{E}_2]/2 \qquad \square$$

B. RESILIENCE AGAINST OTHER ATTACKS

1) **User(U) Impersonate Attack**: The adversary $\mathscr{A}$ may try to impersonate itself as valid user U by forging the initial message m_1.When impersonating, $\mathscr{A}$ tries to produce a fake but valid message $m_1 = \{W_i, \alpha_1, T_1\}$ on behalf of U. For this $\mathscr{A}$ chooses a random number γ_i and can evaluate $V_i = \gamma_i \cdot P$ but computing $W_i = sk_i\gamma_i$. The adversary needs the private key of U. Furthermore, by theorem 1, $\mathscr{A}$ cannot form a valid message m_1 that can fufill both $\psi_i^* = H_1(ID_i\|s\|sk_i)$ and $\alpha_1' = MAC_{\beta'}[ID_i, T_1^*, V_i', \psi_i]$ without the possession of private key sk_i and the secret token ψ_i of U with non-negligible advantage. Hence, the proposed LA-IMDCN is resilient against impersonate attacks.

2) **CN-gateway Impersonate Attack**: The adversary $\mathscr{A}$ may try to impersonate CN-gateway. For this $\mathscr{A}$ tries to form the reply message $m_2 = \{R, S, \alpha_2 T_2\}$ by forming a new time stamp T_2 and sending to U the forged but valid message m_2. But this message m_2 is

formed by $\mathscr{A}$ much pass $\alpha_2 = MAC_{\beta'}[V_i, T_2, R, S]$. As by theorem 2, it has been proven that $\mathscr{A}$ cannot construct a valid message m_2 without having access to master secret key s. $\mathscr{A}$ possesses negligible advantage to perform these operations. Hence $\mathscr{A}$ cannot impersonate on behalf of a CN-gateway.

3) **Key Compromise Impersonation Attack** According to the proposed LA-IMDCN scheme, $\mathscr{A}$ can get the private key of U and pose as the non-compromised CN-gateway on its behalf.. $\mathscr{A}$ possess the private key and the related parameters $\{\psi_i, Q_i, sk_i, ID_i\}$ of U. $\mathscr{A}$ waits for ID_i to make login requests, and once made, it disables the request. $\mathscr{A}$ examines the login information $\{C, W_i, \alpha_1, T_1\}$. $\mathscr{A}$ may form the response message $m_2 = (R, S, \alpha_2, T_2)$ by generating fresh time stamp T_2 sending the forged but genuine message m_2 to U. For this $\mathscr{A}$ must construct m_2 which must pass $\alpha_2 = MAC_{\beta'}[V_i, T_2, R, S]$. Furthermore decryption of C and computation of $V_i = (s + Q_i) \cdot W_i$, the adversary must need the master secret key s. By theorem 2, $\mathscr{A}$ cannot construct a valid m_2 without possessing the secret key s. Similarly, $\mathscr{A}$ not able to compute the pair (α_2, S) by possessing the secret key s and other related parameters $\{\psi_i, Q_i, sk_i, ID_i\}$ of U. Hence $\mathscr{A}$ has negligible advantage to construct a verifiable message m_2 without possessing the CN-gateway's private key s. Thus, the proposed scheme LA-IMDCN is resilient against key compromise impersonation attacks.
4) **Man-in-the-Middle Attack** : $\mathscr{A}$ can mount man-in-the-middle attack (MIMA), and for this $\mathscr{A}$ can wait the user U's login. $\mathscr{A}$ captures these parameters $\{C, W_{i,\alpha_1,T_1}\}$ from the communication channel and tries to send the forged message $\{\bar{C}, \bar{W}_i, \bar{\alpha_1}, \bar{T}_1\}$ to the CN-gateway. Similarly $\mathscr{A}$ captures the reply message $\{R, S, \alpha_2, T_2\}$ and try to send the forged message $\{\bar{R}, \bar{S}, \bar{\alpha_2}, \bar{T}_2\}$. According to the theorem 1 cannot capture both forged request and forged reply messages. Thus, the proposed scheme LA-IMDCN is resilient against MIMA attacks.
5) **Reply Attack** In the proposed scheme, the request message $\{C, W_i, \alpha_1, T_1\}$ contains time-stamp T_1 and the cipher of $C = E_{\psi_i}(ID_i, T_1)$. If an attacker replies to an old message or sends the altered message by replacing T_1, then it would be checked through the assigned time-stamp at the recipient end. Hence, it is resistant to reply attacks.
6) **Perfect Forward Secrecy** The session key $k = H_3(ID_i \| \psi_i \| V_i \| R \| S)$ is constructed by using both the secret session parameters V_i, S and long term secret ψ_i. It is computationally infeasible if the long-term and session parameters are exposed to the adversary. Therefore, the proposed LA-IMDCN scheme ensures perfect forward secrecy.
7) **Known Session Key** In LA-IMDCN, the session keys are independent of each other. Additionally, the construction of session key k uses the random number and one-way cryptographic hash function. If we assume one session key, say $k^1 = H_3(ID_i^1 \| \psi_i^1 \| V_i^1 \| R^1 \| S^1)$ is is computed by U, it will not affect the other session key $k^2 = H_3(ID_i^2 \| \psi_i^2 \| V_i^2 \| R^2 \| S^2)$.
8) **Stolen verifier attack**: The verification data (C, W_i, α_1, T_1) is computed as $\alpha_1' = MAC_{\beta'}[ID_i, T_1^*, V_i', \psi_i]$, where $\beta' = H_2(ID_i \| V_i' \| W_i \| T_1)$, $W_i = sk_i \gamma_i$ and $E_{\psi_i}(ID_i, T_1)$. So it is not possible for the adversary to generate the communication data using the stolen data and sends them to the server CN. He would not succeeds, to impersonate a legal user from the next authentication session

VI. CONCLUSION

The paper presents a novel approach to access control using blockchain network through smart contracts by offering granular control and time-bound access for improved security in managing Internet of Medical Devices (IMD). It provides access rights administration by introducing a simplified device removal process through smart contracts, reducing administrative overhead and enhancing system usability. This enhances security, minimizing unauthorized access and data integrity risks in IMD and other sensitive data management scenarios. The proposed scheme LA-IMDCN is resilient against other attacks, including impersonate attacks, man-in-the-middle attacks, and reply attacks and ensures perfect forward secrecy. The communication overhead and computation time have been compared with the schemes proposed by Wang et al. [12], Ni et al. [9], Kumari et al. [7] and Shen et al. [11]. Additionally LA-IMDCN is designed to be compatible, works on any blockchain network that support both transaction and smart contract. The smart contract provides an efficient key update and revocation by reducing computation cost.

REFERENCES

[1] M. Abdalla, P. A. Fouque, and D. Pointcheval, "Password-based authenticated key exchange in the three-party setting," in *Proc. Int. Workshop Public Key Cryptogr.*, 2005, pp. 65–84.

[2] U. Bansal, J. Kar, I. Ali, and K. Naik, "ID-CEPPA: Identity-based computationally efficient privacy-preserving authentication scheme for vehicle-to-vehicle communications," *J. Syst. Archit.*, vol. 123, Feb. 2022, Art. no. 102387.

[3] D. Halperin, T. S. Heydt-Benjamin, B. Ransford, S. S. Clark, B. Defend, W. Morgan, K. Fu, T. Kohno, and W. H. Maisel, "Pacemakers and implantable cardiac defibrillators: Software radio attacks and zero-power defenses," in *Proc. 29th Annu. IEEE Symp. Secur. Privacy*, 2008, pp. 129–142.

[4] J. Kar, "Provably secure certificateless deniable authenticated encryption scheme," *J. Inf. Secur. Appl.*, vol. 54, Oct. 2020, Art. no. 102581.

[5] J. Kar, K. Naik, and T. Abdelkader, "An efficient and lightweight deniably authenticated encryption scheme for e-mail security," *IEEE Access*, vol. 7, pp. 184207–184220, 2019.

[6] N. Kumari, J. Kar, and K. Naik, "PUA-KE: Practical user authentication with key establishment and its application in implantable medical devices," *J. Syst. Archit.*, vol. 120, Nov. 2021, Art. no. 102307.

[7] S. Kumari, M. Karuppiah, A. K. Das, X. Li, F. Wu, and N. Kumar, "A secure authentication scheme based on Elliptic curve cryptography for IoT and cloud servers," *J. Supercomput.*, vol. 74, no. 12, pp. 6428–6453, Dec. 2018.

[8] D. Mingxiao, M. Xiaofeng, Z. Zhe, W. Xiangwei, and C. Qijun, "A review on consensus algorithm of blockchain," in *Proc. IEEE Int. Conf. Syst. Man, Cybern. (SMC)*, Oct. 2017, pp. 2567–2572.

[9] J. Ni, X. Lin, and X. S. Shen, "Efficient and secure service-oriented authentication supporting network slicing for 5G-enabled IoT," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 3, pp. 644–657, Mar. 2018.

[10] O. Pal, B. Alam, V. Thakur, and S. Singh, "Key management for blockchain technology," *ICT Exp.*, vol. 7, no. 1, pp. 76–80, Mar. 2021.

[11] 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.

[12] J. Wang, L. Wu, K. R. Choo, and D. He, "Blockchain-based anonymous authentication with key management for smart grid edge computing infrastructure," *IEEE Trans. Ind. Informat.*, vol. 16, no. 3, pp. 1984–1992, Mar. 2020.

[13] W. Wang, H. Huang, L. Zhang, Z. Han, C. Qiu, and C. Su, "BlockSLAP: Blockchain-based secure and lightweight authentication protocol for smart grid," in *Proc. IEEE 19th Int. Conf. Trust, Secur. Privacy Comput. Commun.*, Dec. 2020, pp. 1332–1338.

[14] W. Yu, F. Liang, X. He, W. G. Hatcher, C. Lu, J. Lin, and X. Yang, "A survey on the edge computing for the Internet of Things," *IEEE Access*, vol. 6, pp. 6900–6919, 2018.

[15] H. Zhang, J. Wang, and Y. Ding, "Blockchain-based decentralized and secure keyless signature scheme for smart grid," *Energy*, vol. 180, pp. 955–967, Jan. 2019.

JAYAPRAKASH KAR (Senior Member, IEEE) received the M.Sc. and M.Phil. degrees in mathematics from Sambalpur University, and the M.Tech. and Ph.D. degrees in computer science (cryptographic protocols) from Utkal University, India. He is a Visiting Researcher with the Department of Electrical and Computer Engineering, University of Waterloo, Canada. Currently, he is a Professor with the Department of Computer Science and Engineering, The LNM Institute of Information Technology, Jaipur, India. He is a Center-Lead of the Center for Cryptology, Cybersecurity and Digital forensics (C3-SDF). His current research interests include cryptographic protocols and primitives using elliptic curve and pairing based cryptography in random oracle and standard model, zero-knowledge proofs, secret sharing, and multi-party computation. He is a Life Member of the International Association for Cryptology Research (IACR) and the Cryptology Research Society of India; and an Associate Member of ACM, the International Association of Computer Science and Information Technology (Singapore), and the International Association of Engineers (USA). He is an advisory and editorial board member of many peer reviewed journals and international conferences. He is an Associate Editor of *Journal of Circuits, Systems and Computers*; and *World Scientific*.

XIAOGUANG LIU received the Ph.D. degree in mathematics from Shaanxi Normal University, Xi'an, China, in 2014. From 2017 to 2020, he was a Postdoctoral Fellow with the School of Computer Science and Engineering, University of Electronic Science and Technology of China (UESTC), Chengdu, China. He is currently a Lecturer with the School of Mathematics, Southwest Minzu University, Chengdu. His current research interests include cryptography, network security, and optimization computation.

FAGEN LI (Member, IEEE) received the Ph.D. degree in cryptography from Xidian University, Xi'an, China, in 2007. He is currently a Professor with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China. From 2008 to 2009, he was a Postdoctoral Fellow with Future University Hakodate, Hokkaido, Japan, which is supported by Japan Society for the Promotion of Science. From 2010 to 2012, he was a Research Fellow with the Institute of Mathematics for Industry, Kyushu University, Fukuoka, Japan. He has authored or co-authored more than 110 papers in international journals and conferences. His research interests include cryptography and network security.