







A Novel Patient-Centric Architectural Framework for Blockchain-Enabled Healthcare Applications

Akhilendra Pratap Singh, *Member, IEEE*, Nihar Ranjan Pradhan , *Student Member, IEEE*, Ashish K. Luhach , *Member, IEEE*, Sivansu Agnihotri, *Member, IEEE*, Noor Zaman Jhanjhi , *Senior Member, IEEE*, Sahil Verma , *Member, IEEE*, Kavita , *Member, IEEE*, Uttam Ghosh , *Senior Member, IEEE*, and Diptendu Sinha Roy , *Senior Member, IEEE*

Abstract—With the proliferation of information and communication technology in every walks of the society, including healthcare services, digitization, and increased sophistication have been gaining pace, digital healthcare alternatives such as electronic healthcare record (EHR) have gained prominence with increased patients' data volume. However, traditional EHR-based systems are plagued by data loss risks, security and immutability consensus over health records, gapped communication among constituted hospitals, and inefficient clinical data retrieval systems, among others. Blockchain has been developed as a decentralized technology that holds the promise to address the aforesaid facilities in EHR-based systems. This article presents a patient-centric design of a decentralized healthcare management system with blockchain-based EHR using javascript-based smart contracts. A working prototype based on hyperledger fabric and composer technology has also been implemented which guarantees the security of the proposed model. Experiments with the hyperledger caliper benchmarking tool provide performance such as latency, throughput, resource utilization, and so on under varied scenarios and control parameters. The results affirm the efficacy of the proposed approach.

Index Terms—Blockchain, chain-code, electronic healthcare records (EHRs), hyperledger.

Manuscript received June 4, 2020; revised July 23, 2020, September 15, 2020, and October 14, 2020; accepted October 29, 2020. Date of publication November 16, 2020; date of current version May 3, 2021. Paper no. TII-20-2730. (Corresponding author: Sahil Verma.)

Akhilendra Pratap Singh, Nihar Ranjan Pradhan, Sivansu Agnihotri, and Diptendu Sinha Roy are with the Department of Computer Science and Engineering, National Institute of Technology Meghalaya, Shillong 793003, India (e-mail: akhilendra.singh@nitm.ac.in; niharpradhan@nitm.ac.in; shivanshuagnihotri27@gmail.com; diptendu.sr@nitm.ac.in).

Noor Zaman Jhanjhi is with the School of Computer Science and Engineering SCE, Taylor's University, Malaysia (e-mail: noorzaman.jhanjhi@taylors.edu.my).

Ashish K. Luhach is with the Department of Electrical and Communication Engineering, PNG University of Technology, Lae 411, Papua New Guinea (e-mail: ashishluhach@gmail.com).

Sahil Verma and Kavita are with the Department of Computer Science and Engineering, Chandigarh University, Mohali 140413, India (e-mail: sahilverma@ieee.org; kavita@ieee.org).

Uttam Ghosh is with the Lovely Professional University Faculty of Technology and Sciences, Phagwara 144402, India (e-mail: ghosh.uttam@ieee.org).

Color versions of one or more of the figures in this article are available at <https://doi.org/10.1109/TII.2020.3037889>.

Digital Object Identifier 10.1109/TII.2020.3037889

I. INTRODUCTION

BLOCKCHAIN is a distributed system used to record and store transactions with shared, immutable records of peer-to-peer transactions stored in a digital ledger. Blockchain-enabled communication environment provides trustworthy and secure transactions with cryptographic primitives. In recent years, blockchain has gained immense attention from different domains, including healthcare domain, owing to its tremendous potential in integrating disjoint stakeholders and to increase the accuracy of electronic healthcare records (EHRs) [1].

Blockchain has several key features such as decentralization, persistency, anonymity, immutability, and security among others [2], [3]. With the current technological development of the Internet, healthcare services are moving from offline mode to online mode. Online accessing, storing, and maintenance of healthcare records have various security issues such as scattered and disjoint health data, interoperability troubles, data security, privacy, scalability [4]–[8]. These have paved the path for blockchain-based healthcare systems' research and implementation.

Various healthcare institutions use centralized systems and maintain their databases to store patients' health records (EHRs). Such EHRs require frequent distribution and sharing between different stakeholders such as hospitals, patients, and clinics. Distributing EHRs is a time- and cost-intensive process. Cloud-based health data management was introduced in the past to address issues of real-time data sharing and access [7]. However, this becomes memory intensive since patients and hospitals need to encrypt the data before sending it to the cloud. To address this drawback, a lightweight blockchain was introduced by Ismail *et al.* [8] which reduces computation and communication overheads by grouping participants of healthcare networks into clusters as per their demographics and by maintaining a single ledger per cluster. Privacy and security issues could be addressed by such a blockchain-based framework for sharing medical imaging data securely [9], [10]. To overcome the problems of confidentiality, access control, data integrity, level of authorization, authentication, and privacy of record, the authors in [11]–[13] suggested new attribute-based and signature-based data encryption–decryption methods, access control mechanisms, and authorization methods; whereas [14] addressed blockchain-based interoperability issues. In the healthcare domain, various researchers

suggested Ethereum smart contracts based blockchain owing to its easy deployment. However, it was rendered rather inefficient and insecure due to its permission-less mode of operation. It can also affect the performance in terms of computing power, scalability, and poor transparency; though transparency can be achieved at the cost of scalability and privacy [15], [27]. A new framework that supported permissioned mode to solve scalability and privacy issues as well as fine-grained access control mechanism was the hyperledger [16], [17]. Blockchain-based patient-centric healthcare systems were suggested in [18] and [19].

For a privacy-focused patient-centric healthcare system, several approaches have been suggested; but these lack access control mechanism. Thereafter, access control enabled frameworks had been introduced in hyperledger-based systems which were then deployed over the network for data access. Declarative access control [create, read, update, or delete (CURD)] is one such mechanism defined within hyperledger framework. It allows performance evaluation for throughput, latency, transaction rate, memory consumption, CPU utilization, and disc read/write operations using hyperledger Caliper as a benchmarking tool. In this article, a patient-centric architectural framework is presented which employs blockchain for next-generation healthcare applications. The proposed model empowers a patient with unprecedented access control for seamless queries and tracking of his/her healthcare information and is, thus, patient-centric unlike any other blockchain-enabled healthcare systems currently available. Among the different stakeholders, documents are very vital for patients to be accessed, which, however, is not readily available at their disposal. For instance, due to the complicated process of obtaining documents, insurance claims become very cumbersome and sometimes a great hassle for patients. Thus, the main highlight of this article is patient-centric architecture. The main contributions of this article are summarized after highlighting the research gaps in the subsequent section.

II. RELATED WORK

In this section, the state-of-the-art research works related to blockchain for healthcare application have been presented. With the advent of blockchain technology, there are a number of blockchain-enabled healthcare applications [20]–[26] that have emerged to show the pertinence and importance of blockchain in healthcare focused on authenticity, data security, data sharing, and data privacy at different levels. Hathaliya *et al.* [20] proposed an improved location and biometric-based access control scheme, namely automated validation Internet security protocol and application (AVISPA) for secure EHR. Huang *et al.* [21] discussed a user-centric scheme that focused on integrity and validation; however, it was severely impaired by scalability restrictions because of sensor constraints. Fan *et al.* [23] presented a healthcare information system, namely MedBlock, that employs distributed ledgers for efficient access and also presented an improved consensus mechanism. Wang and Song [24] presented a combined-attribute and identity-based encryption and signature (C-AB/IB-ES) that delves into intrinsic cryptography that could also attest to the integrity and traceability of medical

data. It has to be noted that the work presented herein neither focuses on novel consensus mechanisms nor does it attempt to enunciate new encryption schemes; rather, it assumes that such mechanisms are inherently present in the proposed scheme. Uddin *et al.* [25] presented an agent-based system for easy retrieval of remotely streamed medical data. In [24], the main focus was on privacy preservation and verification of authenticity with signer's identity that also tracks on-chain and off-chain collaborative storage for efficient storage and verification thereof. However, these are the aspects that have been assumed to be available for this work. Tanwar *et al.* [26] have proposed a permissioned blockchain-based healthcare framework to provide improvement for accessing data among healthcare providers using an algorithm, namely access control policy. The work presented by Tanwar *et al.* [26] has synergy with that presented newly in this article. However, in this work, two new modules, namely chemist and insurance modules, have been included which have completely changed the interplay of the entire architecture. Such a scheme will allow insurance agencies to register, approve, verify, and disburse claims conveniently and efficiently with foolproof authenticity. This is one such scenario and there can be many such scenarios. Besides, the entire design of the work here is founded on a patient-centric design principle where the patients' requirements can be made simple, unlike the works presented by Tanwar *et al.* [26] and Bhattacharya *et al.* [27]. For instance, generation of patients' records can be made available even after the discharge of a patient, across time frame, hospitals, and others. Thus lies the novelty of the work presented herein. At the architecture level, access rights to different stakeholders has also been restricted by setting certain rules in the network which may otherwise lead to compromise of data privacy or even may cause some personal harm to a patient. A few works have delved on patient-centric healthcare systems, such as the work presented by Shen *et al.* [28]. However, none of these works have been found to provide a full end-to-end implementation or had reported performances. Notable exceptions include [6], [13], and [20] that supported both implementation and performance evaluation. The business logic of suggested frameworks [8] was not explicit and implementations were devoid of any smart contracts. On the contrary, the work by Hathaliya *et al.* [20] ignored data privacy, a key factor in the existing EHR frameworks. In [8], [13], [20], and [21], technology-centric frameworks had been presented which is at diametrically opposite end of the spectrum with respect to patient-centric approach, which is the need for upcoming healthcare sectors.

Based on the abovementioned works, it can be concluded that although a lot of attempts had been made to implement blockchain-based healthcare systems and constituent technologies, none of these could account for all the following properties, namely, smart contracts, data privacy and security, patient-centric approach along with implementation and performance of their proposed schemes. To this end, the present work has more potential to address the aforementioned issues while also presenting the implementation and performance evaluation of the proposed scheme. The main contributions of this article include: 1) design of a lightweight access control scheme via hyperledger blockchain applied to the healthcare domain, 2) development

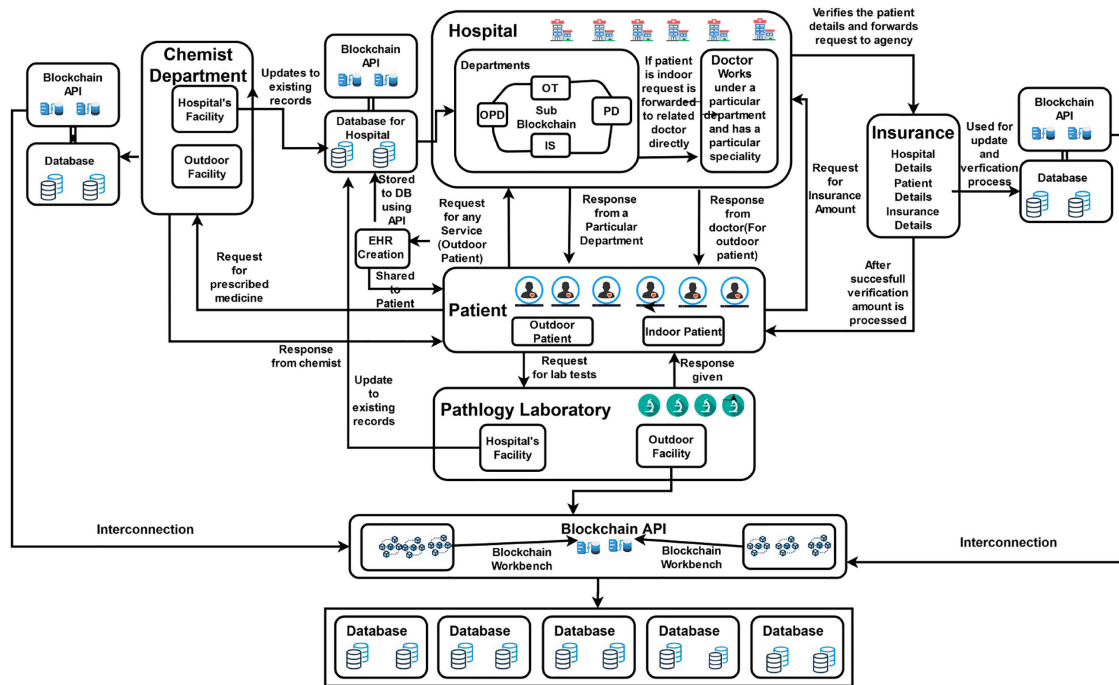


Fig. 1. Patient-centric architectural framework for blockchain-enabled healthcare.

of an architectural framework for such applications, and 3) extensive experimentation in order to show the efficacy of the proposed scheme in terms of various performance metrics such as resource utilization, latency, and throughput, among others.

III. SYSTEM ARCHITECTURE FOR THE PROPOSED FRAMEWORK

Blockchain-based patient-centric healthcare framework is shown in Fig. 1. It depicts the different functional components of the proposed patient-centric, blockchain-based healthcare system, each represented using a rectangular box. The proposed framework presents a common data-sharing platform for disjoint stakeholders of the healthcare system. Blockchain is a ledger for each participant to store health data on the network. Each such component is provided with a blockchain application programming interface (API) that helps in establishing communication among participants and governs the state of the blockchain by interacting with transactions that update the ledgers. The architecture has five software modules, each of which are further depicted in Figs. 2– 5. The first module of the application deals with creating secure, decentralized, and immutable EHRs. The next module deals with maintaining EHR consistency among different participating hospitals and enables patients to share the EHR data with the consent of each patient. The final module encompasses rest of the functionalities including retrieving the full medical history of a patient at a single point, easy verification of medical prescription, availability of EHRs for research purposes, and imparting transparency over patient data. Section III-A outlines the five modules, Section III-B defines the participants and assets of different entities, and Section III-C outlines their requirements.

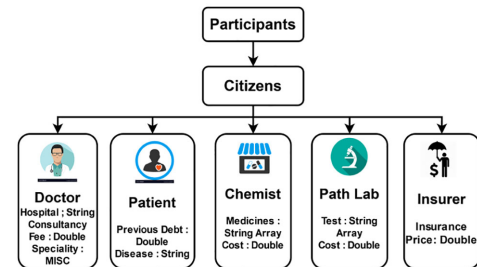


Fig. 2. Participants module.

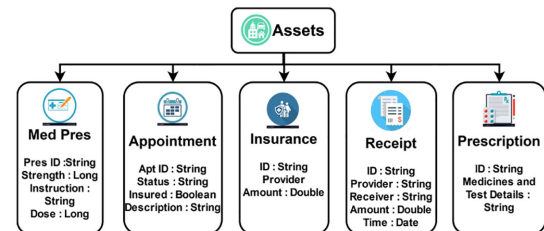


Fig. 3. Assets module.

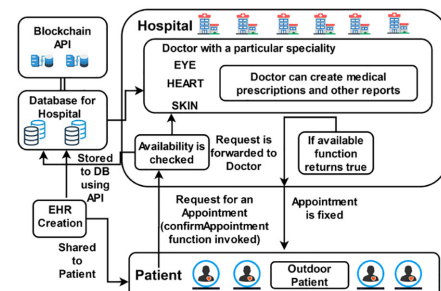


Fig. 4. Appointment module.

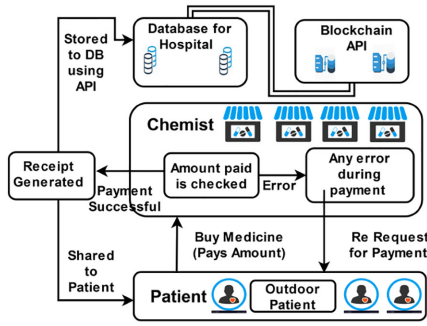


Fig. 5. Chemist module.

A. System Architecture

In the proposed patient-centric, blockchain-based healthcare framework, all modules, assets, and smart contracts are defined to trace the functionality, which works in stand-alone mode as well as in overall functioning framework. The patient registers for an appointment using the client interface with chain code over the communication network. The committed transaction (appointment) is distributed over the network to all stakeholders to ensure security and robustness against data modification or deletion prevention from unauthorized attackers. This transaction is available in the distributed ledger with timestamp and hash value, thus simplifying the verification of a genuine patient. This healthcare record is available to all authorized stakeholders and they may raise the query from other authorized stakeholders over the blockchain-enabled communication network. The patient may also raise a query to the doctors about his appointment, medicine, reports, clinical diagnoses, and many more.

B. Proposed Participants, Assets, and Transactions

The hyperledger composer provides a deeper level of abstraction to develop technical blockchain-based applications. Entities related to a network are participants, assets, transactions, and control rules. These entities combine to set up a complete network. For the proposed healthcare system, a detailed case study has been presented which helps to define the participants and assets required to develop the application. It also defines the transactions required to be executed to perform a particular functionality.

- 1) **Participants:** A participant can be an individual entity or an organization. A participant present in the network can create assets and can exchange assets with other participants. Existing participants in our application are listed in Fig. 2.
- 2) **Assets:** Assets can be any tangible or intangible entity. For example, a receipt is a tangible entity. These assets can be modified by creating different transactions. All the assets related to our application are listed in Fig. 3.
- 3) **Transactions:** A transaction is an invoke result that is submitted for ordering, validation, and committing. In our application, we have worked on two transactions. The first transaction is created to confirm the appointment if a doctor is available and the second transaction is

created to generate the receipt whenever a patient purchases medicines from the chemist. Scenarios related to appointment fixing are listed in Fig. 4. Here an outside patient looks for an appointment under a particular disease category. Doctor with the corresponding category is selected to fix an appointment.

The second functionality is to generate a receipt. A chemist requests for a particular amount to be paid and if the patient successfully pays the amount, a receipt is generated. A complete scenario is depicted in Fig. 5.

C. Interaction Among Participants

1) **Doctor–Patient Appointment:** Patients, doctors, chemists, pathology, and assets (appointment, receipts, etc.) are used in the case study we presented. The doctor ID and patient ID are stored in the hospital database using blockchain as denoted by D and P , respectively. The set d defines different diseases. In this module, a patient requests for an appointment. The functions of the modules are as follows:

A = set of appointment IDs, D = set of available doctors, P = set of incoming patients, d = diseases that can be treated in the hospital, and S = total number of specializations.

PID_i = patient ID of patient i , where $i \in (1, P)$, DID_j = doctor ID of doctor j , where $j \in (1, D)$, AID_{ij} = appointment ID of patient i assigned under doctor j , d_i = particular disease i from set of disease d .

SP_{ik} = patient i is suffering from disease k , SD_{jl} = doctor j has l specialty, where $l \in (1, S)$.

Objective functions are shown in the following equations:

$$R = X + Y + Z \quad (1)$$

$$X = \begin{cases} 1, & \text{if } SP_{ik} = d_i \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

$$Y = \begin{cases} 1, & \text{if } AID_{ij} = A \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

$$Z = \begin{cases} 1, & \text{if } SD_{jl} = SP_{jl} \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

The output depends on the final value of R . R must be 1 to confirm a credible appointment.

2) **Chemist–Patient Interaction:** This module consists of four submodules such as patient, chemist, doctors, and medicines. Medicines purchased by the patient is recorded and a receipt shall be generated by a chemist using blockchain. This functionality is shown in (5). C = set of existing chemists and P = set of incoming patients.

PID_i = ID of patient i , where $i \in (1, P)$, CID_i = ID of chemist i , where $i \in (1, C)$, D_i = previous debt amount on patient i , and A_i = new amount asked by a chemist for a particular set of medicines. Following is the chemist–patient objective function:

$$RP = \begin{cases} 1, & \text{if } P_r = T_i / (A_i + D_j) \geq 0.5 \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

where RP denotes the receipt generation probability.

3) **Patient Checkup:** In this module, a function is executed with predefined constraint such as patient appointment confirmed or waiting and the appointment fee. The output of the function is defined as shown in (6), where A = set of appointment IDs and AS_i = appointment status of patient i . Patient checkup objective function is

$$R = \begin{cases} 1, & \text{if } AS_i = \text{"CONFIRMED"} \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

The output depends on the final value of R . R must be 1 for a valid checkup.

4) **Pathology Lab–Patient Interaction:** The Pathology lab (PathLab) and patient modules comprise three submodules, namely, patient, and doctor. The set of tests collected by PathLab IDs are denoted by L . The transaction amount, doctor ID, and patient ID with a recommendation are defined as the objective function depicted in (7).

PID_i = ID of patient i , where $i \in (1, P)$, LID_i = ID of pathology lab i , where $i \in (1, L)$, d_i = previous debt amount on patient i , and A_i = new amount asked by pathology lab for a particular set of lab reports. Pathology lab–patient objective function is as follows:

$$RP = \begin{cases} 1, & \text{if } P_r = T_i / (A_i + D_j) \geq 0.5 \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

where RP denotes the receipt generation probability.

5) **Insurance Claim:** In this module, patient and insurance provider are mentioned as submodules. The insurance provider gets information from the hospital records. The payment to hospital is completed based on the available balance and terms and conditions. The objective function is defined based on the constraints as shown in (8) and (9).

I = set of insurance providers and P = set of incoming patients. PID_i = ID of patient i , where $i \in (1, P)$, IID_i = ID of PathLab i , where $i \in (1, I)$, D_i = previous debt amount on patient I , A_i = the insurance amount claimed by patient to clear previous debts, and AS_i = appointment status of patient i .

Listing 1: Participant creation.

```
{
  "\$class": "org.ehr.Patient,"
  "debt": 0,
  "disease": "EYE,"
  "id": "1001,"
  "firstName": "Shiva"
  "lastName": "Ram"
}
```

Insurance claim objective function

$$S = \begin{cases} 1, & \text{if } AS_i = \text{"CONSULTED"} \\ 0, & \text{otherwise} \end{cases} \quad (8)$$

Listing 2: Asset creation.

```
{
  "\$class": "org.ehr.Appointment,"
  "appointmentId": "3001,"
  "status": "PENDING,"
  "isInsured": "false,"
  "description": "Please come on time,"
  "group": "EYE"
}
```

Listing 3: Transaction execution.

```
{
  "\$class": "org.ehr.confirmAppointment,"
  "patient": "re-
source:org.ehr.Patient\#1001,"
  "doctor": "re-
source:org.ehr.Doctor\#2001,"
  "appoint": "re-
source:org.ehr.Appointment\#3001"
}
```

Listing 4: Asset updation.

```
{
  "\$class": "org.ehr.Appointment,"
  "appointmentId": "3001,"
  "status": "CONFIRMED,"
  "isInsured": "false,"
  "description": "Please come on time,"
  "group": "EYE"
  "assigned": "resource:org.ehr.Doctor\#2001,"
  "patient": "resource:org.ehr.Patient\#1001"
}
```

Listing 5: Transaction execution.

```
{
  "\$class": "org.ehr.buyMedicine,"
  "patient": "re-
source:org.ehr.Patient\#1001,"
  "chemist": "re-
source:org.ehr.Chemist\#2002,"
}
```

Listing 6: Asset generation.

```
{
  "\$class": "org.ehr.Receipt,"
  "receiptId": "R001,"
  "providerId": "2002,"
  "providedTo": "1001,"
  "amountPaid": 500,
  "logtime": "2020-02-18T06:36:49.6612"
}
```

Algorithm 1: Admin and Participants Enrollment.

Input: Enrolment Certificate (EC) requested from Certification Authority (CA) ;
Output: Access to all nodes C_N, P_N, D_N ;
Initialization: N_{Admin} should be valid node. N_{Admin} can Write , Read, Update, Remove Participants C_N, P_N, D_N ;
while *True* **do**
 if C_N *is valid* **then**
 Add_Node(C_N, B_N) ;
 Grant_access(C_N) ;
 else
 Not_valid(C_N)
 end
 if P_N *is valid* **then**
 Add_Node(P_N, B_N) ;
 Grant_access(P_N) ;
 else
 Not_valid(P_N)
 end
 if D_N *is valid* **then**
 Add_Node(D_N, B_N) ;
 Grant_access(D_N) ;
 else
 Not_valid(D_N)
 end
end
int N: (0 malicious, 1 otherwise) ;
if *activity_node(N)* **then**
 Add_Updates(C_N, P_N, D_N) ;
else
 Remove_Updates(C_N, P_N, D_N) ;
end

$$I = \begin{cases} 1, & \text{if } D_i \leq A_i \\ 0, & \text{otherwise} \end{cases} \quad (9)$$

$$R = S + I \quad (10)$$

where R denotes the final result shown in (10). If $R = 2$, then only the amount can be claimed.

IV. IMPLEMENTATION

This section is the deployment and implementation of the proposed framework.

1) **Deployment Phase:** The hyperledger platform is used to implement the framework for hospital records with security mechanisms. The hyperledger fabric and sandbox are used to implement the proposed architectural framework. Hyperledger is an authentication-based open-source distributed ledger-based platform to implement different smart contracts with constraints and logic for various applications over the online network. The sandbox module is used to implement the smart contracts over the virtual network. Sandbox module is a fully secure and trusted blockchain because participants are known and blockchain is

Algorithm 2: Algorithm on Participant Working. (Appointment Fixing and Buying Medicine)

Input: ID and key requested from N_{Admin} ;
Output: Access to transactions. ;
Initialization: C_N, P_N, D_N should be valid nodes.;
while *True* **do**
 if P_N *in* B_N **then**
 if P_{ID} *not in* B_N **then**
 Create_records(P_{ID}, P_{REC}, B_N) ;
 else
 Read_records(P_{ID}, P_{REC}, B_N) ;
 Update_records(P_{ID}, P_{REC}, B_N) ;
 end
 else
 Not_valid(P_N)
 end
 if D_N *in* B_N **then**
 if D_{ID} *not in* B_N **then**
 Create_records(D_{ID}, D_{REC}, B_N) ;
 else
 Read_records(D_{ID}, D_{REC}, B_N) ;
 Update_records(D_{ID}, D_{REC}, B_N) ;
 end
 else
 Not_valid(D_N)
 end
 if C_N *in* B_N **then**
 if C_{ID} *not in* B_N **then**
 Create_records(C_{ID}, C_{REC}, B_N) ;
 else
 Read_records(C_{ID}, C_{REC}, B_N) ;
 Update_records(C_{ID}, C_{REC}, B_N) ;
 end
 else
 Not_valid(C_N)
 end
end
int N: 0 or 1: Participants availability ;
if (Appointment(D_{ID}, P_{ID})) **then**
 $M_{PID} = \text{Medrecord}(P_{ID}, D_{ID})$;
 if N **then**
 Grant_records(M_{PID}) ;
 Generate_Receipt(P_{ID}, D_{ID}) ;
 else
 NOTIFY("Error!") ;
 end
else
 end
int N: (0 or 1: Payment Status) ;
if (Buying_Medicine(C_{ID}, P_{ID})) **then**
 $M_{PID} = \text{Medrecord}(P_{ID}, C_{ID})$;
 if N **then**
 Grant_records(M_{PID}) ;
 Generate_Receipt(P_{ID}, C_{ID}) ;
 else
 NOTIFY("Error!") ;
 end
else
 end

in permissioned consortium mode. The proposed architectural framework is not specific to the healthcare domain. Java, Go, Node.js, etc. are used to develop contracts and business networks. It has the following phases to deploy and implement the framework.

- 1) Design and deploy of networks: The network has been designed in the first phase of deployment with participants and assets. Creation of business network provides the facility to write the code for the application. After finishing the coding part related to the application, we generated a file with .bna format (banana file). This file was imported over online playground provided by hyperledger community to test the application.
- 2) Testing the application: In this case study, appointment, buying medicine, pathological report, insurance, and doctor diagnosis are implemented with the framework.
- 3) Confirmation of appointment: In the framework, two participants, patient and doctor, perform the transaction as a module. Code listing 1 shows the creation of the patient participant. In the same way, we created a participant doctor asset appointment. Code listing 2 shows the creation of the asset. A transaction is executed to fix the appointment with patient ID and doctor ID as depicted in listing 3. After executing the transaction, appointment registry was successfully updated and appointment status was changed to confirm from pending as mentioned in listing 4.
- 4) Buying medicine: The chemist module is designed and implemented in the framework. This module is tested with the execution of transaction required to buy the medicine as depicted in listing 5. After successful execution of the transaction, a receipt is generated as shown in listing 6.

Listing 7: Rest server response.

```
Request URL
http://localhost:3000/api/Patient/1001
Response Body
{
  "$class": "org.ehr.basic.Patient,"
  "debt": 400,
  "disease": "EYE,"
  "id": "1001,"
  "firstName": "Shiva"
  "lastName": "Ram"
}
Response Code
200
```

2) Implementation of CURD of Records Features Using REST Server: The REST server is used to develop the application interface with the deployed blockchain-based application network and the interface is compatible with client and HTTP. The assets are implemented with features for CURD records. The stakeholders are allowed to raise a query and receive responses from other authorized stakeholders. The administrator reads the patient record using a rest server response which

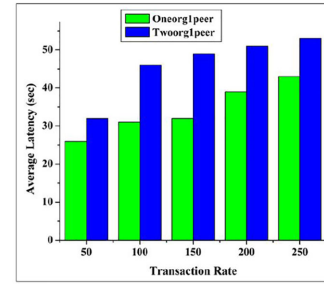


Fig. 6. Average latency with varying transaction rate.

is depicted in code listing 7. The operations such as GET, POST, DELETE, etc., are performed on the participant. Selected operations such as, GET (get patient by ID), are filled up with ID to retrieve the details related to a patient as shown in listing 7.

V. PERFORMANCE ANALYSIS AND DISCUSSION

The performance of the proposed architectural framework is tested and analyzed with benchmark and assessment measurements in this section. The results are analyzed with parameters such as block creation time, endorsement policy, block size, etc. Performance analysis for various cases is performed with the help of latency, throughput, and other parameters. Hyperledger caliper is the benchmark tool utilized for analysis of the developed blockchain-based applications over the network. It can support different hyperledger platforms, such as fabric, Indy, composer, sawtooth, Iroha, among others. In this article, caliper is utilized to verify and execute the performance of the framework. Different parameters, including latency, throughput, CPU usage, traffic in and out, memory consumption, disk write/read, network I/O, etc., are measured for the performance assessment of the framework. The configuration parameters are modified according to appraisal, for example, block size, transaction per second (TPS), support strategy, channel, resource utilization, and record database. The results are obtained using PCs with configurations shown in Tables I–III.

The distributed ledger technology is chosen as fabric. The proposed model is tested for performance analysis with simple writing and querying for transactions. The results obtained are presented in Fig. 6, which shows the latency of querying or reading is less compared to writing. A query transaction in a blockchain network is much faster than that of a writing transaction and the latency increases with transaction rate as well as the scale of operations. Low latency can be ascribed to the higher throughput based on (11)

$$TL = ((CT) * (NT)) - (ST) \quad (11)$$

$$TT = \frac{TCT}{TTS} - (NCN) \quad (12)$$

$$RL = (RR - ST) \quad (13)$$

$$RT = (RO - TT) \quad (14)$$

TABLE I
PERFORMANCE WITH 1000 TRANSACTIONS

Name	Succ	Fail	Send Rate (TPS)	Avg Latency (s)	Throughput
Open	1000	0	50, 100, 150, 200, 250	26.22, 31.00, 32.67, 36.22, 36.23	27, 53, 89, 101, 123
Query	1000	0	50, 100, 150, 200, 250	0.12, 0.44, 0.73, 3.51, 7.62	50, 96, 142, 195, 249

TABLE II
RESOURCE CONSUMPTION AFTER FIRST ROUND

Type	Name	Memory (Max)	Memory (Avg)	CPU % (Max)	CPU % (Avg)	Traffic In	Traffic Out	Disc Write
Docker	peer0.org2.example.com	203.9MB	200.3MB	4.58	3.03	3.3MB	1.8MB	16.3MB
Docker	peer0.org1.example.com	156.6MB	119.3MB	4.28	3.04	3.3MB	2.0MB	16.3MB
Docker	orderer.example.com	19.6MB	18.0MB	1.01	0.26	2.3MB	4.5MB	8.0MB
Docker	ca.org1.example.com	8.3MB	7.6MB	0.13	0.00	1.5KB	0B	0B
Docker	ca.org2.example.com	8.6MB	7.8MB	0.29	0.00	1.4KB	0B	0B

TABLE III
RESOURCE CONSUMPTION AFTER SECOND ROUND

Type	Name	Memory (Max)	Memory (Avg)	CPU % (Max)	CPU % (Avg)	Traffic In	Traffic Out	Disc Write
Docker	peer0.org2.example.com	205.6MB	124.0MB	5.19	3.16	6.7MB	3.6MB	32.5MB
Docker	peer0.org1.example.com	66.3MB	63.1MB	4.39	3.15	6.7MB	3.6MB	32.5MB
Docker	orderer.example.com	24.5MB	22.1MB	1.51	0.26	4.5MB	9.0MB	16.1MB
Docker	ca.org1.example.com	5.5MB	5.5MB	0.20	0.00	729B	0B	0B
Docker	ca.org2.example.com	5.9MB	5.9MB	0.29	0.00	729B	0B	0B

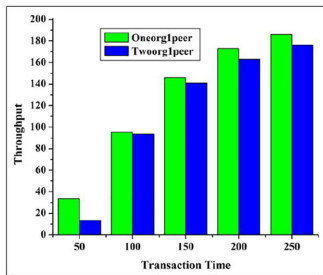


Fig. 7. Throughput with varying transaction rate.

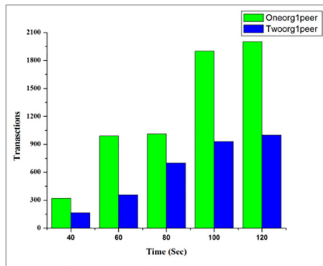


Fig. 8. Number of completed transactions with time.

where TL is transaction latency, CT is confirmation time, NT is network threshold, ST is submit time for transaction, TT is transaction throughput, RL is read latency, and WL is write latency.

While executing Caliper, different parameters such as average CPU consumption, memory, incoming traffic, outgoing traffic, disc read/write, etc., are estimated. Figs. 6–8 depict different variations of peers with org, while CPU consumption is also depicted in Tables I–III. The experiment was conducted in three rounds of transactions (writing and querying) into the blockchain

network in our proposed model with a ledger having 1000 first-round transactions. The distributed ledger here is made from fabric. Hyperledger fabric is a permissioned blockchain network which sets up organizations that take part to achieve consensus. The organizations are referred to as members. Each member in the blockchain network is allowed to set up their own peers for participating in the network. In the network, a peer can act as a participant. Each organization may configure the network with one participant or multiple participants to represent the entire organization. We have considered a network with two organizations both containing one peer each. In the network, One-org1peer is defined as a single peer of 1org network. Two-org1peer defines the chaincode which runs on both peers of the organization, thereby changing performances for latency and throughput. EHR chaincode executes and creates the transaction. Thereafter, the transactions are endorsed and broadcasted to other peers before committing.

Fig. 6 indicates the average latency in seconds. It indicates the latency for querying, and writing success rate of transactions in 1org1peer is lower than that in 2org1peer. Also, as is usual, latency increased as the system scaled up with more organizations and more peers. Fig. 7 shows the throughput against transaction rate. Throughput of 1org1peer network is measured as highest at 186, whereas it is very low for 2org1peer (174). Fig. 8 shows the time under various rounds for successfully completing the transaction. 1org1peer takes about 120 s to reach 2000 transactions where, at the same time, 2org1peer completes 1000 transactions. So it is observed that the transaction time often increases with growth in the organization and peers. Block size is configured per channels. With almost the same condition and varied block size of 10 and 20, we obtain the results as follows. From Fig. 9, we found that more spikes are marked with block size 10, i.e., more number of transactions per second compared to less number of transactions with block size 20. As

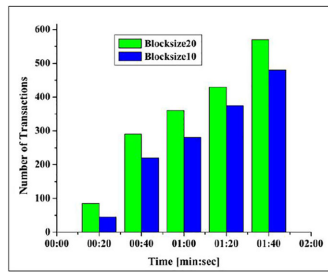


Fig. 9. Effect of blocksize with transactions.

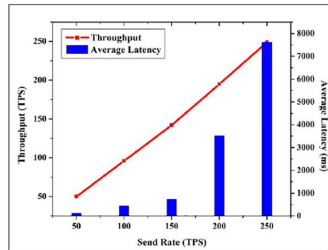


Fig. 10. Throughput and latency for read.

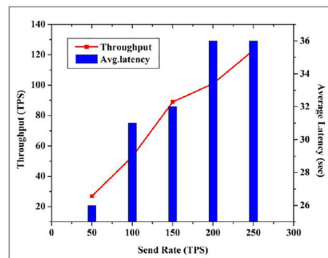


Fig. 11. Throughput and latency for write.

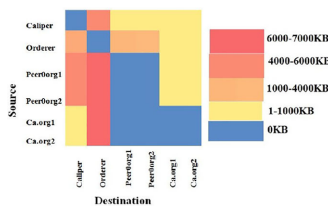


Fig. 12. Heatmap of network traffic in KB.

depicted in Figs. 10 and 11, we found that reading or querying is much faster compared to writing a transaction. The average write latency is 36.23 s, whereas the read latency is 7.62 s. Throughput of read operation is more than that of write operation.

Fig. 12 depicts a communication heatmap between the nodes in the network. Based on the simulation outcome, it can be observed that the data volume transfer between caliper to other peers is high because caliper creates and endorses the transactions. No traffic is represented by the blue fields including the diagonal, and all interpeer communications. There is minimal communication between the peers and the certificate authority, about 10 KB in total. Similarly, the orderer broadcasts the blocks to all the peers resulting in the average traffic of 6.75 MB and org

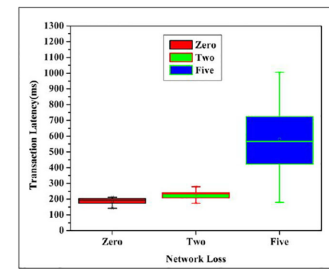


Fig. 13. Transaction latency with varying network loss.

1's peer 0 has an outgoing average traffic of 2.8 MB. Multiple rounds of executions of this measurement cause different distributions of traffic. Fig. 13 depicts that the changes in the number of clients and other network interfaces have a direct impact on the performance of the framework. The box plot represents that the transaction latency increases from 185 to 227 to 580 ms (mean values) with leading to losses of 0%, 2%, and 5%, respectively.

VI. CONCLUSION

This article presented a patient-centric framework for a blockchain-enabled healthcare system that not only addresses the issues of data privacy, authentication, and immutability but also presents a detailed plan for deployment and implementation of the proposed scheme along with a performance study. The implementation was done using the hyperledger platform. Performance evaluation and resource utilization of the proposed model was done using hyperledger caliper. The performance results obtained for parameters like transaction latency, throughput, memory utilization, and CPU utilization and its salient observations indicate that average write latencies exceed read counterparts by around 47% while maximum throughput was scaled at 186 TPS with about a 10 TPS decrease while moving from One-org1peer to Two-org2peer network. It was also observed that a twofold increase in block size led to around tenfold increase in TPS. The client communication through REST server is an added feature in the framework presented. This is a new capability demonstrated in the blockchain-based healthcare domain with the promise to revolutionize next-generation EHR frameworks. In future, the authors aim to extend the work Kafka and practical byzantine fault tolerance ordering services with fault tolerance.

REFERENCES

- [1] M. Holbl, M. Kompara, A. Kamisalic, and L. N. Zlatolas, "A systematic review of the use of blockchain in healthcare," *Symmetry*, vol. 10, no. 10, pp. 1–22, Oct. 2018.
- [2] Z. Liu *et al.*, "A survey on blockchain: A game theoretical perspective," *IEEE Access*, vol. 7, pp. 47615–47643, 2019.
- [3] K. R. Choo, Z. Yan, and W. Meng, "Blockchain in industrial IoT applications: Security and privacy advances, challenges and opportunities," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 4119–4121, Jun. 2020.
- [4] A. A. Siyal, A. Z. Junejo, M. Zawish, K. Ahmed, A. Khalil, and G. Soursoo, "Applications of blockchain technology in medicine and healthcare: Challenges and future perspectives," *Cryptography*, vol. 3, no. 1, pp. 1–16, 2019.
- [5] C. Esposito, A. De Santis, G. Tortora, H. Chang, and K. R. Choo, "Blockchain: A panacea for healthcare cloud-based data security and privacy?" *IEEE Cloud Comput.*, vol. 5, no. 1, pp. 31–37, Jan./Feb. 2018.

- [6] P. Pasquale *et al.*, "An edge computing, Internet of Things, and Big Data analytics applications for healthcare Industry 4.0," *IEEE Trans. Ind. Informat.*, vol. 15, no. 1, pp. 454–456, Jan. 2019.
- [7] A. Celesti, O. Amft, and M. Villari, "Guest editorial special section on cloud computing, edge computing, Internet of Things, and Big data analytics applications for healthcare Industry 4.0," *IEEE Trans. Ind. Informat.*, vol. 15, no. 1, pp. 454–456, Jan. 2019.
- [8] L. Ismail, H. Materwala, and S. Zeadally, "Lightweight blockchain for healthcare," *IEEE Access*, vol. 7, pp. 149935–149951, 2019.
- [9] Y. Zhang, Z. Zheng, H. N. Dai, and D. Svetinovic, "Blockchain for industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3514–3515, Jun. 2019.
- [10] N. Griggs, O. Ossipova, C. P. Kohlhos, A. N. Baccarini, E. A. Howson, and T. Hayajneh, "Healthcare blockchain system using smart contracts for secure automated remote patient monitoring," *J. Med. Syst.*, vol. 42, no. 7, 2018, Art. no. 130.
- [11] H. Wang and Y. Song, "Secure cloud-based EHR system using attribute-based cryptosystem and blockchain," *J. Med. Syst.*, vol. 42, no. 8, 2018, Art. no. 152.
- [12] X. Zhang and S. Poslad, "Blockchain support for flexible queries with granular access control to electronic medical records (EMR)," in *Proc. IEEE Int. Conf. Commun.*, Kansas City, MO, USA, 2018, pp. 1–6.
- [13] R. Guo, H. Shi, Q. Zhao, and D. Zheng, "Secure attribute-based signature scheme with multiple authorities for blockchain in electronic health records systems," *IEEE Access*, vol. 6, pp. 11676–11686, 2018.
- [14] G. G. Dagher, J. Mohler, M. Milojkovic, and P. B. Marella, "Ancile: Privacy-preserving framework for access control and interoperability of electronic health records using blockchain technology," *Sustain. Cities Soc.*, vol. 39, pp. 283–297, May 2018.
- [15] E. Daraghmi, Y. Daraghmi, and S. Yuan, "MedChain: A design of blockchain-based system for medical records access and permissions management," *IEEE Access*, vol. 7, pp. 164595–164613, 2019.
- [16] P. Sajana, M. Sindhu, and M. Sethumadhavan, "On blockchain applications: Hyperledger fabric and Ethereum," *Int. J. Pure Appl. Math.* vol. 118, no. 18, pp. 2965–2970, 2018.
- [17] H. Malik, A. Manzoor, M. Ylianttila, and M. Liyanage, "Performance analysis of blockchain based smart grids with ethereum and hyperledger implementations," in *Proc. IEEE Int. Conf. Adv. Networks Telecommun. Syst.*, 2019, pp. 1–5.
- [18] M. Valenta and P. Sandner, "Comparison of Ethereum, hyperledger fabric and corda," *Blockchain Center*, pp. 1–8, Jun. 2017.
- [19] A. A. Omar, M. S. Rahman, A. Basu, and S. Kiyomoto, "Medibchain: A blockchain based privacy preserving platform for healthcare data," in *Proc. Int. Conf. Secur. Privacy Anonymity Comput. Commun. Storage*, Springer, Cham, 2017, pp. 534–543.
- [20] J. J. Hathaliya, S. Tanwar, S. Tyagi, and N. Kumar, "Securing electronics healthcare records in healthcare 4.0: A biometric-based approach," *Comput. Elect. Eng.* vol. 76, pp. 398–410, 2019.
- [21] H. Huang, T. Gong, N. Ye, R. Wang, and Y. Dou, "Private and secured medical data transmission and analysis for wireless sensing healthcare system," *IEEE Trans. Ind. Informat.*, vol. 13, no. 3, pp. 1227–1237, Jun. 2017.
- [22] B. Liu, M. Liu, X. Jiang, F. Zhao, and R. Wang, "A blockchain-based scheme for secure sharing of X-ray medical images," in *Proc. Int. Conf. Secur. Intell. Comput. Big-Data Serv.*, Springer, Cham, Dec. 2018, pp. 29–42.
- [23] K. Fan, S. Wang, Y. Ren, H. Li, and Y. Yang, "Medblock: Efficient and secure medical data sharing via blockchain," *J. Med. Syst.* vol. 42, no. 8, 2018, Art. no. 136.
- [24] H. Wang and Y. Song, "Secure cloud-based EHR system using attribute-based cryptosystem and blockchain," *J. Med. Syst.*, vol. 42, no. 8, 2018, Art. no. 152.
- [25] M. A. Uddin, A. Stranieri, I. Gondal, and V. Balasubramanian, "A patient agent to manage blockchains for remote patient monitoring," *Stud. Health Technol. Inf.*, vol. 254, pp. 105–115, 2018.
- [26] S. Tanwar, K. Parekh, and R. Evans, "Blockchain-based electronic healthcare record system for Healthcare 4.0 applications," *J. Inf. Secur. Appl.*, vol. 50, 2020, Art. no. 102407.
- [27] P. Bhattacharya, S. Tanwar, U. Bodke, S. Tyagi, and N. Kumar, "BinDaaS: Blockchain-based deep-learning as-a-service in Healthcare 4.0 applications," *IEEE Trans. Netw. Sci. Eng.*, to be published, doi: [10.1109/TNSE.2019.2961932](https://doi.org/10.1109/TNSE.2019.2961932).
- [28] B. Shen, J. Guo, and Y. Yang, "MedChain: Efficient healthcare data sharing via blockchain," *Appl. Sci.*, vol. 9, no. 6, 2019, Art. no. 1207.



Akhilendra Pratap Singh (Member, IEEE) received the B.Tech degree in computer science & engineering from Uttar Pradesh Technical University Lucknow, Uttar Pradesh, India, in 2006, the M.Tech. degree in information security from Motilal Nehru National Institute of Technology Allahabad, Uttar Pradesh, India, in 2011, and the Ph.D. degree in information technology from Indian Institute of Information Technology Allahabad, Uttar Pradesh, India, in 2017.

He is currently an Assistant Professor with the Department of Computer Science and Engineering, National Institute of Technology Meghalaya, Shillong, India. He has authored or coauthored a large number of various research papers in international and national journals and conferences of high repute. His research interests lie in service-oriented computing, wireless sensor network, network security, network forensics, and machine learning.

Dr. Singh is associated with more than 10 international journals of repute as reviewer.



Nihar Ranjan Pradhan (Student Member, IEEE) received his B.Tech in information technology, in 2005 and M.Tech. in computer science & engineering, in 2011 from GIET University, Odisha, India. He is currently working toward the Ph.D. degree with Department of Computer Science and Engineering, National Institute of Technology, Meghalaya, India.

His research area of interest includes the field of blockchain technology, wireless sensor network, and security.



Ashish K. Luhach (Member, IEEE) received the Ph.D. degree in computer science from the Department of Computer Science, Banasthali University, Rajasthan, India, in 2015.

He is currently a Senior Lecturer with the Papua New Guinea University of Technology, Lae, Papua New Guinea. He has more than a decade of teaching and research experience. He also worked with various reputed universities and also holds administrative experience as well.

He has authored or coauthored more than 80

research papers in reputed journals and conferences, which are indexed in various international databases.

Dr. Luhach is a member of CSI, ACM, and IACSIT. He has also edited various special issues in reputed journals and is an Editor/Conference Cochair for various conferences. He is also an editorial board member of various reputed journals.



Sivansu Agnihotri (Member, IEEE) received the M.Tech. degree in computer science from the Department of Computer Science and Engineering, National Institute of Technology, Meghalaya, India, in 2020.

His research interests include blockchain technology and network.



Noor Zaman Jhanjhi (Senior Member, IEEE) is currently an Associate Professor with Taylor's University, Penang, Malaysia. He has edited/authored more than 20 research books with international reputed publishers, earned several research grants, and has a great number of indexed research articles on his credit. He has supervised several postgraduate students, including master's and Ph.D degrees in IT UTP, Malaysia, in 2014.

Dr. Jhanjhi is an Associate Editor for *IEEE Access*, Moderator of *IEEE TechRxiv*, Keynote Speaker for several IEEE international conferences globally, External Examiner/Evaluator for Ph.D. and master's degrees for several universities, Guest Editor of several reputed journals, member of the editorial board of several research journals, and active TPC member of reputed conferences around the globe. Recently, he has been recognized among the top 1% of reviewers globally by WoS/ISI (Publons) for the year 2019.



Sahil Verma (Member, IEEE) received the B.Tech and M.Tech degrees from Maharishi Markandeshwar University, Mullana, India, in 2012, and the Ph.D. degree from Jaipur National University, Jaipur, India, in 2017, all in computer science and engineering.

Some of his research findings are published in top-cited journals such as IEEE, ACM, Elsevier, Springer, and Wiley and various international conferences of repute. He has many research contributions in the area of cloud computing, Internet of Things, and vehicular *ad hoc* networks, WSN, MANET, etc.

Dr. Verma is a member of ACM and IAENG, and editorial board member of many international journals. He has chaired many sessions in reputed international conferences abroad and in India.



Kavita (Member, IEEE) received the B.Tech and M.Tech degrees from Maharishi Markandeshwar University, Mullana, India, in 2012, and the Ph.D. degree from Jaipur National University, Jaipur, India, in 2018, all in computer science and engineering.

Some of his research findings are published in top-cited journals such as as IEEE, ACM, Elsevier, etc. She has many research contributions in the area of cloud computing, Internet of Things, and vehicular *ad hoc* networks, WSN, MANET, etc.

Dr. Kavita is a member of ACM and IAENG, and editorial board member of many international journals.



Uttam Ghosh (Senior Member, IEEE) received the Ph.D. degree in electronics and electrical engineering from the Indian Institute of Technology Kharagpur, West Bengal, India, in 2013.

He is an Assistant Professor of Practice with the Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN, USA. He has Postdoctoral experience at the University of Illinois at Urbana-Champaign, Champaign, IL, USA, Fordham University, New York, NY, USA, and Tennessee

State University, Nashville, TN, USA. He has authored or coauthored 50 papers at reputed international journals including IEEE Transactions, Elsevier, Springer, IET, Wiley, InderScience, and IETE, and also in top international conferences sponsored by IEEE, ACM, and Springer. His main research interests include cybersecurity, computer networks, wireless networks, information-centric networking, and software-defined networking.

Dr. Ghosh is a member of AAAS, ASEE, ACM, and Sigma Xi. He is actively editing two edited volumes on Emerging CPS, Security, Machine/Machine Learning with CRC Press, Chapman Hall Big Data Series.



Diptendu Sinha Roy (Senior Member, IEEE) was born in India. He received the B.Tech. degree in electronics and communications engineering from Kalyani University, Kalyani, India, and the M.Tech. degree in computer science and the Ph.D. Eng. degree from Birla Institute of Technology, Ranchi, India, in 2005, 2010, respectively.

He is currently an Associate Professor with the Department of Computer Science Engineering, National Institute of Technology, Meghalaya, India. His current research interests include software reliability, grid

computing, and reliability analysis.