SHealth: A Blockchain-Based Health System With Smart Contracts Capabilities

MANAF ZGHAIBEH, UMER FAROOQ, NAJAM UL HASAN, AND IMRAN BAIG, (Senior Member, IEEE)
Department of Electrical and Computer Engineering, Dhofar University, Salalah 211, Oman
Corresponding author: Manaf Zghaibeh (mzghaibeh@du.edu.om)

ABSTRACT Governments rely on collected data to analyze their citizens’ health requirements and needs. Such data is usually of scattered manner gathered by several agencies and entities that fall under different business models and goals. Hence, the scattered nature of this data creates a burden on governments in precisely planning adequate health bills that guarantee the wellbeing of their citizens. Furthermore, the issue of synchronizing and having a real-time access to such data exacerbates the problem even more. Therefore, a system that guarantees the integrity, accessibility, correctness and security of medical data in a highly synchronized environment is crucial for governments to accurately project their future needs. In this paper, we present Smart-Health (SHealth), a blockchain-based health management system. In its crux, SHealth is a private multi-layered blockchain integrated with a multi-tiered addressing scheme that defines the privileges and permissions of entities in the system. Blockchain-based systems assure security, reliability, availability, resistance against tampering and malicious attacks, seamless integration, and easy data management. SHealth proposes a complete autonomy to its users. Through a user-friendly graphical interface it capitalizes on smart contracts to initiate various requests and inquiries pertaining to patients such as appointments, medical tests, medications, medical procedures, or history. SHealth is simple, robust, efficient, secured, and completely automated. All the stakeholders in the system can access the health related data stored in a distributed database without compromising its authenticity. SHealth covers all the possible scenarios in health systems from which some of them are explained in this work.

INDEX TERMS Blockchain, health records, asymmetric cryptography, multisignature, smart contracts, consensus mechanism.

I. INTRODUCTION
In order to optimize the health bill, governments require input data to analyze the needs and requirements of their citizens [1]. In many countries such data are usually available in a scattered manner and hosted by several entities such as hospitals, medical centers and doctor offices for different purposes and goals. Hence, there exists a gap between what the government perceives regarding the true and exact needs of its citizens and their actual health requirements. This gap precipitates in governments having unrealistic projections when preparing their health bills and expenditures in order to provide a dignified level of comfort and assistance to its people including but not limited to medications, medical procedure, hospital care, inoculation, or vaccination. Moreover, governments usually tend to analyze certain diseases or illnesses and the magnitude of their presence and spreading within specific age, sex, or geographical area. However, the absence of a unified national database that holds the comprehensive records of patients circumvents such attempts and efforts. Furthermore, having the records scattered and distributed without any mean of compiling them at a central location impedes this process especially with limitations in synchronization and real-time access. As a result of that, a national database that contains the health records of the residents of the country is imminent to establish in order to facilitate the aforementioned causes.

Such databases had existed in several countries around the world under different implementations and purposes. Some were created and managed by independent health insurance providers to track the health cost of their members [2]. On the other hand, in countries where universal health care is being provided by the government, those databases are commonly created by the health provider, i.e., ministry of health or...
department of health. However, the scattered nature of those databases with the persisting problems of inter-connectivity, sharing, synchronization and most importantly privacy limit the optimized utilization of such databases [3], [4]. This issue even aggravates knowing that the access to those databases is only limited to entities that fall under the dome of national provider such as hospitals and public medical centers, whereas, health providers that operate privately may establish and maintain their own databases of records.

From another point of view, in many current health system models, the individual who is the basic brick of the health system is being kept out of the loop in terms of accessing his records freely [5]. Furthermore, this is related to the scattered nature of the patient’s records with different health providers which makes it inconvenient and cumbersome to compile or gather. Moreover, in some scenarios, providers may deny the individual from accessing his complete medical history under different arguments and pretenses.

On another venue, the issue of privacy and security has always paralleled the handling of medical records of patients. Recently, the advancement in sensors, low power devices and better communication systems have changed the way we perceive about modern health care systems. Smart fitness devices are rapidly becoming an integral part of our daily lives. The amount and nature of data handled by various sensors and devices in modern health care systems is really of a very critical nature. Due to the sensitive nature of health related data, access to such critical data must be limited to the individuals and entities that are directly associated with them, consequent to request and approval. Because unauthorized access to the personal health data may expose the concerned person or group of people to potentially life threatening situations [6], [7]. Integrity of the health related data is another important aspect. A compromise on the integrity of the data might lead to false conclusions eventually endangering the life of a person [8]. Hence, the scattered structure of the health related data at different entities which adopt different security measures make the efforts of guarding them against exposure and tampering futile especially with the advent of ever more sophisticated security attacks on IoTs related to healthcare systems [9]–[12].

Blockchain technology has emerged in the last decade and has profoundly addressed concerns such as availability, distribution, trust, and privacy [14], [15]. This technology has ignited research in different platforms to implement its fascinating characteristics [30]. It has circumvented the hurdle of the trust that undermined the performance of trustless distributed-nature systems by providing a tamper-proof and time-stamped record keeping properties. In addition to cryptocurrency, the technology has been adopted in several sectors such as but not limited to education [16]–[18], energy [19]–[21], governance [22]–[25], and healthcare [13], [26]–[29].

This paper presents Smart-Health, a framework for a comprehensive blockchain-based health system that encloses the patient and all entities involved in providing him with health services such as health providers and health insurance partners.

Smart-Health (SHealth) is a private blockchain hosted and maintained by different entities that comprise the health management system. This private blockchain is a concatenated database, distributed ledger, that contains the records of all users and is securely accessible and available to all entities in the system [30], nodes [31]. Each user in the system has a unique and private cryptographic key that is maintained by an application, a wallet, installed on the user’s smart phone or tablet. With this cryptographic feature, the wallet enables the user to access his records in a read mode without the need to obtain any consent from other nodes. In SHealth there are two types of nodes, a health provider and a health partner. A health provider is a licensed node in the system that offers medical services. On the other hand, a partner is a node such as a health insurance provider that provides the means of record keeping. The user in SHealth grants access using his wallet to provider nodes that request his records. In case a new medical entry has to be appended to the user’s records in the system by a provider node, both of the user and his partner, i.e., a health insurance node, must jointly approve the new entry. Each node in SHealth, a provider or a partner, must obtain a complete updated copy of the blockchain. Moreover, each node has its own address that represents its identity in the system which is consequently derived from its own private cryptographic key.

Throughout this paper we will demonstrate the following:

- SHealth is a lightweight and fully automated health management system. It comprises of nodes that maintain the blockchain, clients that hold a fraction of the blockchain, and IoT interfaced instruments that listen to requests from the network and report results back to it. All appointments, requests, inquiries can be initiated and communicated digitally in the system by entities using a user-friendly graphical interface.
- SHealth capitalizes on the versatility of smart contracts hence adding another level of automation to the system and reducing the processing overhead at the participating nodes. Although the network will be handling smart contracts generated by large number of entities, however, the short-term life of each smart contract reduces the overhead of the network [32], [33].
- SHealth is a modular multilayered system with each layer having specific privileges and functionalities. Hence, adding or removing entities to any layer in the system is an entirely independent procedure and does not affect its performance.
- The modular nature of SHealth makes it compatible with the Hyperledger Fabric [34]. That is, SHealth itself can be considered as another module of a more universal hyperledger system. The system can work alongside any other system that is part of hyperledger such as driving licenses, passport, education, and several other systems that may comprise a national hyperledger.
The remaining of the paper is organized as follows: In section II we present an overview of the proposed system. Section III discusses the major scenarios in SHealth. Related work is discussed in section IV and we conclude our paper in section V.

II. SYSTEM OVERVIEW

In this section we introduce the architecture of our proposed SHealth system and present some of the terminologies that are adopted in its paradigm.

Figure 1 portrays an abstraction of this blockchain-based system where its architecture has been segregated into four distinctive layers.

The first layer is the government layer. As the name suggests this is the highest authority in this system. The fundamental role of this layer is to govern the access to the blockchain by 1) sanctioning new nodes 2) and admitting new users to the network based on their national identifiers. Sanctioning a new node is an on-site bureaucratic process that inaugurates the node as a licensed unit in the system with its own SHealth identification, i.e., address in the blockchain. All functioning nodes that had been previously sanctioned will reside in the government layer and hence considered as government nodes. The term node denotes health providers and partners. A “provider” represents any health providing entity such as hospitals, medical centers, doctor offices, and pharmacies. On the other hand, the term “partner” refers to the entity that provides the financial means or support for the treatment of its clients such as a national health coordinator or a health insurance company. A partner in SHealth is responsible for updating its clients’ records in the blockchain and maintaining their actual health records in its database, the shadow network. Every record in the partner’s database is labeled with its client’s multisignature address with that partner and linked to the client’s address in the blockchain.

Furthermore, admitting new users constitutes issuing them SHealth addresses based on their government’s issued identifications. According to that, the government layer will be in direct contact with other non-SHealth government agencies that are responsible for maintaining the National Identifier Database, i.e., (such as a Social Security Number or Social National Number). For instance, when parents are registering their newborn in SHealth system, they will have first to submit an application containing the newborn’s unique national identifier which was obtained initially from the concerned government bureau. Consequently, SHealth’s government layer will have to validate the records provided in the application with the government agency responsible for maintaining the National Identifiers Database.\footnote{Since the government layer has to access the National Identifier Database in a read-only mode, it could be perceived as a node in another blockchain that holds the government’s national identifiers.}

In SHealth, sanctioned and operational nodes maintain a full copy of the blockchain and can sanction new nodes into the system, admit new users to the network, access the blockchain to retrieve records of patients based on their addresses in the blockchain, and initiate/execute smart contracts in the blockchain.

The second layer in our system is the users layer. The term “user” signifies any individual who has a personal record of registration in the government layer either as a care receiver: a client, or as a care giver: a personnel who works at the node such as a doctor, a nurse, a lab technician, or a manager. Each user is identified in the blockchain by his unique address.

Additionally, SHealth clients are issued 2-of-2 multisignature addresses in the system that hold their medical transactions (for further details, see section II-C). A SHealth client does not have a sole authorization to modify his records in the blockchain, yet, he can associatively append new transactions to his records with his partner when requested by the provider as we will discuss later.

Users communicate with the system using an API, the SHealth Wallet, which is an application that is available to them from trusted SHealth locations such as providers, partners, or related websites. The wallet has several functions that include holding the user’s keys, initiating requests, and viewing records.

The third layer in SHealth is the IoT terminals layer. It hosts the sanctioned IoT medical devices and instruments that operate within the provider’s node and run specific medical/health procedures for the clients such as hematology, ultrasound, X-ray, MRI, or others. They are attached to the system via IoT interfaces to report the results of their conducted procedures. To sanction an IoT apparatus in SHealth, its provider has to register it in the government layer first, hence, generating it a unique SHealth identity. Devices in this layer are considered as light nodes, Simple Payment Verification (SPV) clients. They listen to the transactions in the network and operate...
when they are addressed directly by nodes through smart contracts as we will explain later, thus, they do not download the complete blockchain.

The last layer in the SHealth system is the blockchain itself. While the actual records of the clients are being kept in the shadow network at their partners side, SHealth blockchain holds only pointers to those transactions. Each confirmed transaction is registered at the client’s addresses in the blockchain.

Similar to other blockchain-based applications and systems, SHealth incorporates the basic terminologies that are implemented in those systems. The following subsections will shed the light on the essential themes that we adopt in SHealth.

A. ADDRESSES AND TIERS
Privileges and responsibilities of entities in SHealth differ based on their address class. A tier 1 node is a core part of the government layer in the sense that it can sanction new nodes and users. Partners in SHealth are designated as tier 1 nodes. In addition to that, providers such as major medical centers can be designated as tier 1 nodes as well. However, the difference is that each partner will be maintaining its own shadow network, i.e., its clients’ actual database, whereas the blockchain will be maintained globally by all nodes.

Tier 2 nodes represent providers such as doctor offices, clinics, or pharmacies. Such nodes can admit users to the system and participate in the verification and approval of transactions. Hence, they hold a full copy of the blockchain, however, they do not take part in sanctioning new nodes.

A tier 3 address covers all users in general. Those addresses belong to clients or service giver such as patients, doctors, nurses, lab technicians, accountant, or managers. They do not have any sanctioning privileges neither they take part in the consensus mechanism. Yet, they participate in retrieving records and initiating and executing smart contracts.

A tier 4 address is basically a 2-of-2 multisignature address. This class of addresses belongs to the records that hold the clients’ transactions in the blockchain. Those addresses are co-signed by the clients and their partners. Finally, tier 5 addresses are reserved for the IoT terminals. They are embedded in the smart contracts which are triggered by the IoT-ready medical instruments.

B. ASYMMETRIC CRYPTOGRAPHY
Generating cryptographic addresses is known as public-private key generation or asymmetric cryptography. In this process, the system randomly generates a private key that should be, as the name implies, stored privately and in secrecy. The private key is used for creating the digital signatures required to prove the ownership of records: applying the private key to a transaction to generate a numerical signature. Furthermore, the private key is also used to decrypt messages that were encrypted with its public key.

On the other hand, the public key is derived from the private key and is known to other entities besides its owner. It is used by other entities to encrypt messages that are addressed to its owner. Upon reception, the addressee decrypts the received message using his private key. The public key is also used as an address in the system, hence, transactions in the network could be directed to the user’s public key. However, for added security, it is a good practice not to use the public key as an address but rather to generate addresses from the public key using a hashing algorithm.

One key characteristic of this method of encryption is that it is virtually impossible to generate the private key from the public key. Thus, the privacy of users in the system is perfectly maintained. On the other hand, if the private key was compromised, any person or entity would have access to the records of its user.

As an example, suppose that Alice is visiting a doctor’s office for a checkup. She signs a request using her private key and sends it through the network to the node she is visiting. The node verifies her signature with Alice’s public key and hence retrieves her previous records from the blockchain if needed. After the checkup, both of the node and Alice digitally sign the new visitation information with their keys to be appended to the blockchain.

C. MULTISIGNATURE SCHEME:
In Bitcoin [35], Ethereum [36], and other cryptocurrencies, the user could hold his coins either in his account at the broker’s site or at his own personal digital wallet. In either case, when the user is about to receive some digital coins from another party, all what he has to do is to provide the sender with his address in the blockchain which is basically a 26 to 35 alphanumeric string, the public-key hash, without revealing any of his personal information. Alternatively, when the user wishes to send some of his digital coins to another person, he has to lock his transaction with a locking script that determines the conditions of releasing the transferred funds. Then, the user forwards the transaction to his next-hop neighbors in the network. The recipient, who is addressed in the transaction by his own public key hash, can consequently decrypt the transaction using the unlocking script that satisfies the conditions of the locking script and hence obtain the transferred funds. The terms locking and unlocking scripts are the formal expressions for what is generally known as encrypting and decrypting the transaction respectively. Hence, the sender locks the transaction by encrypting it using the client’s public key who in his turn unlocks the transaction by providing a combination of his public and private keys.

This scheme of transactions between parties is one of the standard scripting defined in the Bitcoin protocol and the most commonly used type, Pay to Public Key Hash: P2PKH. It is considered a 1-of-1 signature script indicating that only one signature is required to encrypt the transaction and another signature is required to decrypt it.

Since we are dealing with sensitive data such as medical records, privacy is an issue with the utmost importance in this
matter. Furthermore, having one entity in the system possessing a complete ownership over such private data is repudiated even if that entity is the owner of the data himself. Rationally, the user should not be allowed to alter his records under any circumstances. The significance of adopting blockchain is that it does not allow any party to alter the previously appended records. However, the system allows users to read and to append new information to the records upon approval. Yet, when dealing with medical records, it should not be permissible to any one entity (a patient or a doctor) to modify the patient’s record exclusively. As a result of that, a multisignature scheme is also required in SHealth in addition to the P2PKH signature scheme.

In this regard, Bitcoin and Ethereum introduce other fascinating functionalities of involving multiple users in cashing one particular fund, i.e., multiple users share one specific account and must provide approval when some funds arrive to their account [30], [37]. One of the schemes is called the multisignature or the m-of-n signature script. Here, ‘n’ denotes the number of parties who can sign the transaction and ‘m’ denotes the minimum number of parties required to make the transaction valid. For instance, if Alice, Bob, and Carol adopted a 2-of-3 signature scheme among themselves, then it takes any two of them to jointly sign the transaction to make it valid. For example, Alice and Bob may sign the transaction without involving Carol and the network will accept it. Hence, the locking script is the requirement that two out of three parties are needed to accept the transaction while the unlocking script is presenting the approval of two of those parties.

As previously mentioned, SHealth adopts a 2-of-2 multisignature blockchain addressing scheme for its clients. The two co-signees are the client himself and the partner. Fundamentally, only after the multisignature process, the transaction can be accepted by the blockchain. On the other hand, providers and partners will be assigned P2PKH addresses in the blockchain. However, each entity (user or node) will be issued a single private-public key pair.

D. WHAT DOES THE BLOCK CONTAIN?
SHealth blocks contain all transactions that took place in the network. The word transaction here refers to any 2-of-2 signed record that has been created for a particular client in the SHealth system such as a doctor visitation, a surgical procedure, or a prescription. Our vision for the system is to hold all of those medical records that the client had previously had. For example, a patient’s x-ray image, MRI image, or a pharmacy prescription will be digitally saved in his record in the chain. However, all those records when digitized as data will massively augment the size of the chain. Hence, it is more efficient to append pointers to those records rather than appending them to the blockchain. Therefore, in SHealth, while the main blockchain hosts only pointers about the client’s activity in the network, it deploys a parallel chain of records where each record belongs to a particular client and holds all of his actual digitized data.

For example, when a new provider node joins the network it has only to download the blockchain that contains the pointers of the clients’ records. Consequently, when a patient visits a provider for a medical procedure, the provider can obtain the client’s entire history by downloading his records from the parallel network through the associated partner.

The integrity of the records in the parallel network is also maintained since each record is labeled with its own cryptographic key and is stamped with the 2-of-2 signing script. In addition to that, after appending new data fields to a record in the parallel network, a Merkle root will be generated for the record and will be appended to the transaction in the main blockchain [38]. Therefore, when a node has to access the record of a given client, it downloads its digitized history from the shadow network using his address in the network and the 2-of-2 signing script. The node then checks the integrity of the digitized record by calculating its Merkle root value and matching it to the field saved in the blockchain. At this point, after the client gets his treatment from the medical center he is visiting, he and his partner will have to append his visitation to the blockchain and to the shadow network. Therefore, any procedure that the user had undergone in the node will be digitized and appended to his downloaded record along with the newly calculated hash value.

Accordingly, the two parties will have to co-sign the transaction which contains the indicators about the visit, the timestamp, and the Merkle root value that is stored in the record in the shadow network. After signing the transaction, it will be broadcasted to the network for approval. As a result of that, the blockchain holds the record of last visitation with the most recent calculated hash value of the digitized data. Hence, the updated record will be uploaded to its place in the shadow network.

If the calculated hash does not match what had been recorded in the blockchain, the record is assumed to be tampered with. Thus, in case the shadow record had been altered maliciously, the system would discover it.

The shadow network is possibly implemented as a concatenated data structures following the paradigms of DHTs. For instance, in a Chord-like shadow network each record will be accessed by its address in the network and the 2-of-2 signing script. The node then checks the integrity of the digitized record by calculating its Merkle root value and matching it to the field saved in the blockchain. At this point, after the client gets his treatment from the medical center he is visiting, he and his partner will have to append his visitation to the blockchain and to the shadow network. Therefore, any procedure that the user had undergone in the node will be digitized and appended to his downloaded record along with the newly calculated hash value.

E. SMART CONTRACTS:
The concept of smart contracts was originally introduced in 1994 suggesting to code contracts, bonds, and legal documents and run them in computers for execution, i.e, a bank mainframe [40]. Before delving into the need for smart contracts in our system, let us imagine how a patient’s visit to the medical center will pan out in recent days without blockchain.
implementation. Consider a patient with flu-like symptoms visiting a doctor’s office for checkup. The office will register the patient after getting his credentials and retrieve his local medical history if available. After the examination, the doctor determines that the patient requires further tests and hence refers the patient to the nearby hospital in order to undergo the suggested tests. Consequently, the patient visits the hospital, registers himself, waits for his papers to be processed, then has the suggested tests done. The doctor receives the test results and accordingly prescribes the patient some medication from the local pharmacy in town. Accordingly, the patient will go to the pharmacy, submit his prescription and identification, and obtain his medication.

In the above scenario we deliberately ignored the role of the insurance partner. However, we can fathom how involving another entity in the above procedure might complicate the processing, the approvals, and the amount of paper work associated with a regular visit. In real life this scenario is not that difficult yet it is, as we all might have experienced, a bit stressful.

What blockchain technology promises us is to make trustless networks run in decentralized, transparent, secure, fast and high availability manner. Additionally, this technology throws another advantage on top of its architecture that will make its functionalities even more systematic: smart contracts [41]–[43].

A smart contract is basically an executable script that resides on top of the blockchain that will be executed only when certain conditions are being met or if it has been triggered by an external element [41]. Hence, the smart contract itself has its own address in the network. To execute the smart contracts by triggering, nodes in the network issue transactions addressed to those contracts directly [44]. The elegance of smart contracts is that they basically allow us to write any logic in code while setting the conditions and expected outcomes for that logic.

Let us go back to the previous example and elaborate more on how it would be implemented in SHealth.

First, the patient visits the doctor’s office providing his credentials through the API installed on his handheld device. The node along with the patient’s partner, accepts his request and retrieves his information from the network through the multisignature script. After the registration, the doctor examines the patient who suffers flu-like symptoms. Here, the doctor requests the patient to have a blood test to determine whether the symptoms are related to a bacterial or a viral infection. This request is written as a smart contract and placed in the system having its own address as follows:

1) Admit the patient.
2) Perform the blood test.
3) Read the white cell count.
4) If the count is normal or low and the lymphocyte count is high, then:
   a) Comment: Viral infection, take antipyretic every 6 hours, Vitamin C twice a day.

b) Initiate a smart prescription triggered by the patient’s key with the following medications: antipyretic and vitamin C.

5) Elseif the white cell count is high and the granulocytes count is high, then:
   a) Comment: Bacterial infection, take antibiotic every 12 hours, take antipyretic every 6 hours.
   b) Initiate a smart prescription triggered by the patient’s key with the following medications: antibiotic and antipyretic.

The above smart contract sequence is executed by the clinic, the hematology instrument and the pharmacy. The patient upon his arrival to the clinic will add his signature to the smart contract with his API, thus, initiating his admission procedure. The nurse in the clinic will draw the blood sample and place it in the IoT-ready hematology instrument. The hematology instrument which is addressed in the smart contract will perform the analysis and consequently address the results to the pharmacy. Based on the results, a smart prescription is initiated and executed at the pharmacy by dispensing the medication.

Noticeably the above contract is primitive from a medical point of view. One key feature for smart contracts is that they should be deterministic and written to cover all possible outcomes. In our previous example however, one major outcome among others was not covered for the purpose of illustration which is when the results of the analysis turn out negative for the viral and bacterial infection.

Although the functionality of smart contracts is auspicious, however, a special care must be taken when implementing them. In permissionless blockchain, any person can join the network and hence access the smart contracts’ source code. Even though the code is only executable by the addressees in the contract, however, this still is considered as a threat and may become a tool for users with malicious intentions to undermine the system. Moreover, the immutability of smart contracts exacerbates the situation even more. That is, since the smart contract can not be changed after its deployment in the network, it is impossible to fix any later revelation regarding problems, security issues, loopholes or bugs. Therefore, an extensive effort must be put during the design of the smart contracts to enhance their security considering all previous known vulnerabilities, coding, security, privacy and performance issues [45].

Immense effort have been put in order to optimize smart contracts in permissionless blockchain and enhance their security and privacy [46]. Accordingly, several tools have been developed to eradicate coding and security issues associated with the design of smart contracts such as: SmartCheck [47], Oyente [48], Zeppelin [49].

In a permissioned network where access is limited to sanctioned users, the risk of having unauthorized use or access is abated. However, similar to any other online system, application, or service, opportunities for identity theft are present and can never be eliminated. Yet, certain measures can be taken
in order to minimize their risks such as associating access to the network with biometrics.

F. CONSENSUS MECHANISM

In blockchain networks, transactions are processed by participants and broadcast (flooded) through the entire network using some gossip protocol. All nodes in the network receive those transactions and pool them in their memories for later verification and packing. However, up to this point, those transactions are not considered as part of the blockchain, not until they get verified and confirmed. Here, the procedure of verification is based on predetermined conditions that are set by the network such as the correctness of the syntax of the transaction, its size, and its inputs and outputs.

The process that makes a transaction part of the blockchain is called mining and is only performed by nodes that hold the mining function. The concept of mining is that the mining node will select a group of transactions from the mining pool based on some priority criteria and gather them all in a single block. This block will hold a pointer to the last attached verified block in the chain, i.e., last block mined, which the mining node had already received and acknowledged from the network. At this point, the mining node will attempt to attach its block (with its selected transactions) to the chain. The incentive behind this process is to get rewarded by the network. For example, in Bitcoin the node which succeeds in attaching its mined block to the chain will be rewarded with 12.5 Bitcoins in addition to fees for its processed transactions. However, other mining nodes are also striving to attach their own blocks with their selected transactions to the chain in order to get rewarded. With this fierce competition among the nodes to get the block attached to the network, some sort of consensus among the nodes is required to manage this competition and to eliminate collusion. Bitcoin protocol lays out a robust policy in this area called the Proof of Work consensus mechanism PoW [50]. The essence behind this policy is that the node which is willing to attach the block to the chain must show that it has spent intensive effort in solving a complex mathematical puzzle. This puzzle can only be solved by trial and error method to find a 32-bit number, a nonce, in the block’s header to match a predetermined target set by the network. Consequently, with millions of possibilities to trial, this power-hungry intensive and expensive process provides a guarantee that the miner did not cheat or collude. After finding the nonce, the node will broadcast the mined block (with its transaction attached) to the network. Once other nodes receive the mined block, they will cease their current mining process, confirm the solved nonce in the received block, attach the block, and commence the next mining process.

Why such an intensive process is required has to do with the inherited P2P attacks. Imagine that the network does not require this mechanism to attach the next block but rather a consensus from the participating nodes, i.e., a voting process to select who will mine the next block. This opens the door widely for collusion amongst nodes, especially to that one particular attack which P2P networks are prone to: the Sybil attack [51]. A malicious node might affect the votes by accessing the system using multiple identities that cast their votes in favor of one of them.

A not-all-the-time practical solution to overcome the Sybil attack is to whitelist the users of the network. Having a permissioned network defies this attack since all participants are formally registered and approved beforehand and therefore no user can access the network with multiple identities [52].

SHealth is a private blockchain network where all entities, nodes and users, that participate in the network are whitelisted and sanctioned by the government layer. Rationally, the risk of malicious behavior in such private networks is considerably minimal comparing to other permissionless networks where there is no burden of entry on the users. Nevertheless, whether the blockchain is private does not entirely eradicate the need for a reliable consensus mechanism in the network.

Designating SHealth as a life-critical system and the nature of its transacted data mandate having the system accessible in reliable, accurate, and timely manner. This leads us in this section to address the issue of fault-tolerance and the ability of the system to accurately serve its purpose when some disruptive behavior is reflected by its components. What matters to us in such disruptive behavior is what characterized as Byzantine failures rather than fail-stop failures [53], [54]. The former is defined as a malicious, could be collusive, behavior reflected by some components of the system. The later defines failures affected by some components and detected by others leading to stopping the system, i.e., a malfunction.

Accordingly, SHealth is required to be a fault-tolerant system. In definition, a system is an f fault tolerant if it can operate correctly when no more than f failures occur concurrently [53]. This can be achieved by reaching a state machine system in which the processes of the system are replicated and run on each state of the distributed system starting at the same initial sate, executing the same requests and producing the same output. For Byzantine failures, 2f+1 copies of the processes are required for the system to operate reliably and correctly.

A perfect fit for SHealth consensus mechanism requirement is the Practical Byzantine Fault Tolerance consensus mechanism, PBFT [55], [56]. PBFT is the most prominent consensus mechanism for permissioned blockchains and is currently implemented in the modular hypeledger fabric [57]. This mechanism requires 3f+1 non-faulty nodes that will ensure the semantic of the system when f failures occur concurrently, where f is one third of the total number of nodes. In PBFT, for every block to be sealed and attached to the chain one node has to be elected for the task by at least two thirds of the nodes. That is, PBFT can only turn faulty when more than one third of the participants behave maliciously.

Other key characteristics of PBFT is that it requires low power consumption comparing to the power-hungry PoW.
mechanism and is less complex and non-discriminatory comparing to Proof-of-Stake PoS [36], [58]. Furthermore, PBFT does not mandate multiple confirmations once the transactions are approved which drastically reduces the overhead comparing to other consensus mechanisms.

Finally, in applications where participants are whitelisted and formally identified such as healthcare and national identification systems, the risk of having more than one third of the nodes exhibiting a Byzantine behavior diminishes [59]. Accordingly, implementing consensus mechanisms such as PoW and PoS becomes excessive and superfluous. Simply put, the price paid in adopting PBFT is undoubtedly insignificant than the cost of implementing resource-hungry mechanisms such as PoW and PoS.

### III. SCENARIOS

The proposed SHealth system covers all the possible scenarios of a health care management system. Some of those scenarios are discussed in this section. A summary of notations and their meanings used for the discussed scenarios is presented in Table 1.

#### A. A NEW NODE JOINS SHEALTH

The efforts for licensing and approving nodes into SHealth follow the bureaucratic procedures set by the entity which is responsible for overseeing and regulating their operation, i.e., a Ministry of Health or Department of Health in some countries.

After receiving its approval from the concerned bodies, a new node can be inaugurated into the system by setting up its SHealth wallet and generating its public-private key pair ($\text{Pub}_{\text{nd}} - \text{Pri}_{\text{nd}}$) and address $\text{Adr}_{\text{nd}}$. As mentioned earlier, the private key will never be revealed to the outside world while the public key will be used for recognizing the node’s signature in the system. Consequently, the node generates the QR codes containing its public key and address and securely delivers them to the government layer. Upon receiving the secured information, the government layer scans the QR codes and registers the information in its records along with the identification of the inaugurated node. Consequently, a signature over $\text{Pub}_{\text{nd}}$ and $\text{Adr}_{\text{nd}}$ is generated by the government layer and transmitted as a blockchain transaction: $\text{Sig} = \text{Alg}_{\text{sig}}(\text{Alg}_{\text{hs}}(\text{Pub}_{\text{nd}}, \text{Adr}_{\text{nd}}), \text{Pri}_{\text{gov}})$. Hence, the network will first hear about the new node while listening to the first transmitted transaction that records the existence of the new node in SHealth. Figure 2 represents the Business Process Management (BPM) model for this scenario.

Please note that the above procedure is applicable for all kind of nodes: providers and partners.

#### B. A NEW USER JOINS THE SYSTEM

Similar to the previous procedure, a new user can join SHealth by obtaining the user’s wallet from the system and generating his public-private key pair ($\text{Pub}_{\text{us}} - \text{Pri}_{\text{us}}$) and his tier 3 address $\text{Adr}_{\text{us}}$. The user then generates the QR codes that correspond to his public key and address and securely delivers them to a tier 1 or tier 2 node along with his application to join the system. The node scans the user’s codes and matches his declared information with his national identification. Following a successful matching, the node registers the new user in its database and later using its own private key signs over the user’s public key and address: $\text{Sig} = \text{Alg}_{\text{sig}}(\text{Alg}_{\text{hs}}(\text{Pub}_{\text{us}}, \text{Adr}_{\text{us}}), \text{Pri}_{\text{nd}})$. This signature is announced to the network as a blockchain transaction, hence, concluding the user’s registration. This scenario is depicted in the BPM in figure 3.

### TABLE 1. Notations and abbreviations.

| Notation and Abbreviation | Meaning               |
|---------------------------|-----------------------|
| $\text{gov}$             | Government            |
| $\text{nd}$              | Node                  |
| $\text{pt}$              | Partner               |
| $\text{pr}$              | Provider              |
| $\text{inst}$            | Medical instrument    |
| $\text{usr}$             | User                  |
| $\text{cl}$              | Client                |
| $\text{Pub}_{\text{Alice}}$ | Alice’s public key       |
| $\text{Pri}_{\text{Alice}}$ | Alice’s private key     |
| $\text{Adr}_{\text{Alice}}$ | Alice’s address        |
| $\text{Sig}$             | Signed message        |
| $\text{Alg}_{\text{sig}}$ | Signing algorithm      |
| $\text{Alg}_{\text{hs}}$ | Hashing algorithm      |
All users go through this admission process in the system. However for the clients, the registration takes a step further which is associating the client with a partner. Thus, the next phase in the client’s set up procedure is to generate his tier 4 address in the system along with his selected partner. At this point, the client provides his partner’s address $Adr_{pt}$ to the processing node in order to assign him his tier 4 address and conclude his registration. The assumption here is that the client has been previously registered with a selected partner and that the partner has provided the client with a QR containing his address $Adr_{pt}$ and public key $Pub_{pt}$. On this account, the node signs with its private key $Pri_{nd}$ over the client’s public key $Pub_{cl}$, his address $Adr_{cl}$, the partner’s public key $Pub_{pt}$ and his address $Adr_{pt}$: $\text{Sig} = \text{Alg}_{sig}(\text{Alg}_{hs}(Pub_{cl}), Adr_{cl}, (Pub_{pt}, Adr_{pt}), Pri_{nd})$.

The processing node announces the signature as a blockchain transaction which will be recognized by all nodes in the system. This concludes the creation of the 2-of-2 address pertaining the client and his partner $Adr_{cl}$. The BPM in figure 4 illustrates this scenario.

### C. A CLIENT VISITS A PROVIDER

If Alice is feeling unwell and wishes to visit her doctor, she selects the doctor’s office from the provider list from her handheld device and digitally requests an appointment by initiating a message addressing the doctor’s office $Adr_{pr}$:

$$(Adr_{Alice}. \text{Alg}_{sig}(\text{Alg}_{hs}(\text{Request}), Pri_{Alice})).$$

The doctor’s office receives the message, extracts Alice’s address $Adr_{Alice}$, and then retrieves her public key $Pub_{Alice}$ from the system. After verification, the provider using Alice’s public key $Pub_{Alice}$ encrypts its own public key $Pub_{pr}$ and sends a reply message to Alice: $(\text{Alg}_{sig}(\text{Alg}_{hs}(Pub_{pr}), Pub_{Alice}))$.

Alice receives the message and decodes it using her private key $Pri_{Alice}$ to retrieve the provider’s public key $Pub_{pr}$: $Pri_{Alice}((\text{Alg}_{sig}(\text{Alg}_{hs}(Pub_{pr}), Pub_{Alice})))$.

Now Alice has the provider’s public key.

As shown in the BPM in figure 5, to reserve an appointment, Alice initiates a smart contract and sends it through the network using her API. The contract which is received by all active nodes will only be triggered and hence executed by the addressed provider. The pseudo code this scenario is described in script 1.

#### Script 1

Require:
Unlocking script: $<\text{Alg}_{sig}(\text{Alg}_{hs}(), Pri_{pr})> \&\& <\text{time}>$

Set:
Locking script: $<Pub_{pr}> \&\& <\text{time}>$

main():

if time then
  get().signature$_{pr}$;
  if 1 then
    Response: “Accept request”;
    destruct contract;

Accordingly, the provider who is receiving other requests from other clients as well initiates a smart contract allocating appointments for Alice and the other clients. This contract will be triggered by all addressed clients individually as it is explained in script 2:

The next phase in this scenario is the physical visitation. Alice attends to the doctor’s office for her appointment and provides the QR containing her address $Adr_{pt}$ and public key $Pub_{Alice}$ in the system. The provider at this point has to involve the partner in the process. Hence, the provider extracts the partner’s address $Adr_{pt}$ and retrieves his public key $Pub_{pt}$ from the network. With both addresses in his possession, the provider now can access the 2-of-2 address and its records pertaining Alice if needed. The provider now requests both Alice and her partner to co-sign the visitation records in order to be attached to the 2-of-2 address and the shadow network. Both of Alice and the partner sign the request to append to the record with their signatures: $(\text{Alg}_{sig}(\text{Alg}_{hs}(\text{Request}), Pri_{Alice}), \text{Alg}_{sig}(\text{Alg}_{hs}, (\text{Request})Pri))$

### D. PROVIDER IS REQUESTING FURTHER TESTS

After her successful registration, Alice is being examined by the doctor who requests her to have an X-ray image at the local medical center, i.e., another provider. The doctor office initiates a new smart contract addressed to the medical center...
containing the desired tests and Alice’s address. This scenario is described in script 3.

From its side, the medical center initiates a reservation smart contract for group of its clients who are to get some medical procedures and transmits it through the network in order to assign them appointments. This smart contract includes clauses that set the time for each appointment and other instructions addressed to the patients. Alice among other clients has to confirm her appointment by signing the contract, script 4.

Now Alice visits the medical center for her scheduled procedures and follows the same registration steps mentioned before.

After her registration, the medical center initiates another smart contract addressing Alice $Adr_{Alice}$, the X-ray instrument $Adr_{Inst(X-ray)}$, and the personnel administering the the instrument $Adr_{Inst(X-ray)}$. Their signatures are required by the smart contract within the time limit for the medical procedure to take place. Hence, Alice within the time limit along with the responsible personnel sign the contract to run the procedure. This scenario is presented in script 5.
The call in the above script invokes the following script in order to register the visitation and its results in the 2-of-2 address associated with the current partner. The next step for Alice is to obtain a new 2-of-2 address with the newly selected partner. Hence, she follows the exact procedure mentioned in section IV-B in generating a tier 4 address with the selected partner. Alice is to obtain a new 2-of-2 address with the newly selected partner. Hence, she follows the exact procedure mentioned in section IV-B in generating a tier 4 address with the selected partner. This portability of records is achieved and maintained in SHealth.

**F. RECORDS PORTABILITY**

As mentioned before, the term partner refers to an entity that facilitates the logistics of treatments for its members and keeps track of them. In some countries where universal health care policies are implemented, there exist one partner that essentially keeps records of all its members’ activities. On the other hand, in other countries multiple partners exist and offer their coverage services to their members. Therefore, it is very common for a client to change his association from one partner to another.

Since the partner’s presence is crucial for accessing the client’s records in SHealth due to the 2-of-2 signature scheme, changing the partner should not affect the permanence of accessing the records. That is, accessibility of records must be guaranteed regardless who the partner is. This portability of records is achieved and maintained in SHealth.

If Alice deems to change her partner, she visits a tier 1 or tier 2 node and provides the node with the required keys and signatures to obtain her complete history from her 2-of-2 address associated with the current partner. The next step for Alice is to obtain a new 2-of-2 address with the newly selected partner. Hence, she follows the exact procedure mentioned in section IV-B in generating a tier 4 address with the selected partner. Up to this point, Alice has two 2-of-2 addresses in the system: \texttt{Adr}^{ptCurrent} and \texttt{Adr}^{ptNew}. Now, with the processing node having the history of Alice and her new 2-of-2 address, it appends her records to the new address as a transaction that is propagated through the network. By the end of this procedure, the network is aware of Alice’s new address.

Please note that for multiple partners, it is possible to increase the dimension of the multisignature address. For instance, if a user is getting his coverage from 2 partners, then the network can generate a 3-of-3 address for him.

**E. DOCTORS PRESCRIBES MEDICATION**

After analyzing the image, the doctor decides to prescribe a cough medicine to Alice. He initiates a smart contract for Alice to dispense the medication addressing Alice and a group of pharmacies, script 7.

The final smart contract called in the above procedure is to register the record of the dispensing the medication in the blockchain and the shadow network. This scenario is presented in script 8.

By the conclusion of this medical consultation, the blockchain and the shadow network eventually carry records including: Alice’s visit to the doctor’s office, a record of her visit to the medical center, a record of her X-ray image, a record of her diagnosis and her medication.

The call in the above script invokes the following script in order to register the visitation and its results in the 2-of-2 address and the shadow network, script 6.

With the X-ray image attached to Alice record in the shadow network, she returns back to her doctor for the diagnosis. The doctor accesses Alice’s 2-of-2 address in the blockchain and retrieves the X-ray image from the shadow network accordingly.

**Script 6**

**Require:**
Unlocking script: \((Alg_{sig}(Alg_{hs}(.), Pri_{Alice}), Alg_{sig}(Alg_{hs}(.), Pri_{pt}))\) \&\& \& \& \langle time \rangle

**Set:**
Locking script: \(<Pub_{Alice}^{pt}> \&\& \langle time \rangle\)

main():
get().signature\_Alice;
if 1 then
get().signature\_pt;
Response: “Procedure registered”;
Response: “X-ray image attached to shadow network”;
else
destruct contract;

**Script 7**

**Require:**
Unlocking script: \((Alg_{sig}(Alg_{hs}(.), Pri_{Alice}) \&\& (Alg_{sig}(Alg_{hs}(.), Pri_{pharm1}) \&\& Alg_{sig}(Alg_{hs}(.), Pri_{pharm2})) \&\& \langle time \rangle\)

**Set:**
Locking script: \(<Pub_{Alice} > \&\& (Pub_{pharm1} > \&\& Pub_{pharm2} > \langle time \rangle > \&\& \langle time \rangle \rangle\)

main():
get().signature\_Alice;
if 1 then
get().signature\_pharm1;
Response: “Medicine dispensed”;
Call: Script 8
else
destruct contract;

**Script 8**

**Require:**
Unlocking script: \((Alg_{sig}(Alg_{hs}(.), Pri_{Alice}), Alg_{sig}(Alg_{hs}(.), Pri_{pt}))\) \&\& \& \& \langle time \rangle

**Set:**
Locking script: \(<Pub_{Alice}^{pt}> \&\& \langle time \rangle\)

main():
get().signature\_Alice;
if 1 then
get().signature\_pt;
Response: “Procedure registered”;
Response: “Medication record attached to shadow network”;
else
destruct contract;
IV. RELATED WORK
In this section we briefly present some of the recent proposed health management applications and systems that are based on the blockchain technology.

A proposal for a blockchain-based medical system for secure sharing, storage and access to medical data is introduced in [26]. The paper presents a lightweight scheme for sharing medical records among doctors from various hospitals while ensuring their security in terms of privacy protection and resistance to data manipulation. The proposed scheme also presents a mechanism to match similar symptoms exhibited by different patients in different locations. The system helps the patients create session keys for their future communication on the disease. A major drawback of this mechanism is that it relies on the delegated proof of stake consensus mechanism in order to maintain the sanity of the ledger.

On the other hand, [27] proposes a privacy preserving data sharing system by implementing two blockchains: a private blockchain that hosts the personal health information and a consortium blockchain to hold the indexes for the health records. The system provides a tamper-proof records’ saving and a privacy protection. The security in the scheme is provided by the use of three algorithms, including the SHA-256 algorithm (for computing hash values) followed by the Elliptic Curve Cryptography (ECC) algorithm (for generating asymmetric keys) and the Advanced Encryption Standard (AES) algorithm for encrypting data. However, the use of three different security algorithms makes the proposed system computationally very complex to support a large number of requests. Furthermore, the mechanism lacks data integrity verification capabilities.

BBDS is yet another blockchain-based data sharing paradigm for medical records in a cloud environment [28]. The paper presents a basic layout for a medical data system that allows the users mainly to access the medical records and sharing them while maintaining their privacy. The authors later presented MeDShare, a blockchain-based medical records system that is enabled to monitor malicious data access [29]. MeDShare tracks and records all activities in the system by employing smart contracts. Furthermore, the system revokes the access of entities that exhibit malicious or record-tampering behavior. Both systems suffer a major drawback which is saving sensitive data on the cloud which exposes the system to several risk issues.

The authors in [60] present another blockchain-based health record system that also relies on cloud storage. One key characteristic of the proposed system is having a machine learning capability for inspecting data. This data inspection is crucial to the elimination of noise in order to produce reliable data. Similar to other blockchain-based health schemes that system ensures that users can safely store their personal health records. All of the above mentioned schemes however have a major vulnerability as they save the sensitive data on the cloud which may expose the system to major security risks by third parties. Moreover, no mechanism to ensure the correctness of such records is discussed.

Yet another health record sharing scheme based on blockchain technology is proposed in [61]. The presented system allows searchability functions based on symmetric encryption and attribute based encryption. It stores the hash values of the personal health records in the blockchain while deploying the smart contracts to save their related index. This, according to the authors guarantees the system having a high level of access control without relying on third parties. However, it is unclear how the system will handle the flood of smart contracts that roam in the network for the purpose of saving records and wait to be executed.

SHealth similar to the above mentioned schemes proposes a health management system that is based on blockchain technology. The system however principally differs in condensing its elements and entities into three major groups: users, nodes, and IoT devices. Furthermore, SHealth guarantees the privacy and the integrity of its clients records by associating each client with two addresses with different permissions and privileges. The first address allows the client to solely access his records in privacy without the ability to alter them. While the second address maintains the integrity of the clients’ records by allowing new entries to be appended only by cosigning them with the concerned partner. Finally, SHealth is fully automated system with reliance on easy-to-use API initiated smart contacts. Through a user-friendly graphical interface, clients can initiate reservations, doctors can request medical tests and medication, and medical instruments connected to the system through IoT interfaces can be commanded through requests.

V. CONCLUSION
In this paper we presented SHealth, a multi-layered and multi-tiered modular blockchain-based health management system. Each layer in SHealth comprises of different stakeholders of the system whereas each tier identifies the privileges and responsibilities of the entities in the associated tier. SHealth defines three distinctive layers: the government, the users, and the IoT devices layer. The government layer encompasses providers and partners where the first represents entities such as hospitals and medical centers and the later delineates entities that provide the support and the logistics for clients such as health insurance companies. The second layer hosts individuals who are part of the system either as clients or personnel. The third layer contains all medical instruments or devices that are connected with the system through IoT interfaces. Each entity in SHealth like provider, partner, user, or client has a unique address that identifies it in the blockchain. In addition to that, clients have another address which is accessible associatively by them and their partners. SHealth utilizes smart contracts which are scripts that equip the system with fully automated and secured management capabilities. Moreover, SHealth deploys the PBFT consensus mechanism which is known to have low overhead and requires less resources than other consensus mechanisms.
In addition to the modular nature of SHealth, PBFT makes the system compatible with the Hyperledger Fabric.

In this work, we have covered various scenarios that can be encountered in a real life health system by implementing smart contracts.

REFERENCES

[1] H. Ahvvenniemi, A. Huovila, I. Pinto-Seppä, and M. Airaksinen, “What are the differences between sustainable and smart cities,” Cities, vol. 60, pp. 234–245, Feb. 2017.

[2] M. Mettler, “Blockchain technology in healthcare: The revolution starts here,” in Proc. IEEE 18th Int. Conf. e-Health Netw. Appl. Services (Healthcom), Sep. 2016, pp. 1–3.

[3] A. Haapio, “Towards sustainable urban communities,” Environ. Impact Assessment Rev., vol. 32, no. 1, pp. 165–169, Jan. 2012.

[4] X. Zhang, “Toward a regenerative sustainability paradigm for the built environment: From vision to reality,” J. Cleaner Prod., vol. 65, pp. 3–6, Feb. 2014.

[5] A. Azaria, A. Ekblaw, T. Vieira, and A. Lippman, “MedRec: Using blockchain for medical data access and permission management,” in Proc. 2nd Int. Conf. Open Big Data (OBD), Vienna, Austria, Aug. 2016, pp. 25–30.

[6] M. Zhang, A. Raghunathan, and N. K. Jha, “Trustworthiness of medical devices and body area networks,” Proc. IEEE, vol. 102, no. 8, pp. 1174–1188, Aug. 2014.

[7] A. Mohsen Sia, S. Sur-Kolay, A. Raghunathan, and N. K. Jha, “Physiological information leakage: A new frontier in health information security,” IEEE Trans. Emerg. Topics Comput., vol. 4, no. 3, pp. 321–334, Jul. 2016.

[8] D. Halperin, T. S. Heydt-Benjamin, B. Ransford, S. S. Clark, B. Defend, W. Morgan, K. F. Tu, K. Kohno, and W. H. Maisel, “Pacemakers and implantable cardio-defibrillators: Software radio attacks and zero-power defenses,” in Proc. IEEE Symp. Secur. Privacy (SP), Oakland, CA, USA, May 2008, pp. 129–142.

[9] H. Salmani and M. M. Tehranipoor, “Vulnerability analysis of a circuit layout to hardware trojan insertion,” IEEE Trans. Inf. Forensics Security, vol. 11, no. 6, pp. 1214–1225, Jun. 2016.

[10] E. Y. Vasserman and N. Hopper, “Vampire attacks: Draining life from smart contracts,” in Proc. 13th Int. Conf. e-Health Netw. Appl. Services, Nov. 2011, pp. 150–156.

[11] S. Tanwar, K. Parekh, and R. Evans, “Blockchain-based electronic health care system for healthcare 4.0 applications,” J. Inf. Secur. Appl., vol. 20, Feb. 2016, Art. no. 102407.

[12] J. Vora, A. Nayyar, S. Tanwar, S. Tyagi, N. Kumar, M. S. Obaidat, and J. P. C. Rodrigues, “BHEEM: A blockchain-based framework for securing electronic health records,” in Proc. IEEE Globecom Workshops (GC Wkshps), Abu Dhabi, United Arab Emirates, Dec. 2018, pp. 1–6.

[13] R. Lai and D. L. K. Chuen, “Blockchain—From public to private,” in Handbook of Blockchain, Digital Finance, and Inclusion, vol. 2, D. L. K. Chuen and R. Deng, Eds. New York, NY, USA: Academic, 2018, ch. 7, pp. 145–177.

[14] A. Ojo and S. Adebayo, “Blockchain as a next generation government information infrastructure: A review of initiatives in D5 countries,” in Government 3.0—Next Generation Government Technology Infrastructure and Services. Cham, Switzerland: Springer, 2017, pp. 283–298.

[15] G. Chen, B. Xu, M. Lu, and N.-S. Chen, “Exploring blockchain technology and its potential applications for education,” Smart Learn. Environ., vol. 5, no. 1, p. 1, Dec. 2018.

[16] P. Ocheja, B. Flanagan, and H. Ogata, “Connecting decentralized learning records: A blockchain-based learning analytics platform,” in Proc. 8th Int. Conf. Learn. Anal. Knowl., Mar. 2018, pp. 265–269.

[17] M. Turkanovic, M. Holb, K. Kocic, M. Mericko, and A. Kamisic, “EduCTX: A blockchain-based higher education credit platform,” IEEE Access, vol. 6, pp. 5112–5127, 2018.

[18] S. Cheng, B. Zeng, and Y. Z. Huang, “Research on application model of blockchain technology in distributed electricity market,” in Proc. Int. Conf. New Energy Future Energy System (NEFES), vol. 93, Nov. 2017, Art. no. 012065.

[19] M. Yulea and S. N. G. Gourisetii, “Blockchain for smart grid resilience: Exchanging distributed energy at speed, scale and security,” in Proc. Resilience Week (RWS), Sep. 2017, pp. 18–23.

[20] S. Kushch and F. P. Castrillo, “A review of the applications of the blockchain technology in smart devices and dis-tributed renewable energy grids,” Adv. Distrib. Comput. Artif. Intell. J., vol. 6, no. 3, p. 75, 2017.

[21] M. Swan, “Blockchain for business: Next-generation enterprise artificial intelligence systems,” in Advances in Computers, Amsterdam, The Netherlands: Elsevier, 2018.

[22] S. Ólens and A. Jansen, “Blockchain technology as a support infrastructure in e-government,” in Electronic Government (Lecture Notes in Computer Science), Cham, Switzerland: Springer, 2017, pp. 215–227.

[23] R. Hanifatunisa and B. Rahardjo, “Blockchain based e-voting recording system design,” in Proc. 11th Int. Conf. Telecommun. Syst. Services Appl. (TSSA), Oct. 2017, pp. 1–6.

[24] E. Yavuz, A. K. Koc, U. C. Cabuk, and G. Dalkiliç, “Towards secure e-voting using ethereum blockchain,” in Proc. 6th Int. Symp. Digit. Forensic Secur. (ISDFS), Mar. 2018, pp. 1–7.

[25] X. Liu, Z. Wang, C. Jin, F. Li, and G. Li, “A blockchain-based medical data sharing and protection scheme,” IEEE Access, vol. 7, pp. 118943–118953, Oct. 2019.

[26] H. Li, L. Zhu, M. Shen, F. Gao, X. Tao, and S. Liu, “Blockchain-based data preservation system for medical data,” J. Med. Syst., vol. 42, no. 8, Aug. 2018, Art. no. 141153.

[27] X. Li, C. Sifah, A. Smahi, S. Amofa, and Z. Zhang, “BBDS: Blockchain-based data sharing for electronic medical records in cloud environments,” Information, vol. 8, no. 2, p. 44, 2017.

[28] Q. Zhang, E. B. Sifah, Y. Shen, X. Jiang, and J. Wan, “Smart contract-based access control for the Internet of Things,” IEEE Internet Things J., vol. 6, no. 6, pp. 1594–1605, Apr. 2019.

[29] D. Mohanty, “Deploying smart contracts,” in Ethereum for Architects Developers, Berkeley, CA, USA: Apress, 2018, pp. 105–138.

[30] Hyperledger Fabric. Accessed: Apr. 15, 2020. [Online]. Available: https://www.hyperledger.org/projects/fabric

[31] S. Nakamoto. Bitcoin: A Peer-to-Peer Electronic Cash System. Accessed: Apr. 15, 2020. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[32] V. Buterin. (2014). On Stake. [Online]. Available: https://blog.ethereum.org/2014/07/05/stake/

[33] A. M. Antonopoulos and G. Wood, Mastering Ethereum: Building Smart Contracts and Dapps, 1st ed. New York, MA, USA: O’Reilly Media, 2018.

[34] R. C. Merkle. “A digital signature based on a conventional encryption function,” in Advances in Cryptology—CRYPTO (Lecture Notes in Computer Science), vol. 293, C. Pomerance, Ed. Berlin, Germany: Springer, 1988.

[35] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, “Chord: A scalable P2P lookup service for Internet applications,” in Proc. Conf. Appl., Technol., Architectures, Protocols Comput. Commun. (SIGCOMM), New York, NY, USA: ACM, 2001, pp. 149–160.

[36] N. Szabo. (1994). Smart Contracts. [Online]. Available: http://szabo.best.vwh.net/smart_contracts.html

[37] H. Christidis and M. Devetsikiotis, “Blockchains and smart contracts for the Internet of Things,” IEEE Access, vol. 4, pp. 2392–2303, 2016.

[38] P. Bailis, A. Narayanan, A. Miller, and S. Han, “Research for practice: Cryptocurrencies, blockchains, and smart contracts; hardware for deep learning,” Commun. ACM, vol. 60, no. 5, pp. 48–51, Apr. 2017.

[39] G. Bigi, A. Bracciali, G. Meacci, and E. Tuosto, “Validation of decentralised smart contracts through game theory and formal methods,” in Programming Languages With Applications to Biology and Security (Lecture Notes in Computer Science), vol. 9465. Cham, Switzerland: Springer, 2015.

[40] T. T. A. Dinh, R. Liu, M. Zhang, G. Chen, B. C. Ooi, and J. Wang, “Untangling blockchain: A data processing view of blockchain systems,” IEEE Trans. Knowl. Data Eng., vol. 30, no. 7, pp. 1366–1385, Jul. 2018.

[41] M. Alharby and A. V. Moorsel, “Blockchain based smart contracts: A systematic mapping study,” in Proc. Comput. Sci. Inf. Technol. (CS IT), Aug. 2017, pp. 125–140.
[46] N. Atzei, M. Bartoletti, and T. Cinoli, “A survey of attacks on Ethereum smart contracts,” in *Proc. Int. Conf. Princ. Secur. Trust, Apt*. 2017, pp. 164–186.

[47] S. Tikhomirov, E. Voskresenskaya, I. Ivanitskiy, R. Takhaviev, E. Marchenko, and Y. Alexandrov, “SmartCheck: Static analysis of ethereum smart contracts,” in *Proc. 1st Int. Workshop Emerg. Trends Softw. Eng. Blockchain WETSEB*, May/June 2018, pp. 9–16.

[48] L. Luu, D.-H. Chu, H. Olickel, P. Saxena, and A. Hobor, “Making smart contracts smarter,” in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur. CCS*, 2016, pp. 254–269.

[49] B. K. Mohanta, S. S. Panda, and D. Jena, “An overview of smart contract and use cases in blockchain technology,” in *Proc. 9th Int. Conf. Comput. Commun. Netw. Technol. (ICCCNT)*, Jul. 2018, pp. 1–4.

[50] N. Atzei, M. Bartoletti, and T. Cimoli, “A survey of attacks on Ethereum smart contracts,” in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur. CCS*, 2016, pp. 254–269.

[51] F. Schneider, “The state machine approach: A tutorial,” in *Distributed Computing*, vol. 4, pp. 299–319, Dec. 1990.

[52] F. Schneider, “The state machine approach: A tutorial,” *Fault-Tolerant Distributed Computing*, New York, NY, USA: Springer, 1990, pp. 18–41.

[53] M. Castro and B. Liskov, “Practical Byzantine fault tolerance,” in *Proc. OSDI*, vol. 99, 1999, pp. 173–186.

[54] M. Castro and B. Liskov, “Practical byzantine fault tolerance and proactive recovery,” *ACM Trans. Comput. Syst. (TOCS)*, vol. 20, no. 4, pp. 398–461, Nov. 2002.

[55] H. Sukhwani, J. M. Martinez, X. Chang, K. S. Trivedi, and A. Rindos, “Performance modeling of PBFT consensus process for permissioned blockchain network (Hyperledger Fabric),” in *Proc. IEEE 56th Symp. Reliable Distrib. Syst. (SRDS)*, Hong Kong, Sep. 2017, pp. 253–255.

[56] V. Buterin. (2014). Slasher Ghost, and Other Developments in Proof of Stake. [Online]. Available: https://blog.ethereum.org/2014/10/03/slasherghost-developments-proof-stake/

[57] K. Zheng, Y. Liu, C. Dai, Y. Duan, and X. Huang, “Model checking PBFT consensus mechanism in healthcare blockchain network,” in *Proc. 9th Int. Conf. Inf. Technol. Med. Edu. (ITME)*, Oct. 2018, pp. 877–881.

[58] X. Zheng, R. R. Mukkamala, R. Vatrapu, and J. Ordieres-Meré, “Blockchain-based personal health data sharing system using cloud storage,” in *Proc. IEEE 20th Int. Conf. e-Health Netw., Appl. Services (HealthCom)*, Ostrava, Czech Republic, Sep. 2018, pp. 1–6.

[59] S. Wang, D. Zhang, and Y. Zhang, “Blockchain-based personal health records sharing scheme with data integrity verifiable,” *IEEE Access*, vol. 7, pp. 102887–102901, 2019.

**Manaf Zghaibeh** received the B.S. degree in electrical engineering from Damascus University, Damascus, Syria, in 1997, and the M.Sc. degree in telecommunications management and the Ph.D. degree in electrical engineering and telecommunications management from the Stevens Institute of Technology, Hoboken, NJ, USA, in 2004 and 2009, respectively. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, Dhofar University, Oman.

**Umran Baig** (Senior Member, IEEE) received the Ph.D. degree from Universiti Teknologi PETRONAS (UTP), Malaysia, in 2012. He is currently an Associate Professor and the Chairperson with the Electrical and Computer Engineering Department, College of Engineering, Dhofar University, Salalah, Oman. He authored and coauthored several journal articles and conference papers so far, which have received more than 500 citations with and H-index of 11. His research interests cover many aspects of the physical, communication and networking layers of wireless and mobile communication networks. He is a member of a number of international organizations, including the IEEE, ACM, Pakistan Engineering Council (PEC), Hong Kong, Pakistan and IAENG, and also serving as a Regular Reviewer for a number of internationally renowned the IEEE, Elsevier, and Springer journals.

**Umran Baig** received the Ph.D. degree in informatics and electronics from Universite Pierre et Marie Curie, Paris, France, in 2011. He is currently serving as an Assistant Professor with the Electrical and Computer Engineering Department, Dhofar University, Salalah, Oman. He has authored one book and coauthored many peer-reviewed, impact factor research articles. His research interests include reconfigurable architectures, fog computing, and scheduling techniques for multicore processing systems. He serves as the reviewer for many international conferences and journals of the domain.

**Najam Ul Hasan** received the M.S. degree in computer engineering from the National University of Science and Technology (NUST), Islamabad, Pakistan, in 2014, and the Ph.D. degree in information and communication engineering from Sejong University, Seoul, South Korea, in 2008. He is currently working as an Assistant Professor with the Department of Electrical and Computer Engineering, Dhofar University, Salalah, Oman. He has authored and coauthored several journal articles and conference papers so far, which have received more than 500 citations with and H-index of 11. His research interests cover many aspects of the physical, communication and networking layers of wireless and mobile communication networks. He is a member of a number of international organizations, including the IEEE, ACM, Pakistan Engineering Council (PEC), Hong Kong, Pakistan and IAENG, and also serving as a Regular Reviewer for a number of internationally renowned the IEEE, Elsevier, and Springer journals.

**Umran Baig** received the Ph.D. degree from Universiti Teknologi PETRONAS (UTP), Malaysia, in 2012. He is currently an Associate Professor and the Chairperson with the Electrical and Computer Engineering Department, College of Engineering, Dhofar University, Salalah, Oman. He authored and coauthored 33 journal articles, one book chapter and 20 conference papers; these articles have received more than 600 citations. His research interests cover many aspects of the physical, medium access, and networking layers of wireless and mobile networks. He is also a Chartered Engineer (CEng)— a member of British Engineering Council, and a member of the Institution of Engineering and Technology (IET), The U.K. He is an Editor-in-Chief of IEEE journal and an Associate Editor of IEEE ACCESS. He is also serving as a designated Reviewer for many international reputed journals of the IEEE, Elsevier, IET, and Springer.

**Manaf Zghaibeh** received the B.S. degree in electrical engineering from Damascus University, Damascus, Syria, in 1997, and the M.Sc. degree in telecommunications management and the Ph.D. degree in electrical engineering and telecommunications management from the Stevens Institute of Technology, Hoboken, NJ, USA, in 2004 and 2009, respectively. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, Dhofar University, Oman.