Chain-Net: An Internet-inspired Framework for Interoperable Blockchains

SIDRAH ABDULLAH, NED University of Engineering and Technology, Pakistan
JUNAID ARSHAD and MOHAMMED ALSADI, Birmingham City University, UK

Blockchain has introduced new opportunities with the potential to enhance systems and services across diverse application domains. Fundamental characteristics of blockchains such as immutability, decentralisation, transparency and traceability have a profound role in this. However, integration with contemporary systems and among disparate blockchain-based applications is a non-trivial challenge primarily due to differences with respect to platforms, consensus mechanism, and governance. Although this challenge has received some attention from the research community, it requires careful analysis to analyse existing work and ascertain gaps to achieve effective and efficient solution to this challenge. This paper presents a thorough systematic review of existing research within blockchain interoperability highlighting significant contributions. Leveraging this analysis, the paper presents an internet-inspired framework (Chain-Net) to facilitate interoperability within blockchain-based systems whereby two systems within independent Blockchain networks can securely exchange data with each other. This is achieved by using gateway module at each network. This module is lightweight node registered by Blockchain network, equipped with discovery service to lookup a target blockchain, and is responsible for forwarding cross-chain transactions to gateway module at the target blockchain. Gateway module plays a vital role in Chain-Net model, as it holds a cross-chain transaction in a pending state until a confirmation is received from the target blockchain, thus maintaining the record integrity between the two chains. The paper presents our efforts to evaluate the proposed blockchain interoperability framework against a success criteria based on our analysis of the blockchain interoperability challenge.

CCS Concepts: • **Computer systems organization** → *Distributed architectures*; • **Networks** → *Network design principles.*

Additional Key Words and Phrases: Blockchain,Distributed ledger technologies, Interoperability, Interoperable blockchains

1 INTRODUCTION

Distributed ledger technologies such as blockchain have attracted significant attention in recent years due to their ability to achieve trustworthy computations in a trustless environment. Since the inception of Bitcoin, blockchains have witnessed exponential growth spanning across diverse application domains such as healthcare, voting, identity management and many others [6, 37]. Fundamental characteristics of blockchains such as immutability, decentralisation, transparency and traceability have a profound role in this. Since blockchain is now implemented in various domains and sectors including healthcare, finance and shipping industry, it is essential that blockchain-based solutions have the capability to adapt and cater to several stakeholders and clients.

Blockchain-based systems enable a high degree of customisation with respect to the participation, platforms, consensus algorithms, governance structure, security, and automation [1]. For instance, a supply chain management solution may have different requirements to those for an e-voting system. One one hand, this high degree of customisation facilitates widespread application development however, on the other hand, it leads to complex heterogeneous systems which may not be inter-operable. Such heterogeneity can be at multiple levels such as public vs consortium blockchains, consensus algorithm, block size, block generation rate etc.

The figure 1 illustrates the interoperability gap in the blockchain structures. For instance, considering the scenario of digital healthcare system, it is expected that each healthcare provider may be running its own blockchain infrastructure for its own patients, medical technicians and other service providers. If any user plans to switch from one healthcare system to the another, there is no autonomous method of transferring or sharing the data among several healthcare

Authors' addresses: Sidrah Abdullah, NED University of Engineering and Technology, Karachi, Pakistan; Junaid Arshad; Mohammed Alsadi, Birmingham City University, Birmingham, UK.

systems. This gives rise to number of challenges such as; disparity in ledgers, incompatible interfaces, and heterogeneous governance mechanism which can result in silos that are unable to communicate in an efficient manner.

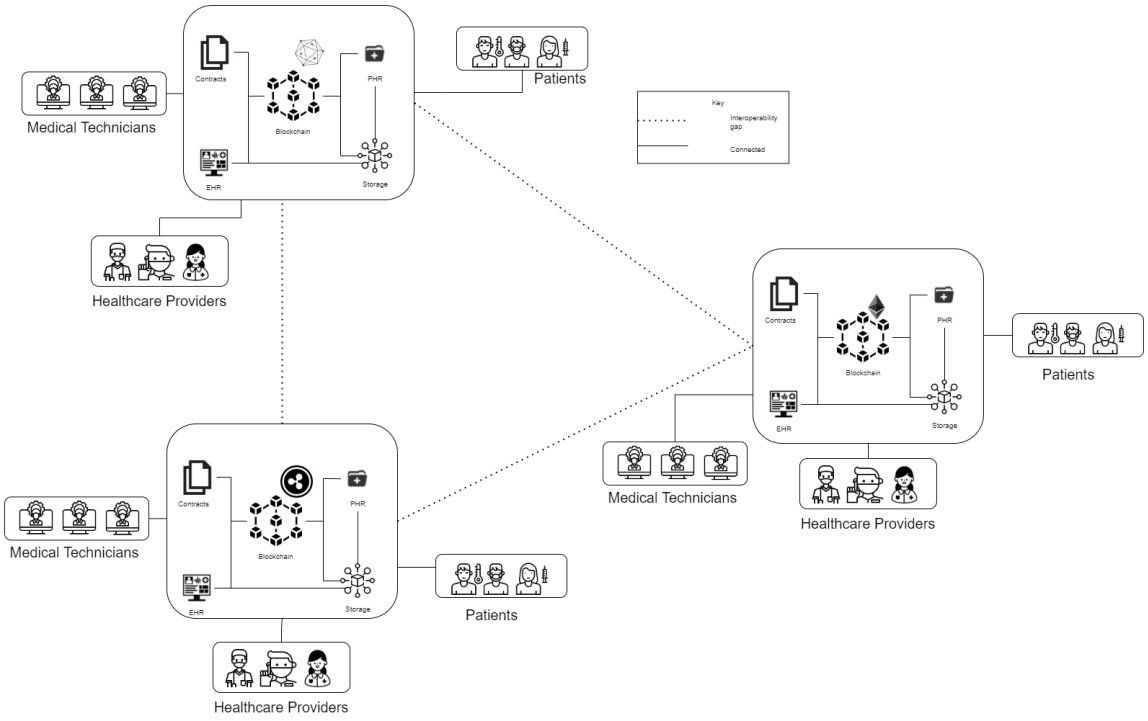

Fig. 1. Interoperability gap in contemporary blockchain-based infrastructures

The scenario explained above illustrates the challenges that can be addressed by interoperability in blockchain. Analysing these challenges, one can refer to the analogy of the open Internet where different hosts (with heterogeneous software and hardware profiles) communicate with each other in a seamless manner. Therefore, some researchers such as [32] argue that blockchains may be considered as new global systems that operate like the Internet. However, instead of transmitting packets of information, blockchains move values or digital assets. The main purpose of cross communication among blockchain systems is to enable exchange or to retrieve information between different networks. Although interoperability of records between systems is crucial, the current architecture of blockchain including other limitations such as scalability and latency [36] does not support interoperability between heterogeneous blockchain systems [32].

Interoperability in blockchain will not only provide flexibility, but it also covers the biggest blockchain research challenge: blockchain scalability. Through interoperability, the blockchain can offload the transaction to other chains and promote privacy to the transactions [21]. Hence, it is crucial to focus on the interoperability bridge between blockchains to enable users to transfer and exchange assets over different blockchains. It improves blockchain scalability and allows users to perform transactions over different blockchains, and create new business opportunities and use cases.

1.1 Problem Statement

Although each blockchain architecture has an immutable ledger that stores the history of assets, the transactions reach a consensus using different algorithms. Hence, their underlying architecture is different from each other. Therefore, the participating blockchain in an interoperable architecture should have a full control of their assets.

Introducing interoperability without violating the underlying structure and assumptions is a key challenge. It is expected from any interoperable framework that cross communication between blockchain systems should not modify any fundamental behavior of the blockchain system. Therefore, the proposed framework must protect the unique characteristics of each blockchain. Moreover, the cross-communication process between blockchain architectures must follow the protocol employed by the respective blockchain system such as a consensus mechanism. To overcome these issues, this paper proposes a cross-communication model that works on the principal of Internet architecture across blockchains. The framework consists of a gateway nodes that are responsible for exchanging the assets between the blockchains. The gateway node, itself, is a part of respective blockchain architecture, and hence, is not a central authority. The solution enables interoperability between the blockchain architectures with minimum effort, ensuring the authenticity of the information generated, and does not alter the heterogeneous nature of blockchains involved in the system.

1.2 Major Contributions

This paper proposes a blockchain interoperability framework leveraging the principles of internet routing protocols. Specifically, this paper makes the following major contributions:

- Conduct a systematic review of the state of the art within blockchain interoperability research to resent a critical analysis of existing work and open challenges. We have analysed 546 articles within this research topic and utilised a systematic process to filter the most relevant to present a critical overview of the state of the art solutions.
- Propose a blockchain interoperability framework (Chain-Net) which facilitates cross-blockchain communication across heterogeneous blockchain infrastructures. The proposed framework leverages the critical analysis of the state of the art and is inspired by how the open Internet functions to establish seamless communication channels across disparate computing platforms.
- Present a thorough evaluation of the Chain-Net framework with respect to security objectives, cost, scalability and vulnerability analysis. The evaluation has highlighted the framework's ability to facilitate interoperability among blockchains whilst meeting requirements such as security, cost and scalability.

To estimate the effectiveness of the proposed framework, a blockchain-based healthcare scenario has been considered as shown in Figure 1. This is because healthcare solution has benefited in numerous ways from the DLT-based healthcare solutions, such as storing Electronic Health Records (EHR) in distributed ledger. The paper presents the framework in EHR scenario where human to human contact is required. However, the proposed framework can also be applied to other scenarios, such as supply chain and logistics.

1.3 Organization

The rest of the article is organized as follows: Section 2 briefly describes the term interoperabilty, blockchain interoperability definitions and cross-chain transactions. Section 3 presents research methodology employed to search and screen related review articles. Section 4 presents critical review of related work, while section 5 proposes the

Chain-Net framework for cross-chain communication network. Section 7 presents the analysis, evaluation and design goals. Section 8 concludes this paper and proposes future research directions.

2 INTEROPERABILITY AND CROSS-BLOCKCHAIN COMMUNICATION

Currently, blockchain technology is growing and gaining interest dramatically which leads to an increased number of independent and unconnected blockchain-based systems. Each system is different from the other in terms of used platform, consensus mechanism, and data governance. These differences result in isolated systems which operate in silos, thus eliminating the ability of creating new added-value and business opportunities since it's not possible to establish a communication between these systems. Hence, it's very important to provide a mechanism through which these isolated independent systems can interact with each other.

The section describes the two major terms: interoperability and cross-chain communication. To achieve interoperability in the blockchain network using the internet-based routing protocol, it is essential to consider cross-chain communication. This section further focuses on the how the cross-chain communication can serve as an underlying architecture for Internet of blockchains.

2.1 Interoperability

Interoperability can be defined as the ability of a system to perform inter-operation functions between entities or nodes. Interoperability in blockchain can be defined as a platform independent communication between blockchain nodes belonging to different platform. It is crucial that blockchains are interoperable as the lack of interoperability leads to independent frameworks or architectures that are unable to communicate with each other, thereby hampering system use. On the other hand, interoperable systems , facilitate seamless integration across diverse systems in the network making the evolution of new use cases possible [16].

Interoperability relies on the concept of two blockchains which need to share data in order to work together. For instance, blockchain X accepts transactions from blockchain Y, as it does not violates X's rule. Therefore, in order for a blockchain system to have interoperability, it should be independent enough to implement its own standard ledger, consensus protocols, and API. Also, the tokens should be permitted to travel over the blockchain network similar to IP datagrams [16].

2.2 Cross-Bockchain Communication

Since the proposed framework employs the cross-chain and cross-blockchain communication protocol to provide communication between interoperable blockchains, it is vital to discuss cross-chain and cross-blockchain communication protocol.

Cross-chain communication protocol is an open-source protocol that allows communication between the packets of different blockchain networks. This protocol aids in simple transfer of packets over the network using smart contracts without delving into the details of underlying architecture [10]. Although interoperability in blockchain involves two blockchain networks exposing their internal states to one another, cross-chain communication takes care of this general purpose interoperability with its three-phase atomic procedure.

In a cross-chain communication based blockchain environment, the blockchain requires plug-and-play connection facility. No blockchain requires heavy architecture change to join the interoperable network. Secondly, the communication is done on the basis of best effort basis. If, for instance, the packet failed to reach the destination, it is transmitted back to the source network, and party involved lose their funds. Thirdly, the connectivity of each blockchain is via black

boxes, or gateway nodes. These nodes are responsible to parsing the information and transmitting it to the required destination. Thus, the cross-chain communication protocol comprises of these properties [11]. However, inherently the blockchain is an 'append-only' model, and the nodes within the network can only append using their own consensus mechanism [4, 12, 32, 36]. Therefore, in the cross-chain communication protocol, the change is not reflected directly on any blockchain system. Instead the gateway nodes takes care of the underlying details.

The cross-chain communication which is based on transactions will ensure the authenticity of the request. Therefore, the smart contract based cross-chain communication network will not alter the heterogeneous nature of the blockchains in the network. Therefore, a user-driven framework using transactions is proposed in this paper.

The section has covered brief overview on interoperability in general, interoperability in blockchain, and cross-chain communication protocol. Section 5 provides more detailed explanation of CCCP with respect to the proposed framework.

3 RESEARCH METHODOLOGY

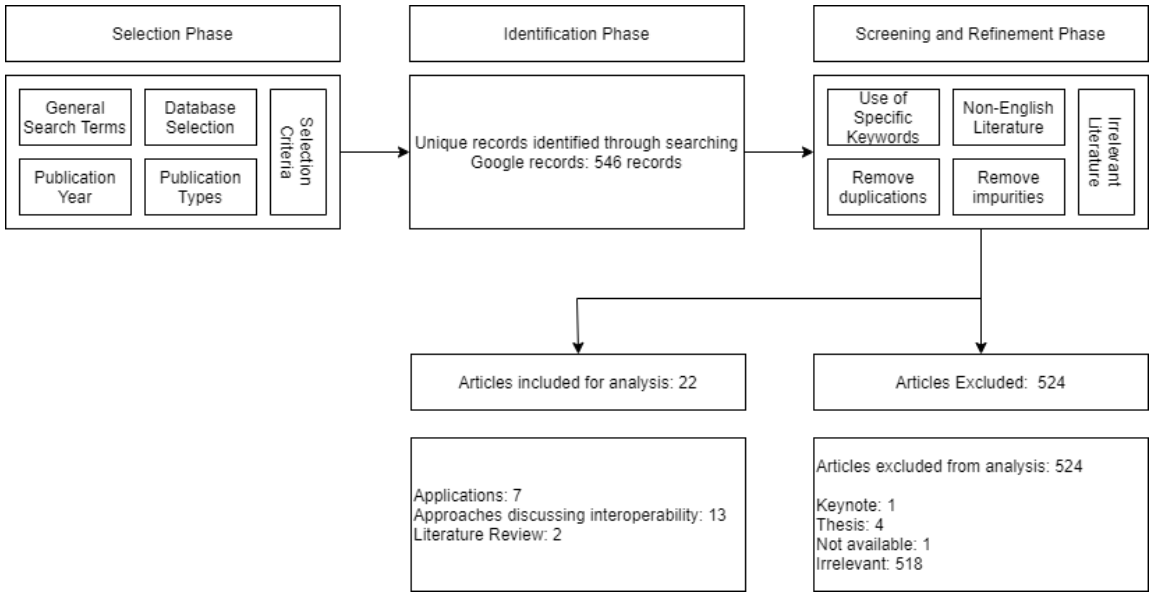

Fig. 2. Research methodology used to conduct systematic review of state of the art

In this section, the research methodology is discussed comprehensively, and illustrated to visualize the whole process. The first step was to conduct an online research for the domain related research available on the internet. The research was carried out on IEEE Xplore, CiteSeerX, Science Direct Google Scholar and Microsoft Academics.

The first step was to search "interoperability in blockchain" without any filter of the research year. The search returned approximately 13,500 records from Google Scholar, 634 records from Science Direct and 294 results from Microsoft Academic. The same search was refined with a condition that filters the paper by year. All the records from 2018 and on wards were searched from the databases which yielded the following results: 10,700 records, 613 records and 201 records from Google Scholar, Science Direct and Microsoft Academic respectively.

From the above research, it can be seen that there was not much difference in the records, and more work has been done on interoperability in blockchain since 2018. The search was further refined by altering the keyword to

"Application level interoperability in Blockchain", as it is the main objective of the paper. Similar to the above search, this search was also carried out in 2 phases: with no filter and with filter on year. The records decreased considerably. In the next step, search keywords were restricted to blockchain, interoperability and application-level to gain search results that align better with the objective of this paper. In this query, the papers were filtered by searching for these keywords in the title of the paper and the existence of these keywords anywhere in the article. It can be seen that number of articles has decreased drastically. A final search was carried out by limiting the year to 2018 and up, and using the exact phrases that are chosen for keywords and the result was available only on Google Scholar, which was 546 records. The website was also Google Scholar as it has the most number of records as compared to other databases. Total 546 records were found from this search. The search results were exported to an excel file. These records were checked manually and were reduced to 44 papers for further analysis.

Figure 2 refers to the research methodology carried out in order to search for the relative papers. Selection phase, from the given figure, consists of general search terms, publication year, publication types, database selection and selection criterion. The phase leads to an identification phase in which unique records were identified, which were 546 records from Google. These records were further screened in the screening and refinement phase, leading to the exclusion of 524 keynote, thesis, and irrelevant articles. Only 22 related articles were included consisting of 7 articles with application interfaces, 13 on discussion of interoperability and 2 literature reviews.

4 RELATED WORK

In order to facilitate the exchange of transactions and cross-chain communication between blockchains, there has been an active research in both the industry and academia on the development of blockchain protocols and architecture. This section presents the survey on related development and research on existing interoperability architecture.

The authors in [29] conducted the study which discusses the need for generic interoperable blockchain protocol that is capable of exchanging tokens and data. However, the authors in [20] claimed the interoperability to be an impossible factor in the classic blockchain network. In [28], the researchers proposed the new definition of heterogeneous blockchain interoperability which states that two blockchains are interoperable if they are able to exchange cryptocurrencies. Furthermore, in [25], blockchains are interoperable if they can exchange information in a unified manner. In addition, [18] defines interoperability as blockchains not only capable of data exchange with smart contracts, but also capable of interaction using smart contracts designed by different designers. The exisitng solutions have studied the blockchain interoperability as a possible exchange of assets (from one blockchain to another), however, the secuity of the cross-chain communication has not been tested yet.

Since the development of blockchain, the main focus of improvement has always been its security, and privacy and performance of blockchain with databases. There are several blockchains that cannot communicate with each other due to because of their designs. [13] has proposed their vision of an interoperable blockchain in which underlying smart contracts can communicate with each other. They have also proposed partial solutions to problems such as cross-chain communication, access control, and cross-chain transactions [13]. The authors in [15] have also discussed lack of cross-communication of blockchain. Thus, they have proposed a solution keeping in mind the history of changing trends of the internet. The goal of the proposed design principle is to bring manageability, interoperability and survivability for blockchain systems [15]. Design goals on the interoperable blockchains have also started emerging, but the seamless implementation is yet to be achieved. Also, the interoperability does not only amalgamate blockchain portability. It also supports blockchain scalability [5]

4.1 Public Connectors

Vitalik Buterin, the founder of Ethereum, described three method of blockchain interoperability: the notary scheme, relay and hash-locking [7]. In **notary scheme**, a set of trustworthy entities allow information sharing across multiple blokchains. The **relay strategy** requests a blockchain to be responsible for verifying the information of the other blockchain in the blockchain network. The third strategy, **hash-locking**, inter-locks multiple operations on different blockchains using the hash. Although all three strategies claim to provide an interoperable blockchain environment, these strategies fail to meet the standard of scalability and security of the network. Apart from the interoperability types discussed by Buterin [7], other two major approaches to interoperability are: Blockchain of Blockchains and Hybrid Connectors.

The application of **Notary Schemes** is represented by Hyperledger Cactus [23], which routes requests around mulitple blockchains using a routing API. However, the solution is centralized which destroys the idea of blockchain, a decentralized ledger. Therefore, the second type, **Relay Schemes**, are equipped with smart contracts capable of accessing assets on the second blockchain in the interoperable network. This design is decentralized by definition. One-way relays are strictly unidirection, and two-way relays are bidirectional. The solutions for this type of interoperable blockchain is Polkadot [33], Cosmos [19] and ChainLink [10]. Cosmos and Polkadots are two interoperability solutions that share the fundamental concepts of creating an interconnected network of blockchains [19, 33]. Furthermore, Polkadots not only offers value exchange, but also data exchange. The blockchains that connects are called parachains and they all are linked to a central connector called relay chain. The parachans use consensus mechanism of the Polkadot. However, they have the liberty to maintain their own structure and function [33]. However, this approach requires blockchains to be built upon the Cosmos or Polkadot platforms which is a hurdle in the enterprise development of blockchain networks. Also, such solutions impose higher execution costs when transferring assets from main-chain to side-chain and vice versa.

Hence, the third type of interoperable blockchain solution, Hashed Time-Lock Contracts (HTLC), enable many atomic state changes to take place on the heterogenous blockchains simultaneously, where both are either rolled back or committed. The [26] study proposed the Hashed Time-Lock Contracts (HTLC) which provide cross-blockchain atomic transactions by ensuring that two transactions are either executed or cancelled after a certain amount of time. In addition, [22] proposed a scalable and secure HTLC by utilizing multi-hop locks. However, these HTLC-based solution only enable asset transfer, and not the token exchange. To overcome this issue, an HTLC-based solution, Lightning Network [27], consists of a layer protocol built on top of a cryptocurrency to support token exchange.

The Public Connectors category identify interoperability solutions supporting cryptocurrencies, and include notary schemes, chains and relays, and hash-locking mechanisms. Table 1 presents a summary of the solutions proposed, commonly classified as Public Connectors, for interoperable blockchain network.

4.2 Hybrid Connectors

Hybrid Connectors are more commonly described as interoperability solutions which are responsible for delivering an abstraction layer [1] that can interact with the decentralized Applications (dApps) without utilizing APIs [14]. This category includes Trusted Relays, Blockchain Agnostic Protocols and Blockchain Migrators.

Trusted Relays refer to an environment where a blockchain is discovered using a registry. In this type of solution, each node acts as a gateway entity that is responsible for validating the transaction. While Blockchain of Blockchains

Solution	Approach	Drawbacks	Integrity	Consensus
Hyperledger Cactus [23]	Notary Scheme	Centralized Server	Yes	Kafka
Polkadot [33]	Sidechain and Relay Nodes	Costly BFT	Yes	GRANDPA
Cosmos [19]	Sidechain and Relay Nodes	Costly BFT and Centralized Hub	Yes	Tendermint
ChainLink [10]	Relay Nodes	No support to shared state	Yes	PoS/PoH
Lightning Network [27]	HTLC	Only for micropayments	Yes	PoW
Loom Network [2]	Sidechain	Closed Source Solution	Yes	DPoS
Wanchain [3]	HTLC	WAN not completely decentralized	Yes	Galaxy Consensus
Chain-Net	Internet architecture	Cross-industry Interoperability	Yes	PoA

Table 1. Solutions to Interoperable Blockchain

aim to provide mechanisms for developers to build dApps for cross-chain communication which migrates data across blockchains. However, this approach requires a common interface interface among ledgers.

An innovative idea of Internet of Blockchains (IoB), where heterogeneous and homogeneous can communicate without any barrier in cross-communication has been proposed by the authors, Tam Vo et al. [31]. The designed model proposes the cross-chain transaction of data, value and state. It also discusses inter-ledger techniques that could achieve interoperability between different ledgers [31].

Another interoperable blockchain design philosophy was presented by the authors in [16]. They have used the internet architecture as the basis for their philosophy to identify the major design principles for an interoperable blockchain. Furthermore, they have discussed key challenges such as survivability (including cryptographic survivability and moving smart contracts for survivability), variety of service types and blockchain systems, reachability, and interconnecting values [16]. They have proposed the implementation of the above discussed principle in [17]. This design philosophy emphasizes on interoperability as a crucial requirement for a standardized blockchain system, which can lower the development cost, provides better reusability, and interoperability. The authors discuss that if the blockchain is the future of the global distributed network of commerce, then it must also meet the requirements of the Internet architecture. According to ARPANET, the fundamental goals of the Internet are: survivability, variety of service types, variety of networks, distributed management of resources, cost effectiveness, ease of attaching hosts and accountability in resource usage [17]. The authors have argued that interoperability is the key to survivability, and the interoperability should exist across the blockchain structure at the mechanical level. They also discussed that an interoperable blockchain should support a variety of networks. This would require a standardized transaction format and syntax and a standardized minimal operations set. Furthermore, an interoperable blockchain becomes complex when a transaction from permissionless blockchain interacts with a transaction from permissioned blockchain. To resolve this issue, the authors have demonstrated the use case which is their design philosophy [17].

Researchers have also proposed interoperable blockchains in the healthcare industry. The authors in [35] suggests the features of dApps for healthcare scenario. The dApps should require minimum integration complexity, flexible design and scalability. Furthermore, in [34], blockchain interoperablity in healthcare is defined on three levels: Foundational, Structural and Semantic. In addition, [9] utilizes Ethereum as an interoperability layer and stores encrypted patient data. However, the solution relies on a centralized service AWS-KMS.

The proposed solutions discussed in this section cover a vast domain of interoperable blockchain. Public connectors offer integrity of transaction but not a variety of use cases. Other solutions implement different use cases, but at the cost

of high complexity for developers. However, there exists no solution with the implementation of Internet of Blockchain concept in the field of healthcare and medicine. This paper proposes a generic interoperable blockchain, Chain-Net Framework. This framework is suitable for cross-chain communications between two independent blockchain networks. Chain-Net Framework is presented in health sector to demonstrate the interaction between two different blockchain networks at different hospitals for managing Electronic Health Records (EHR).

5 CHAIN-NET FRAMEWORK FOR INTEROPERABLE BLOCKCHAINS

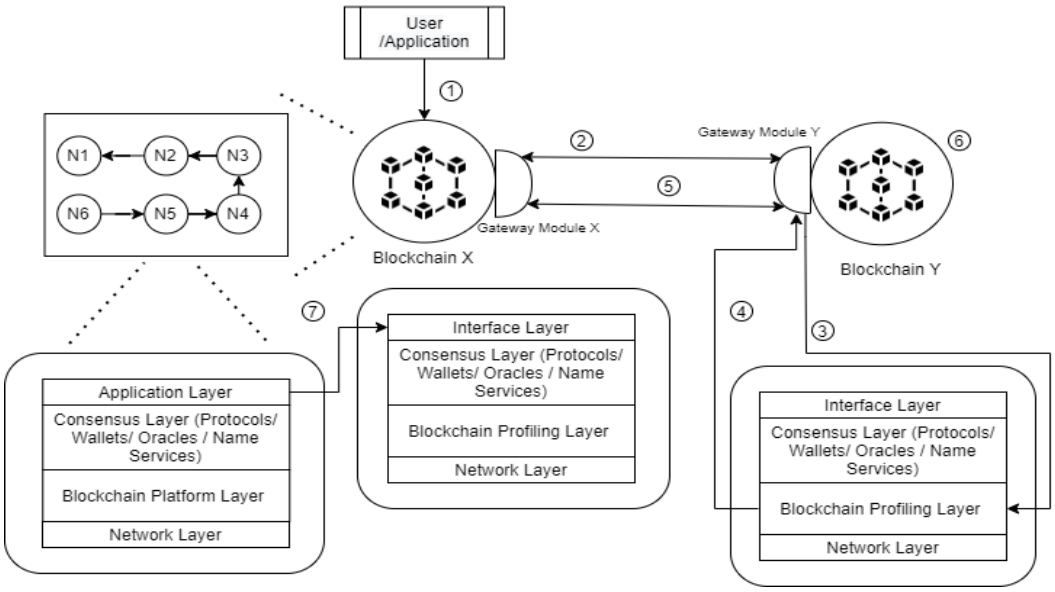

Fig. 3. Chain-Net framework for interoperable blockchains

This section presents the proposed framework on the principle of internet routing protocol, known as Chain-Net. To keep the implementation of the suggested framework with respect to healthcare, the proposed framework is discussed with healthcare implementation However, the framework can be customized to suit the specific needs of the organization or enterprise. This section includes the major elements of the system, design guides, the overall architecture and the sequence of the instructions.

5.1 Blockchain

For the proposed framework, the blockchain ledger used is Ethereum. Ethereum provides a programmable interface and works on Ether currency, which is utilized for transferring assets [8]. The complexity of an Ethereum transaction is calculated in terms of gas units (wei) where 1 wei = e-18 Ether.

5.2 Gateway Nodes

Gateway Nodes are lightweight nodes which are registered by blockchain network. They are designed to access the ledger data, and access patient data for a doctor in the EHR use case. Gateway nodes also acts an interoperability

hub between two blockchains. If another blockchain network is joining, a new gateway node is needed. This way, interconnection between more than two blockchain networks can be achieved. This feature will be investigated comprehensively in the future work. The gateway nodes work on the Layer 3 (Network layer) of the blockchain architecture. The generalized layered structure architecture of blockchain is shown in Figure 4

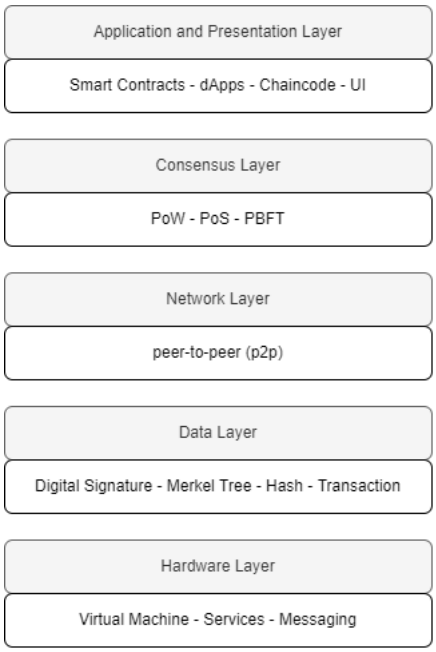

Fig. 4. Layered Structure of the Blockchain Architecture

5.3 Cross-Chain Decentralized Application

Cross-chain dApps act as a primary interoperability hub in the proposed framework. The exchange of information takes place through the Application Programming Interface (API) of the cross-chain dApp with the blockchain individually. The cross-chain transactions are invoked on the dApp and processed at a gateway node in the respective blockchains. The gateway nodes act as the primary interoperability hub that performs the exchange of the information.

5.4 Chain-Net Framework

This section discusses the proposed architecture of an interoperable blockchain. The framework can be seen in the figure 3. This architecture demonstrates interoperability between two blockchains by implementing a gateway module. The framework utilizes two gateway modules, Gateway Module 1 and Gateway Module 2 for two separate blockchains. This module is responsible for hand-shake mechanism, profile exchange and for assuring that same transaction is committed across both the blockchains.

The proposed framework is a simple design at initial level to demonstrate the implementation of interoperability protocol on both the blockchains. The gateway module is responsible for the hand-shaking mechanism. After a request

from the user via dApp is received, it is sent to the gateway module to first search for the existing blockchain profile of the destination blockchain. If the profile exists, it sends back the acknowledgement. Otherwise, it adds the new blockchain in the registry by creating its profile. After this step, a copy of transaction is passed from one blockchain to another via gateway module and is executed on the destination blockchain. The verification is sent back to the sender to execute their copy of transaction. The profile creation is also carried out by the Blockchain Profiling Layer.

In the context of healthcare and EHR, the framework should involve two blockchains with their gateway nodes, doctors or physicians, patients and a healthcare entity such as hospital.

Doctors. The doctors, in EHR system, can request patients' data existing in the system network.

Patients. This entity is registered on the hospital's blockchain, and can restrict data access from the doctor. The patient can also upload their own data as well as the records related to a particular patient are also stored.

Hospitals. Hospitals acts as a established public representation system. Let's assume two hospitals as two different blockchain in the current scenario. Hospital A registers patients and doctors. While Hospital B updates its states on the basis of the communication and request. Both deploy gateway nodes and allow patients to register their data. Furthermore, they also allow doctors to request patient's Electronic Medical Records (EMR).

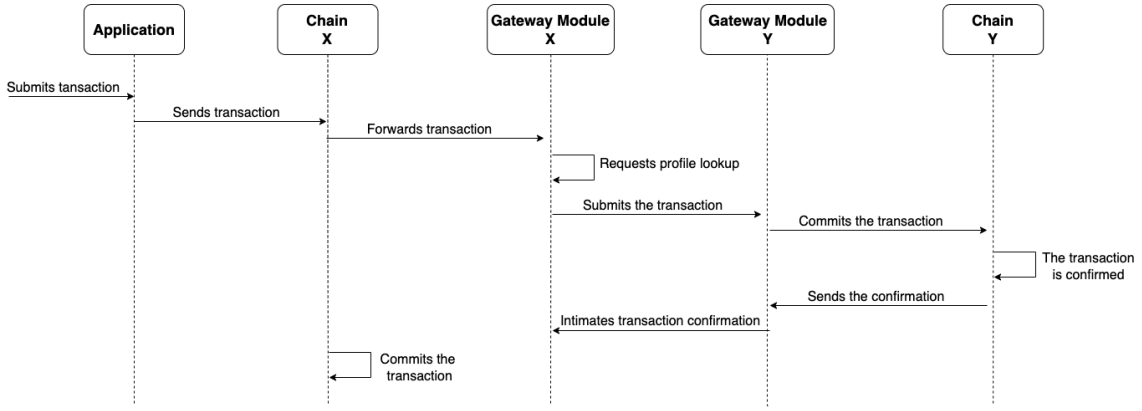

Fig. 5. Sequence Diagram for an Interoperable Blockchain

Figure 5 demonstrates an overall high-level sequence diagram of the proposed model in the EHR scenario. The diagram represents a process of exchanging EMR from one blockchain system to another. The horizontal arrows represent communications with the network. Blockchain X, or the source blockchain is where the information is exchanged. On the hand, the receiver blockchain, Blockchain Y, updates its state based on the transactions. The arrows represent the direction of the communication.

5.5 Sequence of Interactions

The interactions between doctors, patients and hospital is defined in Figure 5 and are discussed below

(1) The cross-chain communication between blockchains begin with a transaction on Hospital X by a doctor through a dApp. The patient data resides in Hospital Y. The doctor generates a token and makes a request using the hash of that token.

(2) The request from the doctor is sent to Gateway Node 1 requesting to fetch the data from Hospital Y. Gateway Node 1, equipped with profiling layer and discovery services, searches for the requested blockchain address.
(3) Upon the discovery of the destination blockchain, this transaction triggers a cross-chain communication in the sender blockchain and becomes final with node consensus.
(4) The next step, which is a state change process in the recipient blockchain.
(5) The patient then makes the self request on Hospital Y for the EMR on behalf of the doctor. The patient's request is sent to the Gateway Module 2. The transaction request is transferred to Gateway Module 2 to perform the required task. In the current scenario, the module fetches the required EMR for the patient. Then it calculates the hash and submits it to Hospital Y blockchain.
(6) In the next step, the patient accepts the doctor's transaction and attach the doctor's original token. The Gateway Module 1 then finally commits the pending transaction ensuring that same transaction is committed on both the chains.
(7) It then broadcasts the transaction confirmation and sends a response to the doctor.
(8) The two gateway modules then establish a connection to transfer the requested documents with its hash. The document is broadcasted to the doctor using the one of the nodes from the system.
(9) The doctor then establishes the connection

In this healthcare scenario, the users: doctors and patients, both are light clients, running a dApp capable of sending transactions to the respective blockchains. A wallet software, based on web3.js, similar to MetaMask, can be created to communicate with the Ethereum platform, a widely-used smart contract platform.

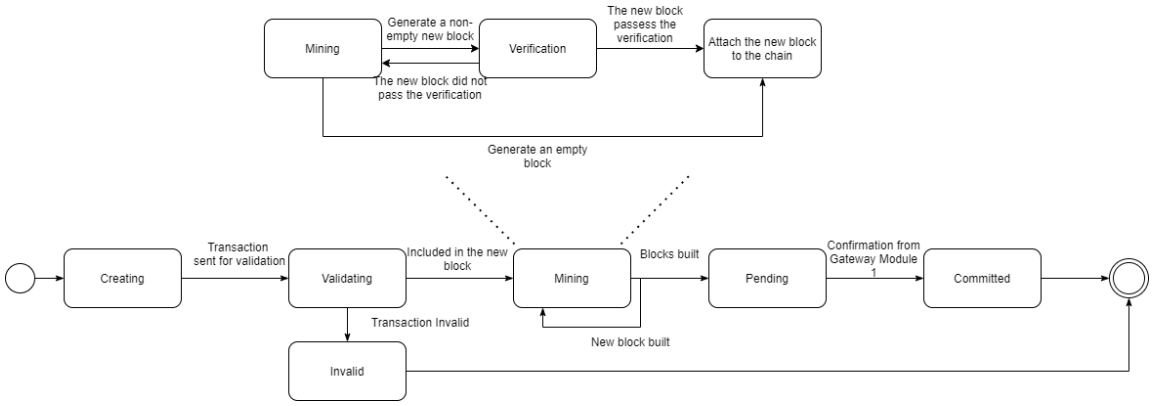

Fig. 6. State Diagram for a Transaction in an Interoperable Blockchain

6 DESIGN AND IMPLEMENTATION OF THE CHAIN-NET FRAMEWORK

This section discusses the design and implementation for the Chain-Net framework specifically focusing on the challenge of interaction between disparate blockchains through the use of a gateway module.

6.1 Chain-Net Design Specification

The implementation covers algorithm for each step of the proposed framework. The algorithm defines the characteristics of each entity involved. The pseudo code is divided into different sections depending on the stage of the transaction. The

first stage is the communication between Blockchain X and gateway module X, the second steps is the communication between the two gateway modules, and the last step is the communication between gateway module Y and Blockchain X.

Algorithm 1 Communication between Blockchain X and Gateway Module X upon the request initiation performed by the entity at Blockchain X

$Tx \leftarrow newTransaction$
$Tx \leftarrow informationRequest$
Tx is encrypted with private key Kr of the user
$Tx \leftarrow Verified$
while $Tx \neq False$ **do**
    Proof of Tx sent to the application as a proof
    $GM1$ requests profile Px lookup in the registry Rx
    if $Px \leftarrow exists$ **then**
        $Px \leftarrow exisitngProfile$
    else if $Px \leftarrow notexists$ **then**
        $Px \leftarrow newProfile$
        $Rx \leftarrow Px$
    end if
end while
$GM1$ commits Tx
$GM1$ includes Tx in block $B1$

The operation on Chain-Net framework begins with the an information query from a blockchain system. Algorithm 1 describes the process between Blockchain X and its Gateway Module X when the doctor initiates an asset request (residing at Blockchain Y) at Blockchain X. This process allows doctors to request the document from an external blockchain. It invokes the smart contracts deployed on the gateway nodes of that blockchain. Then it searches for an existing registry of the external blockchains. If the profile is found, the transaction is verified and included in the block. The network achieves a consensus and selects the next block on the chain until it reaches a desired height. The state of the transaction is hashed in its block header. In case, the registry is not found for a particular external blockchain, the new profile is created and included in the registry. The process is repeated for the blockchain Y.

Algorithm 2 Communication between gateway modules of Blockchain X and Blockchain Y for the asset transfer

if Tx can be verified **then**
    Send token to the gateway node $GM2$ at Blockchain Y as request identifier ri
end if
if ri can be verified at Blockchain Y **then**
    $Dt \leftarrow documentHash$
    $Tx \leftarrow dt$
    $Tx \leftarrow Verified$
    Sends token of the verified transaction Tx and dt to $GM1$
    $GM1$ sends a notification to the requesting entity
end if

Algorithm 2 shows the communication of verified transaction between both the blockchains. The user at Blockchain Y creates the transaction for the requested asset and sends it to gateway module M2. The transaction is validated and

the document hash is attached with the transaction. The verified token is sent to Gateway Module X representing the acceptance of the asset request. Hence, the transaction is then confirmed at Blockchain X, and the entity receives the asset.

Algorithm 3 Submitting an asset of Blockchain Y and requesting an asset

```
The patient Px initiates new transaction for the EMR document
d gets New Document with token dth by patient Px
At index dx, the new document d is added
if ri is received from Blockchain X then
    Tx new Transaction initiated by Px
    r ← newRequest
    r attaches document d
    Tx ← r
    GM2 verifies and attaches Tx in the block
    GM2 notifies Patient Px
end if
```

In Algorithm 1, the entity on Blockchain X could request for an asset from Blockchain Y. However, the asset needs to exist first on the external blockchain. Algorithm 3 describes the asset submission process. In our scenario, the patient submits its EHR to the external blockchain, which can be requested by the doctor on Blockchain X. The patient is capable of attaching the required encrypted document to the transaction upon the arrival of the request identifier from Blockchain X.

6.2 Chain-Net Implementation

The model implementation was completed locally by creating two independent blockchain networks using docker containers on MacBook Pro (CPU is 2.6 GHz and RAM is 16GB) device. Both blockchain networks are EVM compatible, thus solidity programming language was used for writing the smart contracts. For each network, a smart contract and gateway are developed and deployed. Remix IDE[1] was used for development and deployment.

Following the same scenario of EHR, the implementation demonstrates the case where a patient visiting a hospital within BlockchainX needs to register a test at a hospital different from the hospital she already has booked an appointment. These two hospitals are part of different Blockchain networks.

The smart contracts consist of the following functions:

- **addChain**: This function is used to register a new Blockchain network. It takes place in both smart contracts as the registration of a new chain should be completed at both networks in order for them to validate each other.
- **isChainExists**: This function is used to check whether a given Blockchain network is registered. This function is used by both GatewayX and GatewayY as a mandatory control before proceeding with interoperability transactions.
- **addVisit**: This function is used to register a new patient's visit to a hospital in BlockchainX. It's only implemented in BlockchainX.
- **addTest**: This function is used to register a test for the patient. This function can be executed only if the target Blockchain network is registered. AddTest transaction will stay in the pending state waiting for a response from GatewayY. In order for this registration to be completed, a confirmation should be received from target

[1]https://remix.ethereum.org

Fig. 7. A call to addChain function in Blockchain X.

Blockchain (BlockchainY in our case). GatewayX identifier is included as part of the transaction in BlockchainY. Once the transaction is confirmed at BlockchainY, the transaction hash value is sent back to GatewayX which is then committed to BlockchainX ledger by calling the addTest function.

Fig. 8. A call to isChainExist function in Blockchain X.

	HospitalX	HospitalY
Account	0x8b311...2574C0d	0x0e9FA...37f067a
Smart Contract	0x334C...DD580B	0x14Da...581EBB

Table 2. Account and deployed smart contract addresses.

Interactions with smart contracts are illustrated in Figures 7 - 10. Addresses of both Hospitals and the deployed smart contracts are shown in Table 2. Each hospital is responsible for submitting transactions to the corresponding

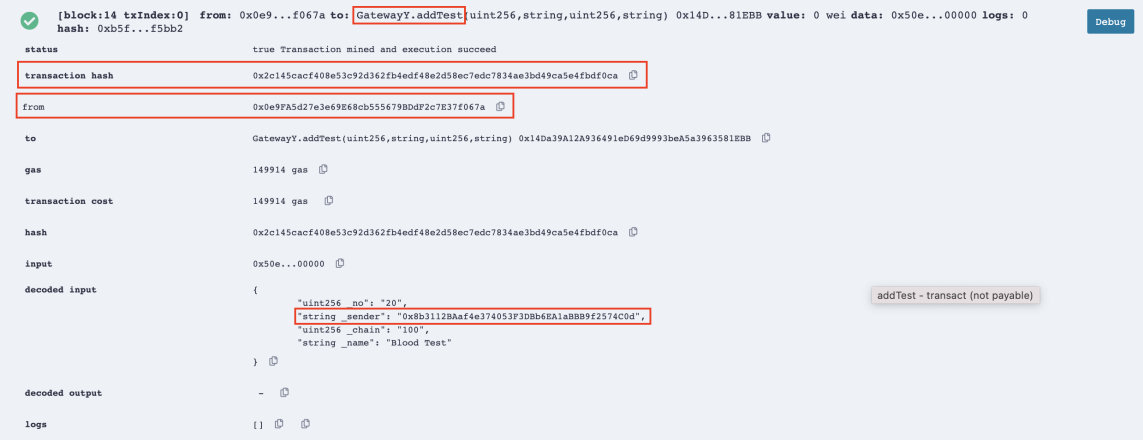

Fig. 9. Adding a new test on Blockchain Y.

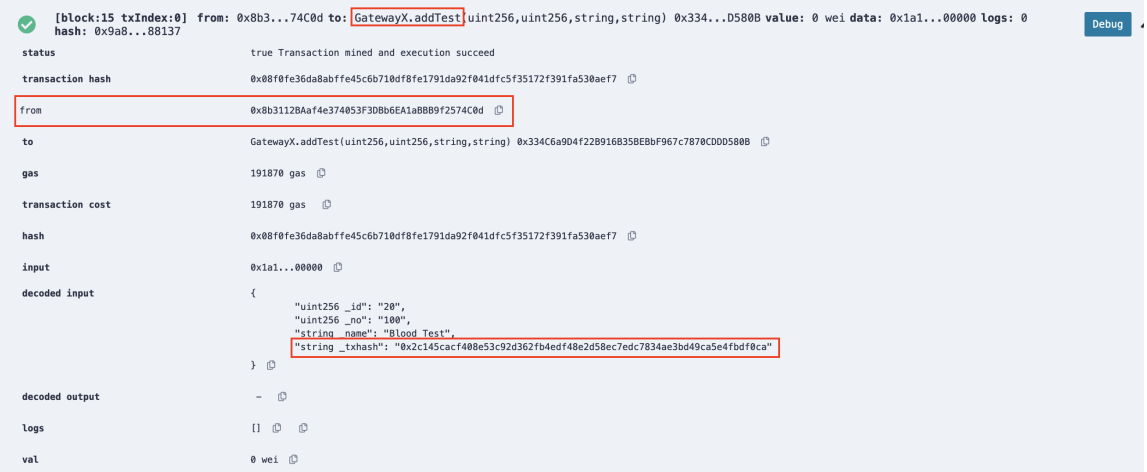

Fig. 10. Adding a new test on Blockchain X.

smart contract. For maintaining data integrity between the two chains, when *addTest* function is called in GatewayY, a reference to GatewayX is included within the transaction. This is used to indicate entity on behalf of which the transaction is created. Once the transaction is committed in BlockchainY, the transaction hash is sent back to GatewayX, which is then included in the payload of the transaction.

7 DISCUSSION AND ANALYSIS

For analysis purposes, this section demonstrates a high-level conceptual model of a Multi-chain Framework in the EHR scenario. The blockchains in this proof-of-concept are independent and distinct, running their own consensus mechanisms. The analysis is performed at a high-level abstraction by considering overall security of the model.

7.1 Security analysis

The following section discusses the security analysis of the proposed model. The framework presented is generic and discussed in the scenario of EHR. However, the actual security depends on the underlying architecture of each heterogeneous blockchain system. Therefore while discussing the threat that might pose to the user, it is assumed that the underlying blockchains are secure. Social Engineering attacks such as phishing or impersonation attacks are out of the scope of this research.

Privacy. The proposed Chain-Net Framework does not require blockchains to disclose or share any sensitive information. The data encryption and decryption works at source and destination, hence it poses no threat or risks. Also, the request from one external chain to another initiates with the exchange of dialogue between entities via face-to-face communication. For example, in the scenario of EHR, patient and doctor would usually communicate via a messaging application, or meet in real-life to initiate the exchange of EHR.

Integrity. Since the framework does not involve any third-party application or service to carry information from one chain to another, there exists no data integrity issue. For instance, the patient is responsible for encrypting their records and translating the receiving the transactions.

Local and Remote Nodes. Since the transaction is signed locally in the blockchain, the node does not hold the private keys in a local wallet. Also, the gateway modules are not external nodes outside of the blockchain system. Hence, there is no risk of Distributed Denial-of-Service (DDoS) attack from the HTTP ports targeting the nodes.

7.2 Data Size Analysis

Blockchains such as Bitcoin and Stellar, are not capable of storing arbitrary data. Hence, there is a limit on the amount of data they can store in the transaction. Considering the scenario of EHR, the patient data might be larger in size than the transaction could hold. Similarly, the same issue could arise in Ethereum due to the deployment of smart contracts. Considering the fact that blockchain technology works as a distributed ledger, and not a full-fledged database, it is inefficient to store huge amount of data on the blockchain. For this reason, a distributed file system could be utilized to store the actual records. The distributed ledger will only be responsible for storing the hash of the data.

7.3 Scalability

The challenge of scalability and interoperability are related in several ways. The proposed framework has provided a cross-chain solution to aid with the issue of scalability. This framework takes the pressure off one blockchain by allowing it to operate on the other blockchain. The smart contracts are managed on different blockchains for different operations, which leverages the speed, and in turn, enhances the scalability. The solution has been tested between two blockchain platforms where gateways are responsible for handling cross-chain transactions. Increasing the number of gateways at each blockchain network would enhance the performance of the whole system. This will be investigated in more details in our future work.

7.4 Cost Analysis

In order to deploy a smart contract and trigger a function in Ethereum, a specific amount of gas should be spent. The gas cost depends on several factors such as function's input and output, size of Solidity cost and its complexity. The actual price paid by users for deploying smart contracts and calling functions depends on gas price. The higher the gas

Item	HospitalX (Gas Used)	HospitalY (Gas Used)	Total Gas	Transaction Fee(ETH)	Total ($)
deploy	623672	694702	1318374	0.00329594208	6.68
addChain	133670	138431	272101	0.00068025396	1.38
addVisit	108073	N/A	18073	0.000045182597	0.091
addTest	191870	149914	341784	0.00085446183	1.73

Table 3. Cost Details.

price the higher opportunity for a transaction to be mined. Table 3 presents the cost details associated with deploying smart contracts and calling the corresponding functions. To calculate the total transaction fee, the following equation is used $Transactionfee(Gwei) = Gasunits(limit) * (Basefee + Tip)$[2]. To improve the readability, the transaction fees are converted to fiat currency (USD).

At the time of writing, the base fee is 0.000005373 *Gwei* and the *Tip* is 2.5 *Gwei*. For example, to calculate the smart contract deployment cost we multiplied the total gas used (1318374) by (2.500005373) and then covert the result to ETH which is converted into USD according to current ETH prices.

We used Slither[3] for smart contract security analysis. Slither is a Solidity static analysis framework written in Python3 and is used to detect security issues in smart contracts by running a suite of vulnerability detectors. Slither runs 76 bug detectors to analyse a smart contract such as shadowing, reentrancy, uninitialsed variables, multiple constructors, Suicidal and many others. A full list of slither's bug detectors along their details can be found at[4]

The results show that no security vulnerabilities have been detected in both transactions. However, there were some informational alerts regarding naming conventions in solidity. The outcome of Slither analysis for both Gatway smart contracts is presented in Figure 11 and 12.

7.5 Interoperability Analysis

For the interoperability analysis of the software systems, the LCIM [30] model has been considered over the past few years. This model consists of conceptual interoperability levels and it is used widely used in several software domains to evaluate the interoperability of the systems. The brief description of the levels is listed below:

- Level 0: indicates no interoperability.
- Level 1: technical interoperability i.e. exchange of bits on the network.
- Level 2: indicates that the devices in the network have understanding of the format of the data being exchanged.
- Level 3: indicates that the devices in the network have understanding of the of the data being exchanged.
- Level 4: the devices are aware of the methods and techniques employed on the other devices.
- Level 5: the devices at this level can exchange data at the interaction of the other system.
- Level 6: this level indicates that the devices are aware of each other's processes, models and information.

However, this model is not considered to be fit for blockchain level interoperability. For instance, level 5 interoperability requires the access from the external system, while in blockchain, the devices cannot be operated by an external system. Therefore, the LCIM for blockchain-based systems proposed by [24] has been considered for the proposed

[2]https://ethereum.org/en/developers/docs/gas/calculating-fees
[3]https://github.com/crytic/slither
[4]https://github.com/crytic/slither/wiki/Detector-Documentation

```
[xxxx@xxxxxxx-mbp  contract % slither gateway_x.sol

Pragma version^0.8.0 (gateway_x.sol#3) allows old versions solc-0.8.9 is not recommended for deployment Reference:
https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-versions-of-solidity
Parameter GatewayX.addVisit(uint256, string, uint256, string)._id (gateway_x.sol#35) is not in mixedCase
Parameter GatewayX.addVisit(uint256, string, uint256, string). _date (gateway_x. sol#35) is not in mixedCase
Parameter GatewayX.addVisit(uint256, string, uint256, string)._patient (gateway_x.sol#35) is not in mixedCase
Parameter GatewayX. addVisit(uint256, string, uint256, string)._hospital (gateway_x.sol#35) is not in mixedCase
Parameter GatewayX. addTest(uint256, uint256, string, string). id (gateway_x.sol#42) is not in mixedCase
Parameter GatewayX. addTest(uint256, uint256, string, string)._no (gateway_x.sol#42) is not in mixedCase
Parameter GatewayX. addTest(uint256, uint256, string, string). _name (gateway_x.sol#42) is not in mixedCase
Parameter GatewayX.addTest(uint256, uint256, string, string)._txhash (gateway_x.sol#42) is not in mixedCase
Parameter GatewayX.addChain(uint256, string, string, string, string)._id (gateway_x.sol#48) is not in mixedCase
Parameter GatewayX.addChain(uint256, string, string, string, string)._hospital (gateway_x.sol#48) is not in mixedCase
Parameter GatewayX.addChain(uint256, string, string, string, string)._url (gateway_x. sol#48) is not in mixedCase
Parameter GatewayX.addChain(uint256, string, string, string, string)._date (gateway_x.sol#48) is not in mixedCase
Parameter GatewayX.addChain(uint256, string, string, string, string)._status (gateway_x.sol#48) is not in mixedCase
Parameter GatewayX.isChainExist(uint256)._id (gateway_x.sol#56) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-solidity-naming-conventions
gateway_x.sol analyzed (1 contracts with 77 detectors), 16 result(s) found
```

Fig. 11. Slither analysis output for Gateway smart contract X.

```
[xxxx@xxxxx-mbp  contract % slither gateway_y.sol

Pragma version^0.8.0 (gateway_y.sol#2) allows old versions solc-0.8.9 is not recommended for deployment Reference:
https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-versions-of-solidity

Parameter GatewayY.is ChainExist(uint256)._id (gateway_y.sol#21) is not in mixedCase
Parameter GatewayY.addChain(uint256, string, string, string, string)._id (gateway_y.sol#27) is not in mixedCase
Parameter GatewayY.addChain(uint256, string, string, string, string)._hospital (gateway_y.sol#27) is not in mixedCase
Parameter GatewayY.addChain(uint256, string, string, string, string). _url (gateway_y.sol#27) is not in mixedCase
Parameter GatewayY.addChain(uint256, string, string, string, string). _date (gateway_y.sol#27) is not in mixedCase
Parameter GatewayY.addChain(uint256, string, string, string, string)._status (gateway_y.sol#27) is not in mixedCase
Parameter GatewayY.addTest(uint256, string, uint256, string). _no (gateway_y.sol#35) is not in mixedCase
Parameter GatewayY.addTest(uint256, string, uint256, string)._sender (gateway_y.sol#35) is not in mixedCase
Parameter GatewayY.addTest(uint256, string, uint256, string)._chain (gateway_y.sol#35) is not in mixedCase
Parameter GatewayY.addTest(uint256, string, uint256, string). _name (gateway_y.sol#35) is not in mixedCase
Parameter GatewayY.getTest(uint256)._no (gateway_y.sol#42) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-solidity-naming-conventions
gateway_y.sol analyzed (1 contracts with 77 detectors), 13 result(s) found
```

Fig. 12. Slither analysis output for Gateway smart contract Y.

framework as shown in the figure 13. As per this model, Level 4 and 5 indicate cross-chain interoperability and enable bidirectional relationship between the nodes in the system.

As per the LCIM for blockchain interoperability, the proposed framework meets the Level 5 interoperability standard. The proposed framework provides integration of heterogeneous networks through gateway nodes, and implements cross-chain functions to handle cross-chain transactions of data.

Fig. 13. LCIM for Blockchain-Based Systems

8 CONCLUSION AND FUTURE WORK

As blockchain-based systems grow, interoperability among them has become a challenge. Focusing on the challenge of interoperable blockchains, this paper first presented critical review of existing approaches to blockchain interoperability including public connectors, hybrid connectors and blockchain of blockchains. Leveraging the analysis of existing literature, this paper has presented an internet-inspired framework (Chain-Net) to facilitate interoperability among blockchains. The Chain-Net framework is inherently scalable and achieves interoperability among diverse blockchains without requiring major changes in their local operation. Our implementation of the Chain-Net framework is currently between two blockchains however further work is underway to extend experimentation with higher number of blockchains. We have also presented evaluation of Chain-Net framework with respect to security properties, cost, and vulnerability analysis to assess its effectiveness.

9 CONFLICT OF INTEREST

On behalf of all authors, the corresponding author states that there is no conflict of interest.

REFERENCES

[1] 2020. Bridging the Governance Gap: Interoperability for blockchain and legacy systems. Technical Report. (2020).
[2] 2022. *Loom network*. Retrieved May 25, 2022 from https://loomx.io/
[3] 2022. *Wanchain*. Retrieved May 25, 2022 from https://www.wanchain.org/
[4] Israa Alqassem and Davor Svetinovic. 2014. Towards reference architecture for cryptocurrencies: Bitcoin architectural analysis. In *2014 IEEE International Conference on Internet of Things (iThings), and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom)*. IEEE, 436–443.
[5] Rafael Belchior, André Vasconcelos, S. Guerreiro, and M. Correia. 2020. A Survey on Blockchain Interoperability: Past, Present, and Future Trends. *ArXiv* abs/2005.14282 (2020).
[6] Rafael Belchior, André Vasconcelos, Sérgio Guerreiro, and Miguel Correia. 2021. A survey on blockchain interoperability: Past, present, and future trends. *ACM Computing Surveys (CSUR)* 54, 8 (2021), 1–41.
[7] Vitalik Buterin. 2016. Chain interoperability. *R3 Research Paper* (2016).
[8] Vitalik Buterin et al. 2014. A next-generation smart contract and decentralized application platform. *white paper* 3, 37 (2014).

[9] Gracie Carter, Ben Chevellereau, Hossain Shahriar, and Sweta Sneha. 2020. Openpharma blockchain on fhir: An interoperable solution for read-only health records exchange through blockchain and biometrics. *Blockchain in Healthcare Today* (2020).

[10] Chainlink. 2021. Cross-Chain Interoperability Protocol (CCIP).

[11] Ratul Antik Das, Md Muhaimin Shah Pahalovi, and Muhammad Nur Yanhaona. 2019. Transaction finality through ledger checkpoints. In *2019 IEEE 25th International Conference on Parallel and Distributed Systems (ICPADS)*. IEEE, 183–192.

[12] Joost De Kruijff and Hans Weigand. 2017. Understanding the blockchain using enterprise ontology. In *International Conference on Advanced Information Systems Engineering*. Springer, 29–43.

[13] T. Dinh, A. Datta, and B. C. Ooi. 2019. A Blueprint For Interoperable Blockchains. *ArXiv* abs/1910.00985 (2019).

[14] Ghareeb Falazi, Uwe Breitenbücher, Florian Daniel, Andrea Lamparelli, Frank Leymann, and Vladimir Yussupov. 2020. Smart contract invocation protocol (SCIP): A protocol for the uniform integration of heterogeneous blockchain smart contracts. In *International Conference on Advanced Information Systems Engineering*. Springer, 134–149.

[15] Thomas Hardjon, Alexander Lipton, and Alex Pentland. 2020. 12. Interoperability of Distributed Systems. In *Building the New Economy* (0 ed.). https://doi.org/10.21428/ba67f642.0499afe0 https://wip.mitpress.mit.edu/pub/xo5rqc22.

[16] Thomas Hardjono, Alexander Lipton, and Alex Pentland. 2018. Towards a Design Philosophy for Interoperable Blockchain Systems.

[17] T. Hardjono, A. Lipton, and A. Pentland. 2019. Toward an Interoperability Architecture for Blockchain Autonomous Systems. *IEEE Transactions on Engineering Management* (2019), 1–12.

[18] Tommy Koens and Erik Poll. 2019. Assessing interoperability solutions for distributed ledgers. *Pervasive and Mobile Computing* 59 (2019), 101079.

[19] Jae Kwon and Ethan Buchman. 2019. Cosmos whitepaper.

[20] Pascal Lafourcade and Marius Lombard-Platet. 2020. About blockchain interoperability. *Inform. Process. Lett.* 161 (2020), 105976.

[21] Claudio Lima. 2018. Developing open and interoperable dlt\/blockchain standards. *Computer* 51, 11 (2018), 106–111.

[22] Giulio Malavolta, Pedro Moreno-Sanchez, Clara Schneidewind, Aniket Kate, and Matteo Maffei. 2018. Anonymous multi-hop locks for blockchain scalability and interoperability. *Cryptology ePrint Archive* (2018).

[23] Hart Montgomery, Hugo Borne-Pons, Jonathan Hamilton, Mic Bowman, Peter Somogyvari, Shingo Fujimoto, Takuma Takeuchi, Tracy Kuhrt, and Rafael Belchior. 2020. Hyperledger Cactus WhitePaper. *URL: https://github. com/hyperledger/cactus/blob/main/whitepaper/whitepaper. md* (2020).

[24] Babu Pillai, Kamanashis Biswas, Zhe Hou, and Vallipuram Muthukkumarasamy. 2022. Level of conceptual interoperability model for blockchain based systems.

[25] Babu Pillai, Kamanashis Biswas, and Vallipuram Muthukkumarasamy. 2020. Cross-chain interoperability among blockchain-based systems using transactions. *The Knowledge Engineering Review* 35 (2020).

[26] Joseph Poon and Thaddeus Dryja. 2016. The bitcoin lightning network: Scalable off-chain instant payments.

[27] Mohamed Sabt, Mohammed Achemlal, and Abdelmadjid Bouabdallah. 2015. Trusted execution environment: what it is, and what it is not. In *2015 IEEE Trustcom/BigDataSE/ISPA*, Vol. 1. IEEE, 57–64.

[28] Eder J Scheid, Timo Hegnauer, Bruno Rodrigues, and Burkhard Stiller. 2019. Bifröst: a modular blockchain interoperability API. In *2019 IEEE 44th Conference on Local Computer Networks (LCN)*. IEEE, 332–339.

[29] Stefan Schulte, Marten Sigwart, Philipp Frauenthaler, and Michael Borkowski. 2019. Towards blockchain interoperability. In *International conference on business process management*. Springer, 3–10.

[30] C. Shanthi, Josephine M.S., and V. Jeyabalaraja. 2016. Analysis of interoperability between mobile apps cross-platforms development using LCIM Model. 9 (01 2016), 2149–2152.

[31] H. Tam Vo, Z. Wang, D. Karunamoorthy, J. Wagner, E. Abebe, and M. Mohania. 2018. Internet of Blockchains: Techniques and Challenges Ahead. In *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*. 1574–1581. https://doi.org/10.1109/Cybermatics_2018.2018.00264

[32] Paolo Tasca and Claudio J Tessone. 2017. Taxonomy of blockchain technologies. Principles of identification and classification. *arXiv preprint arXiv:1708.04872* (2017).

[33] Gavin Wood. 2016. Polkadot: Vision for a heterogeneous multi-chain framework. *White Paper* 21 (2016).

[34] Peng Zhang, Michael A Walker, Jules White, Douglas C Schmidt, and Gunther Lenz. 2017. Metrics for assessing blockchain-based healthcare decentralized apps. In *2017 IEEE 19th International Conference on e-Health Networking, Applications and Services (Healthcom)*. IEEE, 1–4.

[35] Peng Zhang, Jules White, Douglas C Schmidt, and Gunther Lenz. 2017. Applying software patterns to address interoperability in blockchain-based healthcare apps. *arXiv preprint arXiv:1706.03700* (2017).

[36] Zibin Zheng, Shaoan Xie, Hongning Dai, Xiangping Chen, and Huaimin Wang. 2017. An overview of blockchain technology: Architecture, consensus, and future trends. In *2017 IEEE international congress on big data (BigData congress)*. IEEE, 557–564.

[37] Aviv Zohar. 2015. Bitcoin: under the hood. *Commun. ACM* 58, 9 (2015), 104–113.