Blockchain-Based Authentication and Registration Mechanism for SIP-Based VoIP Systems

1st Mwrwan Abubakar
School of Computing
Edinburgh Napier University.
Edinburgh, UK
m.abubakar@napier.ac.uk

2nd Zakwan Jaroucheh
School of Computing
Edinburgh Napier University.
Edinburgh, UK
z.jaroucheh@napier.ac.uk

3rd Ahmed Al Dubai
School of Computing
Edinburgh Napier University.
Edinburgh, UK
a.al-dubai@napier.ac.uk

4th Bill Buchanan
School of Computing
Edinburgh Napier University.
Edinburgh, UK
b.buchanan@napier.ac.uk

2021 5th Cyber Security in Networking Conference (CSNet) | 978-1-6654-0722-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/CSNET52717.2021.9614646

***Abstract*—The Session Initiation Protocol (SIP) is the principal signalling protocol in Voice over IP (VoIP) systems, responsible for initialising, terminating, and maintaining sessions amongst call parties. However, the problem with the SIP protocol is that it was not designed to be secure by nature as the HTTP digest authentication used in SIP is insecure, making it vulnerable to a variety of attacks. The current solutions rely on several standardised encryption protocols, such as TLS and IPsec, to protect SIP registration messages. However, the current centralised solutions do not scale well and cause algorithm overload when encoding and decoding SIP messages. In trying to rectify this issue, we propose in this paper a blockchain-based lightweight authentication mechanism, which involves a decentralised identity model to authenticate the SIP client to the SIP server. Our mechanism uses a smart contract on the Ethereum blockchain to ensure trust, accountability and preserves user privacy. We provided a proof-of-concept implementation to demonstrate our work. Further analysis of this approach's usability, mainly CPU and memory usage, was conducted comparing to IPsec and TLS. Then we discussed our system's security and presented a security analysis. Our analysis proves that our approach satisfies the SIP protocol security requirements.**

***Keywords*—Voice over IP (VoIP), SIP protocol, Blockchain technology.**

I. Introduction

Voice over IP (VoIP) is a popular technology that many end-users and businesses use today to deliver voice communications and media sessions over Internet Protocol (IP) networks. It is also promising because it is more effective to use the internet as a transport channel than to maintain a separate telephone network for calls and another network for data communication. VoIP protocols are classified into two main types. The first is the session control protocols, such as SIP, H.323, etc. The second type is media control protocols, such as RTP, RTCP, etc. The session control protocols do not actually transmit or receive media but are responsible for setting the session and managing it. On the other hand, audio and video streams between end-users are transferred via the RTP protocol by the media server. In VoIP signalling, numerous protocols are utilised, but the Session Initiation Protocol (SIP) [1] is one of the most widely used. SIP is currently used for IoT and telephony systems as both operate on the standard Internet protocol (IP). Because telephones are widely available, users can interact with devices both over the phone and through the internet. Even without internet access, users can call or send SMS commands to IoT appliances remotely. For instance, in smart home automation, IoT devices, such as actuators, switches, household appliances and heating systems, can accept voice commands over the phone [2]. In addition, The SIP protocol is widely adopted today to facilitate all real-time calls and is becoming a vital piece of technology for enterprises telephony systems. The enterprises' phone system is often referred to as a PBX, which stands for Private Branch Exchange. The PBX can run as software that provides telephony services, which can be hosted on a server within the enterprise or in a cloud-based solution entirely handled by a third-party or VoIP provider [3]. The SIP server is considered the main component in any IP PBX system. The traditional SIP-based call flow routes the call via a SIP server, which requires the SIP clients to be registered to ensure that only authorised users can access the telephony services. Therefore, the SIP server will perform the handshake and authenticate SIP clients before connecting and sharing multimedia. SIP authentication is based on a request-response model, in which request messages invoke some functions and receive a response that answers the request. Hence, it allows end-users to negotiate stream parameters, such as encryption key, codecs used in the session, client's capabilities, etc [1].

A. Problem statement

With the rapid pace of businesses that are going digital, more communication will take place over IP infrastructure, resulting in increased SIP deployment. The connection to the Internet, on the other hand, provides hackers with a large attack surface. As a result, most organisations find PBX security to be a difficult task. PBX is still vulnerable to brute-force attacks on SIP accounts with weak passwords [4]. In addition, hackers attempt to gain unauthorised use of telephony system

978-1-6654-0722-9/21/$31.00 ©2021 IEEE

to hijack calls, steal users' information, or perform denial of service attacks to prevent the user from accessing services. Hackers can also use robocalling and auto-dialling software to carry fraudulent activity on the business phone service. Furthermore, attackers can access the enterprises' telephony systems to generate a high volume of international calls [5]. For instance, in 2009, an Australian company's VoIP PBX caused 11,000 international calls in just 46 hours, resulting in a bill of over $120,000.5. In another instance, Michael Smith, a Massachusetts small business owner, discovered that his PBX had been hacked to make $900,000 worth of calls [6]. Previous research in this area by industry organisations such as the Communications Fraud Control Association (CFCA) reported increases in fraud incidences, with subscription and identity fraud being the top attacks [7]. Fraud departments reported a 24% increase in incident workload compared to their previous report. According to the CFCA report, global telecom fraud losses increased by 37%, from $29.2 billion in 2017 to $28.3 billion, or 1.74% of total revenues

B. SIP security

The issue with the SIP protocol is that it was not designed to be secure by default, which makes it vulnerable to hacking. That is because SIP is a text-based protocol similar to HyperText Markup Language (HTML) [1]. This necessitates establishing secure channels between VoIP clients and the SIP server to protect against attacks targeting users' credentials. SIP authentication security relies on several standardised encryption protocols, such as TLS, IPsec, and S/MIME for SIP signalling encryption and SRTP for RTP media encryption [8]. The primary disadvantage of these solutions is their reliance on the public key infrastructure (PKI) to ensure security. The PKI vulnerable to man-in-the-middle attacks because it employs several hash computations and server certificates to ensure security, resulting in overhead and a decrease in performance, rendering the SIP server unable to process incoming requests due to resource exhaustion, making the server more vulnerable to denial-of-service attacks [9]. Additionally, it requires pre-existing user configuration on servers, which is inefficient in terms of scalability. Thus, this necessitates finding new secure, scalable, and lightweight solutions with low computation power, storage capacity requirements, and low latency and communication overhead to overcome the growing impersonation threats and mitigates against the SIP credentials theft.

Blockchain [10], on the other hand, is an emerging technology that provides a decentralised network with no single point of failure and guarantees data immutability via cryptographic functions and consensus algorithms. Blockchain technology and its unique properties open up new possibilities for SIP security that were not available previously. Instead of relying on the current centralised approaches, in this paper, we propose using blockchain and the distributed ledger technology to serve as a globally synchronised, distributed, and immutable ledger to manage the clients' identities and to provide on-chain access control decisions and the other policies.

The remainder of this paper is organised as follows. Section 2 discussed the related work and our contribution. Section 3 provided background on SIP authentication and blockchain technology. In section 4, we present the proposed solution. The system design is presented in section 5. Section 6 looks at the implementation of our system. We provided performance and security analysis of our proposed mechanism in section 7. Finally, we conclude our paper in section 8.

II. Related work and Our Contribution

Since the outbreak of the COVID-19 pandemic, the global VoIP market has seen exponential growth. This can be attributed to the workplace paradigm shift as remote work has become a mandatory requirement for many businesses, resulting in increased deployment of SIP-based VoIP communications. However, the security concerns with the SIP protocol have presented key security challenges. A significant of current researches are focused on security and privacy in the SIP protocol. There have been several efforts [11] [12] [13] to improve the security of the authentication in the SIP protocol. However, the issue with existing solutions is that they are centralised, making them vulnerable to a single point of failure. In addition, when encryption is used for authentication, some complex encryption algorithms will also bring some problems, such as low computational efficiency, increasing hardware power consumption. Alternatively, several research efforts proposed decentralised solutions for the key challenges of the SIP protocol. For instance, the authors in [14] proposed using blockchain technology to improve the security of real-time services, such as video and voice over Long Term Evolution (VoLTE) based on IP Multimedia Subsystem (IMS) networks. The introduced approach involves creating public and private key pairs for VoLTE user devices and then using the Ethereum blockchain to store the public keys. Similarly, the work presented in [15] introduced a decentralised blockchain-based Keystore mechanism that eliminates the use of the traditional Certificate Authority (CA). The blockchain will be holding the cryptographic public keys of users, which the caller can retrieve from the blockchain to ensure the authenticity of the retrieved public key. However, the proposed approach has shown increases in the call setup time compared to the existing secure VoIP solutions. In addition, the authors in [16] proposed the SIPchain. The introduced system proposed leveraging blockchain technology to provide a distributed SIP defence cluster system. SIPchain uses blockchain as a distributed ledger to store a record of an Indicator of Compromise (IOC). This will scale the SIP defences by utilising the Ethereum blockchain to provide an immutable ledger to share attack intelligence between all nodes in the network. However, this approach needs to improve the information dissemination time because public blockchains, such as Ethereum, showed high latency induced by the time needed to include transactions into a block [17]. Hence, it is not appropriate for real-time applications that require urgent access to the ledger. Our approach differs from earlier approaches in that we employ blockchain in a different use case. We utilised blockchain to

enable secure authentication for SIP clients registering with the SIP server and for the access control to telephony services.

A. Our Contribution

Our main contributions in the paper can be summarised as follows.

- We introduced a lightweight weight approach, which involves developing a methodology for registering SIP clients with the SIP server without relying on a centralised authority.
- A decentralised identity model and blockchain-based authentication mechanism for SIP-based VoIP systems.
- A proof-of-concept implementation of the proposed solution.
- We provided performance evaluation and security analysis to demonstrate our system's viability in meeting the SIP security requirements.

III. Background on SIP and Blockchain

Authentication is an essential means of identifying legitimate entities and securing communication in a network. In the current PBX systems, registration is required to access communications tools and services. SIP Protocol depends on a simple challenge-response based authentication mechanism to enables a SIP server to challenge a SIP client [1]. In return, the SIP client needs to provide authentication information in response to that challenge. The SIP client needs first to connect and register to the SIP server by sending a REGISTER message in order to be able to access the telephony services. When the server receives a request from the client, it will send an authentication request to the client. The authentication request will contain a nonce along with the name of the encryption algorithm that the client must use. The client will combine the nonce with user and password information and create a hash out of them. If the correct password is received from the client, a 200 Ok response signifies a successful registration will be sent [1]. The SIP registration process is illustrated in figure 1.

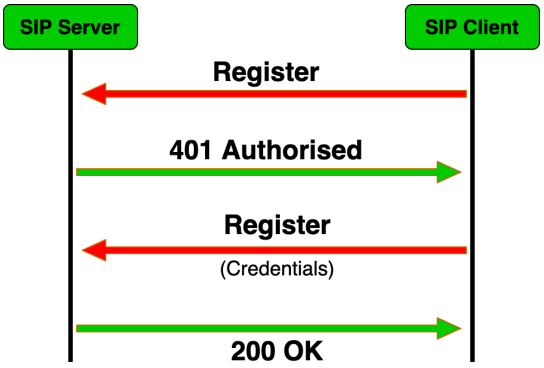

Fig. 1. **SIP Registration**

The problem with the digest authentication used in SIP is the lack of securing the full SIP header and its parameters, making it susceptible to security breaches. The main risk of this mechanism is having the SIP clients credentials stolen, allowing the attacker to impersonate users without any proof that the legitimate client is using these credentials. Then attackers can perform attacks, such as eavesdropping, phishing or ID spoofing to obtain data or carry out additional attacks on their targets.

A. Blockchain

Blockchain technology is a modern invention that has revolutionised the way personal information is distributed securely. Blockchain technology was initially proposed by Satoshi Nakamoto [10] as a solution for data exposure and failure. Its architecture features a shared data structure for storing all transactional history. This distributed database satisfies the features of asymmetric encryption, tamper resistance, and decentralisation requirements. Blockchain is composed of blocks linked together through a chain, with each block consisting of a transaction, transaction counter, and block header. A blockchain's consensus mechanism is one of the most significant and fundamental inventions of blockchain technology. The blockchain's security is implemented using a proof of work concept, in which a transaction is deemed valid only after the system obtains the proof [10]. The proof informs the system that the authorised nodes performed sufficient computational work on the transactions. Mining is the process of creating and adding new blocks. Miners add blocks to a blockchain by using their computing power to solve cryptographic puzzles using hash computation. Each block in a blockchain is uniquely identified using a hash generated using the Secure Hash Algorithm (SHA-256). Additionally, a block's header contains the hash of the block preceding it. The Secure Hash Algorithm generates a cryptographic hash with a fixed length of 256 bits from any length plaintext. [10].

B. Blockchain in VoIP

The first generation of blockchain was focused on ensuring the security of financial transactions. However, the second generation of blockchain is broader in purpose. For instance, it can record data for other applications domains instead of recording financial transactions. Additionally, blockchain can execute and deploy a predefined script called a smart contract [18], such as on the Ethereum blockchain platform. The scripting functionality enables developers to create systems that run on top of the blockchain and thus take advantage of the distributed nature of blockchain technology. As a result, the blockchain can be expanded to be used with emerging technologies such as VoIP, which may require a higher level of verification. Blockchain can improve what is currently considered a complex authentication process in current VoIP systems [19]. Considering the growing number of attacks on VoIP networks, it became necessary for VoIP adopters to adopt blockchain as a solution for many of the security and privacy issues in current VoIP systems. Blockchain can provide a secure authentication method that does not require a centralised server to act as a gatekeeper or a VoIP provider in the middle to authenticate clients. In addition, blockchain

provides a lightweight approach that can help remove the need for the current centralised solutions that rely on several hash computations and server certificates to ensure security. The adoption of blockchain may also enable VoIP providers to provide secure payment services to their customer.

IV. The proposed solution

The primary goal of this work is to create a secure authentication and registration framework for SIP-based VoIP systems through the use of blockchain and the distributed ledger technology. The proposed approach provides a lightweight, secure authentication method for preventing unauthorised end-user access to the SIP server. We will rely on the Ethereum blockchain, which allows us to use smart contracts. We rely on smart contracts to implement on-chain access control decisions and other policies. The smart contract issues the access tokens (challenges) used by the SIP clients to register to the SIP server. Users can communicate with the smart contract by sending transactions signed with their private key. The hash of the used key is interpreted as the user's address and will link the user to their access token. The entropy and storage requirements will be reduced using our proposed method. The smart contract will control the authentication procedure and generate access tokens. As a result, users do not require to save their tokens locally. A simple lookup in the ledger will help verify the authenticity and integrity of all the tokens. Our approach eliminates the need for a central trusted authority. Clients can save their identities directly on their mobile devices, which can then be accessed during the validation phase. This gives users complete control over their identities, rather than having them managed and controlled by a third party. Thus, resolving privacy and reliability concerns. To authenticate with the SIP server, clients must prove their identity by issuing transactions signed by the user's secret key, the hash of which is considered the user's address and is associated with the user's access token. It should be noted that our solution does not affect the SIP-based VoIP architecture.

V. System Design

This Section presents the decentralised architecture and the key details of the proposed authentication and registration mechanism. The proposed blockchain-based SIP telephony system will comprise four main components: the smart contract, SIP client, SIP server, and a decentralised administrative interface. These components are decoupled but can communicate with each other. The system design is presented in figure 2.

A. SIP client

In our model, the SIP client mobile app comprises two main components, the SIP client and the Ethereum wallet. The SIP client is a regular client SIP mobile app that can send and receive SIP messages and provide the traditional call functions of a telephone, such as a dial, answer, reject, call hold, call transfer, etc. The Ethereum wallet application allows user to store their identities straight on their mobile devices, which can be accessed during the validation phase. When the users download the mobile app for the first time on their smartphone phone, they need to generate an empty Ethereum account and get private and public keys. The Ethereum account will be used for signing transactions from the user's account to the smart contract to prove their identity. User issues transactions signed by a private key, whose hash is taken to be the user's address and associated with his or her access token. To interface the Ethereum blockchain with the SIP client, we utilised the Application Programming Interface (API). Our decentralised identity model allows users to manage their own identities rather than have them managed and stored by a third party. For a client to sign transactions from their accounts to the smart contract, the client needs first to fund their accounts with sufficient Ether. The need for Ether is mandatory because for the clients to sign the access token and send transactions from their user account to the smart contract, it requires transaction fees priced in Ether. However, our implementation is not based on the main Public Ethereum network. Instead, we built our implementation based on the Rinkeby test net, which provides a blockchain testing environment with similar characteristics to the main Ethereum public network, but without financial cost, as Ethereum provides a faucet to request free Ethers. The proposed SIP client model and its design do not alter the VoIP architecture based on SIP.

B. Blockchain and the smart contract

The proposed model exploits the advantages of smart contracts and blockchain technology in achieving authentication and authorisation in SIP-based VoIP systems. We use Ethereum-based Blockchain on the proposed model to store information in a distributed manner while maintaining consistency. The reason for using an Ethereum-based blockchain is because it provides an open-source, public, distributed computing platform featuring smart contract (scripting) functionality. The scripting functionality enables developers to write systems over the Blockchain and thus, benefit from the nature of distribution inherited from the Blockchain technology. The smart contract in the Ethereum blockchain is an account holding object that contains distributed code executed by the Ethereum Blockchain autonomously [18]. Clients can interact with the smart contract by issuing transaction signed by their private keys to the smart contract address. In our system, we developed a smart contract to help to store an immutable record of both the clients' policies and authorisation information.

The smart contract is used in our approach to interfacing with data stored on the blockchain. They also offer resilience by executing smart contract code across all blockchain nodes. The smart contracts will be used in our system to implement policies such as on-chain access control decisions and issuing tokens used by SIP clients to authenticate to the SIP server. Using smart contracts ensures that users do not have to store their tokens locally as it helps generate access tokens and manage the authentication process. Thus, it will reduce the overhead associated with issuing, storing, and sharing randomness

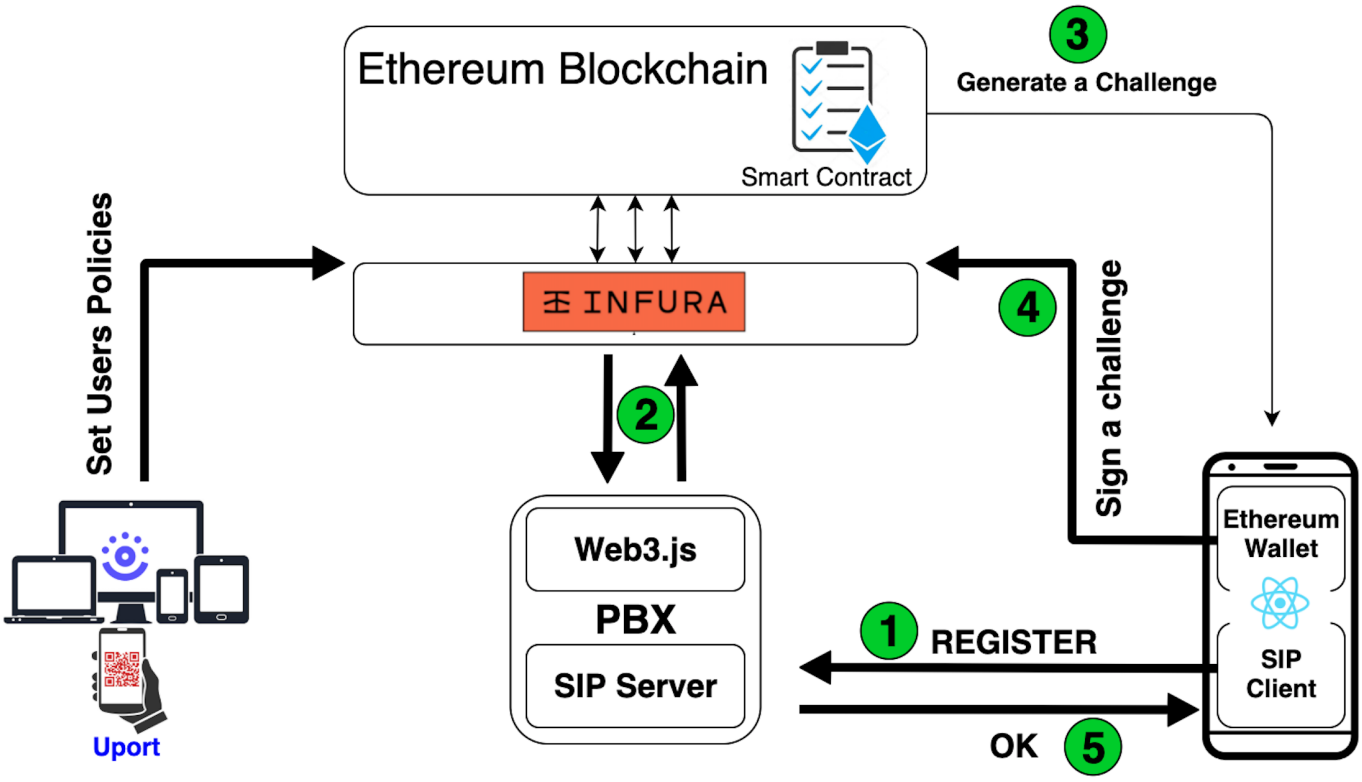

Fig. 2. The proposed System

between SIP servers and the SIP clients. A simple lookup in the distributed ledger can verify the integrity and authenticity of the access token, which acts as a one-time password. We will also use smart contracts in our approach to storing a trusted mapping between the authorised access's public key and its access token. The smart contract is responsible for any operations involved with the authorisation and authentication process, such as setting the users' policies, whitelisting of all addresses that authorised the access token, generating the clients' tokens, and retrieving access tokens signed by clients. The smart contract will also be responsible for registering the SIP client to the SIP server by receiving signed challenges from users and get the result of the verification process into the SIP server. It works similar to the PKI, as the client identity denotes the public part of the asymmetric key pair. Thus, it will guarantee that anyone who has access to the blockchain can verify the authenticity of the message signed by the owner of the keypair.

C. SIP server

In our model, the SIP-enabled PBX software connects to the internet and utilises the SIP protocol to connect endpoints and provide connectivity to external telephone lines and mobile networks. Our SIP-enabled PBX is comprised of two primary components: a SIP server and an Ethereum API. The SIP Server functionality is responsible for registering endpoints, managing SIP calls within a network, and accepting requests to place and terminate calls from user agents, while Ethereum's API will be used to communicate with the blockchain. When the SIP server receives a REGISTER request from a SIP client, it will first extract the client ID, which is the public part of the client key pair. Then the SIP server will send the client's public key to the smart contract. The smart contract will verify the user's permissions in the blockchain. If it is allowed access, the smart contract will generate a challenge and assigned it to that SIP client address. Then the client needs to sign the challenge using the private key and authenticates to the smart contract. This challenge will serve as an access token to authenticate the SIP client to the server. The SIP server will retrieve the result of the verification process and verify the correctness of the procedure. Both the SIP server and the SIP client are Ethereum users and has a public and private key. The SIP server does not have to be physically located at a certain location. In other words, the SIP server might run in the cloud or on a single host.

D. Decentralised administrative interface

IP-PBX today has some remote access method included, which allows the PBX to be maintained and configured via an admin web interface that can be accessed remotely by browsing the PBX domain name or IP address. However, the Administrative Interfaces is a potential security hole as it can be vulnerable to many forms of common network attacks on users accounts and password. A common threat that will affect any website is the attackers who have gained access to a website that a user has an account with. Therefore, attackers

might gain access to the user's password and other accounts that use the same password. As a result of this, the whole system will be compromised [20]. For this, we proposed a decentralised self-sovereign identity model to preserve the privacy of our users. So, therefore, giving them more control over their credentials data rather than have it managed and stored by a third party. In our system, we eliminated the use of username and password to access the administrative interface. Instead, we rely on blockchain and smart contracts to authenticate users and manage users' privileges. Therefore, it makes it impossible for attackers to impersonate the admin user's identity. We utilised the Uport identity app to help with signing transactions from the user's mobile wallet to the smart contract. Our approach assures a high level of confidentiality and integrity as it is backed with the Ethereum account of the user together with its cryptographic proof to allow access to the administrative interface. To access the administrative interface, users need to get authenticated through the smart contract. When a user requests access to the PBX from the web interface, it first needs to click the "Login with Uport" button on our website. The web application will begin by requesting a signing from the user Uport mobile app. The signing request will be requested via a QR code, which needs to be scanned by the user's Uport mobile app. Scanning the QR code using the user's Uport app will allow the user to sign transactions using their private key stored in their Uport identity app. The user will get notified of their request and be asked to confirm their choice by either denying or approving. The approval needs to be confirmed using the user's fingerprint or pin code of their mobile, allowing users to present themself as real people.

VI. Implementation

This section explains the technologies used and the implementation process of our system. The use case developed in this paper involves a decentralised SIP-based Private Branch Exchange (PBX) system that receives a REGISTER request from the SIP clients before they can make calls and access other telephony services. The SIP server software will authenticate and register the SIP client via a smart contract on the Ethereum blockchain. The prototype is made up of four main components, including the SIP client mobile app, SIP server (PBX), a smart contract to help interface data on the blockchain, and the decentralised administrative interface.

A. The decentralised PBX implementation

The implemented PBX system performs two main functions. The first function involves the regular PBX telephony system functionality, which involves sending and receiving SIP messages and the traditional call management functions and services. The second component involves interfacing with data stored in the Ethereum blockchain to perform on-chain access control decisions via the smart contract. To develop our prototype, we implemented our PBX code in the C# programming language. To combine the common activities of writing software into a single application, we utilised Microsoft Visual Studio IDE (Integrated Development Environment). To bring the VoIP functionalities to our application, we utilised Ozeki VoIP SIP SDK [21], a development kit widely used to build professional VoIP communication software. Ozeki VoIP SIP SDK provides a great set of VoIP components, classes, and methods with excellent documentation. We installed .NET Framework on our machine to allow our C# and .NET code to be executed in our operating systems. For this purpose, we utilised the cross-platform implementation of .NET Core to build a core app that runs on macOS. To allow the communication between the SIP server (PBX) and the smart contract, we relied on the Nethereum library [22], the open-source .NET integration library for the Ethereum blockchain. Nethereum library supports the core Ethereum JSON RPC / IPC methods and has an ABI to .NET type encoding and decoding. This will allow us to send transactions from our C# code implementation for the PBX to the smart contract.

B. The SIP client

The blockchain-based SIP client app links two essential elements: the SIP client and the Ethereum wallet. To implement the SIP client, we utilised the open-source react-native SIP client for Android & IOS [23], which using react native PJSIP module to establish communication, sends and receives SIP messages between the SIP client and the SIP-based PBX software. We build a wallet native mobile app using JavaScript. To communicate between the Ethereum blockchain and our application, we utilised the web3.js Ethereum JavaScript Application Programming Interface (API), which interacts with an Ethereum node run on Infura. Infura's APIs and its developer tools provided prompt and scalable API to access the Ethereum networks. The proposed SIP client model and its design do not alter the VoIP architecture based on SIP.

C. Blockchain and the smart contract

To implement our proof-of-concept prototype, we relied on the Ethereum blockchain. The primary impulse behind our choice of Ethereum blockchain is the amount of support provided due to its popularity and the ability to deploy smart contracts to their network. We implemented our smart contract in Solidity [24], the Turing complete language, which is a contract oriented high-level programming that designed to develop smart contracts in Ethereum blockchain systems. To write, evaluate and deploy the smart contract in the Ethereum network, we utilised the Ethereum web browser-based IDE Remix. We have also used debugging tools provided by the Remix IDE to measure gas consumption. The IDE also comes with a compiler that can be used to test the functionality of smart contracts. The role of the contracts in our implementation is similar to the role of the self-signed certificates. It implements any function that carries out operations of the authorisation and authentication process, assisting in the whitelisting of all addresses permitted to request an access token and the production of access tokens for authorised users. Our contract makes use of a mapping data structure indexed by the token to enable mapping between the client's public key and its associated access token. The smart contract functions

include functions such as add, deletes, generate tokens and retrieve access roles. The smart contract can be accessed using its unique address (a 40-characters hex string). The blockchain API facilitates the communication between our application and the smart contract. Instead of running our own Ethereum node, we utilised the API to access the smart contract by calling a REST API endpoint Ethereum node run on Infura over the Internet.

D. The decentralised administrative interface

We implemented the administrative interface together with a decentralised identity model. the web interface helps the admin user to interact with the smart contract to set the users' policies and access data from the smart contract. Our web application consists of various types of resources. These include the HTML templates, JavaScript files, CSS files, images files and server-side implementation code. CSS and images files are used as static resources and particularly influence the display of our application. The server-side is responsible for implementing the application logic. However, some parts of the application logic are possessed on the HTML and JavaScript files, which normally include some sort of templating style. The administrative web interface is built to demonstrate how our decentralised web-based solution can allow users to communicate with blockchain to set clients' policies and managing their identity to subscribe to the SIP server (PBX).

VII. Evaluation

A. Performance analysis

To evaluate the performance of our system, we have implemented the SIP server and SIP clients based on our proposed design. Our measurements setup depends on MacBook air 2018 with 256GB SSD drive, 8GB RAM, 1.6 GHz Dual-Core Intel Core i5. We analysed our approach compared to the current mechanisms used to secure the SIP traffic, such as IPsec and TLS. We utilised the activity monitor application on mac to measure storage usage and processor utilisation. Our evaluation observed that the current TLS consumed a higher memory than our approach because it requires allocating additional buffers. Compared with TLS, the memory utilisation of our approach is around 500 MB less compared with the TLS, while IPsec adds 62 bytes of overhead to every packet, which is increases with multiple applications. In addition, IPsec is not designed for wireless data tunnel "breaks" on roaming or in case of loss of coverage.

Moreover, compared with TLS and IPsec, the computational overhead in terms of CPU is also higher than our approach since cryptographic operations are involved. We observed the computational overhead of the TLS scales up to 91% of the CPU, IPsec protocol utilises about 76% while the CPU overhead of our approach is around 45%. TLS will also cause more overhead for each registration message sent. However, the overhead is varied at runtime depending on the cipher suite and is increases when a certificate uses a larger key length. To measure the overall time of establishing a secure connection, we utilised the internal time function of the system and the Ethereum APIs. Our approach has shown a higher execution time overhead. This an expected issue when adopting the public Ethereum blockchain. This is because of the time needs it to finalise the transactions, which is around 15 seconds on average. However, this can be significantly enhanced by adopting the private blockchain or utilising a reduced mining consensus, reducing the time needed to finals the transactions.

Additionally, any action in the Ethereum blockchain requires a certain amount of gas to send transactions and interact with the smart contract. We calculate the gas cost associated with each event that occurs in the system. To ensure reliable findings, we will use a public test network rather than a private one. Our findings indicate that the most expensive cost is the cost of deploying the smart contract to the Ethereum network. However, this step will be performed only once during the initial setup of the system. Our smart contract deployment cost approximately 0. 000393 ETH Ethers, which equates to $1.28364804 at the average Ether price of $3,266.28 on August 15, 2021 [25]. In comparison, the cost of transmitting transactions and invoking a smart contract function was approximately 0.000085 ETH, this works out to $0.2776338 per transaction. However, this cost is variable, as it is determined by the projected time required to submit the transaction and the necessary storage and processing resources. However, it is important to emphasise that our solution is not dependent on the main Ethereum network due to the associated financial considerations. Rather than that, we constructed our implementation on the Rinkeby test net. The Ethereum test net provides a free blockchain testing environment with features comparable to those found on Ethereum's main public network as Rinkeby provides a faucet for requesting free Ethers to this testing network.

Moreover, when cryptography solutions used, such as TLS and IPsec, the SIP server will act as a single aggregation point for encryption and decryption, which lead to a performance bottleneck. This can be managed by handling encryption/decryption at the endpoints (SIP clients). However, in this case, endpoints must be computationally powerful enough to handle the encryption mechanism, but typically endpoints are less potent than servers.

B. Security of our system

Our model ensures to meet the main three security requirements. This is referred to as the CIA security triad (Confidentiality, Integrity and Availability). Availability refers to having services available on request. Integrity ensures that messages are delivered to their destinations free of change, while confidentiality ensures that the system is only accessible to only those with authorisation. We then analysed the security of our system based on these security requirements.

1) Confidentiality: Through the cryptography technology used to build a blockchain, blockchain technology ensures an extremely high level of security for transactions recorded in the blockchain. To assist in maintaining confidentiality, our system will utilise the symmetric encryption technology

provided by the blockchain to ensure that all access to the system is authorised. To set user policies and authenticate the SIP client to the SIP server, the authorisation will randomly generate a unique token that the user's private key must sign.

2) Integrity: Once data is stored on the blockchain, it is extremely difficult for anyone to alter or modify it. As a result, the system does not permit the modification of user access control rolls or a challenge signed with the user's private key because they are immutable.

3) Availability: Our model ensures a high level of availability for data relating to the verification and authentication processes stored on the blockchain. Every node replicates and updates the transaction data. Even if a node leaves the network accidentally, maliciously, or otherwise becomes inaccessible, the network as a whole will continue to function. Thus, our system will ensure a high level of availability.

4) Auditing: Additionally, each node in the blockchain can approve any changes made in any of the blockchain's linked blocks. This enables organisations to ensure the traceability of blockchain transactions and develop high-quality security intelligence around those transactions, allowing for the auditing and tracking of any data changes.

5) Decentralised trust: Our decentralised model eliminates the need for a trusted third party or intermediary to validate blockchain transactions. Rather than that, the validity and integrity of blockchain transactions are agreed upon through a consensus process. Additionally, because blockchain is a distributed database with no single point of failure, each of its nodes receives a copy of all transaction records, with each transaction accompanied by a digital signature ensuring non-repudiation.

VIII. Conclusions

This paper presented a blockchain-based authentication and registration mechanism for SIP-based VoIP systems. We presented proof-of-concept design and implementation of our approach. Our solution will provide a lightweight approach to facilitate the authentication of the SIP clients to the SIP server in a secure and decentralised way. The proposed approach helps to solve privacy and security challenges in the SIP-based VoIP systems, as it provides a privacy-preserving access control mechanism and facilitates secure users' authentication. This allows the users to have full control over their credentials rather than having them maintained by a third party. We provided security analysis to evaluate the ability of our system to meet the security requirements of the SIP-based VoIP systems. It is feasible that our approach satisfies the security requirements for SIP-based VoIP systems and meet future demands. In addition, we analysed the performance of our system. We observe that our approach provides negligible memory and CPU usage compared with the TLS and IPsec. On the other hand, the proposed approach shows a significant delay since transactions need to be appended to the blockchain but still within an acceptable range. We hope our design provides advantages in the area of user authentication compared to current alternatives.

References

[1] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, "Rfc3261: Sip: session initiation protocol," 2002.

[2] J. Hrabovsky, P. Segec, P. Paluch, M. Moravcik, and J. Papan, "Usability of the sip protocol within smart home solutions," *Communications-Scientific letters of the University of Zilina*, vol. 18, no. 1A, pp. 4–12, 2016.

[3] T. Wallingford, *Switching to VOIP*. " O'Reilly Media, Inc.", 2005.

[4] M. M. Naeem, I. Hussain, and M. M. S. Missen, "A survey on registration hijacking attack consequences and protection for session initiation protocol (sip)," *Computer Networks*, vol. 175, p. 107250, 2020.

[5] N. McInnes and G. Wills, "The voip pbx honeypot advance persistent threat analysis," 2021.

[6] "Introduction to voip fraud," accessed: 2021-06-15. [Online]. Available: https://transnexus.com/whitepapers/introduction-to-voip-fraud/

[7] "Fraud loss survey 2019," accessed: 2021-06-15. [Online]. Available: https://cfca.org/wp-content/uploads/2021/02/CFCA-2019-Fraud-Loss-Survey.pdf

[8] J. Arkko, V. Torvinen, G. Camarillo, A. Niemi, and T. Haukka, "Security mechanism agreement for the session initiation protocol (sip)," *RFC3329*, 2003.

[9] L. Perigo, R. Gandotra, D. Gedia, M. Hussain, P. Gupta, S. Bano, and V. Kulkarni, "Voip security: A performance and cost-benefit analysis," *INFORMATION TECHNOLOGY IN INDUSTRY*, vol. 8, no. 2, 2020.

[10] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," Manubot, Tech. Rep., 2019.

[11] M. Nikooghadam and H. Amintoosi, "A secure and robust elliptic curve cryptography-based mutual authentication scheme for session initiation protocol," *Security and Privacy*, vol. 3, no. 1, p. e92, 2020.

[12] ——, "Perfect forward secrecy via an ecc-based authentication scheme for sip in voip," *The Journal of Supercomputing*, vol. 76, no. 4, pp. 3086–3104, 2020.

[13] J. Jiang and L. Zhao, "A certificate-based authentication for sip in embedded devices," in *Proceedings of the 9th International Conference on Computer Engineering and Networks*. Springer, 2021, pp. 585–590.

[14] E. F. Kfoury and D. J. Khoury, "Secure end-to-end volte based on ethereum blockchain," in *2018 41st International Conference on Telecommunications and Signal Processing (TSP)*. IEEE, 2018, pp. 1–5.

[15] K. D. J. Kfoury, Elie F, "Secure end-to-end voip system based on ethereum blockchain," *Journal of Communications*, vol. 13, no. 8, pp. 450–455, 2018.

[16] A. Febro, H. Xiao, and J. Spring, "Sipchain: Sip defense cluster with blockchain," in *2019 Principles, Systems and Applications of IP Telecommunications (IPTComm)*. IEEE, 2019, pp. 1–8.

[17] V. Buterin *et al.*, "Ethereum white paper," *GitHub repository*, vol. 1, pp. 22–23, 2013.

[18] ——, "A next-generation smart contract and decentralized application platform," *white paper*, vol. 3, no. 37, 2014.

[19] S. Y. Lim, P. T. Fotsing, A. Almasri, O. Musa, M. L. M. Kiah, T. F. Ang, and R. Ismail, "Blockchain technology the identity management and authentication service disruptor: a survey," *Int. J. Adv. Sci. Eng. Inf. Technol*, vol. 8, no. 4-2, pp. 1735–1745, 2018.

[20] A. K. Jain and B. Gupta, "A survey of phishing attack techniques, defence mechanisms and open research challenges," *Enterprise Information Systems*, pp. 1–39, 2021.

[21] C. API and P. API, "Ozeki ozeki voip sip sdk."

[22] "Bringing the love of .net to ethereum," accessed: 2021-06-01. [Online]. Available: https://nethereum.com/

[23] tariq86, "tariq86/rn-sip-app." [Online]. Available: https://github.com/tariq86/rn-sip-app

[24] "Solidity." [Online]. Available: https://docs.soliditylang.org/en/v0.8.5/

[25] Coin Market Cap, "Ethereum." Accessed Aug. 15, 2021. [Online]. Available: https://coinmarketcap.com/currencies/ethereum/