Pak. j. life soc. Sci. (2025), 23(2): 337-345 E-ISSN: 2221-7630;P-ISSN: 1727-4915

Pakistan Journal of Life and Social Sciences

www.pjlss.edu.pk

https://doi.org/10.57239/PJLSS-2025-23.2.00171

RESEARCH ARTICLE

Development of a Data Link Protection Model Based on Algorithmic and Hardware Support

Elena Revyakina[1]*, Andrei Gazizov[2]

[1,2]Don State Technical University, 1, Gagarin Sq., 344002 Rostov-on-Don, Russia

ARTICLE INFO

Received: Apr 24, 2024

Accepted: Jun 13, 2024

Keywords

Blockchain Technology
Information Security
Electronic Signature
Encryption Key
Smart Contract

***Corresponding Author:**

elena.a.revyakina@gmail.com

ABSTRACT

This paper is devoted to the development and research of algorithms and hardware for protecting data transmission channels in information systems. The paper examines in detail the main security threats, analyzes existing methods and means of protection, and suggests new approaches aimed at ensuring confidentiality, integrity and availability of information. The results of the development and implementation of encryption, authentication, and data integrity monitoring algorithms, as well as the design of specialized devices for their implementation, are presented. The results obtained can be applied in the design and modernization of various information systems, which will provide a high level of protection of transmitted data.

INTRODUCTION

A distributed ledger, also known as a shared ledger, is a database that is shared by multiple sites and geographical locations in a peer-to-peer (P2P) network by mutual consent, without the need for a central authority. Each participant becomes a public witness of transactions or data recorded in the distributed ledger. All participants own an identical copy of the data on the shared network, which makes it almost impossible to make changes to the database by one person (Bashir, 2019). A distributed registry is very different from a centralized registry, which is widely used by many institutions. A centralized registry is very susceptible to cyberattacks because it has a single point of failure. On the other hand, the distributed ledger is decentralized, which means there is no need for third-party intermediaries and increased security. Distributed ledger technologies (DLT) require a peer-to-peer network and so-called consensus algorithms to ensure data replication across all nodes. There are various forms of distributed ledgers, one of which is blockchain, which is popular among mainstream users due to its association with Bitcoin and cryptocurrencies. While all blockchains are distributed ledgers, not all distributed ledgers are blockchains. A distributed registry is simply a type of duplicated and synchronized database shared by different regions, servers, and users, without the need for a specific data structure or centralized management.

Different servers interact with each other to store the most recent transaction records (Vashkevich, 2018). A DLT developer can control the structure, privacy, and functionality of a distributed registry, which theoretically makes it anything but decentralized. Advantages of distributed ledgers:

1. Security and transparency-distributed registry technology allows you to make entries in a decentralized registry without involving third parties. The entered records cannot be changed if the organization does not control more than half of the network's computing power. In essence, distributed ledgers are tamper-proof, secure, immutable, and transparent.

2. No intermediaries are required-distributed ledgers significantly reduce operational inefficiency. Eliminating third parties saves you time and money. Such ledgers are good for

financial transactions, as they offer a better alternative to the traditional banking process, which is known for its bureaucracy, high cost, and labor intensity.

3. Security-Distributed ledgers increase the level of security due to their decentralized nature. The information entered in the registry is stored not in one place, but on all computers (nodes) involved in network maintenance. Meanwhile, blockchain technology creates a special type of distributed ledger that typically creates an immutable database shared by a decentralized network, using cryptography to verify and record all actions through a consensus mechanism. Each transaction block is cryptographically linked to its verified predecessor, creating a chain of continuous times tamped data records.

MATERIALS AND METHODS

A blockchain can be defined as a chain of blocks containing certain information. Thus, the blockchain is a registry, i.e. a file that is constantly growing and constantly stores records of all transactions. This process takes place in a secure, chronological (chronological means that each transaction occurs after the previousone) and unchangeable way. Each time an information storage block is completed, a new block is generated (Ivanov, 2017).

Over the past period, blockchain technology has spread and grown in various sectors, and many public and private institutions around the world have begun to think about how they can be applied in practice to develop the services they provide. As mentioned above, a block is a unit of data storage in the blockchain. Each block is securely linked to the previous block using a hash, and if you make changes to one block, you will have to change all subsequent blocks. This method of storage guarantees the immutability of information (Ishchukova et al., 2022).

At the same time, blockchain technology can also be used to create new solutions for data privacy and security. For example, decentralized identity solutions allow users to securely store and access their personal data, while protecting their privacy. Similarly, blockchain-based solutions can be used to authenticate data and protect against data manipulation. By using blockchain technology, users can own their data and have more control over who has access to it. This provides users with greater privacy and security, as well as the ability to monetize their data. Blockchain-based solutions can also be used to improve the security of IoT networks (Minbaleev, 2018).

Using a distributed registry technology, IoT network can be better protected from attackers and data leaks. At the same time, blockchain technology can also be used to create new solutions to ensure confidentiality and data security. For example, decisions for decentralized identification allow users to safely store their personal data and gain access to them, while protecting their confidentiality. In the same way, blockchain -based solutions can be used to verify the authenticity of data and protect against manipulating data. Ethereum is a decentralized blockchain - an open source system that has its own Ether cryptocurrency. ETH works as a platform for many other cryptocurrencies, as well as to perform decentralized smart contracts (Nosirov, 2021).

The purported goal of Ethereum is to become a global platform for decentralized applications, enabling users from all over the world to write and run software that is resistant to censorship, downtime, and fraud. The main difference between Ethereum and Bitcoin is the useof smart contracts. Any action can become such a condition, for example, the transfer of goods, the provision of services, or the appearance of a new record about the shipment of a new batch of computers. For example, a programmer using Ethereumcan program conditions and actions using the built-in scripting language. All records can be verified by interested parties, in this Ethereum is similar to Bitcoin – the system remains transparent and decentralized.

Ethereum has become a pioneer in the concept of a blockchain smart contract platform. Smart contracts are computer programs that automatically perform the actions necessary to fulfill the agreement between several parties on the Internet. They were developed to reduce the need for trusted intermediaries between contractors, thereby reducing transaction costs and at the same time increasing the reliability of transactions (Prai, 2018). The main innovation of Ethereum was the development of the platform, which allowed to perform smart contracts using blockchain, which further enhances the existing advantages of smart contract technology. According to the co

-founder Gavin Wood, the Ethereum blockchain was developed as a kind of "one computer for the whole planet", theoretically capable of making any program more reliable, resistant to censorship and less prone to fraud, launching it on a globally distributed network. The smart contract, also known as the Smart Contract, is a computer algorithm that ensures the control and fulfillment of the obligations of the parties in the process of exchange of assets in blockchain technology. This is a kind of software that monitors the implementation of contractual conditions. Smart contracts are encoded logicians that change digital assets when certain events occur. They use conventional operators "if" where "if" is a requirement to launch certain actions. When the smart contract works on the blockchain, it is automatically fulfilled when all conditions are fulfilled (Raval, 2020). The entire structure of smart contracts is based on checking and confirming by many connected computers. This provides the following advantages of smart contracts:

Safety. Smart contracts provide a high level of security through the use of proven logic and data encryption.

Openness. All smart contracts are public, which means that their code is available for verification and audit to all network participants.

Trust. The use of smart contracts allows participants in the asset exchange to trust the process, as it is fully automated and does not allow manipulation by third parties.

Minimal errors. Thanks to proven and reliable logic, smart contracts are virtually free of the possibility of errors.

However, it is worth noting that the process of executing smart contracts requires the coordinated operation of all computers on the network, which can be expensive. Some blockchain platforms are limited in their capabilities, but Ethereum stands out.

Thanks to the Ethereum Virtual Machine (EVM), which is completely universal software, Ethereum allows you to create and run any application. A special feature of smart contracts on the Ethereum platform is that each smart contract has its own unique address in the blockchain. Instead of inserting code into each contract, the network node executes a transaction that creates and binds a unique address to the contract (Sokolkov, 2020). After that, the contract becomes an integral part of the blockchain and its address remains unchanged. Thus, smart contracts on the Ethereum platform run continuously until the operation is successfully completed. Metamask is a web browser extensionthat provides a convenient way to interact with blockchains and perform transactions on the Ethereum network. It is a powerful tool that allows users to manage their cryptocurrency assets and interact with decentralized applications (dApps) directly from their browser. To use Metamask, you must first install its web extension in the selected web browser. After installation, follow the configuration process, which includes creating a new wallet or importing an existing one, creating a password for the wallet, and saving the secret phrase in a secure location. These steps guarantee the security of the user's wallet and provide access to it only to the owner (Talapina, 2020).

One of the main functions of Metamask is the management of cryptocurrency assets. After setting up the wallet, the user can add various Ethereum tokens, such as Ether (ETH) and ERC-20 tokens, to their wallet. Metamask provides a user-friendly interface for viewing asset balances, sending and receiving funds, and tracking transaction history. Metamask allows users to interact with decentralized applications (dApps) directly from their browser.

The user can automatically connect their Metamask wallet to the DApp, which provides secure and easy authentication. Further, the user can use their cryptocurrency assets to participate in various functions and operations offered by the DApp. To perform transactions on the Ethereum network, the user must sign them using their Metamask wallet (Salnikova, 2019).

The signature of a transaction is an essential part of the safe and authentic execution of transactions. Metamask protects private keys and provides a convenient interface for signing transactions before sending them to the Ethereum network. Metamask also offers a number of advanced features that enhance your wallet's usability and security. This includes the ability to add multiple wallets, manage Ethereum networks (test and main networks), use custom gas

prices, and many other settings that allow users to customize Metamask to suit their individual needs. Metamask is an essential tool for Ethereum users, providing convenient and secure access to the blockchain and cryptocurrency assets.

METHODS

As a result of the study of the basic principles of blockchain technology, as well as the analysis of existing solutions in identity management, the following conclusions can be drawn: consequently, it seems appropriate to develop an employee identification system based on blockchain technology. This will increase the level of security, ensure transparency and reliability of the identification process, and protect confidential data from threats and manipulation. The AS-IS model, in particular the identification system with password storage on a dedicated server, has its drawbacks, namely:

Security risks: if hackers manage to gain access to the server, they will be able to get all the passwords stored on it and use them for unauthorized access to systems and data;

Privacy risks: if passwords are stored on the server in clear text or with a low level of protection, this may lead to the disclosure of confidential information about users.

Insufficient fault tolerance: if the server crashes, users may lose access to their accounts and data, which can lead to serious problems.

Management complexity: if the number of passwords on the server is significant, managing them can become a difficult task, especially if you do not use automated account management tools. In general, storing passwords on a dedicated server can be an unsafe and inefficient way to manage access to systems and data, so we recommend using more secure alternatives. User identification through the block chain has a number of advantages over classical identification methods (Smith, 2002).

Based on the Above, You can Make Requirements for the Designed Systems:

Security. The blockchain uses cryptographic methods to ensure data security and verify the authenticity of user accounts. This allows you to prevent fraud and unauthorized access to your accounts.

Decentralization. The blockchain does not have a centralized point of control, which makes the system more resistant to attacks and more reliable for storing user credentials.

Privacy policy. The blockchain can be used to create anonymous accounts that do not require users to provide personal information.

Speed up the identification process. Blockchain allows you to quickly identify users, which can improve the experience of using the site and increase its attractiveness to users.

Efficiency. The use of blockchain can reduce the cost of maintaining a dedicated server, which can be an important factor for companies with limited budgets.

The process of identifying employees on the site using blockchain technology. includes the following stages:

1. Identification request: An employee sends an identification request on the organization's website.

2. Key generation: After receiving the request, the system generates public and private keys for the employee.

3. Data hashing: an employee provides their data (for example, full name, position, etc.), which is hashed to the blockchain.

4. Creating a block: the system creates a new block in a blockchain with has it with a hasn for an employee, an open key and a tag of time.

5. Verpret check: the system checks the authenticity of the blockchain unit using the consensus mechanism such as Proof of Worki Proof of Stake.

6. Issuance of access token: if the unit has passed authenticity, the system issues an access token that can be used by an employee for authorization on the organization's website.

7. Authorization on the site: The employee uses the issued access token to authorize on the organization's website.

8. Blockchain update: after successful authentication, the system updates the blockchain, adding information about the latest use of access token.

The main difference between user identification with password storage on the server and through the blockchain is how user data is stored Cherckesova et al., 2024. When storing passwords on the server, users enter their usernames and passwords on the site, which then passes them to the server for verification. Passwords are stored in a database on the server, which can become a target for hackers or intruders who can gain access to users ' personal information. This can lead to serious consequences, such as leakage of personal data and financial resources. If a blockchain is used to identify users, the user's data is stored on a distributed database in the form of blocks. These blocks are protected by cryptographic algorithms, and any changes in the blocks are immediately displayed in all copies of the database (Cherckesova et al., 2024). This makes it impossible to change or falsify user data, which provides a higher level of security. In addition, the blockchain also allows you to manage access to user data and configure access rights for each user. This allows for more flexible and precise access control, which can be especially important for mission-critical infrastructure enterprises where security is the highest priority.

Algorithm of the Proposed Approach

In decentralized networks, the identification of users or network nodes is carried out using an electronic signature. Participants in such networks, whether users or nodes, create a couple of keys - a secret and public keys. The information obtained on the basis of a public key allows you to identify the participant in the future. In the Ethereum blockchain, the address (also called the "account") is 160 bits of the hash of the public key, i.e. To create your own address in Ethereum, you need::

1. Create private_keya private_key/public_key key pair for use with the ECDSA scheme (Ethereum uses the secp256k1 elliptic curve).

2. Taking the last 160 bits from the public key hash using the Keccak-256 hash function (Ethereum uses Keccak-256).

3. The resulting value is an address in the Ethereum network, for example: 0xDC25EF3F5B8A186998338A2ADA83795FBA2D695E. The owner of the address must confirm their ownership of it by signing the data with their private key. For example, each cryptocurrency transaction from a specific address contains a public key and an electronic signature, which indicate that the transaction could have been created only by the owner of the secret key on which the public key and address depend.

This method of confirming ownershipof information associated with a secret key is widely used. For example, issuing and verifying HTTPS certificates is based on the same scheme: an HTTPS certificate is a signed hash of the public key with additional data, such as the domain name and expiration date.

Also, such procedures include working with a "qualified EP" (electronic signature), where the registration authority registers a public organization key along with information, such as the name and the TIN. Subsequently, the organization sends reports in electronic form, signing it using its secret key. Taking into account the foregoing, public blockchain, which for the user can act as a cloud service that stores address and related information, is an excellent example of a system that can store information for user identification (Razumov et al., 2023).

In this scheme, all users are authenticated using a private key, while public keys (or, more precisely, their hashes) are stored on the block chain. One of the attractive features of this scheme is that it is the hashes of public keys that are real addresses in the blockchain, and every action in the blockchain, every valid transaction, is proof of ownership of a certain address (i.e., a certain public key). At the same time, all the important properties of blockchains, such as resistance to

attacks, the inability to restrict access and the complete absence of secret information, make it ideal for storing information necessary for user identification. The blockchain is accessible from anywhere, and you can get proof of the existence of an address, as well as confirmation of its validity, from any node of the blockchain anywhere in the world. It is impossible to block all blockchain nodes in the world, and it is impossible to change key information unnoticed, since it is subject to public verification

Therefore, any algorithms that require proof of ownership of the address are perfectly combined with the use of smart contracts, and the address becomes a unique identifier (UID) of the user, which does not require storing other types of shared secrets. In addition, the availability and ability to check such a UID database allow the service to dispense with backing up security information, and in case of failures, it is easy to restore the account database. For public blockchains, a server that provides verifiable information can be launched anywhere without any registration and the need to own crypto currency, since it will simply receive updates from other nodes through the P2P network. This scheme is widely used and requires the user to perform simple one-click actions, similar to" Sign in with Google", but without transferring any data to third-party resources, except for the site itself. Registration in this system is the process of adding information to the blockchain, which connects a public address in the blockchain with a unique identifier and additional data. For a better understanding, we can say that a record is created in a key-value database inside the blockchain. In this record, the key is the public address, which is a unique user identifier in the blockchain. The value associated with this key contains additional data, which may include various user information or other important information. This data structure makes it possible to efficiently store and link information to public addresses on the blockchain. The blockchain, being a decentralized and reliable system, ensures the safety of these records and their availability to all network participants. For this identification system to work, the user must have an Ethereum wallet address. To do this, the user registers in MetaMask and becomes the owner of an Ethereum wallet. The unique address is 0x630666F49e171803Ba1A664eEdAa7450fa4c9DaE.

The following functionality is implemented on the created web page: the SIGN UP button-registers the user, writing down the data to the Sign In-in-line smart contract-allows the user to log in and gain access to the resources of the site. The web page was written using the JavaScript language and CSS styles. The source code was written among Visual Studio Code development and has a structure substituted in Figure 1.

Figure 1. Code Components

Each of the individual files is imported into a shared file called App.js, which is the basis of the web page. To register, the user must fill out a form with a username, password and a 6-digit code, and the Ethereum address isextracted directly from the wallet.

This address is associated with the user name for signature generation via the web3 function token, and the generated signature is hashed (hash1). The password is associated with a 6-digit code to generate another hash (hash2). The two hashes are combined to create the final hash, which is stored inthe smart contract. To log in, the user must connect to the blockchain with the same address that was used during registration, and fill out the login form with the correct username, password, and 6 - digit code. The user's login data is stored as a hash on the blockchain through a smart contract, and each time the user requests access to the website, the hashobtained

from the credentials provided by the user is compared with the hashstored in the smart contract. If they match, then the user has the right to access the website, if not, then access is denied. In this scheme, the user must be associated with the Ethereum address that will be used duringthe registration process, since this address is used to generate a hash that will later be used for logging in. Figure 2 shows all the steps to create a hash of a user's login data from a username, password, 6-digit code, and an Ethereum address.

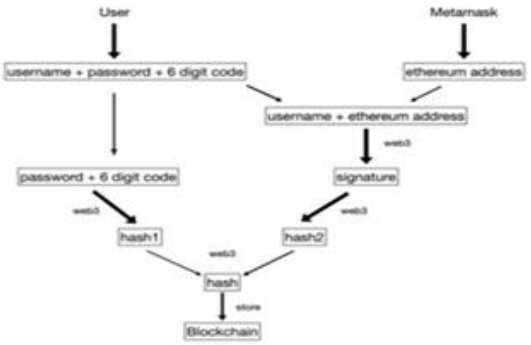

Figure 2. Creating a Hash

When registering in this system, the wallet address is associated with the user name using the MetaMask plugin to generate a signature, and the generated signature is then hashed (hash1). After the Ethereum address is extracted directly from the wallet and linked to the user's name, the MetaMask plugin asks you to confirm the user's registration. To register a user's address in a smart contract, the register() function is called.A snippet of the source code is shown below: function register(string memory _signature) public { require(user[msg.sender].userAddrss == addrss(0x0000000000000000000000000000000000000000), "already registered"); user[msg.sender].signatureHash = _signature; user[msg.sender].userAddrss = msg.sender; nbOfUsrs++; }

RESULTS AND DISCUSSION

After successful registration, the data was recorded in the smart contract. This smart contract is then deployed on the Ethereum blockchain network since this action writes information to the blockchain, the user needs to pay transaction costs in the native cryptocurrency Ethereum (ETH), and despite the fact that the amount of data and commission are insignificant, they are still not zero. Therefore, attracting users to blockchain projects becomes a separate task.

However, there is a good side – first, hacking the site will not give the attacker access to the user's account, since in the blockchain everyone is responsible for their own security. Secondly, registration actions are extremely rare, and a single payment allows the user to use the account for as long as they want, while information about it does not disappear, and the user is responsible for it , not the project. This is the principle of Self-Sovereign Identity. Also, the gas fee is paid only for registration, the login process does not require writing data to the smart contract. Thus, to store a smart contract in the blockchain network, you only need to pay once, and store your identification data securely. Confirmation of signing data with an account does not contain gas, and you will not have to pay for logging in.

Now any node in the blockchain can get information by accessingthe smart contract, which stores the data of registered users by key (for example, 0x13668Ecf257cC15c381b461B9fEDaB5D451c8F7F).

To access the site, the user enters a username, password, and a 6-digit code, after which the MetaMask plugin asks you to confirm ownership of the address by signing a message. After that, the hashobtained from the credentials provided by the user is compared with the hashstored in the smart contract. If they match, then the user has the right to access the website, if not, then access is denied. Users who work with Ethereummust use software that signs transactions.

Therefore, if the user has previously registered or already used Ethereum in the browser, they can sign the data with their private key.

If you use the Metamask browser extension, you can do this in one click. When calling web3 library functions. For js that offer to sign a challenge, the user receives a notification from the Metamask extension asking them to sign the string. The string to sign is provided by the backend. The backend temporarily stores this string to prevent replay attacks if an attacker intercepts the signed message and tries to resend it. The challenge is then sent along with the signature to the backend, which verifies the authenticity of the signature, removes the one-time challenge from its database (excluding replay attacks), and issues the user JWT authorization token. After that, the web application functions as usual. This type of authentication can be implemented using different blockchains, contracts, and different types of signatures.

CONCLUSION

Thus, the flexibility and diversity in the choice of technology allow you to implement authentication in various scenarios that meet various needs and safety requirements. As a result, Proof-OF-Concept was obtained by the decision of the decentralized system of identifying employees on the company's website using the Ethereum network and Metamask plugin. The decentralized system is not subject to the censorship of large structures, such as Google or Facebook. If necessary, censorship, the site must implement it independently, within the framework of its own system, without affecting the user access to other systems. The solution is scalable, since the data with the data is distributed, and anyone can add a new node at any time. The introduction of such a solution for owners of websites does not require significant efforts and difficulties.

Acknowledgments

The article was prepared within the framework of the Russian Science Foundation grant No. 24-28-20502 “Creation of a prototype of a digital catalog of spa and resort architectural objects of the Soviet period using Deep Mapping technology”.

REFERENCES

Bashir I. Blockchain: architecture, cryptocurrencies, development tools, smart contracts. Moscow: DMK Press, 2019, pp: 1-538.

Cherckesova L, Revyakina E, Buryakova O, Gazizov A. Creation of an encryption algorithm resistant to attacks through side channels of leakage. E3S Web of Conferences, 2024, Vol. 583, p. 06011.

Cherckesova L, Revyakina E, Safaryan O, Porksheyan V, Kazaryan M. Analysis of the possibilities of carrying out attacks on the functions of transferring control to operating system console using active intelligence methods. International Research Journal of Multidisciplinary Scope (IRJMS), 2024, Vol. 5, No. 2, pp: 516-534.

Ishchukova EA, Panasenko SP, Romanenko KS, Salmanov VD. Cryptographic foundations of blockchain technology. Moscow: Black and White, 2022, pp: 1-300.

Ivanov VV, Lubova ES. Authentication and authorization. Problems of modern science and education, Moscow, 2017.

Minbaleev AV, Safronov E. Legal nature of blockchain. Bulletin of South Ural State University. Series: Right, 2018, No. 2, p. 94.

Nosirov ZA, Fomichev VM. Analysis of blockchain technologies: the basics of architecture, examples of use, development prospects, problems and disadvantages. Systems of Management, Communications and Security, 2021, No. 2, pp: 37-75.

Prai N. Blockchain. Development of applications. St. Petersburg: BHVPETERBERG, 2018, pp: 1-256.

Raval S. Decentralized applications. Blockchain technology in action. Moscow: HarperCollins Pub Ltd, 2020, pp: 1-192.

Razumov P, Lyashenko K, Checkesova L, Revyakina E, Yengibaryan I, Revyakin A. Development of a system for protecting against DDoS attacks at the L7 level of the OSI model-HTTP Flood. E3S Web of Conferences, 2023, Vol. 402, p. 03008.

Salnikova AV. Blockchain technology as an instrument of copyright protection. Actual problems of Russian law, 2020, Vol. 15, No. 4 (113), pp: 83-89.
Smith RE. Authentication: from passwords to open keys. Moscow: Williams, 2002, pp: 1-432.
Sokolkov GS. Review and comparative analysis of token-based and Session-Based Authentication. Collection of works of the conference, 2021, No. 1, pp: 204-207.
Talapina EV. The use of blockchain in public administration: Prospects for legal regulation. Issues of State and Municipal Administration, 2020, No. 3, pp: 96-113.
Vashkevich AM. Smart contracts: what, why and how. Moscow: Simplorer, 2018, pp: 1-89.
ort and Health Science, 6(4), 395-403.