Global Scientific JOURNALS

GSJ: Volume 12, Issue 6, June 2024, Online: ISSN 2320-9186
www.globalscientificjournal.com

Enhancing Patient Privacy And Data Security in Healthcare Information Systems with Keycloak and Blockchain

Michael Edem Amekuedi[1,2], Solomon Danso Danquah[2], John Lansana[3], Derrick Asante[4]

[1]Engineering Research Center of Digital Forensics, Ministry of Education, Nanjing University of Information Science & Technology, Nanjing, 210044, China.

[2]School of Computer and Software, Nanjing University of Information Science & Technology, Nanjing, 210044, China.

[4] Shenyang Institute of Engineering, No.18 Puchang Road, Shenbei New District. Shenyang, P.R.China 110136.

Keywords: Patient Data Privacy, Health Data Security, Blockchain Technology, Keycloak, Healthcare Informatics.

Abstract

In the digital age, the protection of patient data privacy is a critical concern, particularly with the rise of digitized healthcare records. This paper proposes integrating Keycloak, an open-source identity and access management system, with blockchain technology to enhance the security of patient data. The study identifies the limitations of existing data protection methods and introduces a new framework that combines the strengths of both Keycloak and blockchain. Keycloak ensures robust authentication and authorization, restricting access to sensitive patient information to authorized personnel only. Blockchain technology, with its decentralized and immutable characteristics, provides a secure and transparent method for recording data access and transactions, thereby preventing unauthorized changes and maintaining data integrity. The framework offers a multi-layered security approach, utilizing Keycloak's advanced access control along with blockchain's tamper-resistant properties. This combination aims to improve data privacy, reduce the risk of data breaches, and ensure compliance with regulations such as HIPAA and GDPR. The study includes a thorough analysis of current healthcare data challenges, the architectural design of the proposed system, and a detailed implementation strategy. A case study demonstrating the framework's application in a healthcare setting shows that integrating Keycloak and blockchain enhances patient data privacy and boosts transparency and trust in the healthcare system. Overall, this work contributes to healthcare informatics by providing a scalable, secure solution for managing patient data, paving the way for future advancements in healthcare data security.

1. Introduction

The integration of information technology into healthcare has resulted in substantial changes, primarily via the use of healthcare information systems (HIS). These systems integrate diverse data processing techniques into a uniform framework, improving the quality of patient care and operational efficiency. Health information systems (HIS) go beyond being digital adaptations of conventional medical methods. They showcase the fusion of cutting-edge technology with therapeutic treatment, providing all-encompassing solutions for handling patient data, financial transactions, and administrative procedures. Health Information Systems (HIS) enhance healthcare experiences and provide real-time data access by integrating subsystems like Electronic Health Records (EHRs), Radiology Information Systems (RIS), and Laboratory Information Systems (LIS). This integration supports informed decision-making and individualized patient treatment. The growing digitalization of healthcare, meanwhile, presents difficulties with data security and patient privacy.

Data breaches provide substantial hazards, such as identity theft and financial crime, which need the implementation of strong security measures, training initiatives, and data management systems. Identity and access management (IAM) technologies, such as Keycloak, are essential for guaranteeing secure user authentication and authorization. Keycloak's versatility and functionalities such as user session management and single sign-on (SSO) contribute to the safeguarding of critical health information. Furthermore, blockchain technology, renowned for its unchangeability, distribution, and openness, presents auspicious remedies for securely handling healthcare data. Blockchain technology has the ability to provide a clear and safe record of individuals' medical history, which may possibly transfer the authority of accessing data from healthcare professionals to the patients themselves.

The future of healthcare shows great promise with these technology developments, which jointly improve effectiveness, safety, and patient confidence. Comprehending and incorporating these elements are crucial for successfully navigating the digital revolution in the healthcare industry.

1.1 Contributions

This paper addresses the critical issue of patient data privacy in the digital age by proposing a novel framework integrating Keycloak's access management with blockchain technology. This innovative approach tackles limitations of existing methods by combining robust authentication and authorization with tamper-proof data transaction recording, offering a multi-layered security solution. The framework aims to improve data privacy, reduce breaches, and ensure compliance, ultimately contributing to healthcare informatics by providing a scalable and secure way to manage patient data.

1.2 Organization

The remainder of this paper is organized as follows. Section 2 describes the proposed keycloak and its integration with blockchain in healthcare information systems. Experiment results are shown in Section 3, and Section 4 presents the conclusion of this paper.

2. Related Works

This section will initially delve into healthcare information systems storage, retrieval, and management of patient data. Complexity has increased in health information system development. These systems now support imaging, lab findings, and EHRs.

2.1 Discussion

Despite these hurdles, recent research overwhelmingly emphasizes the substantial benefits over drawbacks. It underscores the urgent need to bolster privacy and security in healthcare information systems, highlighting the significance of this innovation amidst evolving threats and vulnerabilities.

Health information systems contain extensive personal data, including medical and financial records, requiring robust identity and access management solutions. Keycloak, an open-source IAM(identity and access management) framework, is a powerful tool for healthcare organizations to ensure both data security and access efficiency.

2.2 Proposed Method

The proposed solution for enhancing the security of the Health Management System comprises two parts.

1. The User: the individual who communicates their need for resources through the Health Management System.
2. Blockchain and Keycloak: Keycloak acts as the identity manager within the Health Management System, handling all user transaction requests by authenticating and authorizing users. It assigns unique tokens for secure access to specific data and functionalities. Meanwhile, blockchain technology manages all transactions within the system, ensuring secure, transparent, and immutable record-keeping.

This integration of Keycloak and blockchain provides a comprehensive solution for secure identity management and transaction handling in healthcare systems as presented in Fig. 1.

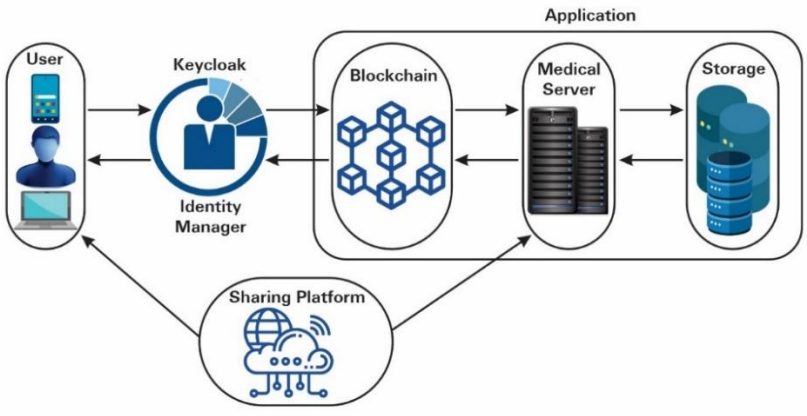

Figure 1: Diagram depicting the proposed method

Fig. 2 shows the authentication procedures for the proposed solution using keycloak. To initiate transactions within the Health Management System, which is configured with keycloak, all transactions have to be authenticated by the keycloak server which is the identity manager.

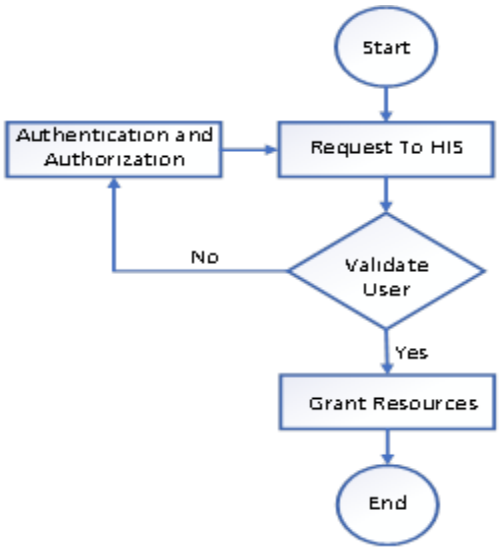

Figure 2: Diagram depicting the flowchart of user authentication with keycloak

The first step in the user verification process is for the user to enter the Health Management System through the Login interface. This user must possess the appropriate credentials. The user is required to give the essential credentials for authentication and authorization using a variety of avenues (such as a username and password, SSO, or MFA, for example. When the user submits their details, the Health Management System will send the request back to the keycloak server. This is because the user does not have a token allocated to them. Within the JWT that the keycloak generates, the user's identity and access claims are included. Everything that the user can access within the system will be determined by the access claims used. The workflow for authentication is formed by this form.

2.3 Research Design

The proposed method employs a mixed-method research design, integrating both qualitative and quantitative approaches. This methodology aims to comprehensively understand current Health Management System (HMS) solutions and assess the impact and efficiency of integrating Keycloak and Blockchain components. Qualitative methods, including interviews and focus group discussions, will gather insights from healthcare professionals, system administrators, and IT experts.

A strategy based on case studies will be utilized to guarantee that the enhancements that have been offered are applicable in the real world. The introduction of Keycloak and Blockchain technology into an existing HMS will be led by a particular healthcare institution that has been chosen to act as the focal point. A full analysis of the system's performance, user experience, and security aspects will be included in the case study. This analysis will be conducted both before and after the integration in question.

2.4 System Landscape

This section, the proposed method entails design principles, components, and relationships between the various modules will be broken down in further detail. A thorough discussion of how Keycloak will manage authentication and authorization, as well as how Blockchain will be deployed to ensure data integrity and transparency.

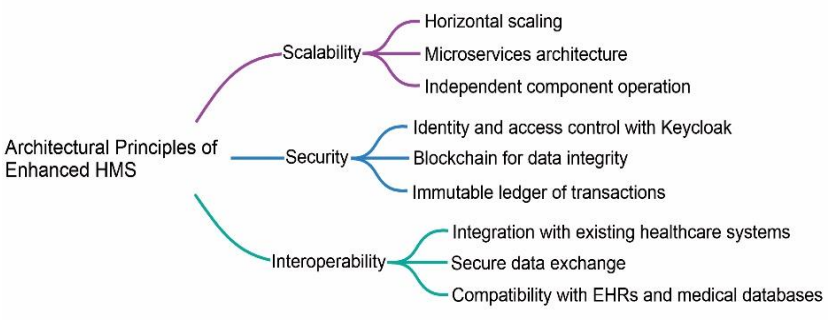

Figure 3: Diagram depicting the Architectural Principles of the proposed solution

Fig 3, extrapolates the architectural design of the proposed Health Management System (HMS) is guided by principles of scalability, security, and interoperability. It is built to handle growing healthcare data demands while ensuring robust security for sensitive information. A key feature is interoperability, allowing seamless integration with existing healthcare infrastructure.

2.4.1 System Components

The enhanced Health Management System (HMS) architecture consists of several key components designed to improve functionality, security, and scalability. The User Interface (UI) offers a responsive and intuitive front-end for healthcare professionals and administrators to manage patient records and appointments. The application layer, integrated with Keycloak, facilitates secure authentication and authorization, ensuring users can only access information relevant to their roles.

The microservices architecture breaks down the system into smaller, independently deployable services, enhancing scalability, fault isolation, and ease of maintenance. Keycloak handles identity and access management, providing robust authentication mechanisms like SSO and multi-factor authentication, and ensuring role-based access control (RBAC). Blockchain integration secures healthcare data and transactions, creating an immutable, transparent record of all activities and providing a tamper-proof audit trail. The database layer combines relational databases with distributed ledger technology to ensure data consistency, reliability, and accessibility, with Blockchain adding an extra layer of security to data storage, thus enhancing trust in the system.

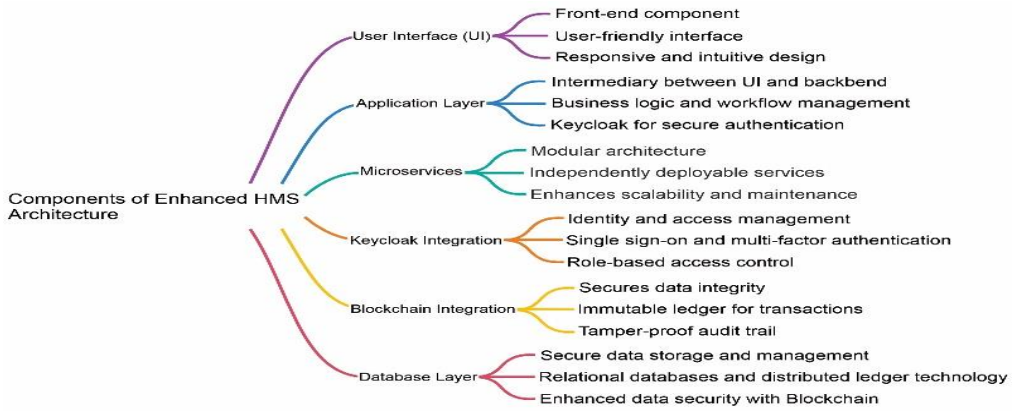

Figure 4: Diagram depicting the components of the Enhanced HMS architecture

Fig. 4 presents the components of the enhanced Health Management System (HMS) architecture, highlighting key elements such as the User Interface (UI), Application Layer, Microservices, Keycloak Integration, Blockchain Integration, and Database Layer. Each component is described with specific functions, emphasizing aspects like user-friendly design, modular architecture, secure authentication, data integrity, and enhanced data security. Together, these components work to improve the system's scalability, maintenance, and overall security.

2.5 Keycloak Implementation

The proposed method provides guide with detailed instructions and diagrams for establishing a robust security framework.

1. Pre-requisites:
   - Download and install Keycloak, ensuring HMS server compatibility.
   - Configure the Keycloak server database (preferably PostgreSQL) for user and realm data.
   - Understand user authentication needs, considering local and LDAP users.
2. Initial Setup:
   - Start Keycloak by running the batch file (`bin/kc.sh start`) on the desired server port.
   - Configure the master realm via the admin console (e.g., `https://127.0.0.1:8090/auth/admin`) to manage settings and other realms.
   - Create a new realm within the master realm for HMS, configuring user federation (importing LDAP users or managing local users).
3. Integration into HMS:
   - Create a client for the HMS, specifying base URL, redirect URLs, and access type (confidential or public).
   - Configure client adapters suitable for the system's framework (e.g., Tomcat adapter for Java).
   - Enable OpenID Connect (OIDC) within the client configuration, specifying endpoints and necessary claims for user authorization.

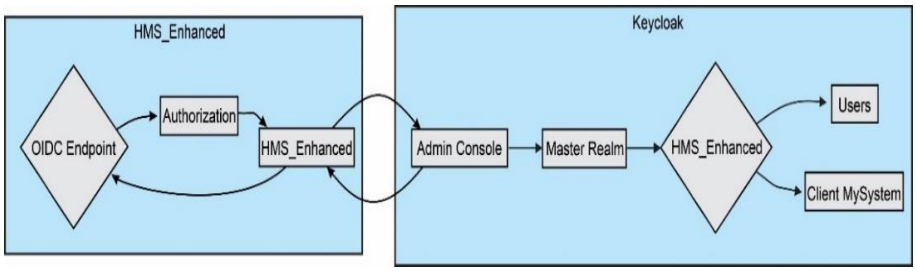

Figure 1: Diagram depicting the overview of the architecture for the HMS

Fig.5, highlights the key components: Keycloak realms, users, client configuration, and the OIDC flow for user authentication and authorization within your system.

4. Advanced Security Configuration:
   - Fine-grained Access Control: Define roles and assign permissions within the client for granular access control for different user groups.
   - Single Sign-On (SSO): Configure SSO for seamless access across multiple applications integrated with Keycloak.
   - Multi-factor Authentication (MFA): Implement MFA for additional security during user login.

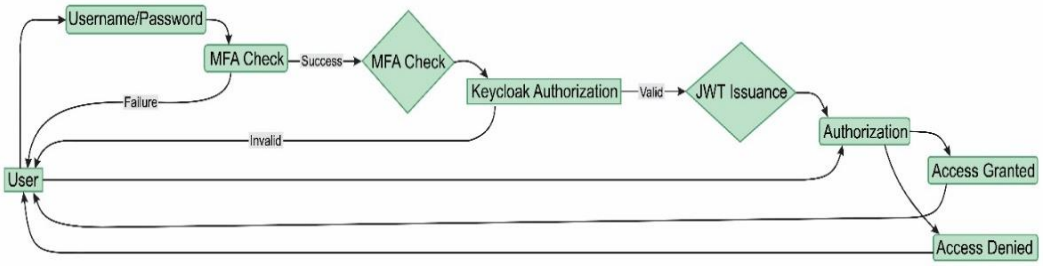

Figure 2: Diagram depicting the advanced security OIDC setting

Algorithm 1: Keycloak

Input:
username: Username or email entered by the user.
Password $(p1)$: Password entered by the user.
Output:
True: Authentication successful.
False: Authentication failed.
User_role : role assigned to user
User_token: token assigned to user

continued

Process:
1. For username in Db
2. TransformFunction $= \sum_{i=1}^{n} p_i$
3. transformed_password = TransformFunction(password)
4. if Keycloak.Authenticate(username, transformed_password):
5. return True, User_role, User_token
6. else:
7. return False
8. end if
9. end For
10. return False

2.6 Blockchain Implementation

The enhanced HMS with Keycloak and Blockchain uses a permissioned Blockchain for its focus on security and privacy. This approach provides increased security and regulatory compliance, suitable for sensitive data. The implementation involves four key stages:

1. Blockchain Platform:
   - The chosen platform is Hyperledger Fabric, known for its flexibility, modularity, and strong community support.
2. Network Setup and Governance:
   - Node Infrastructure: Determine the number and location of nodes based on network size and performance needs; a cloud-based solution is recommended.
   - Membership Management: Define participant onboarding criteria and establish governance rules for managing permissions, consensus mechanisms, and dispute resolution.
   - Security Configuration: Utilize the security configuration done within Keycloak.
3. Data Modeling and Integration:
   - Use Keycloak access tokens for authentication and authorization.
   - HMS API interacts with the Blockchain network using the required SDK for simplified communication.
   - Implement Fully Homomorphic Encryption to protect healthcare information.
4. Smart Contract Development:
   - Use smart contracts for automating specific tasks based on predefined rules.

Algorithm 2: AddTransactionToBlockchain

Input:
transactionData: String - The data of the transaction to be added.
previousBlockHash: String - The hash of the previous block in the blockchain.
blockchain: List<Block> - The list representing the blockchain.
Output:
blockchain: List<Block> - The updated blockchain after adding the transaction.

continued

```
Steps:
1. Begin AddTransactionToBlockchain(transactionData,
previousBlockHash, blockchain):
2.    newBlock.
3.    previousHash = previousBlockHash.
4.    data = transactionData.
5     timestamp = current timestamp.
6.    nonce = 0
7.    while hash is valid
8.       nonce =+ nonce
9.       hash=HashFunction(data+timestamp+previousHash+nonce)
10.      End while
11.      If hash.first_character not 0
12.         Newblock = hash
13.         Blockchain.append(Newblock)
14.     End if
15. End AddTransactionToBlockchain.
```

Keycloak manages access control, while Blockchain smart contracts automate workflow and maintain data integrity as presented in Fig. 7.

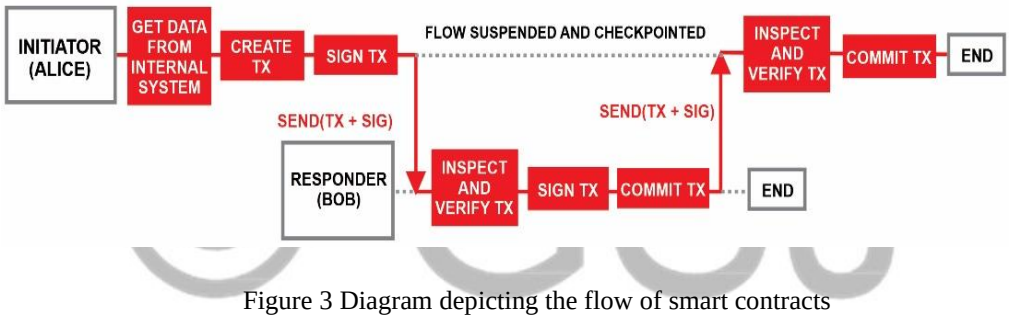

Figure 3 Diagram depicting the flow of smart contracts

3. Experimental Results

The analysis of Keycloak demonstrates how it manages data tampering, access control, and compliance. Understanding the entire process involves acknowledging HTTP status codes, as defined by RFC 9110, which indicate the success or failure of specific HTTP requests. These status codes provide essential feedback on the outcome of interactions within the system.

Table 1 depicting the HTTP status codes

HTTP STATUS CODE	DEFINITION
100 to 199	Informational responses
200 to 299	Successful responses
300 to 399	Redirection messages.
400 to 499	Client-side error responses
500 to 599	Server-side error responses.

1. Information Responses:
   - 100 Continue: The server has received the initial request part, and the client can continue with the remainder.
   - 101 Switching Protocols: The server is willing to change the application protocol being used on this connection.
2. Successful Responses:
   - 200 OK: The request was successful, and the server provided the requested information.
   - 201 Created: The request was fulfilled, creating a new resource.
   - 204 No Content: The server successfully processed the request but has no additional content to send.
3. Redirection Messages:
   - 301 Moved Permanently: The requested resource has been permanently moved to a new location.
   - 302 Found: The requested resource has been temporarily moved to another location.
4. Client-Side Error Responses:
   - 400 Bad Request: The server cannot understand the request due to a client error.
   - 401 Unauthorized: The request requires user authentication.
   - 403 Forbidden: The server understands the request but refuses to authorize it.
   - 404 Not Found: The requested resource could not be found on the server.
5. Server-Side Error Responses:
   - 500 Internal Server Error: A generic error indicating the server encountered an unexpected condition.
   - 502 Bad Gateway: The server, acting as a gateway or proxy, received an invalid response from the upstream server.
   - 503 Service Unavailable: The server is not ready to handle the request, often due to maintenance or temporary overloading.

3.1 Data Tampering Analysis

The proposed system addresses data tampering using Keycloak and Blockchain technology. Data tampering, defined as unauthorized alteration or manipulation of health data, is prevented by ensuring all user transaction requests are checked for authorization and authentication. If a user lacks an authenticated token and user role, the transaction request is aborted as shown in Fig. 7 from section 2.

APIs implemented in Java SpringBoot connects individual endpoints to user roles defined in Keycloak, enhancing data tampering prevention within the Health Information System. As depicted in Fig. 8, the @PreAuthorize annotation in SpringBoot uses the has_Role attribute for client_user and client_admin roles. These roles are created in the Keycloak identity manager, and through Java SpringBoot mapping, users can only access and manipulate data according to their access level.

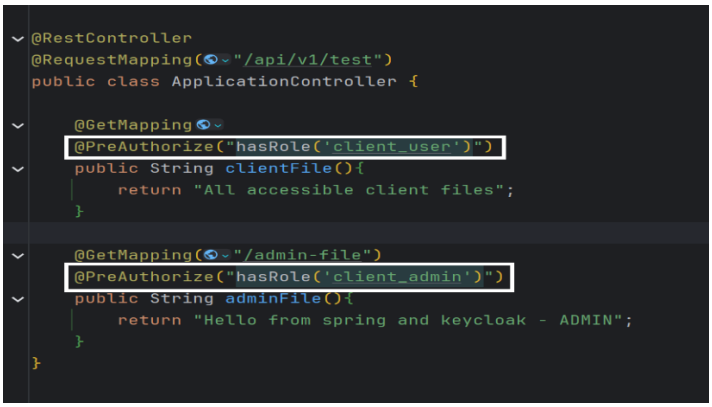

Figure 4: Diagram depicting the role mapping scenario in Java SpringBoot with keycloak

Fig. 9 illustrates how the system blocks unauthorized access to endpoints by enforcing user access levels. Even if a user is authenticated and approved, they cannot control an endpoint if their access level does not permit it. This mechanism protects health data within the Health Information System from unauthorized alterations, ensuring data integrity based on user roles and permissions.

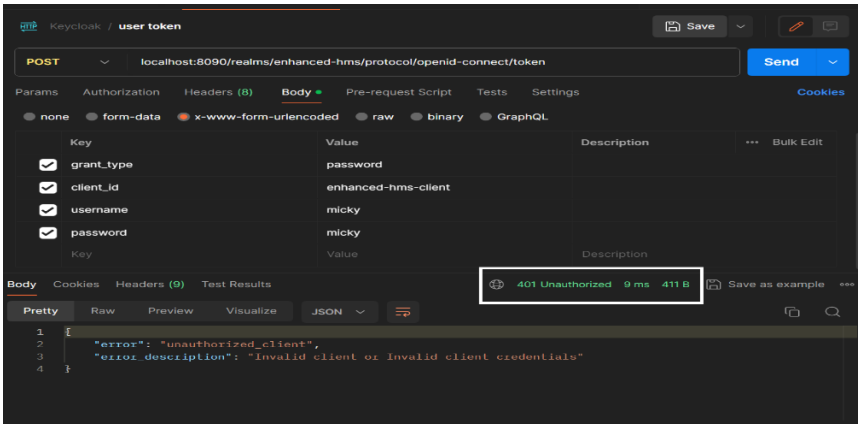

Figure 5: Diagram depicting the prevention of unauthorized access to data

3.2 Access Control

Access control is crucial for security, determining who can access specific data, apps, and resources under certain conditions. It relies on authentication and authorization to verify user identities and grant appropriate access based on factors like device, location, and role. This prevents unauthorized access to sensitive information and reduces the risk of data breaches.

Keycloak enforces access control policies by ensuring configured applications comply. Unauthorized access is blocked, and authentication is managed using access tokens. These tokens contain detailed user information, including access levels. When a user with valid credentials is authenticated, they receive an access token (indicated by an HTTP 200 status code). This token allows the user to perform permitted transactions within the Health Information System.

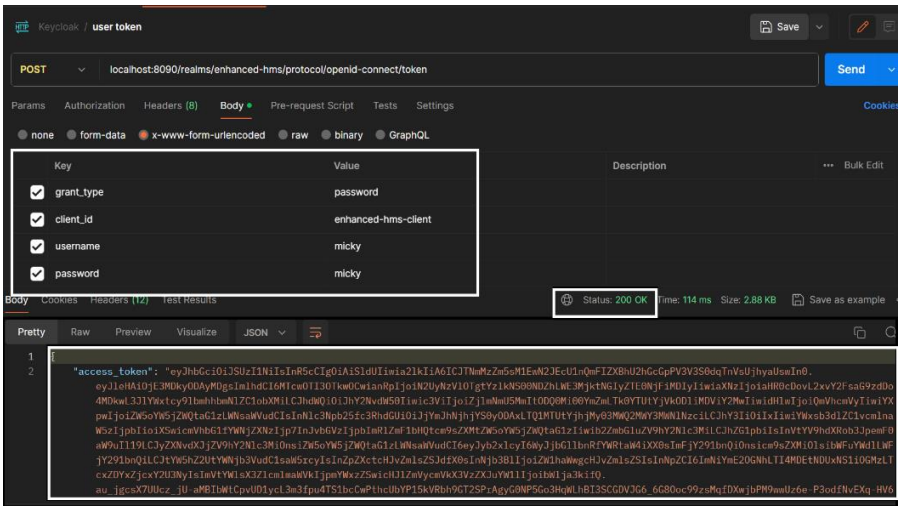

Figure 6 Diagram depicting access token generation for authentication

To fully understand a long access token, the Open ID token API can be used to extract its components. Fig. 4 illustrates this extraction process, revealing that the access token includes the expiration time in milliseconds, the user's UUID, the connected Keycloak client, the refresh token stamp, the client profiling type, and a session created in Keycloak for user activity tracking. This information is available in the token header and can be accessed through SpringBoot Security configuration. The payload of the token, however, remains hidden to ensure user privacy, while Keycloak maintains activity footprints throughout the session.

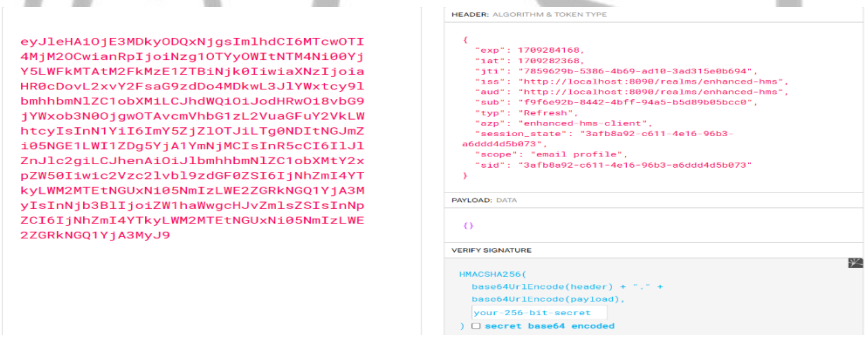

Figure 7 Diagram depicting the access token detail

3.2.1 Fostering Compliance

Fig. 10 through Fig. 12 demonstrate that Keycloak enforces applications to undergo authentication and authorization processes, ensuring adherence to basic security principles. Additionally, Keycloak allows for the integration and enforcement of additional compliance regulations on client systems. It also monitors sessions within the Keycloak server, simplifying comprehensive user management as shown in Fig. 12.

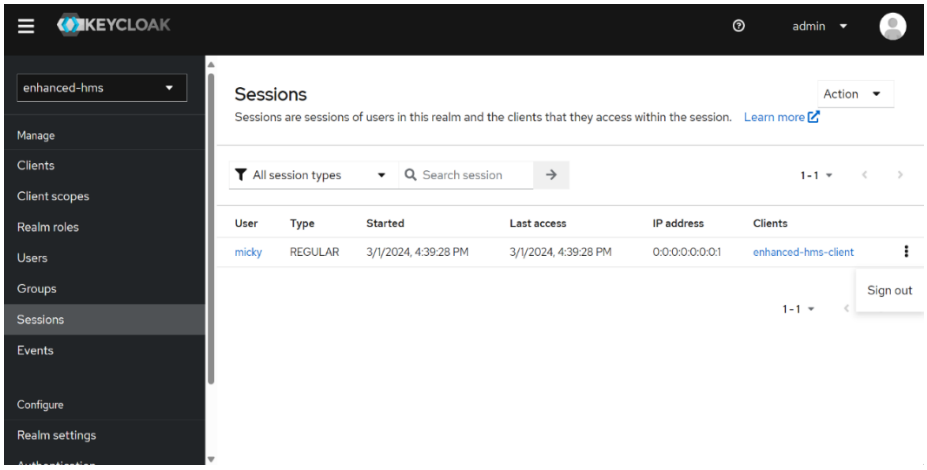

Figure 8 Diagram depicting session tracking in keycloak

3.2.2 Blockchain and Data Handling

Blockchain is integrated with SpringBoot to ensure data integrity within the Health Information System. As shown in Fig. 13, all health data transactions, including deletions, updates, and manipulations, undergo block verification checks to ensure data integrity. Before saving manipulated data, it is validated against the Blockchain validity constraints, ensuring new blocks sync with previous ones. Data is committed only when these validity checks are met.

Figure 9: Diagram depicting blockchain implementation

Fig. 13 illustrates a program, likely named "testblockchain," simulating a blockchain by creating a chain of blocks.

- Block Creation: The program generates three blocks: "firstblock," "secondblock," and "thirdblock," each representing a data storage unit within the blockchain.
- Block Size: Initially, the block size is 1, but it increases to 4, indicating potential data addition or a mechanism for accommodating larger data entries.
- Hashing: The program outputs the hash of each block, using cryptographic techniques to ensure data integrity. Any change to the data results in a different hash, making tampering easily detectable.

- Block Validation: The program validates each block, returning "true," indicating that the block structure and its hash are consistent and correctly formed.

This experiment aligns with the Blockchain Algorithm in 3.10, ensuring data integrity in the Health Information System. The process is foundational for more complex blockchain functionalities. Potential enhancements for further experimentation include:

- Transaction Simulation: Introduce transactions into blocks to demonstrate secure data recording and tracking.
- Peer-to-Peer Network: Simulate a network of nodes to explore blockchain's decentralized aspect, replicating the blockchain on multiple machines and implementing a consensus mechanism.
- Smart Contract Integration: Experiment with simple smart contracts to showcase automated, tamper-proof agreements on the blockchain.

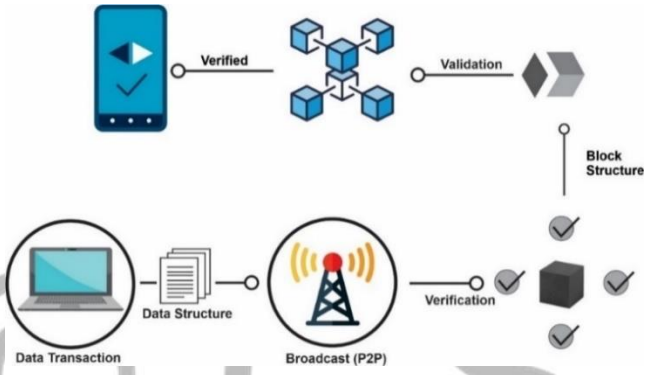

Figure 10 Figure depicting transaction verification in Blockchain

This experiment highlights key blockchain concepts:

- Decentralization: Blockchains are decentralized, with the ledger replicated across a network of computers, eliminating a single point of failure and ensuring data immutability.
- Immutability: Once added, a block's data cannot be altered. Any modification changes the block's hash, making unauthorized data manipulation evident.
- Consensus Mechanism:** In real-world blockchains, participants (miners or validators) verify transactions and add new blocks through a consensus mechanism, ensuring ledger agreement and preventing fraud.

3.3 Information Security Analysis

This section examines potential security risks of integrating Keycloak and blockchain into a Healthcare HIS, identifying threats and proposing mitigation strategies for a secure environment.

1. Threat Modeling
   a) Attackers: Malicious actors targeting patient data for financial gain or identity theft.
   b) Insider Threats: Disgruntled employees or authorized personnel misusing or stealing patient data.
   c) System Vulnerabilities: Software bugs and security misconfigurations in Keycloak or the blockchain platform.

2. Attack Trees: From in Fig. 15, the attack trees include Unauthorized Access, Data Breaches, and Data Tampering.

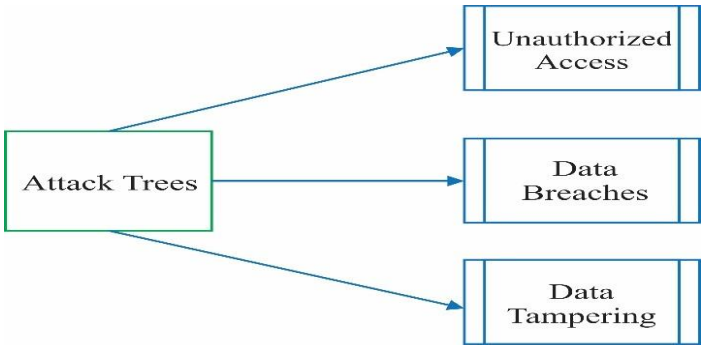

Figure 11 Figure depicting the attack trees

a) Unauthorized Access:
   - Brute-force attacks on Keycloak credentials.
   - Phishing attacks to steal login credentials.
   - Exploiting vulnerabilities in Keycloak or blockchain.

b) Data Breaches:
   - Insider exfiltration of patient data.
   - Man-in-the-middle attacks intercepting HIS communications.
   - Security breaches in the blockchain platform.

c) Data Tampering:
   - Unauthorized alteration of patient data on the blockchain.
   - Insufficient access controls within the HIS.

3. Misuse Cases:
   a) Healthcare worker shares login credentials, bypassing two-factor authentication.
   b) Hacker uses malware-infected device to capture Keycloak credentials.
   c) Malicious actors exploit blockchain zero-day vulnerability to tamper with data.
4. Mitigation Strategies:
   a) Keycloak Security:
      - Enforce strong password policies and multi-factor authentication.
      - Regularly update keycloak software.
      - Implement role-based access control for least privileged access.
   b) Blockchain Security: Similar strategies as keycloak Security to ensure robust protection.
   c) Data Security: Consistent enforcement of strong password policies, multi-factor authentication, and role-based access control.

4. Conclusion

The thesis proposes integrating Keycloak for identity and access management with Blockchain's decentralized, immutable ledger capabilities to enhance patient privacy and data security in Healthcare Information Systems (HIS). This integration, implemented through Java SpringBoot and Spring Security with Oauth2, addresses conventional HIS weaknesses like unauthorized access, data breaches, and

regulatory non-compliance, ensuring secure and tamper-resistant data transactions. The study emphasizes the need for training and legal compliance to effectively deploy this solution, ultimately creating a more secure, efficient, and patient-centric healthcare environment.

Keycloak and blockchain offer innovative solutions to bolster data security and patient privacy in HIS, crucial amid healthcare digitalization. By leveraging Keycloak's single sign-on feature and blockchain's decentralized, immutable ledger, healthcare organizations can ensure patient privacy and regulatory compliance. The future of patient privacy relies on adopting cutting-edge technologies that enhance security and secrecy while facilitating access and collaboration.

Further research is needed to address the challenges and limitations discussed in this thesis. Investigating methods for handling increased capacity, creating oversight rules and guidelines, and promoting specialized knowledge are essential steps for widespread acceptance. Additionally, exploring blockchain's potential beyond data security, such as in medical research and personalized medicine, holds significant promise for the future of healthcare.

References

[1] C. Thapa and S. Camtepe, “Precision Health Data: Requirements, Challenges and Existing Techniques for Data Security and Privacy,” Aug. 2020, doi: 10.1016/j.compbiomed.2020.104130.

[2] J. O. Healthcare Engineering, “Retracted: Research on Data Security and Privacy Protection of Wearable Equipment in Healthcare,” Journal of healthcare engineering, vol. 2023. NLM (Medline), p. 9850693, 2023. doi: 10.1155/2023/9850693.

[3] Harish, V., Ravaut, M., Yi, S., Gutierrez, J., Sadeghi, H., Leung, K., Watson, T., Kornas, K., Poutanen, T., Volkovs, M., & Rosella, L., 2022. Developing Machine Learning Algorithms on Routinely Collected Administrative Health Data - Lessons from Ontario, Canada. International Journal of Population Data Science, 7. https://doi.org/10.23889/ijpds.v7i3.1851.

[4] Li, X., Krumholz, H., Yip, W., Cheng, K., Maeseneer, J., Meng, Q., Mossialos, E., Li, C., Lu, J., Su, M., Zhang, Q., Xu, D., Li, L., Normand, S., Peto, R., Li, J., Wang, Z., Yan, H., Gao, R., Chunharas, S., Gao, X., Guerra, R., Ji, H., Ke, Y., Pan, Z., Wu, X., Xiao, S., Xie, X., Zhang, Y., Zhu, J., Zhu, S., & Hu, S., 2020. Quality of primary health care in China: challenges and recommendations. Lancet (London, England), 395, pp. 1802 - 1812.

[5] Agarwal, A., Joshi, R., Arora, H., & Kaushik, R., 2023. Privacy and Security of Healthcare Data in Cloud based on the Blockchain Technology. 2023 7th International Conference on Computing Methodologies and Communication (ICCMC), pp. 87-92. https://doi.org/10.1109/ICCMC56507.2023.10083822.

[6] Ghafoorian, M., Abbasinezhad-Mood, D., & Shakeri, H., 2019. A Thorough Trust and Reputation Based RBAC Model for Secure Data Storage in the Cloud. IEEE Transactions on Parallel and Distributed Systems, 30, pp. 778-788. https://doi.org/10.1109/TPDS.2018.2870652.

[7] A. G. de Moraes Rossetto, C. Sega, and V. R. Q. Leithardt, “An Architecture for Managing Data Privacy in Healthcare with Blockchain,” Sensors, vol. 22, no. 21, Nov. 2022, doi: 10.3390/s22218292.

[8] Omar, A., Bhuiyan, M., Basu, A., Kiyomoto, S., & Rahman, M., 2019. Privacy-friendly platform for healthcare data in cloud based on blockchain environment. Future Gener. Comput. Syst., 95, pp. 511-521. https://doi.org/10.1016/J.FUTURE.2018.12.044.

[9] H. Liu, R. G. Crespo, and O. S. Martínez, “Enhancing privacy and data security across healthcare applications using Blockchain and distributed ledger concepts,” Healthcare (Switzerland), vol. 8, no. 3, 2020, doi: 10.3390/healthcare8030243.

[10] Fadrique, L., Rahman, D., Vaillancourt, H., Boissonneault, P., Donovska, T., & Morita, P., 2020. Overview of Policies, Guidelines, and Standards for Active Assisted Living Data Exchange: Thematic Analysis. JMIR mHealth and uHealth, 8. https://doi.org/10.2196/15923.

[11] Baca, M., 2022. Health or medical care system?: matters in efficiency. International Jounal of Family & Community Medicine. https://doi.org/10.15406/ijfcm.2022.06.00266.

[12] Villiers, A., & Cuffe, P., 2020. A Three-Tier Framework for Understanding Disruption Trajectories for Blockchain in the Electricity Industry. IEEE Access, 8, pp. 65670-65682. https://doi.org/10.1109/ACCESS.2020.2983558.

[13] Wang, S., Bonomi, L., Dai, W., Chen, F., Cheung, C., Bloss, C., Cheng, S., & Jiang, X., 2020. Big Data Privacy in Biomedical Research. IEEE Transactions on Big Data, 6, pp. 296-308. https://doi.org/10.1109/TBDATA.2016.2608848.

[14] A. Epizitone, S. P. Moyane, and I. E. Agbehadji, “A Systematic Literature Review of Health Information Systems for Healthcare,” Healthcare, vol. 11, no. 7, p. 959, Mar. 2023, doi: 10.3390/healthcare11070959.

[15] K. S. Ahmed and L. Yue, “Improving Intrusion Detection System Using Improved Variational AutoEncoder,” in 2023 8th International Conference on Intelligent Computing and Signal Processing (ICSP), IEEE, Apr. 2023, pp. 215–219. doi: 10.1109/ICSP58490.2023.10248439.

[16] P. Antón, A. Muñoz, and A. Maña, “Authentication and Authorization in Ambient Assisting Living Applications: An Approach for UniversAAL,” 2012, pp. 135–142. doi: 10.1007/978-3-642-35395-6_19.

[17] H. Liu, R. G. Crespo, and O. S. Martínez, “Enhancing privacy and data security across healthcare applications using Blockchain and distributed ledger concepts,” Healthcare (Switzerland), vol. 8, no. 3, 2020, doi: 10.3390/healthcare8030243.

[18] Zheng, X., & Lu, Y., 2021. Blockchain technology – recent research and future trend. Enterprise Information Systems, 16. https://doi.org/10.1080/17517575.2021.1939895.

[29] Wang, Q., Su, M., & Li, R., 2020. Is China the world’s blockchain leader? Evidence, evolution and outlook of China’s blockchain research. Journal of Cleaner Production, 264, pp. 121742. https://doi.org/10.1016/j.jclepro.2020.121742.

[20] Kormpakis, G., Kapsalis, P., Alexakis, K., Mylona, Z., Pelekis, S., & Marinakis, V., 2023. Energy Sector Digitilisation: A Security Framework Application for Role-Based Access Management. 2023 14th International Conference on Information, Intelligence, Systems & Applications (IISA), pp. 1-10. https://doi.org/10.1109/IISA59645.2023.10345842.

[21] P. Guo, W. Liang, and S. Xu, “A privacy preserving four-factor authentication protocol for internet of medical things,” Comput Secur, vol. 137, Feb. 2024, doi: 10.1016/j.cose.2023.103632.

[22] A. A. Monrat, O. Schelén, and K. Andersson, “A survey of blockchain from the perspectives of applications, challenges, and opportunities,” Ieee Access, vol. 7, pp. 117134–117151, 2019.

[23] Divyabharathi, D., & Cholli, N., 2020. A Review on Identity and Access Management Server (KeyCloak). Int. J. Secur. Priv. Pervasive Comput., 12, pp. 46-53. https://doi.org/10.4018/ijsppc.2020070104.

[24] Patra, A., Verma, S., Kumar, S., & Keerthi, P., 2022. A Higher-Level Security Scheme for Key Access ON Cloud Computing. International Journal for Research in Applied Science and Engineering Technology. https://doi.org/10.22214/ijraset.2022.40321.

[25] Chenthara, S., Ahmed, K., Wang, H., Whittaker, F., & Chen, Z., 2020. Healthchain: A novel framework on privacy preservation of electronic health records using blockchain technology. PLoS ONE, 15. https://doi.org/10.1371/journal.pone.0243043.

[26] Divyabharathi, D., & Cholli, N., 2020. A Review on Identity and Access Management Server (KeyCloak). Int. J. Secur. Priv. Pervasive Comput., 12, pp. 46-53. https://doi.org/10.4018/ijsppc.2020070104.

[27] Butpheng, C., Yeh, K., & Hou, J., 2022. A Secure IoT and Cloud Computing-Enabled e-Health Management System. Security and Communication Networks. https://doi.org/10.1155/2022/5300253.

[28] A. B. Kretarta and H. Kabetta, “Secure user management gateway for microservices architecture apis using keycloak on xyz,” in 2022 5th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), IEEE, 2022, pp. 7–13.

[29] Kretarta, A., & Kabetta, H., 2022. Secure User Management Gateway for Microservices Architecture APIs Using Keycloak on XYZ. 2022 5th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), pp. 7-13. https://doi.org/10.1109/ISRITI56927.2022.10052901.

[30] Chatterjee, A., & Prinz, A., 2022. Applying Spring Security Framework with KeyCloak-Based OAuth2 to Protect Microservice Architecture APIs: A Case Study. Sensors (Basel, Switzerland), 22. https://doi.org/10.3390/s22051703.