WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Blockchain Based Logging to Protect Against Malicious Insiders in Remote Health Monitoring System

[1] Mr. P. Rajapandian, [2] Vithanala Manoj Kumar

[1]Associate Professor, Department of computer Applications, Sri Manakula Vinayagar Engineering College (Autonomous), Puducherry 605008, India
[2]Post Graduate student, Department of computer Applications, Sri Manakula Vinayagar Engineering College (Autonomous), Puducherry 605008, India
*Email address: manojkumarvithanala@gmail.com , rajapandian.mca@smvec.ac.in

ABSTRACT-

IoT-based remote health monitoring is a promising technology to support patients who are unable to travel to medical facilities. Due to the sensitivity of health data, it is important to secure it against all possible threats. While a great deal of work has been done to secure IoT device-cloud communication and health records on the cloud, insider attacks remain a significant challenge. Malicious insiders may tamper, steal or change patients' health data, which results in a loss of patient trust in these systems. Audit logs in the cloud, which may point to illegal data access, may also be erased or forged by malicious insiders as they tend to have technical knowledge and privileged access to the system. Thus, in this work, we propose a Cloud Access Security Broker (CASB) model that (a) logs every action performed on user data and (b) secures those logs by placing them in a private blockchain that is viewable by the data owners (i.e., patients). Patients can query the blockchain, track their data's movement, and be alerted if their data has been accessed by an administrator or moved outside the cloud storage. In this work, we practically implement a web application that receives health data from patients, a CASB that securely stores the records in the cloud, and integrate a private blockchain that immediately logs all actions happening in the backend of the web application and CASB. We evaluate the system's security and performance under varying numbers of patients and actions.

KEYWORDS Certainly! Here are some relevant keywords based on your content:

- **Remote health monitoring,IoT devices**
- **Healthcare security**
- **Data confidentiality**
- **Blockchain logging**
- **Insider attack detection**
- **Cloud security**
- **CASB (Cloud Access Security Broker)**
- **Immutable ledger**
- **Access control**
- **Authentication & authorization**
- **Patient data protection**
- **Malicious activity detection**
- **Decentralized healthcare system**
- **Two-factor authentication**
- **Medical IoT integration**

- **Data integrity**
- **User behavior analytics**
- **Encryption & key management**

INTRODUCTION:

The proposed system, "Blockchain-Based Logging to Protect Against Malicious Insiders win Remote Health Monitoring System," is designed to enhance data security and integrity in remote healthcare environments. This system allows patients to upload their health data, which can be monitored by doctors, while administrators oversee user activities and data access. All actions are securely logged on a blockchain to prevent unauthorized After analyzing the requirements of the task to be performed, the next step is toAnalize the problem and understand its context. The first activity in the place is studying the exciting system and other is to understand the requirements and domain of the new system. Both the activities are equally important, but the first activity serves as a basis of giving the functional specifications and then successful design of the proposed system. Understanding the properties and requirements of a new system is more difficult and requires creative thinking and understanding of existing running system is also difficult, improper understanding of present system can lead diversion from solution.

System analysis is a detailed study of the various operations performed by a system and their relationships within and outside of the system. Here the key question is –what all problems exist in the present system? What must be done to solve the problem? Analysis begins when a user or manager begins a study of the program using existing system. During analysis, data collected on the various files, decision points and transactions handled by the present system. The commonly used tools in the system are data Flow Diagram, interviews, etc. Training, experience and common sense are required for collection of relevant information needed to develop the system. The success of the system depends largely on how clearly the problem is defined thoroughly investigated and properly carried out through the choice of solution. A good analysis model should provide not only the mechanism of problem understanding but also the frame work of the solution. Thus it should be studies thoroughly by collecting data about the system. Then the proposed should be analyzed thoroughly in accordance with the needs.

ABOUT THE PROJECT

Busy lifestyles make regular medical checkups difficult for many people, especially for chronic conditions like diabetes and hypertension. Some patients may be less mobile for medical reasons, such as the weak and elderly or those with motion sickness, light sensitivity, or social anxiety. In the recent Covid-19 pandemic, concern about contracting the virus or other illnesses has increased. Remote health monitoring utilizing smart IoT devices could help people unwilling or unable to visit the doctor regularly. Health monitoring IoT devices connect to a mobile app via Bluetooth to share patients' health data with doctors and receive medical suggestions. Such a system, depicted in Figure 1, allows remote medical consultations. Due to the sensitivity of health data and high-security requirements in this domain, a remote health monitoring system must secure user health data at all stages. It is important to ensure (CIA) confidentiality, integrity, and availability of patient data. If patient data is mismanaged or leaked, the lack of privacy will damage the system's reputation, reduce patient trust and hence leave it with few users. All possible threats to patient data must be secured by a successful remote health monitoring system. A large amount of work has been done to secure various aspects of remote monitoring, such as authentication, access control, and secure storage. Notably, Cloud Access Security Broker (CASB) is a complete solution for securing cloud data, monitoring its movement and managing access policies. Several CASB products are available commercially, such as Bit glass CASB , Lookout CASB , CISCO cloud lock and Microsoft Cloud App Security. A CASB provides many security 3 services, including malware detection, cloud configuration, single sign-on for authentication and identity management, user behavior analytics, encryption, key management, and access control. However, even with CASB deployment, insider attacks remain a key challenge. Insider attacks are known to cause significant data breaches. According to the report of Observe it in 2020, 60% of data breaches were caused by insider attacks. According to a survey by Colombia University researchers, 50% of organizations suffered operational disruption because of insider attacks, 48% reported the loss of critical data and intellectual property, and Blockchain Based Logging To Protect A Against Malicious Insiders In Remote Health Monitoring System 37% experienced damage to their brands. These attacks may be perpetrated by a malicious administrator (e.g., disgruntled employees, spies, opportunists looking to expose/sell data for money) who has privileged system access and is familiar with the system policies. As a classical example, a medical device packaging business let go of an employee, Christopher Dobbins, in March 2020. After March, when receiving his last payment, he hacked the company's computer network, gained administrator access, and destroyed 120,000 documents, causing delays in medical equipment delivery. Typical insider attacks in the eHealth domain are tampering, selling, or publishing patients' health data, such as a breach discovered by the Florida hospital where two hospital staff procured patient data sheets, including personal data such as phone numbers, names, and addresses. Two years of data were compromised and possibly used for false insurance claims. The solution to the detection of insider attacks is continuously auditing system activities. For auditing, log data is used to store user actions with timestamps. However, tampering with log data itself is an issue. Malicious administrators with log access can modify log data to cover their tracks after illegally accessing patients' health data. To solve this problem, an immutable logging system is needed. Blockchains present a natural solution for immutability. The blockchain is an immutable, decentralized, and distributed ledger. It consists of several blocks which contain data representing transactions with timestamps. Each block contains the previous block's hash stored in the block's header. The first block in any blockchain is the Genesis block, which does not have any hash of the

previous blockchain. Once data is stored in the blockchain, it can neither be updated nor removed. Because each block contains the previous block's hash, and if any record is updated or removed from the first block, the next connected block hash cannot be matched. If somehow, we match the hash of the first two blocks, and then we need to connect the second with the third block and also the third with the fourth and up to the last block of the blockchain by using the previous block hash matching 4 scheme and this process is very difficult or impossible. Therefore, it is known as an immutable ledger. In this paper, we present a blockchain-based logging system Blockchain Based Logging To Protect Against Malicious Insiders In Remote Health Monitoring System KIET 5 MCA that will be integrated with our proposed CASB architecture to benefit from the security properties of CASB while ensuring visibility into attempted insider attacks. We will deploy a web application that receives patients' health data (from their mobile devices or medical staff, e.g., doctors or nurses) and passes it to the CASB to securely store in a public cloud service. Likewise, the web application also receives data retrieval requests and passes them to the CASB to process according to its access control policies, which in turn passes the data back to the web application. Each action of the web application backend and the CASB, whether it is related to data storage or retrieval, is logged immediately into a private blockchain.

MODULES:

Module 1: User (Patient) Module Forms in this module:

• Registration Form: Patients create their accounts with personal details.

• Login Form: Secure access to the system using credentials.

• Health Data Upload Form: Upload health data manually or through IoT/sensor devices.

• View Own Health Logs Form: Patients can view their submitted health records.

Explanation:

This module allows patients to interact with the system. Patients upload health data which is later stored with integrity using blockchain logging. Authentication ensures only verified users access the system.

Module 2: Doctor Module Forms in this module:

• Login Form: Doctor logs in using credentials.

• View Patient Data Form: Doctors can access assigned patients' health data.

• Prescribe/Comment Form: Doctors can comment or suggest treatment.

• Audit Logs Viewer (Read-only): View blockchain logs to ensure no tampering.

Explanation:

Doctors use this module to monitor patient health remotely. The blockchain ensures the integrity of patient data, preventing internal tampering. Doctors only have access to authorized data.

Module 3: Blockchain Logging Module Forms in this module (admin/backend interface):

• Log Viewer Form: View all blockchain-based transaction logs.

• Block Validator Form: For checking hash values and verifying data integrity.

Explanation:

This module logs every health data submission, access, and modification attempt into the blockchain. Each record is hashed and linked to the previous one, preventing unauthorized modifications and creating a traceable, tamper-proof log.

Module 4: Admin Module Forms in this module:

• Login Form: Admin login for secure access.

• User Management Form: Manage doctor/patient accounts.

• View System Logs Form: Audit blockchain logs and activity trails.

• Malicious Activity Detection Form: Monitor and flag suspicious behaviors.

Explanation:

The admin oversees the platform, manages users, and audits logs. The system uses logging to detect anomalies, e.g., unauthorized access attempts, ensuring insider threats are mitigated.

Module 5: Security & Authentication Module Forms in this module:

• 2FA Setup Form: Set up two-factor authentication for doctors/admins.

• Access Logs Viewer: View history of logins and access attempts.

Explanation:

This module enhances user authentication and maintains access logs, forming the first defense layer against insiders. Combined with blockchain logging, it strengthens security.

Module 6: Data Analysis & Visualization Module (optional/advanced) Forms in this module:

• Health Data Chart Form: Graphical representation of health parameters.

• Risk Alert Form: Auto-alerts based on abnormal readings.

METHEDOLOGY:

PROPOSED SYSTEMS:

The main purpose of this system is to securely store and monitor patient health records while preventing data manipulation by unauthorized or malicious insiders. Blockchain technology is used to maintain a transparent and tamper-proof log of all user activities, ensuring trust and accountability.

Java Technology

Java technology is both a programming language and a platform.

The Java Programming Language

The Java programming language is a high-level language that can be characterized by all of the following buzzwords:

- Simple
- Architecture neutral
- Object oriented
- Portable
- Distributed
- High performance
- Interpreted
- Multithreaded
- Robust
- Dynamic
- Secure

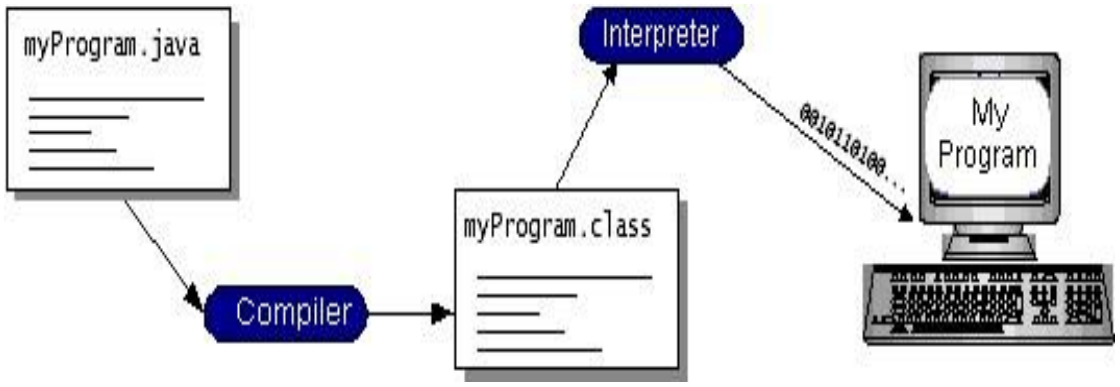

FIGURE 1 JAVA EXECUTION ARCHITECTURE

J2ME (Java 2 Micro edition):-

Sun Microsystems defines J2ME as "a highly optimized Java run-time environment targeting a wide range of consumer products, including pagers, cellular phones, screen-phones, digital set-top boxes and car navigation systems." Announced in June 1999 at the JavaOne Developer Conference, J2ME brings the cross-platform functionality of the Java language to smaller devices, allowing mobile wireless devices to share applications. With J2ME, Sun has adapted the Java platform for consumer products that incorporate or are based on small computing devices.

1. General J2ME architecture

Devices Devices

J2ME uses configurations and profiles to customize the Java Runtime Environment (JRE). As a complete JRE, J2ME is comprised of a configuration, which determines the JVM used, and a profile, which defines the application by adding domain-specific classes. The configuration defines the basic run-time environment as a set of core classes and a specific JVM that run on specific types of devices. We'll discuss configurations in detail in the The profile defines the application; specifically, it adds domain-specific classes to the J2ME configuration to define certain uses for devices.

2. Developing J2ME applications

Introduction In this section, we will go over some considerations you need to keep in mind when developing applications for smaller devices. We'll take a look at the way the compiler is invoked when using J2SE to compile J2ME applications. Finally, we'll explore packaging and deployment and the role preverification plays in this process.

1.SQL Level API

The designers felt that their main goal was to define a SQL interface for Java. Although not the lowest database interface level possible, it is at a low enough level for higher-level tools and APIs to be created. Conversely, it is at a high enough level for application programmers to use it confidently. Attaining this goal allows for future tool vendors to

“generate” JDBC code and to hide many of JDBC’s complexities from the end user.

1. SQL Conformance

SQL syntax varies as you move from database vendor to database vendor. In an effort to support a wide variety of vendors, JDBC will allow any query statement to be passed through it to the underlying database driver. This allows the connectivity module to handle nonstandard functionality in a manner that is suitable for its users.

2. JDBC must be implemental on top of common database interfaces

The JDBC SQL API must “sit” on top of other common SQL level APIs. This goal allows JDBC to use existing ODBC level drivers by the use of a software interface. This interface would translate JDBC calls to ODBC and vice versa.

3. Provide a Java interface that is consistent with the rest of the Java system

Because of Java’s acceptance in the user community thus far, the designers fee they should not stray from the current design of the core Java system.

ARCHITECTURE DIAGRAM:

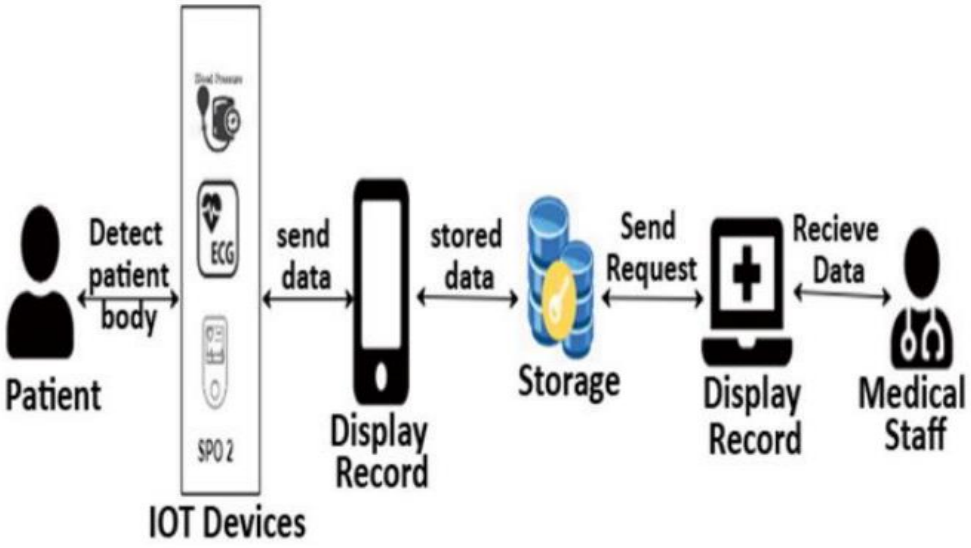

Fig: -1 ARCHITECTURAL DESIGN

The architectural design of the system illustrates the high-level structure and communication flow between various components involved in a real-time chat and calling application. It ensures that the system is modular, scalable, and supports real-time data transmission.

DATA DESIGN

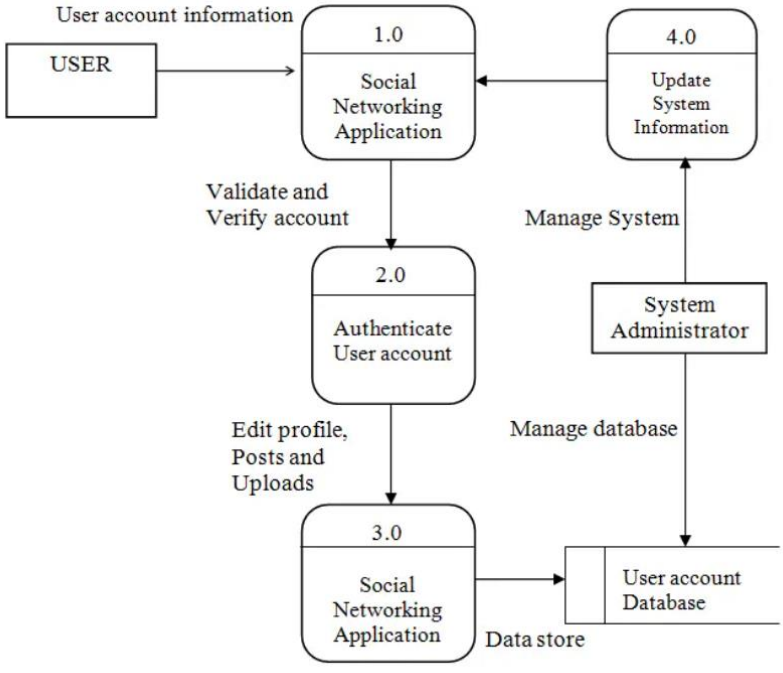

Fig: - 2 DATA DESIGN

The Data Design diagram illustrates how data flows and is managed within a Social Networking Application. It emphasizes the interactions between users, the application, and administrators to ensure smooth functioning of account handling, authentication, content management, and system updates.

USER INTERFACE DESIGN

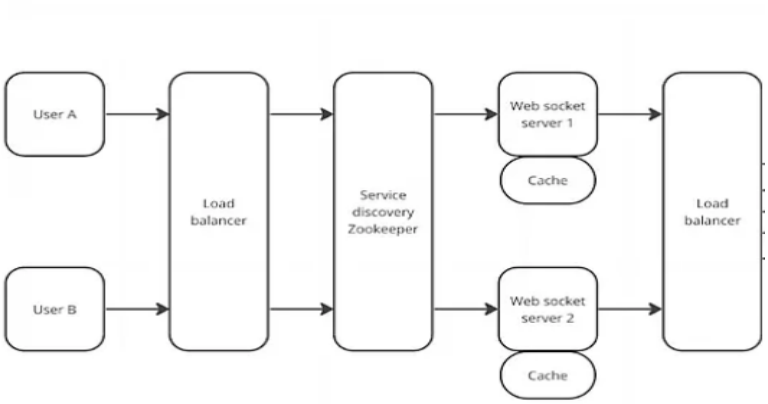

Fig: - 3 USER INTERFACE DESIGN

The User Interface Design illustrates the flow of user interactions with the system through a well-structured backend architecture. This design ensures real-time communication, efficient load handling, and seamless user experience.

CONCLUSION

In this Project, i have presented a private blockchain-based remote health monitoring system to protect against insider attacks. The proposed system offers immutability, distribution, and partial decentralization. The two components of our system are the Cloud Access Security Broker (CASB) for managing real health data and a private blockchain to continuously monitor each user's behaviors for detecting insider attacks. CASB would provide end-to-end security, which includes Authentication, Access Control, and Storage, while all user actions are logged and stored in the blockchain. However, due to blockchain's immutability, tampering or theft of log data is not possible. In addition, any user of the system including the auditors, patients, or doctors can search their log data with ID from the blockchain and detect the administrator's malicious behaviors. Moreover, we practically implemented our system using the Ethereum blockchain and evaluated the performance of the system.

FUTURE ENH ANCEMENTS

In the future, the blockchain-based logging system for remote health monitoring can be enhanced by integrating real-time data collection from IoT-enabled wearable health devices, enabling continuous and accurate health tracking. The incorporation of smart contracts can further automate access control, ensuring that only authorized personnel can view or modify patient data. To strengthen security, AI and machine learning algorithms can be employed to detect and alert on unusual behavior patterns, potentially identifying insider threats early. Additionally, a mobile application can be developed for patients and doctors to improve accessibility and user interaction. The system can also be scaled using more efficient blockchain frameworks to handle large volumes of data and ensure faster transaction processing.

BIBLIOGRAPHY

- S. Sengupta, ‘‘A secured biometric-based authentication scheme in IoTbased patient monitoring system,’’ in Emerging Technology in Modelling and Graphics, 2020, pp. 501–518.
- J. Sun, X. Yao, S. Wang, and Y. Wu, ‘‘Blockchain-based secure storage and access scheme for electronic medical records in IPFS,’’ IEEE Access, vol. 8, pp. 59389–59401, 2020.
- (2022). Bitglass CASB. [Online]. Available: https://www.bitglass. com/casb-cloud-accesssecurity-broker
- (2022). Lookout CASB. [Online]. Available: https://www.lookout. com/products/casb-cloudaccess-security-broker
- Cisco Cloudlock. https://www.cisco.com/c/en/us/products/security/cloudlock/index.html

- Microsoft Cloud App Security. https://www.microsoft.com/enus/security/business/siem-and xdr/microsoft-defender-cloud-apps
- Cloud-Access-Security-Broker-CASB. [Online]. Available: https://www. techtarget.com/searchcloudcomputing/definition/cloud-access-securitybroker-CASB [8] Casb. [Online]. Available: https://www.proofpoint.com/us/threatreference/casb/
- ObserverIT Cost of Insider Threats Global Report 2020. [Online]. Available: https://www.proofpoint.com/us/products/informationprotection/insider-threat-management
- The Colombia University Researchers Perform Survey in 2019. [Online]. Available: https://delinea.com/blog/insider-threats-in-cyber-security
- Real world Insider Attack Example. [Online]. Available: https://www.tessian.com/blog/insider-threats-types-and-real-worldexamples/
- Insider Threats at Hospitals. https://resources.infosecinstitute.com/topic/ insider-threats-athospitals/
- H. Halpin and M. Piekarska, ‘‘Introduction to security and privacy on the blockchain,’’ in Proc. IEEE Eur. Symp. Secur. Privacy Workshops (EuroS&PW), Apr. 2017, pp. 1–3.