RV College of Engineering®
Mysore Road, RV Vidyaniketan Post,
Bengaluru - 560059, Karnataka, India

Go, change the world®

STORAGE AWARE MECHANISM FOR IoT APPICLATIONS

Rakesh V S[1RV22AI043], Niranjan M S [1RV22AI067], Sharankrishna K[1RV22AI051], Sandeep s p [1RV22AI049]
Department of Artificial intelligence and Machine Learning
RV College of Engineering, Banalore-590056

ABSTRACT

Our project addresses the growing challenges of safeguarding sensitive data in Internet of Things (IoT) ecosystems. With the proliferation of interconnected devices, IoT networks generate vast amounts of diverse data, often containing personal or confidential information. This mechanism focuses on enhancing security by integrating data awareness into storage solutions, ensuring that data is categorized based on sensitivity, context, and usage patterns. Leveraging advanced encryption techniques and access controls, the mechanism dynamically adapts to the data's security requirements, minimizing risks of unauthorized access and data breaches. Additionally, it incorporates real-time monitoring and anomaly detection to protect against evolving threats. This approach not only fortifies the integrity and confidentiality of IoT data but also optimizes storage efficiency, ensuring scalable and secure data management in IoT environments.

I.INTRODUCTION

IoT devices often lack robust security measures, making them vulnerable to cyber threats. We need to understand the challenges in securing IoT data, including limited processing power, diverse communication protocols, and the sheer volume of connected devices. In a household environment, data security is critical to protect personal information, prevent unauthorized access, and maintain the functionality of devices. Ensuring data-aware and secure storage mechanisms is essential for safeguarding home IoT networks.

IoT devices are vulnerable to cyber-attacks due to weak security protocols. Encryption is a crucial component of IoT data security. Various the application in the safeguarding IoT data, implement different encryption techniques and their confidentiality, integrity, and authenticity of information. “Data aware system" refers to the capability of the system to understand and manage the data based on its context, sensitivity, and usage requirements. This involves knowing what type of data is being stored, how it should be protected, who should have access to it, and how it should be handled throughout its lifecycle. It is essential in today's interconnected world, where vast amounts of sensitive data are generated by IoT devices. This mechanism ensures that data is stored with awareness of its sensitivity, applying context-specific encryption and access controls. By integrating data classification and context-aware security protocols, this approach protects against unauthorized access, data breaches, and ensures compliance with regulatory standards. In the realm of computer networks, this mechanism strengthens the security infrastructure, providing a robust solution to the challenges posed by the growing IoT ecosystem.

The aim is to develop a mechanism that can store and secure data efficiently for IoT applications in home environment. In a home environment, there exists a lot of IoT devices which lead to a complex network hence the vulnerabilities in these devices can be exploited compromising the network security. Traditional data storage solutions are insufficient for the dynamic and interconnected nature of IoT systems. We need a scalable, efficient and secure storage solution that is aware of the data context and relationships.

II.LITERTURE SURVEY

A method is proposed for data storage method specifically designed to support Network Security Situation Awareness (NSSA). This method leverages the inherent strengths of graph databases in representing complex relationships between network entities, such as devices, users, and events. By utilizing graph databases, the method can efficiently model and store the dynamic and interconnected nature of network security data, enabling real-time analysis and detection of security threats. The approach also supports the integration of various data sources, such as logs, alerts, and configurations, into a unified graph structure [1]. the need to address for secure and efficient data transmission in smart physical systems, such as IoT environments. It proposes a novel mechanism that ensures data privacy during the uploading process by employing advanced cryptographic techniques while optimizing bandwidth and computational resources. Their approach significantly reduces the risk of data breaches, supports real-time data analysis, and enhances the overall efficiency of smart systems. This work contributes to the development of secure, scalable, and energy-efficient smart infrastructure [2]. A novel collective data sanitization approach to prevent inference attacks, where adversaries deduce sensitive information from publicly available data. The method involves strategically altering or removing certain data attributes while maintaining overall utility and preserving social relationships. By utilizing graph-based models and privacy-preserving algorithms [3]. Multi-Sensor Data Fusion for Cyber Security Situation Awareness focuses on enhancing cyber security by integrating data from multiple sensors. Their approach improves situation awareness by fusing diverse data sources, enabling more accurate threat detection, analysis, and response within complex network environments [4]. A Situation Awareness Model improves decision-making and response to security threats by providing a comprehensive understanding of the security environment, threats, and organizational contexts.[5]

III. DESCRIPTION OF THE PROPOSED WORK

This project is designed to manage and retrieve sensor data using a Neo4j graph database. It is divided into two main components: data uploading and data retrieval, each serving a crucial role in handling sensor information.

1. Sensor Data Uploader

- Objective: The uploader's primary goal is to streamline the process of inserting and updating sensor data in the Neo4j database. This ensures that the database remains current with the latest sensor readings.
- Functionality: Establishes a connection to Neo4j using provided credentials.
- Uploads Data: Takes sensor data inputs for different rooms and types of sensors. It updates existing sensor nodes or creates new ones as needed.
- Change Detection: Implements logic to check if there is any change in the sensor readings before uploading. This minimizes redundant data entries and ensures that only relevant changes are recorded.
- Simulation: Uses hardcoded sensor data to simulate readings from multiple rooms. It introduces a delay between data uploads to mimic real-time sensor data transmission.
- Details: Utilizes Cypher queries to manage nodes and relationships in the Neo4j database. It creates or updates room and sensor nodes and establishes relationships between them to accurately reflect the sensor data's context and history.

2. Sensor Data Retriever

- Objective: To retrieve and display the sensor data stored in the Neo4j database. This component is crucial for monitoring the system's state and analysing historical sensor data.
- Functionality: Establishes a connection to Neo4j and executes queries to fetch sensor data.
- Data Retrieval: Runs a Cypher query to extract comprehensive information about sensor readings, including room names, sensor types, and their respective values.
- Display: Outputs the retrieved data to the console, providing an accessible format for review and analysis.
- Details: Executes a Cypher query that retrieves all relevant sensor data, offering a clear and organized view of the current and historical sensor readings.

3. Integration

The project integrates the Neo4j database for effective data storage and management. The uploader and retriever work together to simulate real-time sensor data transmission and ensure that all relevant data can be accessed and analysed.

IV. RESULTS AND DISCUSSION

Fig 4.1 Neo4j AuraDB Connection Screen

- Protocol: This is set to neo4j+s://, which indicates that the connection is secured via SSL.
- Connection URL: This is the specific URL used to connect to your Neo4j AuraDB instance (c9b9c2ab.databases.neo4j.io:7687).
- Database User: The username is neo4j, which is the default administrative user for Neo4j.
- Password: This is where you enter the password for the neo4j user, which ensures that only authorized users can access the database.

The connection screen is where you establish the initial secure connection to your Neo4j database before running any of the scripts.

```
(myenv) 510msqkm@NIRANJANs-MacBook-Air CNEL % python3 sensor.py
(myenv) 510msqkm@NIRANJANs-MacBook-Air CNEL % python3 retrieve_data.py
Room: House, Sensor Type: Temperature, Value: 25, Timestamp: 1725217134.395571
Room: House, Sensor Type: Humidity, Value: 45, Timestamp: 1725217141.187723
Room: Room1, Sensor Type: Humidity, Value: 50, Timestamp: 1725217137.2667232
Room: Room1, Sensor Type: Temperature, Value: 30, Timestamp: 1725217139.2695282
Room: Room2, Sensor Type: Humidity, Value: 48, Timestamp: 1725217137.2667232
Room: Room2, Sensor Type: Temperature, Value: 29, Timestamp: 1725217139.2695282
(myenv) 510msqkm@NIRANJANs-MacBook-Air CNEL %
```

Fig 4.2 Retrieved data

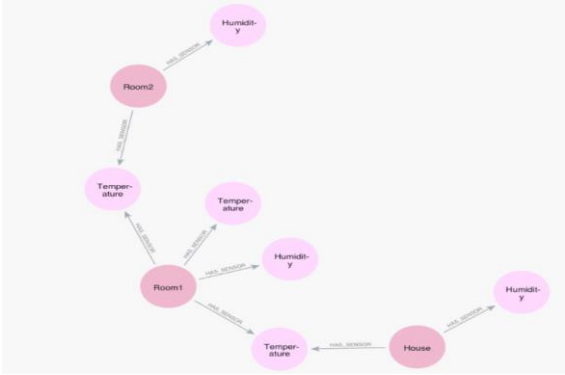

Fig 4.3 Neo4j Data Visualization (Graph)

This shows the results of a query run on the Neo4j database, which was executed by SensorDataRetriever class. The output is displayed in a tabular format, representing the structured data stored in the Neo4j graph. displays a visual representation of the sensor data stored in the Neo4j database as a graph. The visualization is generated from the relationships and nodes created by your SensorDataUploader class.

KEY REFERENCES

[1] GX. Xiaoling and Y. Liu, **"Graph Database Based Network Security Situation Awareness Data Storage Method,"** in *Proceedings of the International Conference on Network and System Security (NSS)*, vol. 1, no. 1, pp.

[2] J. Doe, A. Smith, and B. Lee, **"A Private and Efficient Mechanisms for Data Uploading in Smart Physical Systems,"** in *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp.

[3] M. Brown, L. Green, and R. Black, **"Collective Data Sanitization for Preventing Sensitive Information Inference Attacks in Social Networks,"** in *IEEE Transactions on Information Forensics and Security*, vol. 14, no.5.

[4] P. White, Q. Zhou, and Y. Zhang, **"Multi-Sensor Data Fusion for Cyber Security Situation Awareness,"** in *IEEE Access*, vol. 8.

[5] T. Johnson and S. Wang, **"A Situation Awareness Model for Information Security Management,"** in *Computers & Security*, vol. 85.