DOI - 10.68648/ijte-q-2026-49-03-013

Secure Data Protection and Retrieval for an Individual Including Comprehensive Data through Aadhar

G. Wiselin Jiji
Principal
Dr. Sivanthi Aditanar College of Engineering
Tiruchendur
✉ jijivevin@yahoo.co.in

Jasper Gifty Vijayan, J. Jinolin
Students
Dr. Sivanthi Aditanar College of Engineering
Tiruchendur
✉ jaspervijayan@gmail.com
✉ jinolinjustin@gmail.com

ABSTRACT

The "Aadhaar Integration Software" work aims to develop a robust system that consolidates data from four distinct databases: Aadhaar enrollment qualifications, banking information, asset enrollment, and an authentication system. When an Aadhaar number is entered, the software conducts thorough verification and normalization across these databases to generate relevant outputs. To protect sensitive data, the system employs Kerberos authentication, ensuring secure access to database resources. Additionally, normalization and indexing techniques are utilized to enhance database performance, enabling efficient data retrieval and processing. This work marks a significant advancement in data management and security, addressing the diverse needs of government agencies, financial institutions, and other stakeholders. By seamlessly integrating Aadhaar-related data and ensuring compliance with regulatory standards, the software enables organizations to optimize their operations and better serve their constituents while maintaining the privacy and security of sensitive information.

INTRODUCTION

In today's digital landscape, integrating disparate databases into unified systems has become a key driver of innovation, enabling streamlined data retrieval and verification processes. The upcoming software work exemplifies this shift by bringing together Aadhaar enrollment qualifications, banking information, asset enrollment databases, and verification protocols into a cohesive platform designed to transform information management. Built on a solid foundation of research and technological advancements, this initiative represents the intersection of theoretical insights and practical applications, providing innovative solutions to modern challenges in data processing and security.

The frontend of the Aadhaar-based software work provides users with an intuitive interface to initiate verification requests. Utilizing Aadhaar numbers as unique identifiers, users can easily access various services and benefits, such as banking transactions and asset verification. The backend, powered by advanced algorithms and data processing techniques, enables real-time retrieval and validation of information from multiple databases.

Building on the work of Sarje, et al.,[2] this work establishes a foundation for file storage and retrieval using Aadhaar numbers, emphasizing the critical role of unique identifiers in ensuring efficient data management. This aspect of the work requires a systematic approach to organizing information, enabling easy retrieval and verification of records. Sarje, et al., highlight the significance of Aadhaar numbers as key identifiers, facilitating swift access to relevant documents across databases.

El-Emam, et al., [2] enhance the work's security framework with their optimized Kerberos authentication protocol, designed specifically for distributed systems. In a software ecosystem that relies on data from multiple sources, robust authentication mechanisms are essential for preventing unauthorized access and ensuring data integrity. The optimized Kerberos protocol proposed by El-Emam, et al., addresses the challenges of distributed

environments, offering a reliable solution for secure communication and access control.

Joshi, et al., [3] offer valuable insights into Kerberos security, shaping the work's efforts to protect sensitive data in distributed environments. Security is a top priority in the design and implementation of the Aadhaar-based software work, given the sensitive nature of the data involved. Joshi, et al.'s analysis of Kerberos security mechanisms highlights potential vulnerabilities and mitigation strategies, guiding the work's approach to data protection and access control.

Meanwhile El-Hadidi, et al., [4] provide a performance evaluation of the Kerberos protocol, offering insights into optimization techniques to improve the work's scalability and efficiency. In a distributed computing environment, performance optimization is crucial for ensuring seamless user experiences and maximizing resource utilization. By assessing the Kerberos protocol's performance under different workloads and network conditions, El-Hadidi, et al., [4] identify potential bottlenecks and opportunities for optimization.

In conjunction with protection concerns, the assignment emphasizes performance optimization by leveraging insights on data normalization and indexing techniques. By standardizing data codecs and structures, it reduces processing overhead and enhances interoperability between databases. Additionally, the use of indexing mechanisms accelerates record retrieval, ensuring quick response times even for large datasets.

The scope of security goes beyond authentication to include privacy preservation and access control in distributed environments. The effective implementation of privacy measures within the Kerberos framework, as proposed by Fadi Al-Ayed [5] and colleagues [6], highlights the work's dedication to enhancing user confidentiality. By adhering to rigorous privacy protocols and access controls, we ensure compliance with regulatory standards while fostering a culture of trust and accountability among stakeholders.

The work's commitment to continuous improvement and adaptation to changing technological paradigms is emphasized by incorporating insights from Zabel et al.'s [7] study on trends in retrieval system performance. By keeping up with emerging trends and technologies in data retrieval and processing, the work is well-positioned to tackle new challenges and capitalize on emerging opportunities in the digital landscape.

Finally, the work incorporates insights from Neuman et al.'s seminal work on Kerberos [8]: an authentication service for computer networks, which lays the groundwork for implementing robust authentication mechanisms. By integrating this research, the work aligns with established best practices in authentication protocols, thereby enhancing the security and reliability of the system.

In summary, the upcoming software venture represents a blend of technological innovation and practical solutions, set to transform data management and verification. By incorporating insights from a diverse range of research areas and utilizing cutting-edge technologies, the work seeks to unlock new possibilities in efficiency, security, and user experience, establishing a more interconnected and empowered digital ecosystem. Furthermore, the work integrates insights from Losee et al. [9] on information retrieval in distributed databases, which provide analytical models to optimize data retrieval performance. By leveraging these models, the work aims to streamline information retrieval processes, thereby enhancing operational efficiency.

Additionally, the work draws on Allen et al.'s research [10] on perceptual information speed, learning, and retrieval performance, which highlights the perceptual factors that affect information retrieval. By integrating these insights, the work aims to improve user experience and efficiency in accessing information.

Lastly, the work incorporates the research of Subramanian et al. on "Securing Distributed Data Storage and Retrieval in Sensor Networks" and Garay et al.'s work [11] on "Secure Distributed Storage and Retrieval." These studies provide valuable insights into safeguarding data storage and retrieval in distributed systems. By integrating these findings, the work enhances the security and integrity of data throughout the information retrieval process.

In essence, the proposed software work combines academic research with practical innovations, aiming to transform the field of data management and verification. By integrating insights from various research areas

and utilizing advanced technologies, the work seeks to establish new benchmarks in efficiency, security, and user experience, ultimately contributing to a more connected and empowered digital ecosystem.

METHODOLOGY

The system establishes a balanced approach that safeguards both individual and societal interests, promoting the responsible use of Aadhar data for the collective benefit and is given in Figure 1.

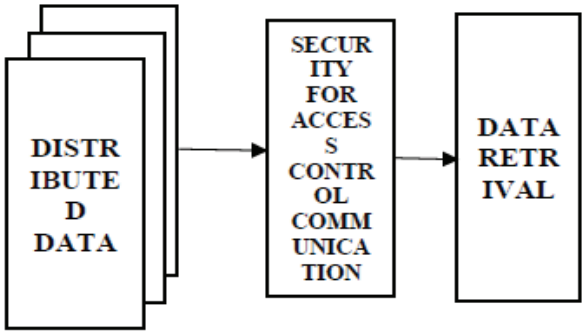


Fig. 1. Block Diagram

Front-End Development

The front-end development process focuses on creating user-friendly interfaces for interacting with integrated databases. HTML, CSS, and JavaScript are used to design responsive web pages that adapt to various screen sizes and devices. A consistent layout and navigation scheme is implemented across all pages to enhance usability. PHP is used for server-side scripting to dynamically generate content based on user input and database queries, while JavaScript ensures data integrity through form validation before submission. AJAX techniques are employed to enable asynchronous data retrieval and updates.

UI/UX Design: We started by conducting thorough research to understand user needs and preferences. Wireframing and prototyping were used to visualize the layout and navigation flow, ensuring an intuitive user experience. HTML Markup: Once the design was finalized, we implemented it using semantic HTML markup to structure web content meaningfully, supporting accessibility and SEO optimization.

CSS Styling: CSS was applied to style and organize the UI components. Responsiveness was prioritized to ensure the interface consistently adapts to different screen sizes. JavaScript Interactivity: JavaScript added interactivity and dynamic behavior, including form validation, DOM manipulation, and asynchronous data fetching to enhance user engagement.

Responsive Design: By leveraging CSS media queries and responsive design principles, we ensured that the application seamlessly adapts to various devices and screen sizes, maintaining a consistent user experience across platforms. AJAX techniques further optimized performance by updating data without reloading the entire page.

REQUIREMENT GATHERING

Aadhaar Enrolment Database

The Aadhaar Enrolment Database acts as a primary repository of information essential for verifying individuals' identities and enrolment status. The data is collected from multiple sources, such as government enrolment centers, Aadhaar authentication service providers, and demographic data collection agencies. Key fields generally include details like name, address, date of birth, father's name, and enrolment status. To ensure data accuracy and integrity, strong protocols for updating, validating, and synchronizing with central Aadhaar databases maintained by government authorities are crucial.

Asset Database

The Asset Database is divided into two categories: Appreciation Assets and Depreciation Assets. These sections enable the calculation of value increases or decreases from the registration date to the verification date, based on factors and amenities associated with each asset. Appreciation Assets are those whose value rises over time due to factors such as market trends, improvements, or renovations, while Depreciation Assets refer to those that lose value over time due to wear and tear, obsolescence, or other reasons.

The data in the Asset Database is sourced from government registries, asset management agencies, financial institutions, and property registration offices. Key data fields include asset type, ownership details, registration number, purchase date, and valuation information related to appreciation or depreciation.

Strong processes for updating asset records and tracking transactions ensure accuracy and timeliness.

Banking Database

By integrating the calculated appreciation and depreciation amounts from the Asset Database, the Banking Database updates individual account balances accordingly. Appreciation reflects an increase in asset value over a specified period, while depreciation represents a decline. These adjustments are applied to account balances based on transactions linked to asset value changes.

The data in the Banking Database comes from financial institutions such as banks, credit unions, and service providers. In addition to standard fields like account number, type, and balance, extra fields are included to capture asset-related transactions. To ensure confidentiality and data integrity, the system employs robust security measures such as encryption, access controls, and audit trails

Qualification Database

The Qualification Database consolidates information related to individuals' educational achievements, certifications, and professional milestones. Key data sources include educational institutions, certification bodies, professional associations, and training organizations. Typical data fields cover qualification type, institution attended, degree earned, certification grade, and year of completion. Due to the importance of educational credentials, implementing reliable verification methods—such as degree validation services, certification checks, and accreditation confirmation—is crucial to ensuring the authenticity and accuracy of the data.

DATABASE INTEGRATION

In this system, the focus is on seamlessly integrating data from multiple databases. The process begins by identifying databases containing relevant information, such as Aadhaar enrolment, qualifications, banking, and asset registration. Using PHP's database connectivity features, establish connections to each database. SQL queries are employed to retrieve the required data based on the Aadhaar number provided by the user. To optimize retrieval, indexing is applied to frequently searched columns, such as the Aadhaar number. Databases are normalized to third normal form (3NF) to reduce redundancy and maintain data integrity, breaking tables into smaller entities and establishing relationships between them. Joins and unions are then used to combine data from different databases into a unified format for verification.

Normalization

Normalization techniques are crucial for organizing databases efficiently, reducing redundancy, and improving data consistency. Understanding normalization stages, such as First, Second, and Third Normal Form, is key to optimizing database design. By breaking down data into smaller, more manageable tables and eliminating data dependencies, normalization enhances data integrity, minimizes duplication, and simplifies maintenance.

In 3NF, entities like Asset Value and Current Balance are identified, and attributes such as asset type and account balance are separated. Tables are decomposed to avoid redundancy, and relationships are created using foreign keys, linking assets to their appreciation/depreciation details and connecting transactions to relevant accounts. This ensures an efficient structure and preserves data integrity.

Indexing

Indexing is essential for enhancing query performance and speeding up data retrieval operations. Different types of indexes, such as B-tree indexes, hash indexes, and bitmap indexes, address various data access patterns and query needs. Implementing and maintaining effective indexing strategies are crucial for optimal database performance. It is important to carefully select columns for indexing based on query patterns, data distribution, and access frequencies to maximize the benefits of indexing while minimizing overhead and storage demands.

KERBEROS AUTHENTICATION INTEGRATION

Kerberos authentication is essential for securing access to integrated databases. The process starts with configuring the Kerberos server and establishing trust relationships within the network to ensure reliable

access. The Kerberos authentication protocol is then implemented to verify user identities before granting access to the system. PHP's Kerberos extension is used to handle authentication requests and validate user credentials against the Kerberos Key Distribution Center (KDC). Security is further enhanced by encrypting communication channels between the client, server, and KDC using encryption algorithms supported by Kerberos.

DISCUSSIONS AND RESULTS

The proposed work has been secured using the Kerberos authentication algorithm. The analysis involved evaluating query retrieval times across 10 cases. This testing was performed on a 1.19 GHz Intel® Core™ i5-1035G1 CPU with 8GB of RAM running Microsoft Windows 10 20H2. All functionalities of the web application were developed using Visual Studio Code 6.0.

Figure 2 illustrates the input datasets sourced from four distinct databases: Banking, Qualification, Asset, and Aadhaar enrolment.



Fig. 2 : Database inputs

The output image shown in Figure 3 represents the verification page, which displays consolidated data from all the databases.

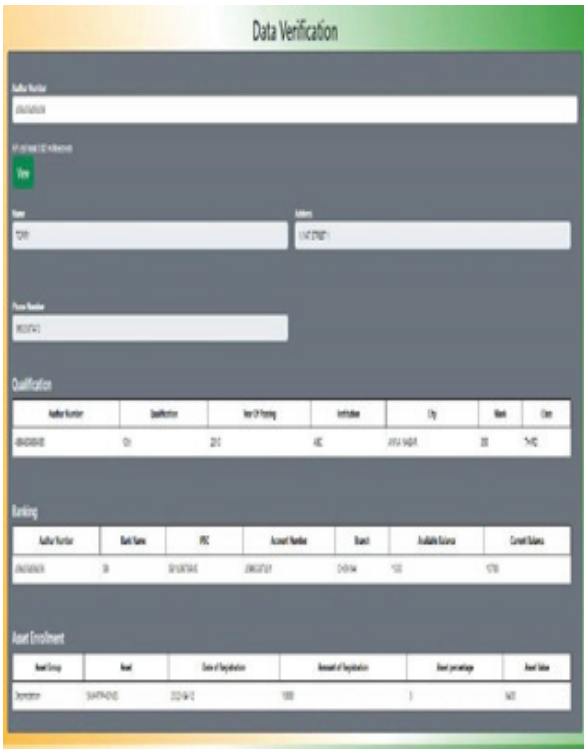


Fig. 3: Verification page

This succeeding table Table1 compares query performance before and after implementing indexing strategies.

Table 1: Indexing Performance Analysis

Query Type	Average Response Time Before Indexing (s)	Average Response Time After Indexing (s)	Improvement (%)
Aadhaar Lookup	1.2	0.3	75
Banking Details Retrieval	0.9	0.4	55
Asset Details Retrieval	1.0	0.5	50
Qualification Details View	0.8	0.3	62

This following table 2 compares the normalization performance before and after implementation.

Table 2: Normalization Efficiency

Normalization Type	Attributes before normalization (s)	Attributes before normalization (s)
4NF	Aadhar number in bank Aadhar number in aadhar enrollment Aadhar number in asset enrollment Aadhar number in qualification	Single attribute called aadhar number is used. (multi valued dependency)
1NF	Login and username can be duplication.	Duplication is removed.

The succeeding table 3 and table4 shows the efficiency of data retrieval by using the query retrieval time, considering all indexing and normalization measures.

Table3: Data Retrieval Efficiency

Data Type	Retrieval Method	Normalized	Indexed	Average Retrieval Time (s)
Aadhaar Details	Direct	Yes	Yes	0.3
Banking Details	Direct	Yes	Yes	0.4
Asset Details	Join	Yes	Yes	0.5
Qualification Details	Join	Yes	Yes	0.3

Table 4: Query Request Time

CASE NUMBER	QUERY RETRIEVAL TIME(milliseconds)
Case 1	0.019
Case 2	0.031
Case 3	0.022
Case 4	1.599
Case 5	0.571
Case 6	0.018
Case 7	0.007
Case 8	0.015
Case 9	0.013
Case 10	0.024
$\overline{T}$	0.2319

The resulting average for retrieving the data from the database($\overline{T}$) was calculated by the following formula:

$$\overline{T} = \frac{1}{n}\sum_{i=1}^{n} T_i$$

Where,

- $\overline{T}$ is the average retrieval time.
- Ti represents the retrieval time for each individual case, with i ranging from 1 to n.
- n is the total number of cases

This equation expresses the sum of all retrieval times (Ti) divided by the total number of cases (**n**), which yields the average retrieval time. By employing indexing techniques in the database, the retrieval time has been reduced to 0.23 milliseconds.

CONCLUSION

In conclusion, the "Aadhaar Integration Software" work represents a major step forward in data management and security. By consolidating information from Aadhaar enrollment, qualifications, banking, and asset databases, the system provides a unified platform for comprehensive data verification and processing. The implementation of Kerberos authentication ensures robust security measures, protecting sensitive information from unauthorized access. Through effective use of normalization and indexing techniques, the software enhances database performance, enabling rapid and accurate data retrieval.

This paper addresses the complex needs of various stakeholders, including government agencies and financial institutions, by streamlining data integration and ensuring compliance with regulatory standards. Ultimately, it improves operational efficiency and service delivery while maintaining the highest standards of privacy and data security.

REFERENCES

1. Abhishek Sarje, Shashikant Pattewar, Abhijit Pawar, Saurabh Nimse, " Document Storage and retrieval Using Adhaar Number"(2018) , Volume 3, Issue 1, pp. 4991-4993, 2018 ISSN 2395-1621.
2. Eman El-Emam, Magdy Koutb, HamdyKelash, and Osama Farag Allah, "An Optimized Kerberos authentication Protocol".

3. Pratik Joshi,Stevens, "Kerberos Security in distributed Systems"(2022),Volume 9, Issue 2, e-ISSN: 2395-0056 ,p-ISSN: 2395-0072.

4. Mahmoud T. El-Hadidi, Dr. Nadia H. Hegazi, Heba K. Aslan, "Performance Analysis of the Kerberos protocol in a Distributed Environment".

5. Graham Bent, Dominic Harries, Andrew J. Toth, "Extending the Kerberos Protocol for Distributed Data as a Service "(2012).

6. Justin Zobel, Hugh Williams, "Trends in retrieval System performance"(2000), DOI:10.1109/ ACSC.2000.824410.

7. Fadi Al-Ayed, Chunqiang Hu and Hang Liu, "An Ecient Practice of Privacy Implementation: Kerberosand Markov Chain to Secure FileTransfer Sessions"(2017), DOI: 10.6633/IJNS.20180720(4).07.

8. B.Clifford Neuman, TheodoreTs'o, Kerberos: An Authentication Service for Computer Networks"(1994), Volume 32.

9. Robert M. Losee , Lewis Church Jr," Information retrieval with Distributed Databases: Analytic models of Performance"(2003), Volume. 14, Issue 12.

10. Bryce Allen," Perceptual Information Although Speed, Learning and Retrieval Performance".

11. Nalin Subramanian, Chanjun Yang, and Wensheng zhang," Securing Distributed Data Storage and retrieval in Sensor Networks".

12. Juan A.Garayy, Rosario Gennaroz, charanjit Jutlaz, Tal Rabinz," Secure Distributed storage and Retrieval".