SecureFile Drive

by
Burak Eymen Çevik

Engineering Project Report

Yeditepe University
Faculty of Engineering
Department of Computer Engineering
2024

Table of Content

1. Introduction

In today's digital era, the rapid expansion of data generation, storage, and sharing has fundamentally transformed how individuals and organizations handle information. The proliferation of the internet and connected devices has fueled an exponential increase in data creation, with global data output reaching unprecedented levels. According to industry reports, the world is expected to generate over 175 zettabytes of data by 2025, highlighting the immense scale of digital information that needs to be managed effectively. This trend has intensified the demand for efficient data management, secure storage, and reliable sharing solutions that can accommodate the growing volume and complexity of data.

Cloud-based storage has emerged as a fundamental response to these demands, offering users extensive benefits, including accessibility from any location, flexible scalability to meet varying needs, and cost efficiency by reducing the reliance on physical infrastructure. Such solutions have enabled smoother collaboration and real-time data availability, revolutionizing the way businesses operate and individuals interact with digital content. However, this convenience brings significant concerns regarding data security and privacy. High-profile cyberattacks, data breaches, and unauthorized access incidents have exposed vulnerabilities in existing storage systems, raising critical questions about data protection and integrity.

Despite advancements in technology, many cloud storage solutions struggle to balance convenience with robust security measures. Users often face a trade-off between ease of access and the assurance that their data is protected from malicious actors. The increasing sophistication of cyber threats, including advanced persistent threats and zero-day exploits, exacerbates this challenge. Furthermore, the stringent requirements of data protection regulations add layers of complexity to data management practices.

The SecureFile Drive Project seeks to address these concerns by delivering a platform that emphasizes both security and user experience. This project incorporates advanced encryption methods, such as AES-256 encryption for data security and RSA algorithms for key management, to safeguard data during transmission and while at rest. By leveraging these encryption standards, the platform ensures that sensitive information remains

confidential and integral, even if intercepted by unauthorized parties. Furthermore, it aligns with legal standards, such as the General Data Protection Regulation (GDPR) and Turkey's Personal Data Protection Law (KVKK), ensuring compliance with international privacy and security norms.

In addition to robust security features, the project prioritizes usability by integrating an intuitive interface developed with Next.js, a modern React framework. This approach enhances performance and provides a responsive, interactive user experience, enabling users to navigate the platform effortlessly. The combination of advanced security and user-friendly design aims to eliminate the trade-offs users typically face, offering a solution that is both secure and convenient.

The SecureFile Drive Project also introduces innovative functionalities, such as metadata tracking and file versioning, to provide users with greater control over their data. These features enable efficient data organization, facilitate recovery of previous file versions, and offer insights into storage usage patterns. By empowering users with these tools, the platform enhances productivity and fosters confidence in data management practices.

This chapter presents a comprehensive overview of the SecureFile Drive Project, encompassing the general problem definition that highlights the solution's necessity. It introduces essential terminology related to data security and the project's technological framework. Additionally, this chapter discusses the motivation and objectives driving the initiative, defines the project's scope and limitations, articulates the specific problem statement, and lists the key requirements guiding the project's development. Finally, it outlines the structure of the document, providing a roadmap for the subsequent chapters that delve into the technical details, implementation strategies, and evaluation of the project.

1.1 General Problem Definition

The relentless growth of digital data in today's interconnected world has intensified the need for secure and efficient storage and sharing solutions. While cloud-based storage services have been developed to meet these demands, they often present significant challenges that compromise their effectiveness. Chief among these challenges are security vulnerabilities and the ever-evolving landscape of cyber threats. Cyberattacks have become increasingly sophisticated, exploiting weaknesses in cloud infrastructures to gain unauthorized access to sensitive data. High-profile data breaches have demonstrated that even well-established cloud services are not immune to such threats, resulting in the exposure of personal and corporate information, financial losses, and damage to reputations.

Privacy concerns also loom large, particularly with the introduction of stringent data protection regulations like GDPR and KVKK. These laws mandate strict controls over how personal data is stored, processed, and shared, requiring organizations to implement robust security measures and obtain explicit consent from users. Failure to comply with these regulations can lead to severe legal and financial repercussions. Unfortunately, many existing cloud storage solutions fall short in fully addressing these compliance requirements, either due to inadequate security features or a lack of transparency in data handling practices.

Limitations in user experience and data management control compound these issues. Users often struggle with monitoring data use, organizing files effectively, and setting specific access permissions. Without intuitive interfaces and comprehensive management tools, productivity can suffer, leaving users feeling disconnected from their data. For organizations, this can lead to inefficiencies, increased risk of data mishandling, and challenges in enforcing internal governance policies effectively.

These multifaceted challenges highlight the pressing need for a cloud storage platform that not only provides robust security and compliance but also enhances user control and experience. Such a platform must address the shortcomings of existing solutions by integrating advanced security measures, offering comprehensive management features, and ensuring compliance with legal standards, thereby restoring trust and confidence in cloud storage services.

1.2 Key Terms

The key terms are listed below to provide clarity on the technical concepts and services referenced throughout the project.

- **Cloud Storage:** Internet-based storage solutions that allow users to save data on remote servers accessible from anywhere with an internet connection.
- **Encryption:** The process of converting data into a coded format to prevent unauthorized access, requiring a decryption key to revert to readable form.
- **AES-256:** An advanced symmetric encryption standard using a 256-bit key, widely regarded as highly secure for protecting sensitive data.
- **RSA:** An asymmetric encryption algorithm used for secure data transmission, key exchange, and digital signatures, relying on a pair of public and private keys.
- **SSL/TLS Protocols:** Cryptographic protocols that establish secure communication channels over networks, ensuring data integrity and confidentiality during transmission.
- **Multi-Factor Authentication (MFA):** A security mechanism that requires users to provide multiple forms of verification before granting access to an account or system.
- **Azure Blob Storage:** A scalable cloud storage solution provided by Microsoft Azure for storing large amounts of unstructured data, such as text or binary data.
- **Azure Entra ID:** Microsoft's cloud-based identity and access management service, formerly known as Azure Active Directory, used for authentication and authorization.
- **Azure Key Vault:** A cloud service that securely stores and manages cryptographic keys, secrets, and certificates, helping to safeguard cryptographic keys and other sensitive data.
- **JWT (JSON Web Token):** A compact, URL-safe means of representing claims to be transferred between two parties, commonly used for authentication and authorization.

- **Versioning:** The practice of keeping multiple versions of files or data, allowing users to access and restore previous states of a file.
- **RBAC (Role-Based Access Control):** A method of regulating access to resources based on the roles of individual users within an organization.
- **SAS Token:** A secure access signature that grants restricted access rights to Azure Storage resources for a specified time interval.

1.3 Motivation and Objectives

The SecureFile Drive Project is propelled by a combination of factors that underscore the urgent need for enhanced security and control in cloud storage solutions. The increasing prevalence of cyber threats and data breaches has eroded trust in existing platforms, as users become more conscious of the risks associated with storing sensitive information in the cloud. High-profile incidents have highlighted vulnerabilities that can lead to unauthorized access, data theft, and significant financial and reputational damage. This environment necessitates a solution that prioritizes security at its core, implementing advanced measures to protect user data against evolving threats.

Simultaneously, the landscape of privacy regulations has evolved, with laws such as GDPR and KVKK imposing strict requirements on how personal data is handled. Organizations are now obligated to ensure data confidentiality, integrity, and availability, with severe penalties for non-compliance. This regulatory pressure drives the need for storage solutions that not only secure data but also provide the tools necessary for organizations to demonstrate compliance and manage data in accordance with legal standards.

User expectations also play a critical role in shaping the project's objectives. In an era where digital services are ubiquitous, users demand platforms that are not only secure but also intuitive and user-friendly. Complex security features must be seamlessly integrated without compromising usability. A platform that balances robust security with an accessible interface can significantly enhance user adoption and satisfaction.

The objectives of the SecureFile Drive Project are thus multi-faceted. The primary goal is to ensure secure data transfer and storage by employing SSL/TLS protocols for data in transit and AES-256 encryption for data at rest. RSA encryption is used for secure key management, adding an additional layer of protection. By utilizing Azure Blob Storage, the project leverages a reliable and scalable infrastructure that meets industry standards for security and compliance.

Another key objective is to enhance user control and experience. Features like file versioning enable users to track changes and recover previous versions, safeguarding against accidental deletions or modifications. The integration of Multi-Factor Authentication (MFA) via Azure Entra ID strengthens account security by requiring additional verification steps. Metadata tracking provides insights into storage usage and file access patterns, empowering users to manage their data more effectively.

Furthermore, the project aims to offer flexible and secure file sharing options. Through the use of SAS tokens and RBAC, users can grant time-based or permanent access to files with precise control over permissions. This functionality is crucial for both personal use and collaborative work environments. By implementing JWT-based authentication, the platform ensures secure session management, maintaining a balance between security and performance.

Overall, the project seeks to create a secure, compliant, and user-centric cloud storage solution that addresses the shortcomings of existing services while meeting the evolving needs of users and organizations.

1.4 Scope and Limitations

The scope of the SecureFile Drive Project encompasses a comprehensive suite of features designed to enhance data security, user management, and overall functionality. In terms of User Management, the platform includes essential features such as user registration and secure login processes. The integration of Multi-Factor Authentication (MFA) through Azure Entra ID adds an additional layer of security, requiring users to provide multiple forms of verification. Password reset and account recovery mechanisms are also incorporated to ensure that users can regain access to their accounts securely in the event of forgotten credentials.

For File Management, the platform provides robust capabilities, including the uploading and downloading of files, deletion and renaming operations, and the critical feature of versioning. Each file is secured using AES-256 encryption to protect data both during transfer and while at rest. Versioning allows users to maintain historical records of their files, facilitating the recovery of previous versions if necessary.

The File Sharing component enables users to share files securely and flexibly. Time-based sharing is facilitated through the use of SAS tokens, allowing users to grant temporary access to files. For more persistent sharing needs, Role-Based Access Control (RBAC) is implemented, providing fine-grained control over permissions and access levels. This ensures that users can collaborate effectively while maintaining control over their data.

Metadata Tracking is another integral aspect of the platform, supporting the analysis of file attributes and usage patterns. This feature provides valuable insights that can inform data management strategies and optimize storage utilization. By understanding how files are accessed and modified, users can make informed decisions about organizing and maintaining their data.

Security measures are deeply embedded throughout the platform. The use of SSL/TLS protocols ensures that data is encrypted during transmission, protecting it from interception. RSA encryption is employed for the secure management of encryption keys, which are stored securely in Azure Key Vault. JWT-based authentication is used for session management, providing a stateless and secure method of verifying user identities.

The platform also emphasizes the importance of a user-friendly interface. By adhering to accessibility standards and designing an intuitive layout, the platform ensures that users can navigate and utilize its features effectively, regardless of their technical proficiency. The frontend of the application is developed using Next.js, a React framework that enables server-side rendering and static site generation for fast, dynamic user experiences. This choice enhances performance and provides a responsive and interactive interface.

Despite its extensive feature set, the project acknowledges certain limitations. Time constraints may impact the depth to which some features can be developed, potentially necessitating prioritization or phasing of functionalities. Budget limitations may restrict the extent of resources available for infrastructure or advanced technologies. The reliance on specific technologies, such as Java, Spring Boot, Next.js, and Azure services, may introduce compatibility challenges with other systems or limit integration options. Additionally, system performance may be affected when handling high volumes of data or ensuring compatibility across a wide range of devices and operating systems. These limitations require careful consideration and strategic planning to mitigate their impact on the project's objectives.

1.5 Problem Statement

In today's rapidly evolving digital landscape, there is a conspicuous gap in the availability of cloud storage solutions that offer both comprehensive security and user-centric control mechanisms. Existing platforms often fall short in providing adequate protection against sophisticated cyber threats, leaving users' sensitive data vulnerable to unauthorized access, breaches, and other security incidents. Additionally, users frequently lack the necessary tools to effectively manage their data, with limited visibility into how their data is stored, accessed, and shared.

This deficiency not only exposes users and organizations to potential security risks but also complicates compliance with stringent data protection regulations such as GDPR and KVKK. The lack of robust security features and insufficient user control can lead to non-compliance, resulting in legal penalties and erosion of user trust.

The SecureFile Drive Project seeks to address these critical shortcomings by developing a cloud storage platform that integrates advanced encryption protocols and stringent security measures. By implementing features such as AES-256 encryption for data at rest, RSA encryption for key management, and SSL/TLS protocols for data in transit, the platform ensures a high level of data protection. The incorporation of comprehensive data management features, including file versioning and role-based permissions, empowers users with greater control over their data.

Furthermore, the platform emphasizes ease of use and accessibility, providing an intuitive interface developed with Next.js that facilitates efficient data management without compromising security. By aligning with best practices in data security and adhering to regulatory requirements, the SecureFile Drive Project aims to create a secure environment for reliable data storage and sharing, ultimately restoring confidence in cloud storage solutions.

1.6 Requirements

The requirements for the SecureFile Drive Project are detailed below, outlining the essential functional and technical specifications necessary for its development and implementation.

Functional Requirements

1. **User Management**
    - Support for account functionalities including user registration, secure login processes, integration of Multi-Factor Authentication (MFA) via Azure Entra ID, password reset mechanisms, and account recovery options to enhance security and user accessibility.
2. **File Management**
    - Efficient handling of files, encompassing uploading and downloading operations, implementation of AES-256 encryption for data security, deletion and renaming capabilities, and versioning features to provide comprehensive control and protection of user data.
3. **File Sharing**
    - Utilization of Shared Access Signature (SAS) tokens for time-limited access permissions and Role-Based Access Control (RBAC) for permanent sharing options, enabling secure and controlled access with detailed permission management for collaborative purposes.
4. **Metadata Tracking**
    - Implementation of tracking and analysis of file attributes and usage patterns to provide insights into storage behavior, supporting data optimization, and facilitating user analytics for better data management decisions.
5. **Notifications and Alerts**
    - Incorporation of email and SMS notifications for security alerts, access attempts, and significant account or file activities, ensuring that users remain informed of key events affecting their data and account security.

Technical Requirements

1. **Infrastructure**
    - Development of the platform using Java and Spring Boot frameworks for scalability and reliability, paired with MySQL or PostgreSQL for robust database management. Integration with Azure services, including Azure Blob Storage for data storage needs, Azure Entra ID for authentication and authorization, and Azure Key Vault for the secure management of encryption keys and other sensitive data.
2. **Security and Encryption**
    - Implementation of SSL/TLS protocols to secure data in transit, utilization of AES-256 and RSA encryption algorithms for data at rest and key management respectively, and adoption of JWT-based authentication to support secure and stateless session management.
3. **Performance and Scalability**
    - Inclusion of asynchronous processing techniques, caching strategies, and load balancing mechanisms to ensure reliable system performance and responsiveness under varying workloads and to support scalability as data volumes and user numbers grow.
4. **User Interface**
    - Design of a responsive and accessible user interface that aligns with Web Content Accessibility Guidelines (WCAG), providing a user-friendly experience across different devices and catering to users with varying levels of technical expertise.
5. **Access Control**
    - Implementation of Shared Access Signature (SAS) tokens for temporary access permissions and Role-Based Access Control (RBAC) for persistent access control, ensuring high-level data protection and offering flexible access options to meet diverse user needs.