WWW.IJRPR.COM

International Journal of Research Publication and Reviews

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

CloudVault - Secure File Encryption and Decryption

[1]Darin Joy Peringalloor, [2]Shreya Patil, [3]Vaishnavi Pawar, [4]Prof. Pallavi Marulkar

Dept. Computer Engineering, Pillai HOC College of Engineering and Technology, Khalapur, HOC Colony Rd, HOC Colony, Taluka, Rasayani, Maharashtra 410207

ABSTRACT:

With growing concerns over data privacy, CloudVault offers a simple yet powerful way to encrypt and decrypt files securely. Our web-based file encryption system ensures that sensitive data remains protected using Fernet symmetric encryption (AES-128 with HMAC integrity verification).The system is designed for effortless file management users can upload files for encryption and later decrypt them as needed. Encryption keys are generated at runtime and not stored, ensuring an extra layer of security, though decryption is only possible within the same session. CloudVault focuses on a structured and efficient file encryption process while maintaining temporary local storage for better security. This paper explores the development, encryption workflow, security measures, and potential improvements, such as persistent key management and enhanced file download capabilities. Designed for personal and business use, CloudVault is an essential tool for safeguarding confidential files and ensuring secure data transmission.

Keywords: Secure File Storage, Data Privacy, AES-128 Security, File Encryption, Secure Decryption, Runtime Key Generation.

Introduction:

In today's digital era, data security is more critical than ever. With rising cyber threats, unauthorized access, and data breaches, traditional methods like passwords and basic encryption no longer provide sufficient protection. Sensitive files require a stronger, more reliable security solution that ensures confidentiality and protection against cyber risks.

CloudVault is a cutting-edge web-based platform designed to provide seamless file encryption and decryption, allowing users to securely store and access their data with confidence. Using Fernet AES-128 encryption with HMAC integrity verification, CloudVault ensures that documents, images, audio, and video files remain protected and inaccessible to unauthorized parties.

Unlike conventional encryption tools, CloudVault encrypts files locally on the user's device before securely storing them in the cloud. This dual-layer approach ensures that sensitive data is protected even before it leaves the system, reducing the risks associated with cyber threats. Each encrypted file is assigned a unique key, which is required for decryption. Since CloudVault does not store encryption keys, only the rightful owner can unlock and access their data, adding an extra layer of security and control.

With a user-friendly interface, robust encryption, and cloud-backed storage, CloudVault bridges the gap between security and accessibility. Whether for personal use, business security, or secure file sharing, CloudVault offers a fast, reliable, and highly secure way to safeguard digital assets in an increasingly interconnected world.

Methodology:

We followed a structured methodology when developing our Warehouse Inventory Management System:

1. **Requirements Analysis:** We began by analyzing the increasing need for secure file storage and encryption. This involved studying cybersecurity challenges, existing encryption methods, and user expectations to define core features.

2. **System Architecture:**

CloudVault follows a three-tier architecture:

- Presentation Layer: A responsive UI built with HTML, CSS, and JavaScript.
- Application Layer: Backend logic using Flask (Python) to handle encryption, decryption, and file management.
- Data Layer: Secure cloud storage for encrypted files, ensuring accessibility while maintaining confidentiality.

3. **Encryption & Security Implementation:**

- We implemented AES-128 encryption using Fernet symmetric encryption, ensuring data integrity with HMAC.

- Files are encrypted locally before being stored in the cloud, preventing unauthorized access.
- Unique encryption keys are generated per file, and only the user with the correct key can decrypt their data..

4. **Module Development:** We developed the system in separate modules:
   - File Upload & Management: Users can upload files for encryption or decryption.
   - Cloud Storage Integration: Encrypted files are stored securely, ensuring access anytime, anywhere.
   - Encryption & Decryption Engine: Secure processing of files using AES 128 encryption.

5. **User Interface Design:** We designed an intuitive interface, making encryption and decryption processes seamless for users with minimal technical knowledge.

6. **Testing and Validation:**
   - Functionality Testing: Ensuring correct encryption, decryption, and storage operations.
   - Security Testing: Validating encryption strength and preventing unauthorized access.
   - User Acceptance Testing: Refining the system based on feedback for a smooth user experience.

7. **Implementation:** After final testing, CloudVault was deployed on a cloud-compatible server, ensuring performance optimization and data security. Future updates will focus on enhancing security, user experience, and additional encryption features.

Existing System:

We have researched these existing systems and the findings were:

1. Cryptomator (Skymatic, 2014) an open-source tool that encrypts files before uploading them to the cloud. It provides transparent encryption for Google Drive, Dropbox, and OneDrive, but lacks advanced access controls or multi-factor authentication for extra security.
2. NordLocker (Nord Security, 2019) a commercial encryption software that automatically syncs encrypted files to the cloud. It provides strong security with zero-knowledge encryption, but free users get limited storage, and key recovery is only available for premium subscribers.
3. AxCrypt (AxCrypt AB, 2001) a file-level encryption software designed for individuals and small teams. It uses AES-128/256 encryption but requires users to manually manage encryption keys. It does not offer built-in cloud storage but integrates with third-party services like Google Drive and Dropbox.

DRAWBACKS OF EXISTING SYSTEM:

- **Limited User Access Controls:**

Most encryption platforms do not offer role-based access, meaning all users must handle encryption keys manually. This increases security risks, especially in shared or business environments where different levels of access are needed.

- **Poor Scalability:**

Existing systems often struggle to handle large files or bulk encryption, leading to performance slowdowns and extended processing times. Some cloud-based encryption tools have file size limits, restricting their usefulness for businesses or professionals handling large datasets.

- **Weak Mobile Accessibility :**

Many encryption solutions have *limited mobile support*, requiring users to *rely on desktops* for encryption and decryption. This prevents *on-the-go* file security and makes encryption tools inconvenient for mobile users.

- **Device Dependencies:**

Many encryption tools store files locally after encryption, making it difficult to access them from multiple devices. This lack of cross-device compatibility limits usability for users who frequently switch between computers and mobile devices.

System Components:

Our Warehouse Inventory Management System consists of three main modules, each with specific access rights:

1. Admin Module
The admin has full control over the system with access to:

- Dashboard**:** Displays product counts, sales statistics, recently added products, and highest/lowest selling products
- User Management: Add, edit and manage system users
- Product Management**:** Add and categorize products with buying price, selling price, and product images
- Product Image Management**:** Upload and manage product images
- Sales Management**:** Record sales transactions and view detailed sales reports

2. Special User Module
Special users have limited access focused on:

- Dashboard**:** View system statistics and performance metrics
- Product Management**:** Add and edit product information
- Media Management**:** Upload and manage product images

3. User (Employee) Module
Regular users can access:

- Dashboard**:** View basic system statistics
- Sales Management**:** Record sales transactions
- Sales Reports**:** Generate and view daily, weekly, and monthly sales reports

Technical Implementation:

Front-end: HTML5, CSS3, JavaScript for dynamic interaction

Back-end: Flask, JSON,OS modules(Python)

Storage:

Algorithm: AES-128 (Advanced Encryption Standard)

Database Structure:

Our system uses several interconnected tables:

· Users – Stores user credentials, authentication details, and roles.
· Files – Stores metadata of uploaded files, including name, size, encryption status, and timestamps.
· Encryption Keys – Stores hashed keys for secure authentication-based access.

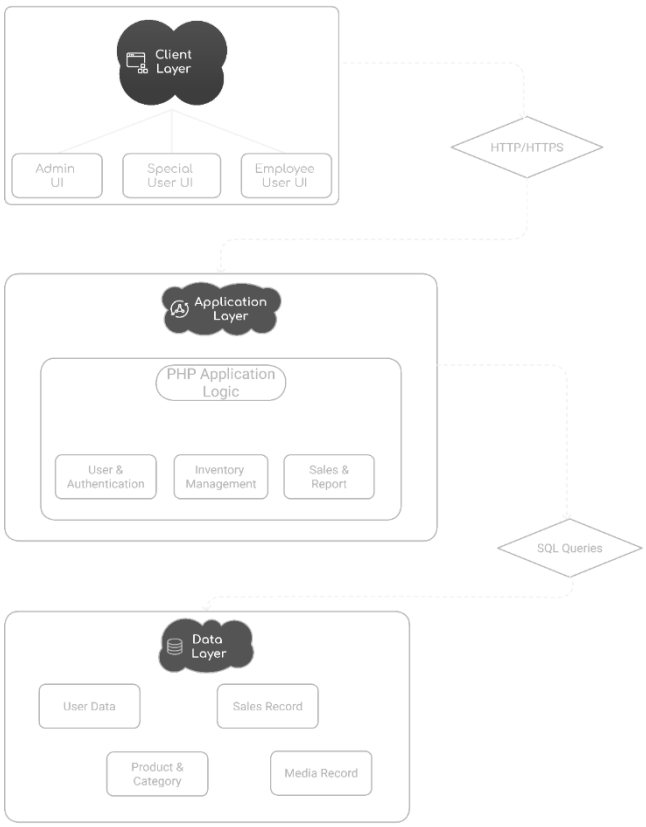
Challenges Faced:

During development, we encountered several challenges:
· Encryption Efficiency: Optimizing AES-128 encryption to ensure fast processing for large files.
· Key Management: Securing encryption keys while allowing easy user access.
· Cloud Storage Security: Encrypting files before upload while maintaining seamless access.
· User Experience: Designing an intuitive UI for smooth file encryption and decryption.
· Cross-Browser Compatibility: Ensuring smooth functionality across different devices and browsers

Results

The implementation of our Warehouse Inventory Management System has been come out as follows:

Fig 1: System Architecture



Our Warehouse Inventory Management System follows a three-tier architecture design that efficiently separates concerns and organizes the application components, as illustrated in Fig. 1.

Fig. 1 System Architecture

The architecture consists of three distinct layers:

1. **Client Layer**: This top-level layer represents the user interface components that different system users interact with. It's divided into three specialized interfaces:
    - Admin UI: Provides comprehensive access to all system functions
    - Special User UI: Offers limited access focused on product management
    - Employee User UI: Presents sales and reporting functionality

All user interfaces communicate with the application layer using HTTP/HTTPS protocols, ensuring secure data transmission.

2. **Application Layer**: The middle tier contains the PHP Application Logic which processes all business rules and application functionality. This layer is organized into three core modules:
    - User & Authentication: Handles login, access control, and user management
    - Inventory Management: Processes product and category operations
    - Sales & Report: Manages sales transactions and generates various reports

The application layer communicates with the data layer through SQL queries to retrieve and store information.

3. **Data Layer**: The foundation tier responsible for data persistence and storage contains four main database components:
    - User Data: Stores user credentials and role information
    - Product & Category: Contains product details and category classifications
    - Sales Record: Maintains all transaction data
    - Media Record: Stores product images and related media files

Conclusion

CloudVault provides a robust and reliable solution for protecting sensitive digital files through advanced encryption techniques. By integrating AES-128 encryption with a user-friendly web interface, it ensures that files remain secure from unauthorized access while allowing users to manage their encrypted data effortlessly. The cloud storage functionality enhances accessibility, enabling users to securely store, retrieve, and share their encrypted files from anywhere.

By leveraging strong encryption standards and eliminating third-party access, CloudVault offers a secure environment for individuals and businesses to safeguard their critical information. Its seamless encryption and decryption process, combined with efficient key management, ensures top-tier data security without compromising usability.

Overall, CloudVault enhances digital privacy, empowers users with complete control over their data, and provides a scalable, efficient, and highly secure platform for modern data protection needs.

REFERENCES:

List all the material used from various sources for making this project proposal

Research Papers:

1. Schneier, B. (1996). Applied Cryptography: Protocols, Algorithms, and Source Code in C. John Wiley & Sons.
2. Stallings, W. (2016). Cryptography and Network Security: Principles and Practice. Pearson.
3. Menezes, A. J., Van Oorschot, P. C., & Vanstone, S. A. (1996). Handbook of Applied Cryptography. CRC Press.
4. Koblitz, N. (1987). Elliptic Curve Cryptosystems. Mathematics of Computation, 48(177), 203–209.
5. Diffie, W., & Hellman, M. (1976). New Directions in Cryptography. IEEE Transactions on Information Theory, 22(6), 644–654.