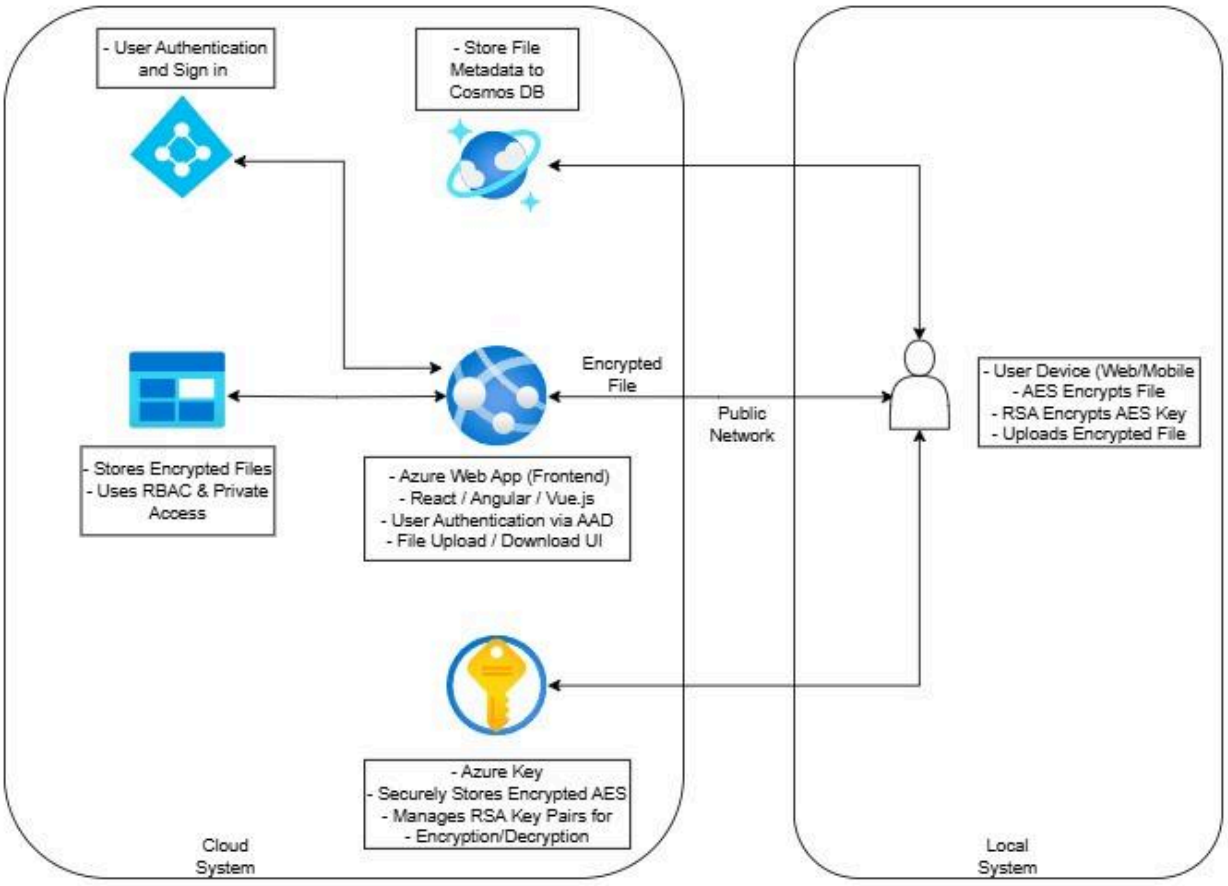
Secure File Storage System using Hybrid Encryption on Azure

Team Members- Rohit Arodi Ramachandra- 002830329
Jasnoor Singh Mac-002840292

Implementing a cloud-based file storage system where files are encrypted before being uploaded. Use AES (for symmetric encryption) and RSA (for key exchange) in a hybrid encryption model. Azure Services: Azure Blob Storage for storing encrypted files. Azure Key Vault for secure key management. Azure Functions for handling encryption and decryption operations. Outcome: A web-based platform where users can securely store and retrieve encrypted files with access control.



The system ensures **confidentiality, integrity, and controlled access** to user files by leveraging **hybrid encryption** and secure cloud services. Users authenticate through **Azure Active Directory (AAD)** before accessing the system. Upon uploading a file, the

AES (Advanced Encryption Standard) encrypts the file locally on the user's device, ensuring that no plaintext data is transmitted. The AES key is further **encrypted using RSA** with the **public key stored in Azure Key Vault**, adding an extra layer of security.

The **encrypted file is stored in Azure Blob Storage**, where strict **RBAC (Role-Based Access Control) policies** prevent unauthorised access. Meanwhile, the **encrypted AES key and file metadata** are stored securely in **Azure Blob Storage**. When a user requests a file, the backend retrieves the **encrypted AES key**, decrypts it using the **RSA private key from Azure Key Vault**, and then decrypts the file using the recovered AES key. The decrypted file is then securely sent back to the user.

This architecture **follows a zero-trust security model**, ensuring that even if **Azure Blob Storage is compromised, attackers cannot decrypt files** without access to **Azure Key Vault**. By integrating **end-to-end encryption, secure key management, and strict authentication**, this system provides a **highly secure and scalable** approach to cloud-based file storage.