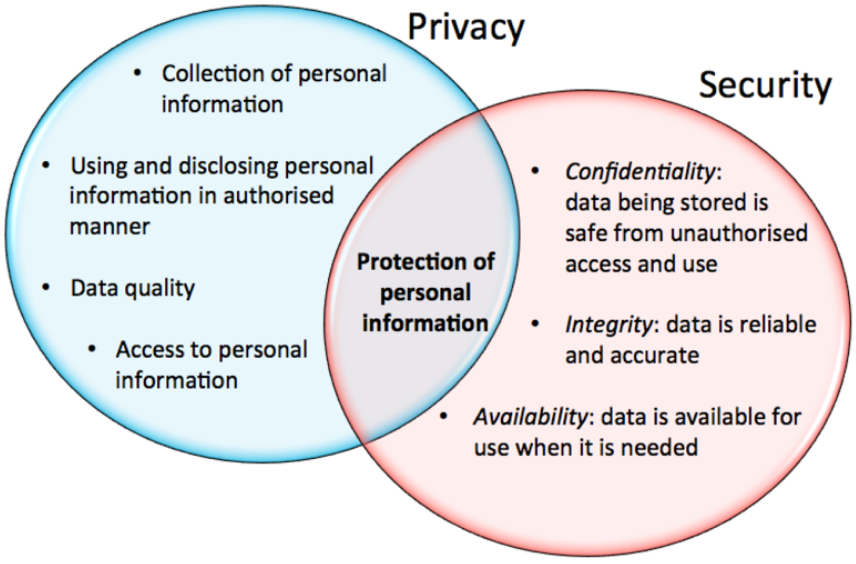
Limitations Security and Privacy

1. Encryption Overhead: The use of Fernet symmetric encryption for sensitive fields like 'age' and 'gender' may introduce computational overhead, affecting the system's performance.
2. Database Trust: The project assumes that the cloud is semi-trusted. While protocols are in place to protect data, a fully malicious cloud provider could still pose risks.
3. Hash Collision: Although unlikely, SHA-256 hashing used for password storage is not entirely collision-free. Two different inputs could, in theory, produce the same hash.
4. Access Control Granularity: The current access control mechanism only differentiates between two types of users ('H' and 'R'). More complex scenarios involving multiple roles and permissions are not covered.
5. Query Completeness: The system verifies the completeness of query results but does not
guarantee 100% accuracy. There's a probability factor involved in detecting whether data items have been removed from a query result.
6. Scalability: The project is designed as a proof of concept and may require additional features and optimizations for large-scale deployment.
7. Vulnerabilities and Exploits: Software and hardware vulnerabilities can be exploited by malicious actors. Regular security audits and updates are necessary to patch vulnerabilities.



Conclusion

The SecureDB Project successfully demonstrates a secure database-as-a-service system with a focus on healthcare information. It incorporates essential security features such as user authentication, access control, data confidentiality, and query integrity. While there are limitations in terms of encryption overhead, trust assumptions, and scalability, the project serves as a robust starting point for building secure database systems. Future work could involve addressing these limitations and adding more features to make the system more versatile and scalable.