Summary of Paper No.31: RESCU-SQL: Oblivious Querying for the Zero Trust Cloud

Student Name: **Jiaqi Jiang** Student ID: **10225501447**

---

1 Research Context

Cloud services are increasingly used for DBMS applications due to their availability and cost-effectiveness. However, high-stakes applications, such as defense and healthcare, face challenges due to confidentiality and regulatory requirements, making it difficult to trust third-party cloud service providers (**CSP**s) with sensitive data.

2 Main Problem Addressed

The paper addresses the challenge of securely querying databases in a zero-trust cloud environment, where users cannot trust CSPs with their private data, as it impacts the adoption of cloud services in sensitive domains where data breaches can have severe consequences.

3 Core Idea and Methods

3.1 Core Idea The core idea is the development of RESCU-SQL, a platform that allows resilient and secure SQL querying on untrusted cloud servers. The system leverages secure multiparty computation (**MPC**) to ensure that queries can be executed without revealing private data to the cloud servers.

3.2 Technics and Methods Technical methods include the use of an authenticated garbling protocol extended to the outsourced setting. A trusted coordinator generates and distributes authenticated secret shares to reduce communication and memory costs, making the protocol efficient.

4 Reviews: Innovativeness and Originality

Combining MPC with a trusted coordinator to achieve secure SQL querying in a zero-trust cloud environment. The approach is original in its extension of the authenticated garbling protocol to an outsourced setting and its ability to tolerate up to $n-1$ corrupted servers.

5 Strengths & Weaknesses

5.1 Strengths

- **Robust Security Model**
  - The paper provides a comprehensive security analysis demonstrating the robustness of RESCU-SQL against various attack vectors, including passive and active adversaries.
  - Commentary: The thoroughness of the security analysis is crucial for high-stakes applications where even minor vulnerabilities can have significant consequences. This depth of analysis builds confidence in the system's security guarantees.
- **Minimized Trusted Computing Base (TCB)**
  - By limiting the trusted computing base to a small, trusted coordinator, the system reduces the trust requirements on the cloud servers.
  - Commentary: A minimized TCB enhances the overall security posture of the system, as fewer components need to be trusted and protected. This design choice also simplifies the security audit process.
- **Scalability**

  - The system is designed to scale with the number of cloud servers, making it suitable for large-scale deployments.
  - Commentary: Scalability is crucial for real-world applications, especially in environments with large datasets and complex queries.

5.2 Weaknesses

- **Performance Overhead**
  - The system incurs a substantial performance overhead compared to plaintext querying, with slowdowns of several orders of magnitude.
  - Commentary: While security is enhanced, the performance trade-off might limit the adoption in scenarios where query speed is critical. For example, real-time analytics or interactive querying might be infeasible with the current performance overhead.
- **Complexity of Implementation**
  - The system's reliance on a trusted coordinator and MPC protocols introduces complexity in implementation and deployment.
  - Commentary: The implementation complexity of RESCU-SQL is significant due to the intricate cryptographic protocols, the need for secure infrastructure, and the challenges of integrating with existing systems while maintaining performance. For example, it is very hard to modify existing DBMS to support secure querying without compromising performance.