Computer Fraud and Security
ISSN (online): 1873-7056

A Residency-Aware Policy Enforcement and Tamper-Evident Auditing Framework for Heterogeneous Multi-Cloud Databases

Sumit Gupta,

Independent Researcher
Email id -sumit.masters2007@gmail.com

Abstract

The increasing adoption of heterogeneous multi-cloud database environments has introduced significant challenges concerning data residency, access control, information security, and audit integrity. Managing sensitive information across different cloud platforms requires coordinated mechanisms that ensure database operations comply with geographical restrictions and organizational security policies. This study proposes a residency-aware policy enforcement and tamper-evident auditing framework designed to strengthen security monitoring across distributed cloud database environments. The framework integrates geographical data residency verification, role-based authorization, policy violation identification, and cryptographic hash-based audit verification. A quantitative, simulation-based research approach was adopted using a researcher-generated synthetic dataset representing database activities across simulated Amazon Web Services, Microsoft Azure, and Google Cloud environments. The dataset was analyzed to examine residency compliance, unauthorized access attempts, audit record integrity, and policy enforcement latency. The proposed approach evaluates database operations against predefined authorization and geographical requirements while using cryptographic verification to identify potential modifications to recorded audit events. The findings provide an analytical demonstration of how these security mechanisms can be integrated into a unified framework for heterogeneous multi-cloud databases. The study contributes to research on cloud security by connecting residency-aware policy enforcement with verifiable auditing. Its results provide a foundation for future validation using real-world cloud audit logs and operational database environments.

Keywords: Access control, Cloud database security, Data residency, Multi-cloud computing, Tamper-evident auditing

1. Introduction

Cloud computing has revolutionized the way organizations store, process and manage digital information. Distributed cloud infrastructures are becoming more and more popular for businesses to use, as they help to improve resource efficiency, ensure service availability, and support database operations in geographically dispersed locations. The increasing adoption of multiple cloud providers has also prompted companies to spread their workloads across various cloud platforms based on their operational needs, storage capacity, and service performance. This shift, however, has created complicated security issues with data confidentiality, access control, regulatory compliance, and the integrity of the data stored. Securing databases in such environments demands security mechanisms that are able to function across various cloud architectures and provide proper control over sensitive information (Singh et al., 2016; Khoda Parast et al., 2021).

Multi-cloud computing provides flexibility because it enables companies to leverage services from multiple cloud providers instead of relying on one infrastructure. Even with these benefits, variations in security setups, admin interfaces, authentication methods, and data management policies can lead to inconsistencies in cloud environments. These differences may increase the difficulty of monitoring database activities and applying uniform security requirements. The security issues with multi-cloud, noting the need to address distributed security issues. Likewise, Megouache et al. (2020) talk about authentication and data-integrity needs in multi-cloud environments. These concerns highlight the need for a coordinated security solution that can control access to the database and safeguard data on a variety of cloud platforms.

Organizations often have to ensure that data stays within geographically authorized areas, which has become an important factor in cloud database security when it comes to data residency. The storage or processing of specific information may be limited across national borders due to regulatory requirements, contractual obligations, and organizational data-governance policies. In cloud environments, however, storage and processing are often spread out across multiple regions,

making it hard to have a clear view of where sensitive data is geographically located. Guo et al. (2016) explored encryption-based solutions to data sovereignty in federated clouds, and the connection between information protection and jurisdictional control. Hummel et al. (2021) have recently explored data-sovereignty models in public cloud settings. In addition, Irain et al. (2017) studied methods to verify data locations in cloud infrastructure. The studies show that geographical restrictions are important to include in cloud security mechanisms.

Another basic need for securing heterogeneous cloud databases is access control. It defines who, what, and how can access and manipulate the stored information. When different database platforms have their own authorization structures, traditional permission-management methods can be challenging to implement in a distributed environment. Permission settings that are not consistent can leave sensitive records vulnerable to unauthorized changes, disclosure or deletion. Namasudra and Roy (2016) discussed secure data-access control mechanisms in cloud computing, and El Sibai et al. (2020) investigated various access-control methods for cloud environments. Golightly et al. (2023) also investigated access-control methods for distributed technologies such as cloud computing, blockchain, and software-defined networking. Additionally, Almutairi et al. (2021) explored access-control systems using blockchain, highlighting the importance of decentralized solutions in security management. Together, this research demonstrates that there is a need for uniform policy enforcement in various database environments.

Access-control mechanisms can prevent unauthorized operations, but they are not necessarily used to identify whether the recorded activities in the database have changed. Organizations also need to have trustworthy auditing tools that can review user activity, detect suspicious changes, and maintain trustworthy evidence of database operations. Traditional centralized audit methods can raise issues about the integrity of audit data, administrative manipulation, and data alteration. Cryptographic verification is a way of establishing the authenticity of changes by linking records to integrity protection mechanisms. Garg and Bawa (2017) explored an auditing protocol for cloud computing based on Merkle hash tree, and Shen et al. (2017) introduced a dynamic structure for public auditing of cloud data. The methods presented here demonstrate ways in which cryptographic structures can enhance the verification of stored information and make it easier to conduct more reliable audit processes.

In recent years, blockchain has also been explored as a tool to enhance transparency and accountability in cloud storage auditing. Li et al. (2020) studied public auditing without trusted auditors, and Wang et al. (2020) studied blockchain-based fair payment mechanisms for public cloud storage auditing. Chen et al. (2022) studied secure deduplication and shared auditing for decentralized storage. Li et al. (2022) gave a wider analysis of cloud-storage integrity auditing, which includes single-copy and multiple-replica approaches. Moreover, Karatas Baydogmus et al. (2018) studied transparent integrity auditing in combination with encrypted deduplication. The focused on practical auditing needs for dynamic data in decentralized storage, while Gudeme et al. (2019) discussed collaborative auditing mechanisms for dynamic data updates. These contributions are the basis for building auditing methods that ensure data integrity and support distributed storage operations.

Even with these advances, residency compliance, access-control enforcement and audit integrity are often considered as distinct security issues. When database operations span multiple providers, storage regions, and database technologies, managing them independently can lead to operational gaps. For instance, a request for access might meet typical authorization criteria and yet be in violation of the geographical storage policy. Likewise, a database operation can be properly denied, but the corresponding audit record can still be susceptible to unauthorized changes. The examples show the importance of an integrated approach that takes into account geographical restrictions, security permissions and audit verification. The general cloud security literature highlights the need for the protection mechanisms to be privacy-enhancing, trustworthy, and coordinated for distributed infrastructures (Butt et al., 2022).

To mitigate these concerns, the present study introduces a policy enforcement and tamper-evident auditing framework for heterogeneous multi-cloud databases that is based on the concept of residences. The proposed research approach integrates geographical data restrictions, role-based authorization decisions and cryptographic audit-integrity verification in a single security model. The framework is explored using a synthetic dataset generated by the researcher, with 5,000 simulated database activity records from three different cloud environments: AWS, Microsoft Azure, and Google Cloud. The database operations, user roles, regional data, authorization results, policy decisions, cryptographic hash values, and enforcement delays are included in the dataset. The assessment is based on the compliance with residency, identification of unauthorized

access, integrity of the audit records, and overhead of policy enforcement. The results are to be used to give a simulation-based evaluation of the possibility of coordinating these security mechanisms in heterogeneous database environments.

The study adds to the discussion on multi-cloud database protection by exploring the relationship between geographical compliance, access authorization and trustworthy auditing. The research does not view these mechanisms as separate controls, but rather as a whole, working together to ensure security throughout distributed database infrastructures. The proposed approach also offers a foundation for research into the implications of operating with uniform security policies on multiple cloud providers. The study aims to contribute to the research of integrated database-security architectures to meet the requirements of data sovereignty, unauthorized access, and integrity of audit records in increasingly distributed cloud computing environments through its simulation-based evaluation.

The current research has two main aims. First, it seeks to design and analyze a security policy enforcement strategy that incorporates geographical information constraints and access control decisions for heterogeneous multi-cloud database systems, focusing on detecting unauthorized operations and possible data-residency violations. Second, it aims to test a tamper evident auditing mechanism using cryptographic hash verification to verify the integrity of the audit record and to explore the policy enforcement latency in simulated AWS, Microsoft Azure, and Google Cloud environments. These goals are combined to explore the effectiveness and operational implications of combining residency compliance, security enforcement and audit integrity in a single multi-cloud database-security solution.

2. Materials and Methods

2.1 Research Design

This study uses a quantitative, simulation-based research methodology to investigate the enforcement of residency-aware security policies and tamper-evident auditing in multi-cloud database environments with heterogeneous clouds. The methodological approach is to assess geographical data restrictions, database access permissions, audit-log integrity and performance of policy enforcement. The investigation doesn't rely on experiments performed on commercial cloud infrastructure, but rather a synthetic dataset created by the researcher that contains simulated database activity records. This way, various security scenarios can be examined in a controlled manner and the selected cloud environments are consistent. The research design is a structured framework that is used to evaluate the proposed framework using descriptive statistical analysis and measurable security-related indicators.

2.2 Dataset Description

The study is based on a synthetic database with 5,000 records of database activity and 24 variables that were specially created for the proposed research. The data set contains three cloud environments: Amazon Web Services (AWS), Microsoft Azure, and Google Cloud. It includes nine simulated database platforms, including relational and non-relational. The attributes that are recorded are the cloud provider, database platform, user role, database operation, data classification, source region, storage region, permitted region, authorization result, policy decision, audit hashes, and enforcement latency. The data set is synthetic and not reflective of data from actual cloud providers or organizational databases.

2.3 Data Preparation and Variable Classification

The data was presented in a structured format to allow for a consistent analysis of the proposed security framework. A unique event identifier was given to each database activity, and relevant cloud, user, operational and geographical attributes were attached to each activity. Cloud provider, database platform, user role, operation type, data classification, and compliance status were all categorical variables. Numerical variables included enforcement latency and event frequencies. Residency compliance was calculated by comparing the recorded storage region to the allowed geographical region for the event in question. The outcomes of authorizations and policy decisions were kept as separate indicators to facilitate analysis of access limitations and policy violations.

2.4 Residency-Aware Security Policy Enforcement

The residency-aware policy enforcement part was tested by the association between data storage locations and allowed geographical areas. For each simulated event, there were source-region, data-storage-region and permitted-region attributes. Any events that occurred outside the permitted area were considered residency violations. Access authorisation was evaluated independently based on role-based permissions and database operation categories. Allow and deny outcomes were used to represent the resulting policy decisions. This assessment allowed for the analysis of geographical compliance

and unauthorized database operations in the simulated multi-cloud environment, giving measurable indicators to evaluate consistency of policy enforcement.

2.5 Tamper-Evident Auditing Mechanism

The auditing part was analyzed by applying cryptographic hash values to individual database activity records. For each event there was a known SHA-256 hash, a recorded hash, and a reference to the previous event hash. The integrity of the audit was verified by comparing the expected and recorded hash value. Records with intentional changes were detected by hash mismatches and were labeled as tampered, while records with matching values were labeled as valid. This process is a simplified simulation of the tamper evident audit verification procedure. It can assist in the investigation of record integrity and traceability, but is not a full-scale production-grade distributed audit ledger.

2.6 Statistical Analysis and Performance Evaluation

Descriptive statistical techniques were used to analyze the dataset to summarize the distribution of cloud activity, geographical compliance, outcomes of unauthorized access, audit integrity, and latency of policy enforcement. The frequencies and percentages were computed to compare the security related outcomes across the three simulated cloud providers and their database platforms. To explore the computational burden of policy decisions, mean enforcement latency was calculated. Valid and tampered record counts were used to summarize audit integrity outcomes. The resulting measurements were then reported in three statistical tables and the corresponding graphs. The analyses were performed on the data generated and the results were discussed in the context of the limitations of simulation-based research.

3. Results

3.1 Distribution of Cloud Database Events by Cloud Provider and Database Platform

Table 1 shows the distribution of 5,000 simulated database events over three cloud service providers and nine database platforms. AWS led the way with 1,982 records (39.64%), Microsoft Azure was next with 1,774 records (35.48%) and Google Cloud with 1,244 records (24.88%). Among the individual platforms, Aurora MySQL had the most events (690 events, or 13.80%), while Cloud SQL MySQL had the fewest events (407 events, or 8.14%). This distribution defines the dataset composition and serves as a foundation for studying security-policy enforcement in heterogeneous multi-cloud database environments.

Table 1. Distribution of Cloud Database Events by Cloud Provider and Database Platform

Cloud provider	Database platform	Frequency (N)	Percentage (%)
AWS	Amazon Aurora MySQL	690	13.80
AWS	Amazon DynamoDB	678	13.56
AWS	Amazon RDS PostgreSQL	614	12.28
Microsoft Azure	Azure Cosmos DB	578	11.56
Microsoft Azure	Azure Database for PostgreSQL	600	12.00
Microsoft Azure	Azure SQL Database	596	11.92
Google Cloud	Cloud SQL MySQL	407	8.14
Google Cloud	Cloud SQL PostgreSQL	411	8.22
Google Cloud	Firestore	426	8.52
Total	All platforms	5,000	100.00

Source: *Authors' calculations based on the researcher-generated multi-cloud residency and security audit dataset.*

Note: *N = 5,000. Percentages are calculated using the total number of event records. The database platforms represent simulated cloud environments rather than actual provider-collected database logs.*

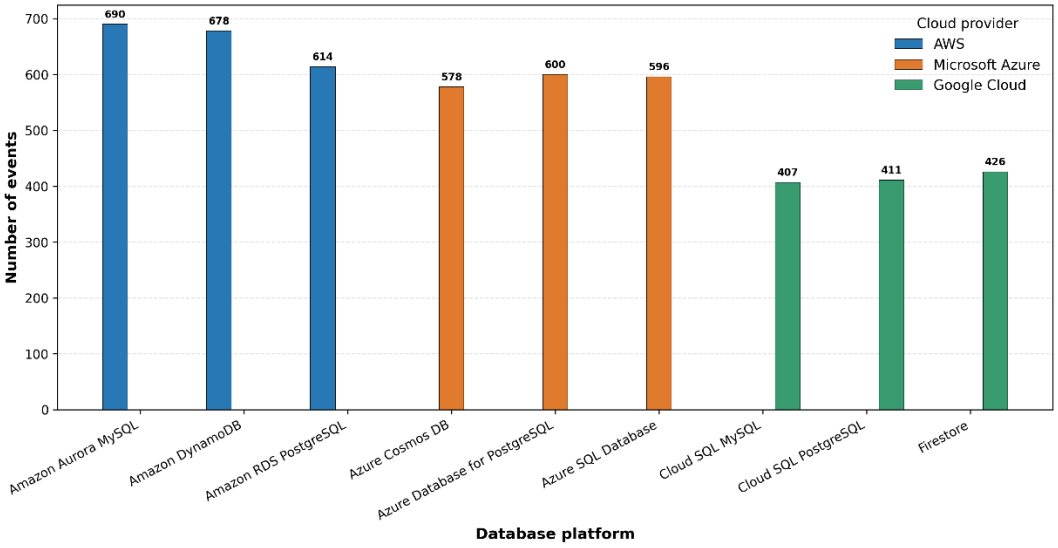

Figure 1. Cloud Database Events Distribution Across Different Cloud Providers and Database Platforms.

Figure 1 shows the distribution of simulated database events in the AWS, Microsoft Azure, and Google Cloud environments. The comparison is based on nine database platforms, showing the differences in the number of events across the selected cloud infrastructures. AWS is the biggest overall contributor, followed by Microsoft Azure and Google Cloud. On the individual platform level, Aurora MySQL has the most events, and Cloud SQL MySQL has the least. The distribution gives an overview of the data set composition and sets the groundwork for analyzing the enforcement of security policies across various database technologies in heterogeneous multi-cloud environments.

3.2 Analysis of Data Residency Compliance and Non-Compliance Across Cloud Providers

Table 2 shows the distribution of events that are residency compliant and non-compliant for the three simulated cloud environments. Of the 5,000 records, 4,795 (95.90%) met the specified geographical storage criteria and 205 (4.10%) were determined to be non-compliant. The top three cloud providers for non-compliant events were AWS with 88, Microsoft Azure with 73 and Google Cloud with 44. When looked at proportionately, AWS had the highest non-compliance rate at 4.44%, while Google Cloud had the lowest at 3.54%. The observations show how regional restrictions can be factored into policy evaluation and offer a foundation for evaluating residency-aware enforcement mechanisms.

Table 2. Data Residency Compliance and Non-Compliance Across Cloud Providers

Cloud provider	Compliant (N)	Compliant (%)	Non-compliant (N)	Non-compliant (%)	Total
AWS	1,894	95.56	88	4.44	1,982
Microsoft Azure	1,701	95.89	73	4.11	1,774
Google Cloud	1,200	96.46	44	3.54	1,244
Total	4,795	95.90	205	4.10	5,000

Source: *Authors' calculations based on the researcher-generated multi-cloud residency and security audit dataset.*

Note: *Residency compliance was determined by comparing the simulated data storage region with the permitted data region for each event. Percentages within each provider are calculated using its respective total event count.*

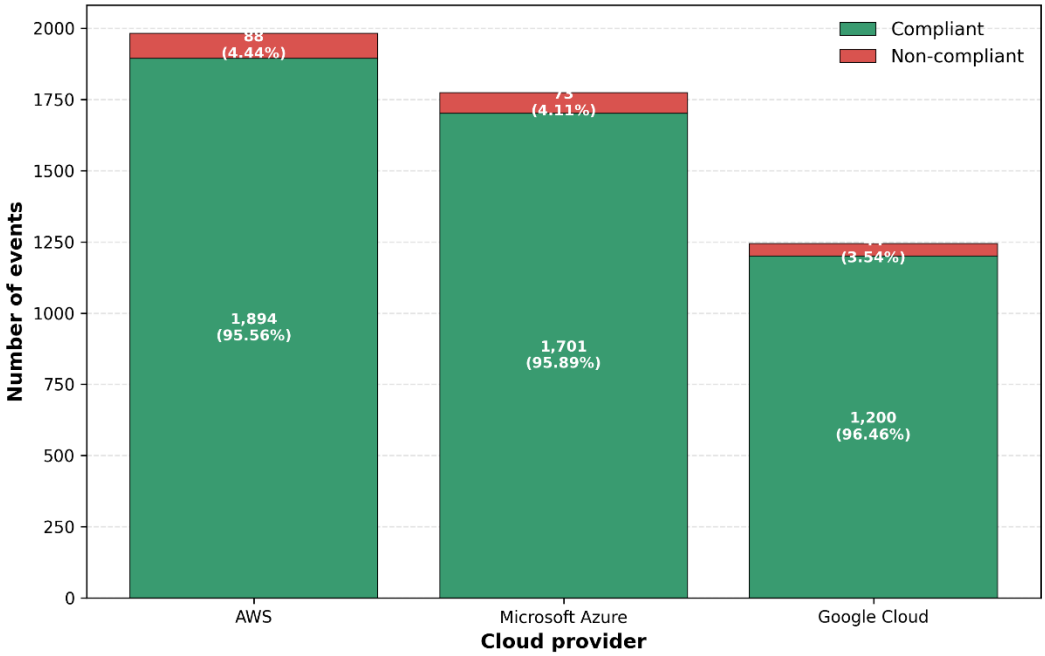

Figure 2 shows a comparative evaluation of the data residency compliance in the three simulated cloud environments. The graphical representation separates compliant and non-compliant database events based on the geographical storage restrictions defined. The results show that the majority of the events are compliant, and a smaller number are in relation to residency violations. AWS has 88 non-compliant events recorded, while Microsoft Azure has 73 and Google Cloud has 44. These observations show the need for geographical restrictions to be included in database access and storage policies. The comparison gives a descriptive basis for assessing the enforcement of residency-aware policies in heterogeneous cloud infrastructures.

3.3 Evaluation of Tamper-Evident Audit Integrity and Policy Enforcement Performance

Table 3 shows the status of the audit and the average latency of enforcement for the three simulated cloud providers. There were 5,000 events recorded, with 4,681 (93.62%) having valid audit hashes and 319 (6.38%) having intentionally added modifications. The most tampered records were registered by Microsoft Azure (124), AWS (116) and Google Cloud (79). In terms of enforcement latency, AWS averaged 38.20 ms, Microsoft Azure 37.52 ms, and Google Cloud 37.69 ms. The mean latency for the overall was 37.83 ms. The results are descriptive evidence for audit integrity verification and policy-enforcement overhead in the simulated environment.

Table 3. Tamper-Evident Audit Integrity and Policy Enforcement Performance Across Cloud Providers

Cloud provider	Valid audit records (N)	Valid (%)	Tampered records (N)	Tampered (%)	Mean enforcement latency (ms)
AWS	1,866	94.15	116	5.85	38.20
Microsoft Azure	1,650	93.01	124	6.99	37.52
Google Cloud	1,165	93.65	79	6.35	37.69
Total / Overall	4,681	93.62	319	6.38	37.83

Source: *Authors' calculations based on the researcher-generated multi-cloud residency and security audit dataset.*

Note: *Audit integrity was assessed using the expected and recorded SHA-256 event hashes. Tampered records represent intentionally injected modifications in the synthetic dataset. Enforcement latency is reported as the arithmetic mean for each cloud provider.*

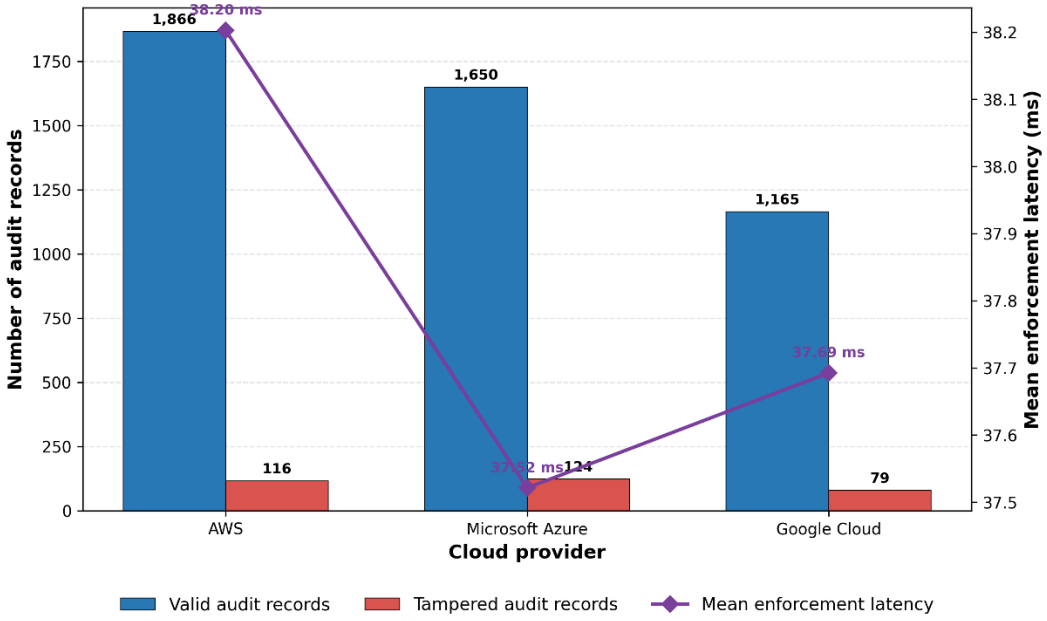

Figure 3. Comparing Audit Log Integrity and Policy Enforcement Latency between Cloud Providers

Figure 3 shows audit-log integrity results and the average latency for policy enforcement on AWS, Microsoft Azure, and Google Cloud. The graphical comparison displays the distribution of valid audit records and intentionally modified audit records, as well as the average enforcement time for each provider. AWS has 1,866 valid audit entries and 116 tampered entries, while Microsoft Azure and Google Cloud have 1,650 and 1,165 valid entries, respectively. The average enforcement latency is between 37.52 and 38.20 milliseconds. This comparison gives an overview of the simulated relationship between audit integrity verification and policy enforcement overhead for heterogeneous cloud database environments.

4. Discussion

The results of this study emphasize the need for data residency controls, security policy enforcement and tamper-evident auditing in heterogeneous multi-cloud database environments. The analysis of 5,000 simulated database events illustrates the security risks that can arise when data access activities, geographical storage needs and audit mechanisms are deployed across multiple cloud platforms. The findings showed that 4.10% of the simulated events were found to be non-compliance of residency and 48.70% of the events were found to be unauthorized access attempts. Also, 6.38% of the audit records were found to be tampered. While these results are generated from synthetic data and not from real cloud environments, they represent the kinds of security conditions that a coordinated multi-cloud framework should be able to detect and handle.

The residency compliance results illustrate the importance of geographical restrictions in cloud database security. Information that is stored or processed outside of an authorized jurisdiction can pose regulatory and organizational risks, especially when sensitive information is shared across multiple cloud providers. One solution to this challenge is to implement a policy enforcement mechanism that takes the residency of the data into account, by checking the storage area that the data is permitted to be stored in against the actual or requested storage area before granting an operation. This is in addition to the traditional access control method, which focuses on whether or not a user is allowed to take a specific action. This means that security decisions can take into account the identity and privileges of the requester and the geographical conditions for the data. The significance of such controls is in line with the overall problems highlighted in the study of cloud data integrity and security architectures.

The unauthorized operations identified further highlight the importance of robust identity management and privacy-protecting access verification. Users can access databases using various authentication methods and authorization models in multi-cloud scenarios. This means that different platforms may have different rules and regulations, which can lead to potential access issues. Yan and Gui (2021) explored identity-based public integrity auditing while maintaining user privacy, highlighting the need for accountability and protection of user-related information. Likewise, Xu et al. (2021) introduced a privacy-preserving cloud storage auditing scheme for data sharing. These studies offer relevant background

to the discussion on integrating identity verification, privacy protection and auditability into a single security architecture. In the current context, authorization is taken into account along with residency conditions, and the simulated policy mechanism can determine whether the operations are in violation of the defined access conditions.

The audit integrity findings also show the potential for cryptographic verification of the integrity of the recorded events in the event that they are modified without authorization. Of these simulated data, 319 records were identified as tampered, which is 6.38% of all events. Tamper evident mechanism using cryptographic hashes can be used to compare the hash of a record with the previous hash to determine if changes have occurred. If linked hashes are employed, changes can also impact the verification of subsequent records. In this regard, Yang et al. (2022) explored provable data possession in cloud storage, highlighting the importance of cryptographic methods in ensuring the integrity of stored data. While provable data possession and event-log auditing serve different technical purposes, both highlight the importance of ensuring that there is evidence that data have not been wrongly altered.

The findings also indicate the need to take distributed auditing mechanisms into account in multi-cloud environments. Zhang et al. (2022) proposed a multi-cloud storage auditing scheme based on blockchain to identify faults. Their work is relevant because in a multi-provider environment, there is a need for mechanisms that can support the verification process across distributed storage locations. This general direction can be extended by an auditing framework that is aware of the residency of the events being audited, and that also records whether the operation that produced the event conforms to geographical and authorization policies. However, the present study does not simulate a deployed blockchain network but a simplified hash-based simulation. Thus, the results should not be interpreted as evidence of the implementation or evaluation of blockchain consensus, decentralized validation, or production-grade immutable storage.

The policy enforcement latency results give another view on the operational aspects of security controls. The overall simulated mean latency was 37.83 milliseconds, with provider-level means of 38.20 milliseconds for AWS, 37.52 milliseconds for Azure, and 37.69 milliseconds for Google Cloud. The simulated policy evaluation process resulted in similar response times for the three provider categories, as indicated by these relatively close values. But they do not prove that the performance of the services provided by these providers is equivalent. In reality, the latency would vary with network conditions, database structure, policy complexity, geographical distance, workload, and the use of logging and cryptographic verification. Zhou et al. (2018) list the design issues of cloud integrity schemes, which is useful for comparing the security assurance and computational cost.

Designing auditing mechanisms should also take into account privacy and efficiency. Xu et al. (2021) and Yan and Gui (2021) show how auditing methods can be developed to consider privacy preserving data sharing and user identity protection. These are important considerations because the audit records can also include sensitive information about users, operations, and organizational activities. It is therefore important to limit access to audit metadata, safeguard personally identifiable information and establish suitable retention and disclosure policies in a practical framework. In addition, Yang et al. (2022) offer an alternative approach to minimizing storage overhead by using compressed provable data possession. These methods could help optimize future audit storage efficiencies, but would need to be assessed on their own for continuous multi-cloud event monitoring.

In general, the study recommends that residency-aware authorization and tamper-evident auditing should be considered complementary security functions, not separate ones. Access policies define the conditions for allowing or denying access to an operation, residency rules define the conditions for the geographical part of the operation, and cryptographic verification ensures the reliability of the resulting audit evidence. Integrating these functions can provide a more comprehensive basis for monitoring heterogeneous cloud database activities. The synthetic nature of the data set is, however, an important limitation. The results show the analytical behavior of the proposed framework under the set simulated conditions, but not its performance in the presence of real cyberattacks or production workloads. The framework should be validated in future research with authorized operational audit logs, larger and more diverse workloads, and implementations in real cloud database services. Other tests should also cover false positive rates, detection accuracy, scalability, policy conflicts, and overhead of cryptographic verification. This validation would offer more confidence in the framework's real-world applicability for secure and compliant multi-cloud database management.

5. Conclusion

This study proposed a policy enforcement and tamper-evident auditing framework for heterogeneous multi-cloud database environments, which is based on the concept of a residency. The proposed solution includes geographical data residency verification, access authorization, security policy monitoring, and cryptographic audit validation, which addresses the important challenges of managing distributed cloud data. The framework incorporates these mechanisms to offer a structured way of recognizing unauthorized database operations, violations of residency, and verifying the integrity of recorded audit activities. The simulation-based analysis shows the potential of the integration of these security functions in a single monitoring architecture. The research points out that traditional user authentication and access control are not enough for effective multi-cloud security. It also involves ongoing attention to data location, regulatory needs and audit record reliability. The proposed framework provides a conceptual and analytical contribution, linking these requirements in a shared security approach. The use of a synthetic dataset, however, restricts the generalizability of the results to real cloud infrastructures. The framework should then be tested in future research with authorized operational cloud logs and real deployments of databases. Other investigations should also take into account advanced cryptographic auditing, automated policy management, scalability, and detection accuracy under various workloads. Overall, this research provides a foundation for developing more transparent, accountable, and residency-compliant security mechanisms for heterogeneous multi-cloud database systems.

References

1. Almutairi, Suzan & Alghanmi, Nusaybah & Mostafa, Muhammad. (2021). Survey of Centralized and Decentralized Access Control Models in Cloud Computing. International Journal of Advanced Computer Science and Applications. 12. 10.14569/IJACSA.2021.0120243.
2. Butt, Umer & Amin, Rashid & Mehmood, Muhammad & Aldabbas, Hamza & Alharbi, Mafawez & Albaqami, Nasser. (2022). RETRACTED ARTICLE: Cloud Security Threats and Solutions: A Survey. Wireless Personal Communications. 128. 387-413. 10.1007/s11277-022-09960-z.
3. Chen, Jiannan & Wang, Ying & Huang, Zhaohui & Ruan, Conghao & Hu, Chunqiang. (2022). A Decentralized Public Auditing Scheme for Secure Cloud Storage Based on Blockchain. Wireless Communications and Mobile Computing. 2022. 10.1155/2022/3688164.
4. El Sibai, Rayane & Gemayel, Nader & Bou abdo, Jacques & Demerjian, Jacques. (2020). A Survey on Access Control Mechanisms for Cloud Computing. Transactions on Emerging Telecommunications Technologies. 10.1002/ett.3720.
5. Garg, Neenu & Bawa, Seema. (2017). RITS-MHT: Relative indexed and time stamped Merkle hash tree based data auditing protocol for cloud computing. Journal of Network and Computer Applications. 84. 1-13. 10.1016/j.jnca.2017.02.005.
6. Golightly, Lewis & Modesti, Paolo & Garcia, Rémi & Chang, Victor. (2023). Securing Distributed Systems: A Survey on Access Control Techniques for Cloud, Blockchain, IoT and SDN. Cyber Security and Applications. 1. 100015. 10.1016/j.csa.2023.100015.
7. Gudeme, Jaya & Pasupuleti, Syam & Kandukuri, Ramesh. (2019). Review of remote data integrity auditing schemes in cloud computing: taxonomy, analysis, and open issues. International Journal of Cloud Computing. 8. 20. 10.1504/IJCC.2019.097893.
8. Guo, Cheng & Zhuang, Ruhan & Jie, Yingmo & Ren, Yizhi & Wu, Ting & Choo, Kim-Kwang Raymond. (2016). Fine-grained Database Field Search Using Attribute-Based Encryption for E-Healthcare Clouds. Journal of Medical Systems. 40. 10.1007/s10916-016-0588-0.
9. Hummel, Patrik & Braun, Matthias & Tretter, Max & Dabrock, Peter. (2021). Data sovereignty: A review. Big Data & Society. 8. 205395172098201. 10.1177/2053951720982012.
10. Irain, Malik & Jorda, Jaques & Mammeri, Zoubir. (2017). Landmark-based data location verification in the cloud: review of approaches and challenges. Journal of Cloud Computing. 6. 10.1186/s13677-017-0095-y.
11. Karatas Baydogmus, Gozde & Akbulut, Akhan. (2018). Survey on Access Control Mechanisms in Cloud Computing. Journal of Cyber Security and Mobility. 1-36. 10.13052/2245-1439.731.
12. Khoda Parast, Fatemeh & Sindhav, Chandni & Nikam, Seema & Yekta, Hadis & Kent, Kenneth & Hakak, Saqib. (2021). Cloud Computing Security: A Survey on Service-based Models. Computers & Security. 114. 10.1016/j.cose.2021.102580.

13. Li, Angtai & Chen, Yu & Zhou, Xiaokang & Shimizu, Shohei. (2022). A Survey on Integrity Auditing for Data Storage in the Cloud: From Single Copy to Multiple Replicas. IEEE Transactions on Big Data. 8. 1428-1442. 10.1109/TBDATA.2020.3029209.
14. Li, Song & Liu, Jian & Yang, Guannan & Han, Jinguang. (2020). A Blockchain-Based Public Auditing Scheme for Cloud Storage Environment without Trusted Auditors. Wireless Communications and Mobile Computing. 2020. 1-13. 10.1155/2020/8841711.
15. Megouache, Leila & Zitouni, Abdelhafid & Djoudi, Mahieddine. (2020). Ensuring user authentication and data integrity in multi-cloud environment. Human-centric Computing and Information Sciences. 10. 10.1186/s13673-020-00224-y.
16. Namasudra, Suyel & Roy, Pinki. (2016). Secure and efficient data access control in cloud computing environment: A survey. Multiagent and Grid Systems. 12. 69-90. 10.3233/MGS-160244.
17. Shen, Jian & Shen, Jun & Chen, Xiaofeng & Huang, Xinyi & Susilo, Willy. (2017). An Efficient Public Auditing Protocol With Novel Dynamic Structure for Cloud Data. IEEE Transactions on Information Forensics and Security. 12. 2402-2415. 10.1109/TIFS.2017.2705620.
18. Singh, Saurabh & Jeong, Young-Sik & park, Jong. (2016). A Survey on Cloud Computing Security: Issues, Threats, and Solutions. Journal of Network and Computer Applications. 75. 10.1016/j.jnca.2016.09.002.
19. Wang, Hao & Qin, Hong & Zhao, Minghao & Wei, Xiaochao & Shen, Hua & Susilo, Willy. (2020). Blockchain-based fair payment smart contract for public cloud storage auditing. Information Sciences. 519. 10.1016/j.ins.2020.01.051.
20. Xiao, Jie & Huang, Hui & wu, Chenhuang & Chen, Qunshan & Huang, Zhenjie. (2023). A collaborative auditing scheme with dynamic data updates based on blockchain. Connection Science. 35. 10.1080/09540091.2023.2213863.
21. Yan, Hao & Gui, Wenming. (2021). Efficient Identity-Based Public Integrity Auditing of Shared Data in Cloud Storage With User Privacy Preserving. IEEE Access. 9. 45822-45831. 10.1109/ACCESS.2021.3066497.
22. Yang, Yang & Chen, Yanjiao & Chen, Fei & Chen, Jing. (2022). An Efficient Identity-Based Provable Data Possession Protocol With Compressed Cloud Storage. IEEE Transactions on Information Forensics and Security. 17. 1359-1371. 10.1109/TIFS.2022.3159152.
23. Zhang, Cheng & Xu, Yang & Hu, Yupeng & Wu, Jiajing & Ren, Ju & Zhang, Yaoxue. (2022). A Blockchain-Based Multi-Cloud Storage Data Auditing Scheme to Locate Faults. IEEE Transactions on Cloud Computing. 10. 2252-2263. 10.1109/TCC.2021.3057771.
24. Zhou, Lei & Fu, Anmin & Yu, Shui & Su, Mang & Kuang, Boyu. (2018). Data integrity verification of the outsourced big data in the cloud environment: A survey. Journal of Network and Computer Applications. 122. 10.1016/j.jnca.2018.08.003.