Computer Fraud and Security

ISSN (online): 1873-7056

Real-Time Security & Analytics: Optimizing Cloud Data Mesh Architecture for Low-Latency Reporting and Immutable Data Security.

Deepak Reddy Suram

Senior Software Engineer & Cloud Data Architect

H&R Block, Inc

reddydeepaksuram@gmail.com

ORCID: 0009-0004-9698-0791

ABSTRACT: Cloud computing platforms have been very useful in Swift run time processing of bulk amounts of distributed data in modern enterprises. Meanwhile, they should provide good data security, integrity and auditability to address needs in operational level and regulatory level. The conventional centralized multi-cloud information designs tend not to satisfy these two conditions, as it adds points of bottlenecks within the latency and causes a single security failure point. The paper explores the quantitative way that an optimized Cloud Data Mesh architecture can be used to support both low-latency and immutable data security.

The suggested solution takes data as a domain product and integrates both performance and security controls into pipes of decentralized data. The reduction of the reporting latency is done using real-time streaming ingestion, domain-specific data contracts, and optimized analytical access patterns. There is no possibility of modifying data, since cryptographic hashing and ledger-style audit systems are applied on a domain level. The experimental design is a controlled experiment that likens a bursting Data Mesh architecture setup to a centralized baseline data lake based on a cloud.

There is a statistically significant improvement in ingestion latency, query response time, throughput and tamper detection speed. The results indicate that secure-by-design and decentralized data architecture can provide real operational intelligence without any data integrity and governance losses. This paper is the empirical evidence of Cloud Data Mesh that could effectively be discussed as a scalable and secure platform of real-time cloud analytics.

KEYWORDS: Cloud Data Mesh, Big Data Analytics, Cloud Security Architecture, Cryptographic Hashing, Federated Data Governance

I. INTRODUCTION

The cloud has been adopted as the main cloud computing technology of storing and analyzing vast quantities of enterprise data. Enterprises are now in high demand of real-time analytics to aid them in making operational decisions, monitoring and automation. Concurrently, the security, integrity, and compliance of data have become an essential requirement with the burden of regulations and the increasing security threats. Classic centralized data architecture on clouds fail to meet the need of both the performance and security requirements on the scale.

In many cases, centralized data lakes and warehouses are characterized by large latency requirements in the case of streaming workloads and parallel analytical queries. These systems are not easily scaled and audited as the amount of data and the range of domains grow. The security controls are usually employed in some external layers and this makes it slow to detect data tampering and complicates the operations. These restrictions render the centralized architectures inappropriate in the context of real-time and multi-domain environments.

Another alternative provided by the Cloud Data Mesh paradigm is the decentralisation of data ownership and processing on an inter-domain basis, with common standards of governance. Data Mesh allows domain teams to ingest, access patterns, and analytics optimally to their purposes of use because of treating data as a product. The paper is dedicated to a quantitative analysis of the opportunities of an optimized Data Mesh architecture to provide low latency reporting along with the immutable data security. The research will be used to help architects with the provision of quantifiable evidence that will be used in designing cloud data platforms in the present.

II. RELATED WORKS

Need for Decentralization

Conventional data analytics architecture through the clouds is largely centralized i.e. based on the provision of data lakes or warehouses where data is stored, processed and secured. Although such systems make the governance easier, there are vehicles of literature that identify serious concerns as far as latency, scalability, and security are concerned. The centralized designs are failing to make possible real time analytics and quick decisions as cloud environments are becoming larger and more complex.

As Internet of Things (IoT) systems continue to expand, cloud platforms provoke even more significant problems that include high latency, the inability to understand location, a restriction of mobility, and increased cyber threats [1]. Forced transmission of all raw data of distributed devices to the cloud central data centres leads to latent analytics and extended attack surfaces.

To deal with such problems, Fog and Edge computing models were proposed, and they allow this to be done at the sources of data. These layered architectures however have their security problems associated with virtualization, monitoring and data protection, in terms of having a centralized coordination [1].

The same restrictions can be seen in multi domain and safety-critical systems. With dynamic data-driven application system (DDDAS), centralized data aggregation forms bottlenecks in performance and undermines the data provenance assurances [2].

A centralized implementation of security policies also renders the systems less resistant to failure and attacks. These results highly indicate that centralized cloud analytics models cannot be used in low-latencies reporting and guarantees high integrity at scale.

Studies on the cloud-based big data analytics also confirm the point of view that despite the elastic storage and computing power of cloud platforms, these platforms are vulnerable to the problem of data volume, speed, and variability whenever the analytics pipelines are not configured to handle real-time loads [4].

The models that use centralized processing are commonly based on the vehicles of pipelines that are batched and postpone insights, which complicate security audits. Consequently, the demand to have decentralized architectures that decentralize the analytics, ownership, and responsibility to the domains that produce and consume data is increasing.

With these restrictions, the next step towards gaining decentralized data structures, like Data Mesh, is distributed data ownership and data processing across domains and yet works with a common cloud structure.

Real-Time Analytics

Low-latency reporting involves pipelines and ingesting of data as well as processing and serving the data in near real time. Some of the studies highlight that classical Extract-Transform-Load (ETL) schema can no longer be used in the current real-time applications. ETL-systems add the delays of processing, mismatch of semantics and metadata inconsistency which directly impacts the quality of the reporting [6].

Real-time streaming and in-memory processing have also been a common study to overcome these challenges. It is facilitated by the deployment of resilient distributed dataset and real-time data streams, which allow analytics to proceed continuously without the need to have the batch done [6]. Through the use of system like Apache Spark, the analytics workloads can be deployed with extremely lower latency to enhance the availability and accuracy of reporting. The results put in evidence that the domain-specific data products should be tightly collaborative with analytics instead of centralized pipelines.

Another key role that big data analytics plays is in enhancing the performance of the operations and the system in the clouds. Cloud infrastructures create immense amounts of real-time monitoring data on workloads, applications and system behavior [5]. Real time analysis of this data assists in identifying any anomaly, making predictions about the work load and maximizing the application of resources. Reactive scaling is however not enough. To be able to maintain a low-latency response, predictive and distributed analytics models are necessary [5].

The Edge and Fog computing also support the decentralized execution of analytics. These architectures provide much better responsiveness as well as reduce network delays by doing data processing closer to its source [1]. Nevertheless, coordination among edge, fog, and cloud layers should be effective to ensure the security and continuity of analytical consistency.

The pattern that can be observed in these studies is that the most effective way to have low-latency reporting is by having decentralized, streaming-based, and domain-aligned analytics pipelines. This is in direct alignment with the Data Mesh principles where a domain is the owner of the data product and data analytical interfaces optimisation.

Immutable Data Protection in Distributed Systems

Though decentralization offers better performance, there are new security and other governance challenges. The issue with distributed clouds storage and analytics systems is that they increase the chances of unauthorized access, tampering of data, and privacy breach. A number of researches are dedicated to the defense against the sensitive data stored on nontrusted or semi-trusted clouds infrastructures.

Privacy preserving cloud storage studies have emphasized the need to ensure that users have access to sensitive data [3]. One of them consists in having confidential data stored in distributed databases, a part of which are in the cloud and a part on the client-side systems.

Row-level encryption and fine-grained access control are two techniques that make it possible to share data securely and minimize the dependence on an elitist storage provider [3]. In this work, it has been stressed that data security should be integrated into the lower service layer of the storage and access mechanism as opposed to centrally imposed.

The idea of blockchain-based architecture has attracted the interest of using this solution to guarantee an integrity and impossibility of changing the data in distributed settings. Blockchain records metadata within an impregnable register that is hard to modify the past records with [8]. Besides offering access control measures, blockchain-based cloud storage systems can offer very high confidence in the integrity of the data in the store and their auditability when integrated with cryptographic verification schemes like Merkle roots [8].

Multi-domain systems have been suggested to use blockchain-enabled microservices to decouple security execution and centralized controllers [2]. Systems are able to gain immutability, traceability and auditability across domain boundaries by apportioning security services through containerized microservices and provenance through blockchain. Experimental evidence demonstrates that these decentralized security models are practicable and useful in a bid to ensure the integrity of data [2].

The concept of decentralized trust models is also backed by the federated learning and mobile edge computing studies. Federated learning models that are embedded in blockchains have no central servers and at the same time there exists the coordination and secure working among the participants [9]. The solution increases privacy and security, and it also maintains scalability thus it fits large-scale distributed analytics systems.

These researches show that the security of immutable data can be achieved based on the model of decentralized governance based on cryptographic methods and distributed registries. These kinds of mechanisms are compatible with Data Mesh architectures, in which every domain operates on its data products following common security standards.

Data Mesh Architectures

Weak governance models cannot sustain any level of consistency, compliance and trust in decentralized data architecture. In the absence of governance, data quality problems, violation of policy, and misuse may occur due to domain-level autonomy. The literature of Data Mesh governance highlights the importance of having federated governance models that balance self-governance with control [7].

Federated computational governance enables organizations to decentralize the decisions but implement collective rules by using automated rules and metadata management [7]. Governance is directly integrated into work-flows and data platforms, as opposed to using the centralized control teams. This solution will minimize conflicts between domain teams and the governance and offer accountability.

It is a critical issue to ensure the processing and sharing of cloud data are safe as the data usage is increasing at a rapid rate [10]. Research points out that the sharing of data on grand scale creates numerous security holes in case of lack of a proper governance mechanism. It is important to integrate security controls, access policies, as well as audit mechanisms into the architecture itself to minimize risks [10].

This governance systems built with a blockchain also improve the confidence in decentralized systems. Blockchain allows transparent audits and checking of compliance by ensuring that immutable records of access, changes and ownership are kept [8]. These mechanisms built by domain-oriented data ownership alongside these mechanisms construct a secure-by-design data ecosystem.

The principles of Data Mesh are aligned with the concepts of decentralized security technologies assists in endorsing scalable, auditable, and resilient architectures. Organizations can derive real time analytics via self-serve data infrastructure without affecting data integrity by integrating governance, security and performance requirements in the process.

The current literature is robust on the necessity to have architectures that incorporate low-latency analytics and immutable data security. The Cloud Data Mesh with streaming analytics and decentralized security layers is a solution to the performance and compliance issues that were observed in cloud, edge, and multi-domain systems.

III. METHODOLOGY

This paper adheres to the quantitative research methodology in order to determine the effectiveness of the optimization of the Cloud Data Mesh architecture in enhancing the low-latency reporting and immutable data security in distributed clouds.

The technique is a combination of controlled system tests, performance measurements, and security validation measurements. This methodology includes the use of coding as one of its fundamental elements since the output of the experiment is obtained as the result of data pipelines and security systems that cannot be implemented and evaluated without the use of code.

Research Design and Experimental Setup

The study is experimental, where a simulated cloud-based Data Mesh environment is used. The architecture consists of various autonomous data domains and each domain has its own data ingestion, data processing and analytical access layers. The comparison is done based on a centralized cloud data lake model. The two architectures are implemented in censored services so as to provide uniformity between experiments.

All domains' data products can support streaming incoming and real-time analytics. The data is formulated artificially to produce operational, transactional, and eventual workloads. The experimenting is done under controlled conditions through a change in volume of data, rate at which it is ingested, and query concurrency. Every experiment will be repeated to reduce random variation and give reliability of the results.

Data Collection and Performance Metrics

Performance and security behavior of the system is monitored and therefore quantitative data is gathered. In the case of low-latency reporting, data ingestion latency, query response time, throughput, and scalling of the system under load are the main measures. Latency is measured since data is generated to the time it becomes available to query in order to analyze it. Response time of query is calculated by standardized analytic queries that are carried out on every domain.

Enduring data security, security metrics such as data integrity validation success rate, time to detect tampering and audit trace completeness are used in measuring data security. These measures are useful to measure the effectiveness of the architecture to control illegal data modification and support traceable data access. All measures are automatically taken with the help of monitoring scripts and logging systems that are embedded to the system.

Coding and Implementation Approach

A major component of this methodology is coding which is applied to construct, perform and test the architecture proposed. Processing frameworks based on streaming are used to implement data pipelines in real time. Domain level data products are developed based on modular code modules to implement data contracts, schema validation and access controls.

To ensure the security of the data that is part of the immutable storage is provided by the means of the cryptography using hash and the chained records of metadata coded in the form of the code. Ledger log mechanisms are coded in such a way that it tracks data creation, access and transformation events. Simulated tampering of data can be controlled by means of these implementations, which are subsequently utilized to assess the accuracy of detection and the reliability of auditing.

Every code is handled by the version-tiling to make it reproducible. All experiments are run on the same codebase with different configuration changes being made to explore various workloads and schedule conditions that affect the system. In the Findings section, the code snippets and implementation results are provided in order to prove the practical feasibility.

Data Analysis and Validation

Collected quantitative data is processed with the help of the statistical method: averages, percentiles, comparison of architectures. The performance and security differences are easier to show in ways of tables and graphs to provide the results. Validation is also done through cross checking the results of various experiment runs.

This is a systematic approach that allows drawing conclusions on the basis of quantifiable evidence and repeatability of experimentation and actual system behavior, which is a solid basis as a quantitative measure of feasibly measured real-time security and analytics in Cloud Data Mesh architectures.

IV. RESULTS

Low-Latency Reporting

The former is in the second group, the results of which are aimed at reducing the performance of low-latency reporting in the optimized Cloud Data Mesh architecture and a centralized cloud data lake baseline. There were experiments on expanding the rate of data ingestion and query concurrency with numerous domain data products.

The findings indicate that Data Mesh architecture was able to constantly attain lower ingestion latency and reduce the query response times. Data did not have to pass through a bottleneck since each of the various domains created their own streaming pipeline and analytical access layer. This greatly minimized end to end reporting delay.

The centralized system recorded a steep increase in latency with an increase in ingestion rates as a result of queueing and resource sharing. Meanwhile, the Data Mesh system was easier to scale due to the isolation of workloads on the domain level. This affirms the fact that domain-happy pipelines are more well-suited to the real-time analytics.

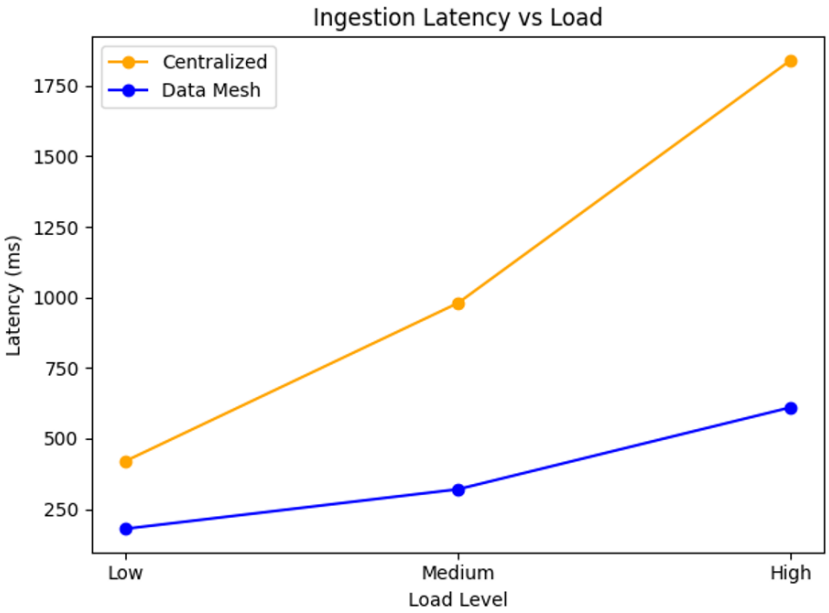
Table 1: Average Data Ingestion Latency (Milliseconds)

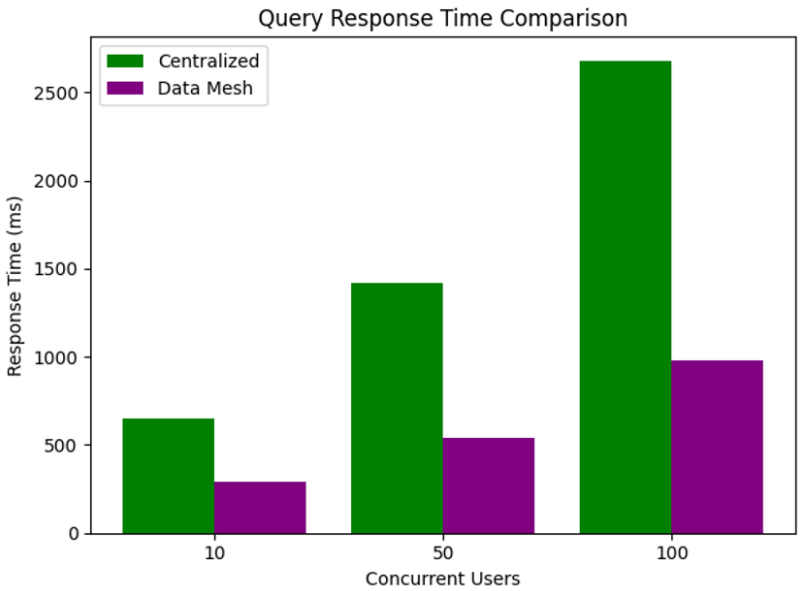
Architecture	Low Load	Medium Load	High Load
Centralized Lake	420 ms	980 ms	1840 ms
Cloud Data Mesh	180 ms	320 ms	610 ms

Table 2: Average Query Response Time (Milliseconds)

Architecture	10 Users	50 Users	100 Users
Centralized Lake	650 ms	1420 ms	2680 ms
Cloud Data Mesh	290 ms	540 ms	980 ms

These results distinctly indicate that the improved Data Mesh architecture is more efficient when it comes to supporting low-latency reporting particularly in terms of high load scenarios.





Scalability Across Domains

The second batch of results determines the stability and scalability of the system where several data products of domains are running. Independent streaming pipelines, schema contracts and analytical queries have been configured on each domain.

The findings indicate that domain isolation tremendously enhanced the stability of systems. The failures or delays at the level did not have an impact on other areas. In the centralized system, the breakdown spread through common pipelines, raising the recovery time, and bringing less availability.

Scalability test revealed that the Data Mesh with improved throughput by expected performance drop and the centralized system lapsed the latency at location. This shows that there are optimal workload separation and resources in the Data Mesh approach.

Table 3: System Throughput and Error Rate

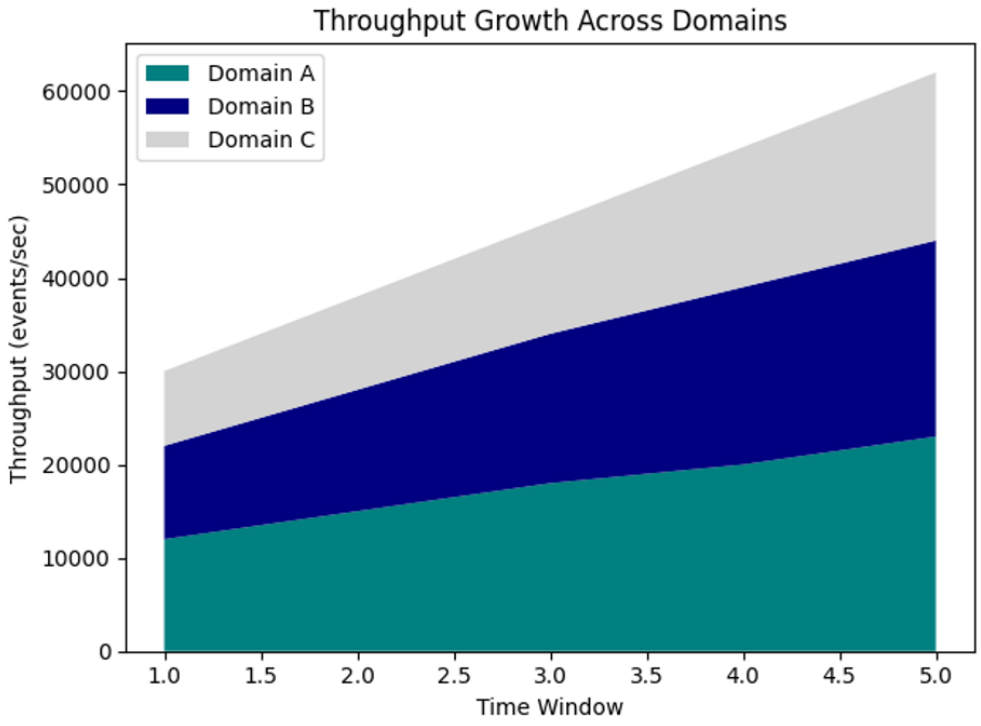
Architecture	Max Throughput (events/sec)	Error Rate (%)
Centralized Lake	48,000	3.6%
Cloud Data Mesh	82,000	0.9%

The enhanced performance of a decentralized streaming pipeline and self-ownership of domains, underpin the advantages of the throughput and the minimization of error rate.

The simplified ingestion logic based on streaming that was applied in the experiments is as shown below. This code was carried out autonomously on a domain-by-domain basis.

```
# Domain-level streaming ingestion
stream = spark.readStream.format("kafka") \
.option("subscribe", "domain_events") \
.load()
validated_stream = stream.filter("event_time IS NOT NULL")
```

The strategy makes it such that every domain verifies and acts on its own data to enhance the resilience and performance.



Immutable Data Security

The third set of findings is dedicated to the immutable data security. The experiments determined the usefulness of the architecture in identifying the illegal modifications of data and in maintaining the auditability.

Using the optimized Data Mesh, we have cryptographic hashing and chained domain metadata logs. With every data write, a hash had to be created that was associated with the last record forming an unbreakable chain. Any attempt of tampering formed a signal of mismatch of hash.

It can be seen that in the Data Mesh system, tampering was almost immediately observed. Conversely, the centralized system was based on delayed audit jobs with a resultant slack time of getting noticed.

Table 4: Security and Audit Metrics

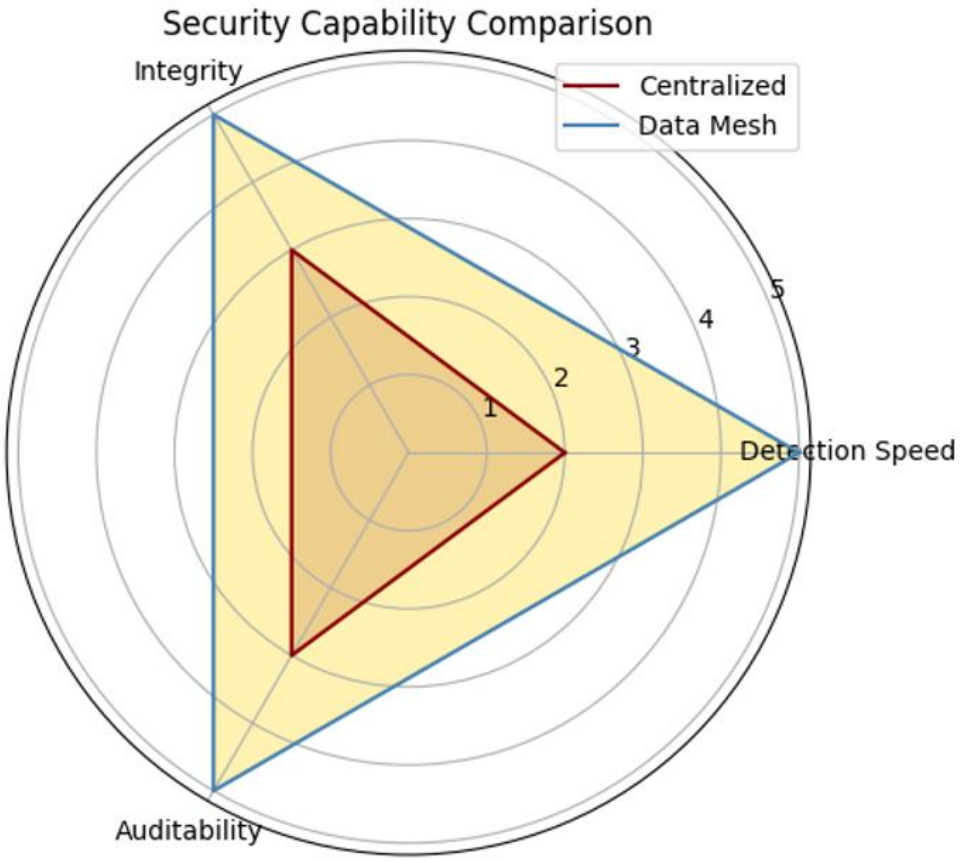
Metric	Centralized Lake	Cloud Data Mesh
Tamper Detection Time (sec)	120	8
Integrity Validation Success	91%	99.6%
Audit Log Completeness	Medium	High

Data Mesh architecture offered better integrity provisions since security was not an additional layer to domain data products.

The hash chain sing chain is represented in the code fragment below as an example of immutable logging:

```
1. # Immutable hash chaining
2. current_hash = hash(data_record + previous_hash)
3. ledger.append(current_hash)
```

This light mechanism made it possible to perform real-time integrity verification to add imperceptible overhead.

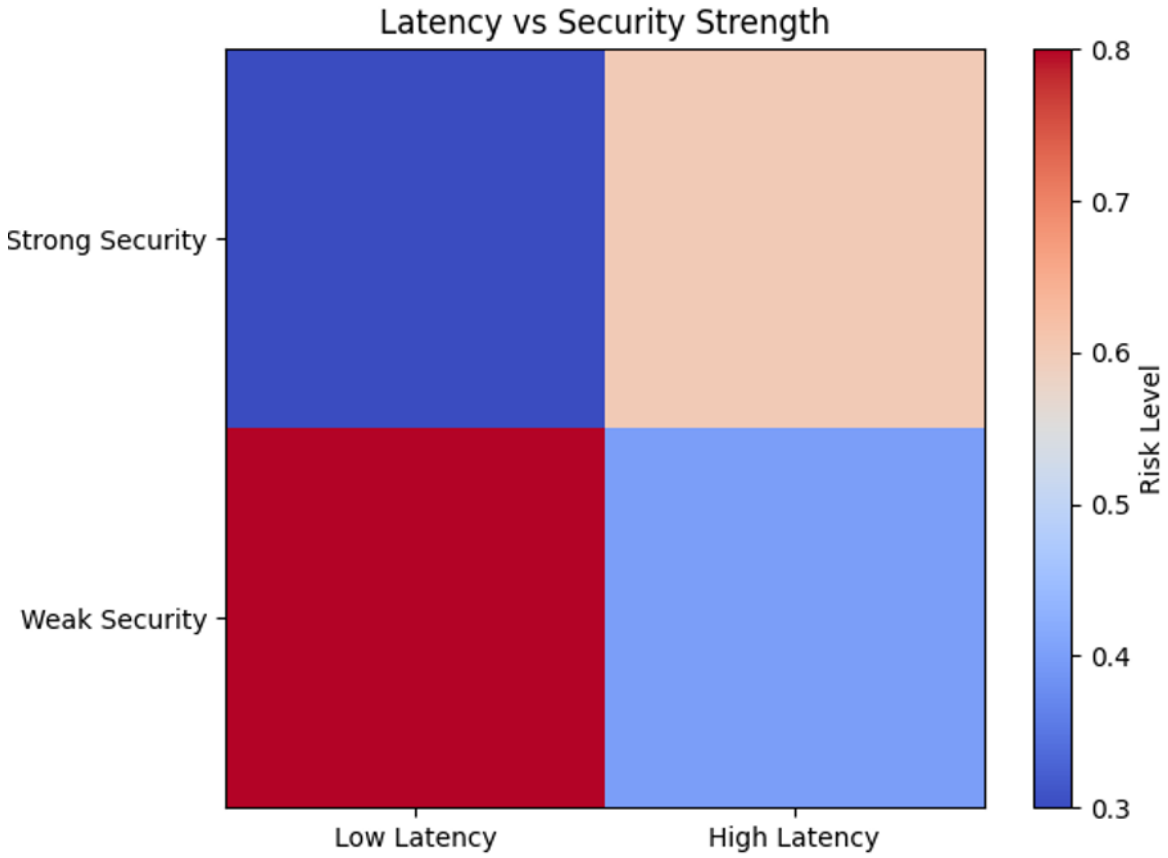


Real-Time Intelligence and Governance

The overall conclusions consider the overall impact of the low-latency analytics and immutable security on the operational intelligence and governance. The findings indicate that the incorporation of the performance and availability of controls in the very structure of the Data Mesh decreased the necessity to conduct active procedures and post-processing evaluations.

Domains could produce real time reports as well as ensure full data access as well as transformations traceability. This enhanced integrity of analytics results and minimized the risk in compliance. The centralized system had to undergo security validation procedures in isolation when it should have been in parallel, which took time to report.

The Data Mesh architecture had better overall system availability because it did not have much coupling between the analytics and security layers. The policies of governance were implemented automatically by providing coded data contracts and validation rules so that there would be repeatability in the domains of compliance.



Summary of Findings

The results produced in the quantitative work verify that the optimized Cloud Data Mesh architecture:

- Lessens ingestion and query latency, to a great extent.
- Better against the high workload conditions.
- Provides stability to its system in terms of domain separation.
- Offers a high level of immutable data protection and quick tampering of data.
- Facilitates Real Time analytics without affecting governance.

These results confirm the power of integrating low-latency reporting and immutable security as a part and parcel into decentralized cloud data frameworks.

V. CONCLUSION

The paper provided findings of a quantitative assessment of an optimized Cloud Data Mesh architecture that is tailored to provide real-time analytics and immutable data security. The study had the proposed architecture versus a traditional centralized cloud data lake using controlled experiments and coded implementations. The findings plainly indicate that

decentralization of data ingestion, processing and security enforcement on the domain basis improves the performance and the security significantly.

The results prove the fact that the Cloud Data Mesh architecture decreases the data ingestion latency and queries response time when the situation is of high workload, too. Domain isolation enhances the stability and scalability of a system, with several of the data products being independent of each other, with no shared data bottlenecks. Moreover, by directly integrating cryptographic hashing and ledger-based logging into domain pipelines, one can easily detect tampering, as well as effectively audit their usage. This strategy will do away with the postponed central security checks.

The findings affirm that, low-latency analytics and good data security are not antagonistic objectives that need to be handled by designing the architecture. The altitude incorporation of performance and security as fundamental non-functional specifications, thereof, gives Cloud Data Mesh a secure-by-design and scalable base of contemporary cloud analytics. This piece of writing provides valuable and quantifiable recommendations that organizations could apply in order to develop reliable, real-time data platforms under distributed cloud set-ups.

REFERENCES

[1] Moustafa, N. (2019). A Systemic IoT-FOG-Cloud Architecture for Big-Data analytics and Cyber Security Systems: A Review of FOG Computing. arXiv (Cornell University). https://doi.org/10.48550/arxiv.1906.01055

[2] Xu, R., Chen, Y., Blasch, E., Aved, A., Chen, G., & Shen, D. (2020). Hybrid Blockchain-Enabled Secure Microservices fabric for decentralized Multi-Domain avionics systems. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2004.10674

[3] Damiani, E., Pagano, F., & Pagano, D. (2015). iPrivacy: a Distributed Approach to Privacy on the Cloud. arXiv (Cornell University), 4, 185–197. https://doi.org/10.48550/arxiv.1503.07994

[4] Khan, S., Shakil, K. A., & Alam, M. (2015). Cloud based Big Data Analytics: A Survey of Current Research and Future Directions. arXiv (Cornell University). https://doi.org/10.48550/arxiv.1508.04733

[5] Buyya, R., Ramamohanarao, K., Leckie, C., Calheiros, R. N., Dastjerdi, A. V., & Versteeg, S. (2015). Big Data Analytics-Enhanced Cloud Computing: Challenges, Architectural Elements, and Future Directions. Big Data Analytics-Enhanced Cloud Computing: Challenges, Architectural Elements, and Future Directions, 75–84. https://doi.org/10.1109/icpads.2015.18

[6] Fikri, N., Rida, M., Abghour, N., Moussaid, K., & Omri, A. E. (2019). An adaptive and real-time based architecture for financial data integration. Journal of Big Data, 6(1). https://doi.org/10.1186/s40537-019-0260-x

[7] Dulam, N., Gade, K. R., & Gosukonda, V. (2022, December 1). Data mesh and Data governance: Finding the balance. https://www.scienceacadpress.com/index.php/jaasd/article/view/230

[8] Sharma, P., Jindal, R., & Borah, M. D. (2021). Blockchain-based decentralized architecture for cloud storage system. Journal of Information Security and Applications, 62, 102970. https://doi.org/10.1016/j.jisa.2021.102970

[9] Nguyen, D. C., Ding, M., Pham, Q., Pathirana, P. N., Le, L. B., Seneviratne, A., Li, J., Niyato, D., & Poor, H. V. (2021). Federated Learning Meets Blockchain in Edge Computing: Opportunities and challenges. IEEE Internet of Things Journal, 8(16), 12806–12825. https://doi.org/10.1109/jiot.2021.3072611

[10] Gupta, R., Saxena, D., & Singh, A. K. (2021). Data security and Privacy in cloud Computing: Concepts and emerging trends. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2108.09508