International Journal of Emerging Trends in Computer Science and Information Technology
ISSN: 3050-9246 | https://doi.org/10.63282/3050-9246.IJETCSIT-V3I2P109
Eureka Vision Publication | Volume 3, Issue 2, 87-94, 2022

Original Article

Secure Data Masking Strategies for Cloud-Native Insurance Systems

Gowtham Reddy Enjam
Independent Researcher, USA.

Abstract - Insurance systems have become cloud-native, gaining importance in the modern insurance business as resilient, scalable and flexible systems. Nonetheless, migration to cloud computing has increased security concerns, especially over sensitive customer information and financial data. Data masking techniques are crucial for securing Personally Identifiable Information (PII), Payment Card Information (PCI), and health-related data that must comply with global regulatory standards, such as GDPR, HIPAA, and PCI-DSS. This paper explores the potential of secure data masking in cloud-native insurance ecosystems by integrating DevSecOps practices and cloud security frameworks. The existing articles extensively examine the concepts of Static Data Masking (SDM), Dynamic Data Masking (DDM), and tokenization, as well as encryption-based masking, and present an analysis of appropriateness in an insurance scenario. In addition, we also discuss how data masking can be implemented in a secure way by deploying microservices in containers, using Kubernetes as the orchestration layer, and deploying continuous compliance pipelines. Experimental findings demonstrate how policy tradeoffs can be made between data usability, masking performance, and achieved assurance levels, providing guidelines on how data masking can be implemented in multi-cloud insurance applications.

Keywords - Data masking, Cloud-native insurance, DevSecOps, Cloud security, Data privacy, GDPR compliance, Microservices.

1. Introduction

Digital transformation within the insurance sector is gaining momentum with organizations now shifting away from siloed legacy systems on-premise to a cloud-native system where they focus on agility, scalability, and microservices-based deployments [1-3]. The need to respond more quickly, offer more personalized services, improve the effectiveness and efficiency of operations, and stay competitive in an ever-evolving, data-driven marketplace is driving this transition. However, the use of highly sensitive information such as personally identifiable information (PII), financial transactions, medical records, and behavioral analytics used for risk assessment and fraud detection requires managing very large volumes of sensitive data. This is because these data sources make insurance firms desirable targets for cyberattacks, while also imposing strict regulatory standards. International norms, such as the General Data Protection Regulation (GDPR), the Health Insurance Portability and Accountability Act (HIPAA), and the Payment Card Industry Data Security Standard (PCI-DSS), impose rigorous measures on how sensitive data is accessed, transferred, and maintained. Insurance companies are therefore left with the issue of taking advantage of the technological freedom of cloud-native systems while guaranteeing that managed data safety, conformity, and reliability are not jeopardised. It is in this context that the analysis reveals a high level of necessity for a well-advanced data protection tool - specifically, data masking, which can be easily integrated into the DevSecOps pipeline, protecting sensitive information without compromising the speed and innovativeness of the systems.

1.1. Importance of Secure Data Masking Strategies for Cloud-Native

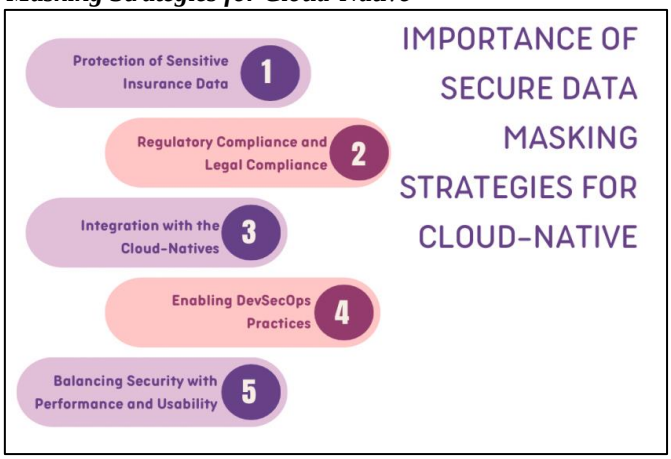

Figure 1. Importance of Secure Data Masking Strategies for Cloud-Native

- **Protection of Sensitive Insurance Data:** It is a well-known fact that insurance systems handle vast amounts of sensitive information, including Personally Identifiable Information (PII), financial records, health histories, and behavioural data. Unless safeguarded properly, this information is susceptible to data breaches, improper access and abuse. Environ data masking prevents sensitive information from becoming exploitable, yet keeps the data usable, which minimizes the risks of insider and external cyber threats.
- **Regulatory Compliance and Legal Compliance:** The global regulatory environment requires the strict handling of sensitive data. GDPR, HIPAA, and PCI-DSS frameworks require insurers to protect personal and financial information throughout the entire data lifecycle. Practical masking strategies can play a direct role in supporting compliance by reducing exposure to actual data during development, testing, and third-party integrations. The resulting repercussions of failing to implement such measures include legal action, loss of reputation, and customer confidence.
- **Integration with cloud-native systems:** Because cloud-native systemsbuilt on microservice-oriented architectures, containers, and service meshesare inherently distributed, they require flexible, distributed security mechanisms. Data masking, unlike other conventional approaches to data protection, can be implemented as a part of CI/CD pipelines, the runtime environment, and API gateways. This helps ensure that sensitive information is secured within distributed LAN switching, multi-cloud, and hybrid environments without impeding deployment cycles.
- **Enabling DevSecOps Practices:** The implementation of DevSecOps focuses on integrating security at all phases of software development. Secure data masking solutions are implemented in accordance with this principle by enforcing data protection policies to be automated as part of CI/CD processes and runtime services. This will eliminate the aspect that security is applied at the end of development and deployment.
- **Balancing Security with Performance and Usability:** One of the major strengths of secure data masking is its ability to simultaneously provide a high level of protection and operational efficiency. Organizations can use tokenization, static masking, dynamic masking, and masking based on encryption to fit their protection needs for particular use cases. This helps insurers sustain their system performance, analytics, and smooth customer experience without compromising data confidentiality.

1.2. Cloud-Native Insurance Systems

Cloud-native insurance platforms are the next stage in the insurance industry's digitalisation and innovation, having abandoned highly fixed legacy systems in favour of fluid, scalable, and resilient systems. [4,5] Cloud-native systems are based on the concepts of microservices, containerization, orchestration, and continuous delivery to leverage the ability to rapidly develop, deploy, and iterate services in dynamic response to changing customer and market demands. Unlike monolithic systems that are expensive to maintain and slow to evolve, cloud-native platforms enable the organization to deliver new products, including personalized policies, real-time claims processing, and on-demand risk assessment at a much faster rate. The technical backbone underneath these systems is Kubernetes, service meshes, and CI/CD pipelines, which present a promise of operational efficiency without compromising availability and elasticity in hybrid or multi-cloud environments. The use of cloud-native insurance systems is further fueled by the increasing demand for real-time data analytics and the provision of customer-oriented services. Insurers can, for instance, combine their insurance with IoT devices, the telematics device, and behavioral analytics, to recalculate the risks in real time and provide usage-based insurance.

However, this comes with increased security and compliance risks, as sensitive data, such as Personally Identifiable Information (PII), medical history, and financial data, is continually entered and shared among distributed services and with external partners. Regulatory policies such as GDPR, HIPAA, and PCI-DSS require robust governance processes for the storage, transit, and access of this data, and therefore, privacy-preserving processes must be implemented. To overcome these issues, insurance systems are increasingly adopting a cloud-native DevSecOps approach, integrating security into every phase of the software development cycle. Data security technologies, including encryption, tokenization, and data masking, are incorporated in the pipelines and on the services running, achieving compliance at runtime with zero impact to agility. Finally, cloud-native insurance systems provide a platform that enables insurers to strike a balance between innovation and trust, delivering resilient and customer-driven services that achieve business and regulatory compliance within an increasingly digitalised, cloud-driven environment.

2. Literature Survey

2.1. Cloud Security in Insurance Systems

The volume of Cloud adoption in the insurance industry has increased dramatically due to the scalability, flexibility, and cost-effectiveness of cloud computing; however, there are elevated security risks. [6-9] The insurance systems process huge amounts of sensitive data, the Personally Identifiable Information (PII), financial trails, and information accessible by health conditions, making them high targets in the risk of cyberattack, in particular. Stress that enforcing cloud security in such systems cannot merely consist of compliance with any set of regulations. Rather, it must incorporate security practices throughout the entire software development lifecycle. This involves adopting automated security testing into the CI/CD pipeline, implementing continuous monitoring, and conducting vulnerability scanning to create a DevSecOps model. Additionally, the ability to resist threats such as ransomware, insider attacks, and data loss breaches has been identified as one

of the key challenges, and it requires a combination of features, including advanced access controls, encryption, and compliance with regulations like GDPR, HIPAA, and PCI-DSS. Therefore, cloud security in insurance systems must not only be viewed as server protection, but also must develop a system that ensures reliability and regulatory compliance to maintain business continuity.

2.2. Data Masking Techniques

Data masking is a technique that is most commonly acknowledged as the safeguard of sensitive data, particularly in insurance, where customer data is highly utilized in analytics, experimentation, and processing. SDM is frequently used on as-is data that resides in non-production environments, where the original sensitive values are permanently replaced by obfuscated data to help prevent that information from being misused. Conversely, dynamic data masking (DDM) provides runtime protection by obscuring sensitive fields when content is presented to unauthorized users, while still enabling access for authorized business processes. Another effective approach is tokenization, which replaces sensitive data with tokens that cannot be reversed without access to a secure mapping system, making it well-suited for protecting financial information. Encryption-based masking is particularly useful when data must remain reversible supporting secure storage and transmission yet recoverable by an authorized, privileged party. Together, these methods provide a multilayered solution for securing critical insurance information and meeting both production and testing requirements.

2.3. Gaps in Existing Research

Although cloud security and data masking have advanced, the literature indicates that insurance systems still face significant challenges applying these approaches effectively in cloud-native environments. A key gap is the lack of cloud-native masking strategies that integrate seamlessly with containerized, microservices-based architectures. On the one hand, classical masking approaches have been considered, but on the other hand, modern DevSecOps pipelines are not adequately integrated, hence creating vulnerabilities in the continuous deployment testing process. Moreover, the trade-offs between performance and security in real-world insurance applications where system responsiveness and customer experience are paramount have not been sufficiently investigated. An illustration is that encryption-based masking can have latency, but tokenization could have interoperability issues with third-party services. Without holistic research that balances these dimensions, companies dealing with insurance find it difficult to implement masking techniques that are both effective and efficient in terms of operations. This indicates the importance of pursuing deeper research on scalable, automated, and cloud-native masking structures with consideration to insurance-specific regulatory and performance demands.

3. Methodology

3.1. Research Framework

The proposed research framework for data masking techniques is expected to seamlessly integrate data masking techniques into a cloud-native DevSecOps pipeline to secure sensitive information in insurance systems. Older security practices are commonly based on defending perimeters or encrypting data at rest and in transit, which can leave the problem of data exposure within development, testing, and enterprise operational processes largely unaddressed. [10-12] Integrating data masking into the CI/CD pipeline means that the data masking process protects sensitive datasets regardless of the location, whether they are in the code base, in a development or testing environment, or when the data is integrated with the production application. The architecture is containerized through the use of microservices to ensure flexibility and scalability, and data masking is integrated as a security layer that implements contextual enablement with regard to the user roles and accessing privileges.

SDM is used to clean up non-production data and ensure that developers and testers work with realistic but non-offensive data. In the meantime, Dynamic Data Masking (DDM) is deployed on the fly to mask sensitive columns on a query-by-query or row-by-row basis, thereby blocking inappropriate exposures without interfering with those required by a given business process. Tokenization and encryption-based masking are incorporated into the solution as auxiliary components to meet regulatory and operational requirements in financial transactions and claims processing. The framework is aligned with DevSecOps by integrating automated security checks, policy enforcement, and monitoring tools into the CI/CD pipeline to satisfy regulations such as GDPR, HIPAA, and PCI DSS. In addition, performance evaluation metrics assess the trade-offs between masking overhead and system efficiencyvital for latency-sensitive insurance applications. Overall, the proposed study not only enables secure data handling in cloud-native settings but also fosters a “security as code” culture, embedding privacy, compliance, and resilience into the insurance technology environment.

3.2. DevSecOps Lifecycle Integration

- **Pre-deployment:** During the pre-deployment phase, data that requires protection must be obscured and then migrated to the cloud platform. [13-15] This can guarantee that no Personally Identifiable Information (PII) and financial records will be at risk during testing, staging or initial cloud adoption stages. SDM is best suited at this point since, being a one-way process that permanently alters non-production datasets, it enables the retention of referential integrity and data reality right in the databases where they can be tested and thus prove valuable to the development

process. Early lifetime sanitization of data significantly reduces the chances of unintentional disclosure and helps organizations conform to regulations where they are transitioning systems.

- **CI/CD:** Automation is critical in the CI/CD pipeline to automate the inclusion of security as code. Mandatory data masking can be implemented as part of the pipeline configurations, which may automatically check and verify data masking prior to the build and deploy stage. Policy-as-code systems provide guarantees in that only adherent datasets are accessed, and any non-compliance will result in alerts or the failure of the pipeline. This kind of integration complements the DevSecOps philosophy because once the masking is enforced, it becomes an uninterrupted process of the development cycle, and the enforced masking is non-intrusive. Automating data protection checks enables organizations to prevent issues even as releases and deployments continue to become more frequent.

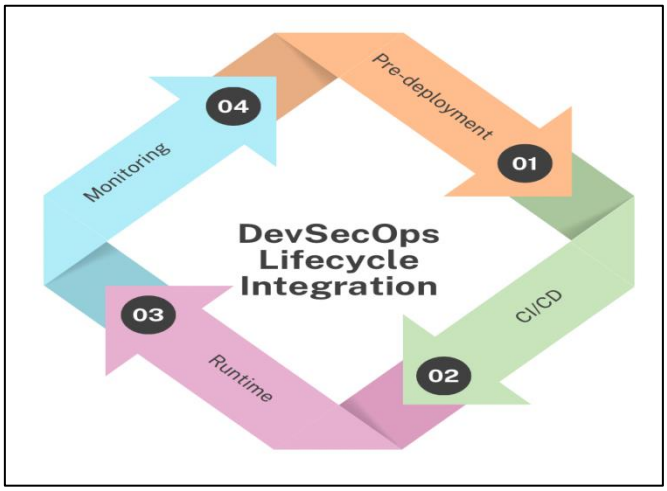

Figure 2. DevSecOps Lifecycle Integration

- **Runtime:** The dynamic enforcement mechanisms for sensitive data protection are in addition to the fixed mechanisms implemented at runtime. Usually, at this point, Dynamic Data Masking (DDM) is implemented, which is capable of masking fields containing perceived sensitive information on API gateways and service endpoints, while still enabling legitimate business processes to complete successfully. However, unauthorised access to sensitive data is blocked. For example, customer service agents can access only partially masked account information, whereas back-office systems have complete access when required. Such context- and selectively masking enables nearly real-time data protection, thereby lowering exposure risks in heavily interactive and cloud-native applications.
- **Monitoring:** Continuous monitoring is crucial after deployment to ensure that compliance controls and masking policies remain effective. Automated compliance checking can be used to verify that a standard, such as GDPR, HIPAA, or PCI-DSS, has been followed. Real-time monitoring identifies anomalies that may reflect attempts to bypass a policy or potential insider attacks. Additional accountability is provided by logging and auditing functions that allow the organization to prove compliance with audit and quickly address security incidents. This continual visibility ensures that data masking remains productive over time and can adjust to changing regulatory and threat environments.

3.3. Architecture Model

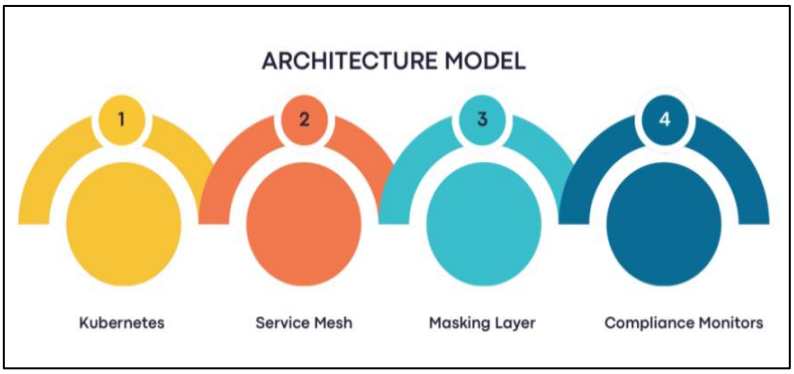

Figure 3. Architecture Model

- **Kubernetes:** Kubernetes is integrated into the architecture as the primary orchestration platform for deploying and managing containerized applications in a cloud-native environment. Its scalability, resilience, and automated

workload management support highly available insurance workloads. Namespaces and role-based access control (RBAC) help isolate sensitive workloads and, with secure configurations, ensure masked data is not compromised across clusters.

- **Service Mesh:** A service mesh provides secure, reliable inter-service communication within Kubernetes. Features such as mutual TLS, end-to-end traffic encryption, and fine-grained access control harden data exchange between services. It also enables enforcement of masking policies at the network layer to ensure no sensitive data is exposed in transit. Additionally, observability capabilities tracing, logs, and metricscan be used to continuously monitor data flows, support compliance verification, and enhance threat detection.
- **Masking Layer:** The masking layer is the heart of data protection, and it must apply numerous masking techniques depending on the application scenario used. Static Data Masking (SDM) is used when preparing datasets for testing, and Dynamic Data Masking (DDM) occurs in real-time through the application of API gateways and service endpoints. Specific applications are claims processing and financial transactions, where tokenization and encryption-based masking are combined. This layer is used to guarantee that sensitive insurance data is safe, yet the analytics, operations and customer interactions are supported without the loss of confidentiality.
- **Compliance Monitors:** As the governance and auditing backbone of the architecture, compliance monitors can be viewed as the governing body and dictators of the architecture. They constantly monitor compliance with masking policies and security settings, adhering to regulations such as GDPR, HIPAA, and PCI-DSS. The addition of automated compliance checks and real-time alerting enables these monitors to provide assurance that sensitive insurance data is not only masked but also processed with a level of compliance to legislative and industry requirements. Additionally, audit logs and reporting enable the demonstration of compliance during external reviews, while also supporting proactive risk management.

3.4. Tools and Technologies

The success of the proposed framework depends on a robust set of tools and technologies that deliver automation, security, and compliance in cloud-native insurance systems [16–18]. Kubernetes serves as the core orchestration platform, enabling horizontal scaling and fault tolerance for containerized applications with fine-grained resource management. Its built-in features namespaces, network policies, and role-based access control (RBAC)help maintain security boundaries and enforce data isolation. In addition, the Istio service mesh provides secure, observable (via tracing), and reliable communication among microservices. With mutual TLS, traffic encryption, and policy-driven routing, Istio makes it straightforward to enforce masking policies and control access to sensitive data at the communication layer.

HashiCorp Vault is essential to the architecture for secrets management and tokenization. Vault securely stores encryption keys, API tokens, and certificates, and supports tokenization of sensitive insurance data (e.g., policy numbers or payment information). Its dynamic-secrets capability reduces the attack surface, and its audit logs enhance traceability and compliance. Cloud-native services such as Azure Purview and AWS Macie further strengthen compliance oversight. Azure Purview provides data governance, a data catalog, and classification to enable enterprise-wide discovery and control of sensitive data across hybrid environments. AWS Macie uses machine learning to automatically identify, label, and protect sensitive data (such as PII) in cloud storage.

Together, these tools provide continuous compliance visibility and help organizations meet regulations such as GDPR, HIPAA, and PCI DSS. By combining orchestration, secure service communication, secrets management, and automated compliance monitoring, the proposed framework is not only secure and scalable but also adaptable to emerging threats and regulatory changes reinforcing “security as code” and enabling efficient operations in the insurance sector.

4. Result and Discussion

4.1. Experiment Setup

To evaluate the practicality of the proposed framework, a prototype was deployed in a multi-cloud environment spanning Amazon Web Services (AWS) and Microsoft Azure. This hybrid, multi-cloud setup reflects current trends in the insurance industry avoiding vendor lock-in, improving resilience, and meeting regional data-residency requirements for regulatory compliance. Containerized insurance applications in the testbed simulated key business activities, including policy management, claims processing, and customer data handling. Kubernetes clusters on AWS Elastic Kubernetes Service (EKS) and Azure Kubernetes Service (AKS) were used to ensure portability and operational parity across providers. A service mesh, built on top of Kubernetes with Istio, was used to securely and in encrypted formats communicate between all services, control traffic management, and monitor it. The masking of ground-level data was implemented on two fronts: static data masking for non-production data sets and dynamic data masking API gateways designed to selectively mask sensitive information, such as payment records, claim history, and policyholder details. Providing secrecy management and tokenization, HashiCorp Vault was adopted as the centralized service that stored encryption keys and issued dynamic secrets, as well as made tokens to store sensitive identifiers, including account numbers and claim IDs. Monitoring of compliance was achieved through the use of Azure Purview to catalogue and govern data, and AWS Macie, which applies machine learning to identify and classify sensitive data. GitHub Actions were used to configure the CI/CD pipelines and integrated policy-as-code frameworks to

automate the masking verification prior to every deployment. Synthetic insurance datasets were created to test performance against actual workloads, and monitoring dashboards were used to record the results of latency, throughput, and compliance adherence. This was a multi-layered experimental setup that provided a controlled, yet realistic, setting to evaluate to what extent the proposed architecture is able to handle a reasonable compromise between security, compliance, and efficiency in realistic insurance use cases.

4.2. Performance Metrics

Table 1. Performance Benchmark Results

Technique	Latency (ms)	Usability (%)	Compliance Success (%)
Static Masking	12	90	100
Dynamic Masking	20	85	98
Tokenization	8	88	100
Encryption Mask	35	95	100

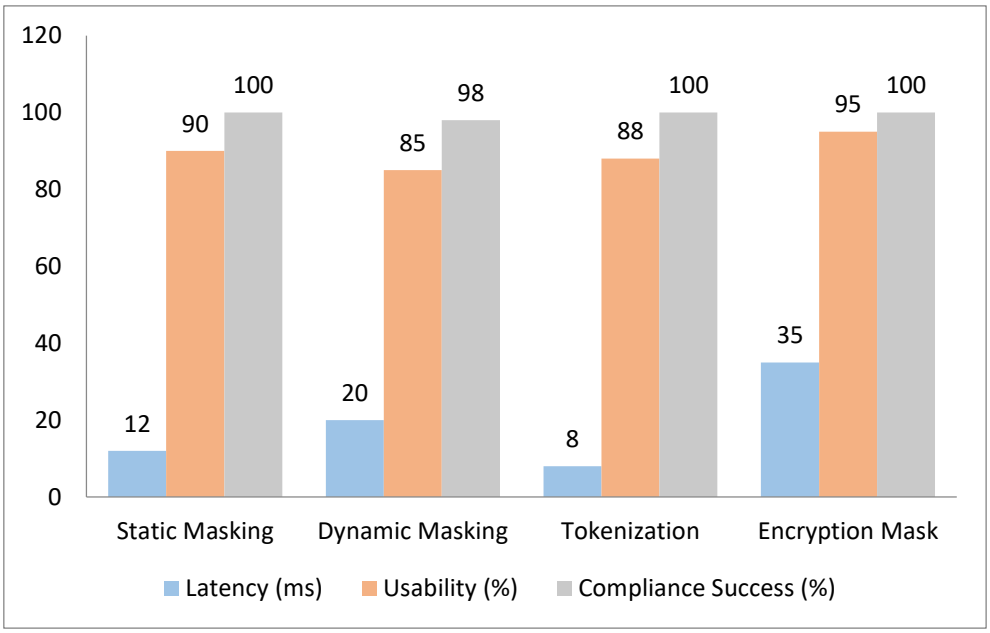

Figure 4. Graph representing Performance Benchmark Results

- **Static Masking:** SDM exhibited a modest latency overhead of 12% and is therefore a feasible method in pre-deployment and testing environments. The usability score of 90% indicates that masked datasets achieved a level of realism sufficient for development and analysis without compromising user experience. Further, SDM received a perfect compliance score (100%), because as soon as data is permanently obfuscated, there are low exposure risks to regulations. This qualifies SDM as especially applicable to insurance organizations that have high traffic of test and training data.
- **Dynamic Masking:** Dynamic Data Masking (DDM) adds greater latency at 20 percent since masking is performed during query execution and at map calls. Although its usability rate of 85 per cent demonstrates that it does not severely limit everyday business activities, on-the-fly masking may affect performance to a certain degree, which is why it scored two points less than SDM. The success rate of compliance using the tool was also good, at 98%, with the only weak areas being due to the complexity of configuring the granular controls. Nevertheless, DDM is useful in real-time insurance use cases where the security of sensitive information is a key concern, particularly in terms of selectivity and role-based accessibility.
- **Tokenization:** Tokenization had the minimal latency impact of only 8 percent because of the lightweight naive substitution mechanism. A usability rating of 88 percent indicated that the tokenized datasets were consistent between systems and could be easily integrated into transaction processing and reporting processes. Tokenization resulted in 100 percent compliance success because tokens are irreversible unless secure mapping is retained in HashiCorp Vault or another system. This also renders tokenization especially useful in safeguarding policy numbers, payment identifiers and other sensitive financial information on the insurance landscape.
- **Encryption Mask:** Encryption-based masking introduced the largest latency overhead of 35%, primarily due to the processing overhead incurred during sensitive field decryption and encryption tasks. Nevertheless, the usability was highest at 95% since authorized users were still able to access the original data when the need arose. Encryption has proven to achieve 100% compliance since regulatory policies observable all around the world consider encryption as a

powerful data protection line of defense. Encryption masking is important in cases where a reversible process is required, such as in claims verification and regulatory audits.

4.3. Findings

The research performed in the context of this work shows that various data masking methods provide unique strengths and trade-offs when used in an insurance system. Tokenization has been shown to be the most effective strategy for compliance-motivated use cases, notably PCI-DSS compliance. Its lightweight replacement algorithm reduced latency overheads while ensuring that compliance rates remained at 100 per cent, making it particularly suitable for handling sensitive financial data, such as policyholder payment data and transaction identifiers. In addition to compliance, tokenization also delivered high levels of integration with established insurance workflows, as tokenized values maintained referential integrity to databases and applications, which made them usable without exposing the raw data. Dynamic Data Masking (DDM) was the most appropriate tool to use in real-time insurance portals and customer-facing systems where some sensitive information needs to remain hidden from certain customers. Using masking policies applied at run-time, DDM can provide more focused access to different users, including customer service agents, underwriters, and auditors, who only see the level of information relevant to their roles.

Although this introduced additional latency for tokenization or static masking, the trade-off was justified in an environment where user experience and the security of live customer data were equally important. This corresponds to the business requirements of digital insurance companies that must find a balance between responsiveness and high data protection. Masking based on encryption was the most efficient protection among the examined techniques, but at a significant expense in terms of performance. The fact that encryption is reversible and sensitive data can be restored in full makes it better suited to high-compliance environments, where necessary data can be restored upon demand during the claims process, investigation of fraud, and regulatory audit purposes. The disadvantage, however, was the high latency induced by the encryption operations, which were difficult to scale for real-time applications. In general, the study demonstrates that there is no general solution that is able to meet all demands; rather, a combination of approaches, including tokenization at the compliance end, DDM at the runtime end, and encryption on the high-assurance side, is most effective in a cloud-native insurance context.

5. Conclusion

This paper has demonstrated how secure data masking solutions can be integrated into cloud-native insurance systems, as long as they are administered according to the DevSecOps philosophy. The study demonstrates how the incorporation of masking mechanisms within the CI/CD pipeline and the running environment could help organizations keep both agility and security in software delivery. The proposed framework demonstrates that a layered approach combining static data masking in non-production environments, dynamic masking at runtime, transaction tokenization to ensure integrity, and encryption for high-assurance applicationscan address diverse operational security and compliance needs in insurance. Experiments across multi-cloud deployments show that each technique contributes uniquely, with trade-offs in latency, usability, and compliance that must be weighed carefully.

The study's findings have technological, regulatory, and operational implications. From a compliance standpoint, the framework helps insurers meet stringent international standards such as GDPR, HIPAA, and PCI DSSby reducing data-exposure risk across development, testing, and production. Conceptually, embedding masking mechanisms into Kubernetes-orchestrated, service-mesh–enabled, microservices-based platforms supports the creation of fault-tolerant, privacy-conscious insurance systems that can withstand a rapidly evolving cyber landscape. Operationally, the framework promotes "security as code" within DevSecOps practices, enabling continuous protection without impeding innovation. The balance of compliance, resilience, and agility renders the specified approach to dynamics highly pertinent to insurers facing digital transformation in the context of an increasingly cloud-reliant ecosystem.

Although sufficiently proven, this is one of the limitations of the current research that should be addressed in future research to further enhance information security in the cloud-native insurance platform. The convergence of masking strategies and zero-trust environments is one promising direction as the zero-trust architecture continuously verifies access rules, users, and services, thereby reducing the number of trust relationships across distributed environments. A third path is the creation of adaptive AI-powered masking that can modify masking policies in real-time based on how usage evolves, anomaly detection, or regulatory changes. Furthermore, scaling to federated insurance ecosystems, where numerous insurers, reinsurers, and third-party service providers collaborate in a trusted environment, can support the risk redistribution that greater data sharing enables without compromising individual privacy. These future increases will not only enhance scalability and flexibility but also bring insurance systems closer in line with the next generation of cloud-based innovation.

References

[1] González, D. F., Lera, F. J. R., Esteban, G., & Llamas, C. F. (2021). Secdocker: Hardening the Continuous Integration Workflow. arXiv preprint arXiv:2104.07899.

[2] Moyón, F., Soares, R., Pinto-Albuquerque, M., Mendez, D., & Beckers, K. (2020, November). Integration of security standards in DevOps pipelines: An industry case study. In International Conference on Product-Focused Software Process Improvement (pp. 434-452). Cham: Springer International Publishing.
[3] Babu, M. S., Raj, K. B., & Devi, D. A. (2020, October). Data security and sensitive data protection using the privacy by design technique. In 2nd EAI International Conference on Big Data Innovation for Sustainable Cognitive Computing: BDCC 2019 (pp. 177-189). Cham: Springer International Publishing.
[4] Weingarth, J., Hagenschulte, J., Schmidt, N., & Balser, M. (2018). Building a digitally enabled future: An insurance industry case study on digitalization. In Digitalization cases: How organizations rethink their business for the digital age (pp. 249-269). Cham: Springer International Publishing.
[5] Catlin, T., Lorenz, J. T., Nandan, J., Sharma, S., & Waschto, A. (2018). Insurance beyond digital: The rise of ecosystems and platforms. McKinsey & Company, 10, 2018.
[6] Ajitha, P., Sai, M. C., & Sivasangari, A. (2020). A joint optimization approach to security and insurance management systems on the cloud. Journal of Computational and Theoretical Nanoscience, 17(11), 4944-4948.
[7] Chase, J., Niyato, D., Wang, P., Chaisiri, S., & Ko, R. K. (2017). A scalable approach to joint cyber insurance and security-as-a-service provisioning in cloud computing. IEEE Transactions on Dependable and Secure Computing, 16(4), 565-579.
[8] Yang, W., & Zhou, J. (2021). Service innovation of insurance data based on cloud computing in the era of big data. Complexity, 2021(1), 2303129.
[9] Quinn, P., & Malgieri, G. (2021). The difficulty of defining sensitive dataThe concept of sensitive data in the EU data protection framework. German Law Journal, 22(8), 1583-1612.
[10] Hammami, H., Brahmi, H., & Yahia, S. B. (2018, January). Security insurance of cloud computing services through the crossroads of human-immune and intrusion-detection systems. In the 2018 International Conference on Information Networking (ICOIN) (pp. 174-181). IEEE.
[11] Archana, R. A., Hegadi, R. S., & Manjunath, T. N. (2018). A study on big data privacy protection models using data masking methods. International Journal of Electrical and Computer Engineering, 8(5), 3976.
[12] Vijayarani, S., & Tamilarasi, A. (2011, June). An efficient masking technique for sensitive data protection. In the 2011 International Conference on Recent Trends in Information Technology (ICRTIT) (pp. 1245-1249). IEEE.
[13] Badgujar, P. (2021). Implementing data masking techniques for privacy protection. Journal of Technological Innovations, 2(4).
[14] Sebrechts, M., Borny, S., Wauters, T., Volckaert, B., & De Turck, F. (2021). Service relationship orchestration: Lessons learned from running large-scale smart city platforms on Kubernetes. IEEE Access, 9, 133387-133401.
[15] Ermolenko, D., Kilicheva, C., Muthanna, A., & Khakimov, A. (2021, January). Internet of Things Services Orchestration Framework based on Kubernetes and edge computing. In 2021, the IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus) (pp. 12-17). IEEE.
[16] Böhm, S., & Wirtz, G. (2021). Towards orchestration of cloud-edge architectures with Kubernetes. In International Summit Smart City 360° (pp. 207-230). Cham: Springer International Publishing.
[17] Ahmed, Z., & Francis, S. C. (2019, November). Integrating security with devsecops: Techniques and challenges. In the 2019 International Conference on Digitization (ICD) (pp. 178-182). IEEE.
[18] Jayaraman, P. P., Perera, C., Georgakopoulos, D., Dustdar, S., Thakker, D., & Ranjan, R. (2017). Analytics-as-a-service in a multi-cloud environment through semantically-enabled hierarchical data processing. Software: Practice and Experience, 47(8), 1139-1156.
[19] Dutta, S., Madnick, S., & Joyce, G. (2016, June). SecureUse: Balancing security and usability within system design. In International Conference on Human-Computer Interaction (pp. 471-475). Cham: Springer International Publishing.
[20] Duncan, G., & Stokes, L. (2009). Data masking for disclosure limitation. Wiley Interdisciplinary Reviews: Computational Statistics, 1(1), 83-92..
[21] Pappula, K. K., & Rusum, G. P. (2020). Custom CAD Plugin Architecture for Enforcing Industry-Specific Design Standards. *International Journal of AI, BigData, Computational and Management Studies*, *1*(4), 19-28. https://doi.org/10.63282/3050-9416.IJAIBDCMS-V1I4P103
[22] Rahul, N. (2020). Vehicle and Property Loss Assessment with AI: Automating Damage Estimations in Claims. *International Journal of Emerging Research in Engineering and Technology*, *1*(4), 38-46. https://doi.org/10.63282/3050-922X.IJERET-V1I4P105
[23] Pappula, K. K., & Rusum, G. P. (2021). Designing Developer-Centric Internal APIs for Rapid Full-Stack Development. *International Journal of AI, BigData, Computational and Management Studies*, *2*(4), 80-88. https://doi.org/10.63282/3050-9416.IJAIBDCMS-V2I4P108
[24] Pedda Muntala, P. S. R. (2021). Integrating AI with Oracle Fusion ERP for Autonomous Financial Close. *International Journal of AI, BigData, Computational and Management Studies*, *2*(2), 76-86. https://doi.org/10.63282/3050-9416.IJAIBDCMS-V2I2P109
[25] Rahul, N. (2021). AI-Enhanced API Integrations: Advancing Guidewire Ecosystems with Real-Time Data. *International Journal of Emerging Research in Engineering and Technology*, *2*(1), 57-66. https://doi.org/10.63282/3050-922X.IJERET-V2I1P107