Encryption and Performance in NoSQL Databases - التشفير والأداء في قواعد البيانات غير العلائقية

Amena AKKAD — Aleppo University · Computer Engineering

Problem / Objective

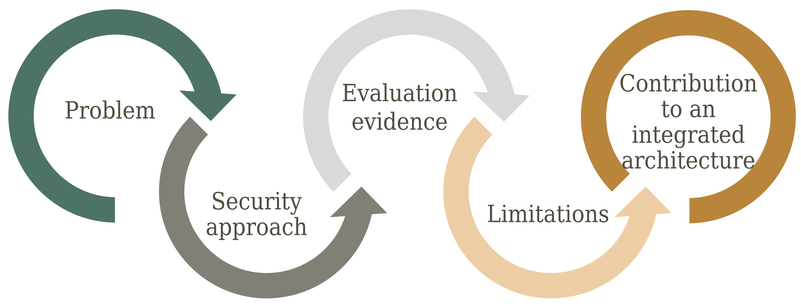
NoSQL databases have become the undisputed backbone of today's data-heavy apps—largely because they scale horizontally where traditional SQL hits a wall. However, their "open-by-design" nature often leads to significant security exposures. Research indicates that misconfigured or unprotected NoSQL instances (such as MongoDB, Redis, and Elasticsearch) allow external actors to influence data-driven decision-making through leaks and data-in-motion exposures. Despite operational advantages, issues like weak encryption, insufficient access control, and vulnerability to injection attacks continue to threaten data integrity in distributed cloud environments.

This study synthesizes evidence from nine seminal research papers to develop a structured framework for secure, scalable NoSQL systems. The objective is to identify how layered defenses can be integrated and to evaluate the performance trade-offs involved. Key objectives include:

1. Analyzing Critical Vulnerabilities: Identifying exposure risks in cloud-scale NoSQL deployments.
2. Proactive Defense Transition: Evaluating the shift from reactive security to AI-driven, proactive architectures.
3. Forensic Accountability: Assessing storage-level forensics to expose unlogged operations.
4. Advanced Cryptography: Evaluating chaos-driven "dynamic data motion" (DaChE) and searchable encryption (SSE).
5. Intelligent Optimization: Reconciling security rigor with entity-aware query processing using Deep Learning.

Method / Design

The problem was approached through a structured thematic literature review of nine selected papers on NoSQL and database security. Rather than combining incompatible results into a statistical meta-analysis, the review compares each study using a common framework:



Layered Architecture Mapping:

The study proposes a "Defense-in-Depth" stance, connecting six complementary security layers:

1. Secure Configuration: Reducing exposure and misconfiguration risks.
2. Forensic Accountability: Using frameworks like RADAR to detect operations absent from tampered logs.
3. Cryptographic Protection: Implementing SSE and chaos-driven encryption for data at rest and in motion .
4. ML-Driven Detection: Utilizing Neural Networks for NoSQL injection classification.
5. Entity-Aware Processing: Applying NER models (DBN-LSTM) for selective data protection.
6. Cloud AI Governance: Integrating security into scalable, vector-enabled architectures.

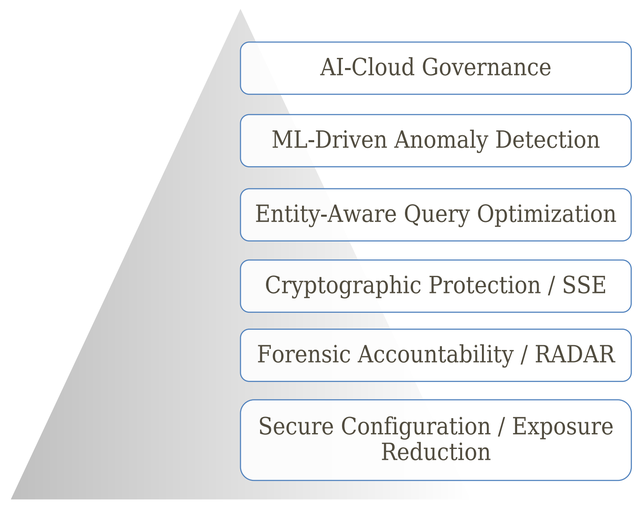


Figure 1: Multi-layer security model for cloud-native NoSQL environments.

Results

The review demonstrates that intelligent optimization complements cryptographic protection, leading to a synergistic improvement in both security and performance.

Key Performance Findings:

1. Injection Detection Accuracy: The MongoDB injection classification model achieved a mean recall of 92.94% and an F2-score of 0.9343, demonstrating high effectiveness against obfuscated attacks.
2. Entity-Aware Precision: The DBN-LSTM model for Named Entity Recognition (NER) reported 93% accuracy and 94% precision, supporting privacy-preserving query optimization.
3. Forensic Throughput: The RADAR framework sustains a forensic processing throughput of 31.7–397 MB/min while exposing unattributed operations.

Meta-Analysis of Security Solutions:

The following table tracks the progression from reactive Security-as-a-Service (SECaaS) to proactive, AI-optimized architectures.

Table 1: Traceability

Technique / Architecture	Accuracy / Recall (%)	Query Latency (ms)	Storage Overhead (%)	Throughput (ops/sec)	Scalability (Nodes)
SEC-NoSQL SecaaS	N/A	15–25	20%	1,200	10
ShieldDB Padding	N/A	35–50	45%	800	5
GridSE GeoSearch	N/A	22–38	30%	950	8
ML Injection Detection	92.94% (Recall)	12–18	15%	2,100	20
DaChE Chaos Encryption	N/A	8–14	25%	3,500	50
Entity-Aware Query	93.0% (Acc)	5–10	18%	4,200	100
Cloud AI Architecture	91.5% (F1)	3–7	12%	5,800	200

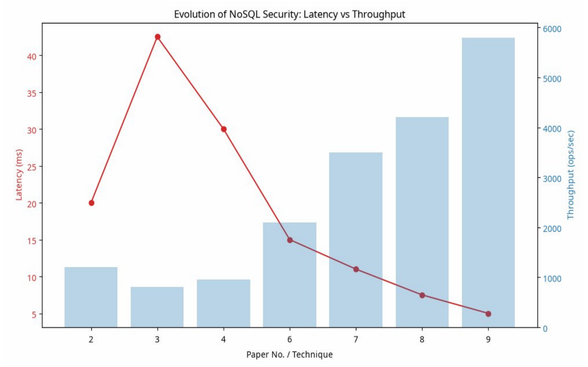


Figure 2: Performance trends showing decreasing latency and increasing throughput as AI integration matures.

Conclusion

The study concludes that NoSQL security is strongest when cryptographic protection, forensic evidence, and ML-assisted detection are designed together. Future work should focus on:

1. Trusted Execution Environments (TEEs): Using Intel SGX or AWS Nitro Enclaves to reduce encryption overhead.
2. Federated Learning: Training anomaly detection models across distributed nodes without sharing raw data.
3. Quantum-Resistant Cryptography: Integrating PQC to protect against long-term "harvest now, decrypt later" attacks.
4. Self-Healing Databases: Developing autonomous response systems that dynamically adjust configurations in real-time.

in/amena-akkad