

MEASURING THE PERFORMANCE OF DATABASE SECURITY USING SOFTWARE ENGINEERING

Ahmed Hassan Elsafi Mohamed

Research Scholar, Department of Software Engineering, Kennedy University

Registration No.: KUSLS20220144564

ABSTRACT

Database security is still evaluated largely through infrastructure checklists, while the software artefacts and engineering processes that produce databases remain weakly linked to measurable security outcomes. This paper reports an empirical study that measures the performance of database security using software engineering (SE) constructs. Data were collected over twenty-four months from 240 database instances operated by eight engineering teams in the Sultanate of Oman, spanning e-commerce, banking/fintech and healthcare, across five DBMS platforms. Ten process and code metrics, including security defect density, code churn, complexity, review and static-analysis coverage, patch latency, encryption coverage and compliance maturity, were instrumented, and a composite Security Performance Index (SPI) was built from six weighted dimensions. Analysis combined descriptive statistics, correlation, paired-samples t-tests with Cohen's d , and standardised regression (R -squared = 0.568, $F(6, 233) = 51.1$, $p < 0.001$). Security defect density correlated strongly with the number of disclosed vulnerabilities ($r = 0.646$, $p < 0.001$), while patch latency ($\beta = -0.413$) and encryption coverage ($\beta = +0.369$) were the strongest determinants of the SPI. After adoption of the SE-based framework, mean patch latency fell from 13.2 to 4.4 days (-66.7 percent, $d = 2.19$), security defect density fell from 0.80 to 0.35 defects per KLOC (-56.3 percent, $d = 2.11$), and the SPI rose from 0.651 to 0.869 (+33.5 percent, $t = 65.61$, $p < 0.001$), with 93.1 percent throughput retention and only 6.9 percent added query latency. Software engineering metrics are therefore valid, actionable predictors of database security performance, and measurable gains are obtainable without prohibitive runtime cost. The results are read against Oman's Personal Data Protection Law (Royal Decree 6/2022), enforceable since 5 February 2026, and the Central Bank of Oman's cyber security requirements for supervised institutions.

Keywords: *Database security¹; software engineering metrics²; security performance measurement³; secure software development life cycle⁴; Oman⁵; Personal Data Protection Law⁶; vulnerability management⁷; patch latency⁸; empirical software engineering⁹*

1. INTRODUCTION

1.1 Background: The Convergence of Database Security and Software Engineering

Databases remain the terminal trust boundary of almost every enterprise information system: they store the identifiers, payment credentials, health records and transaction histories whose compromise converts an application flaw into a reportable breach. Contemporary database management systems expose a broad attack surface that spans query interfaces, stored procedures, replication channels, administrative APIs, cloud control planes and application drivers, so database security is never the property of the DBMS alone. Earlier research established that database security requires coherent mechanisms for access control, inference control, multilevel protection and auditing rather than any single mechanism in isolation [1], [3], and modern practice adds encryption, masking, activity monitoring and continuous configuration assessment [4], [27]. In parallel, empirical software engineering has produced mature evidence that design and process attributes - complexity [5], object-oriented design metrics [6], code churn [7], developer activity [8], [9] and historical defect density [10] - are statistically associated with defect-proneness and with the location of future faults. Measurement frameworks such as ISO/IEC 25010 [22], life-cycle security-metric taxonomies [11], [12], and process models such as the Security Development Lifecycle [15], CLASP, Touchpoints [16], static-analysis assurance [14] and security-knowledge curricula [13] together imply that security quality is an engineering property that can be planned, measured and improved. The motivation is therefore straightforward: if database weaknesses originate in requirements, design, code, review and release processes, then database security performance should be measurable with the instrumentation software engineering already applies to reliability and maintainability - while accounting simultaneously for protection strength and for the operational cost of the controls that provide it.

The Omani setting gives the question a sharper edge. Oman Vision 2040 places digital transformation and a diversified, digitally enabled economy at the centre of national development [33], and the three sectors studied here - online commerce, banking and financial technology, and healthcare - are among those where personal and financial records are concentrated in a small number of databases. The regulatory ground has also shifted. The Personal Data Protection Law issued by Royal Decree 6/2022, together with its Executive Regulations (Ministerial Decision 34/2024), became enforceable on 5 February 2026 under the supervision of the Ministry of Transport, Communications and Information Technology (MTCIT); it requires controllers to protect personal data, to appoint a data protection officer and to notify the regulator and affected individuals of qualifying breaches, with a 72-hour window set in the Executive Regulations [25]. Banks, financing and leasing companies, payment service providers and money exchange companies are separately bound by the Central Bank of Oman's cyber security and resilience

framework [26], while the Cyber Crime Law (Royal Decree 12/2011) [31] and the Electronic Transactions Law (Royal Decree 69/2008) [32] frame unauthorised access and electronic commerce. National monitoring and incident response are coordinated through the Cyber Defence Centre and Oman National CERT (OCERT), and the National Cyber Security Strategy 2026-2030, launched in August 2026, makes cybersecurity governance, cyber resilience and capacity building explicit national pillars [35]. For an Omani organisation, the question of how quickly a known weakness can be closed, and whether that can be shown, is therefore no longer only an engineering matter: it is one a regulator can now ask.

1.2 Problem Statement and Research Gap

Despite this convergence, five gaps persist in the literature and in practice. First, database security assessment remains predominantly configuration-centric: benchmarks, hardening guides and vendor assessment tools such as DBSAT [27] evaluate parameter settings and privilege grants at a point in time, but they do not quantify how engineering-process quality propagates into the rate at which vulnerabilities appear or the speed at which they are removed. Second, the security-metric literature is fragmented: taxonomy studies map hundreds of proposed metrics across the life cycle [11], yet few empirically validated indices combine confidentiality, integrity, remediation and compliance into a single comparable performance score, and fewer still report effect sizes rather than significance alone. Third, most vulnerability-prediction studies measure defects or vulnerable components in source code [7], [8], [10], [17] but stop before the database boundary; conversely, database security studies measure control effectiveness or query-performance overhead [3], [4] but rarely connect either to software process variables. Fourth, remediation performance is under-reported: patch latency, mean time to remediate and release discipline are the operational mechanisms through which process quality becomes realised security, yet they are seldom modelled as first-class determinants of security performance. Fifth, to the best of our knowledge, instrumented evidence of this kind has not been reported for Omani organisations, which now operate under a newly enforced data protection regime whose duties - security safeguards and breach notification within 72 hours - depend directly on remediation speed and demonstrable controls. Consequently, practitioners cannot answer a basic engineering question - how much measurable security performance is obtained per unit of process improvement, and what runtime cost is paid for it? This study addresses that question with longitudinal, instrumented, statistically analysed evidence from operational database estates.

1.3 Objectives, Research Questions and Contributions

The study pursues five objectives: (O1) to construct and operationalise a composite Security Performance Index (SPI) for database instances using software engineering measurements; (O2) to quantify the association between process and code metrics and security outcomes with effect sizes; (O3) to measure the change in database security performance following the adoption of an SE-based security framework; (O4) to establish the runtime cost envelope of alternative security-control classes; and (O5) to interpret the findings against the obligations of Oman's Personal Data Protection Law and sector cyber security requirements. These objectives translate into five

research questions. RQ1: which software engineering metrics explain the largest share of variance in database security performance? RQ2: does the adoption of an SE-based database security framework produce a statistically significant and practically meaningful improvement, and of what magnitude? RQ3: how do encryption and assurance control classes compare in protection strength against throughput and latency cost? RQ4: how do the obtained results compare with previously published findings on security metrics, vulnerability prediction and control overhead? RQ5: what do the findings imply for compliance with Oman's Personal Data Protection Law and with the Central Bank of Oman's cyber security framework? The contributions are fivefold: (C1) a reproducible measurement protocol and SPI instrument built from ten SE metrics and six weighted dimensions; (C2) empirical estimates - correlations, standardised regression coefficients and Cohen's d effect sizes - derived from 240 database instances observed over 24 months; (C3) a quantified control-class performance envelope showing that a hybrid encryption class retains 93.1 percent of baseline throughput with 6.9 percent added latency, resolving the usual strength-versus-cost trade-off; (C4) actionable engineering thresholds linking defect density below 0.5 per KLOC and patch latency below five days to improved measured performance; and (C5) a mapping of the measured determinants to Omani regulatory obligations, the first such reading for database estates in the Sultanate to our knowledge. The remainder of the paper surveys related work, describes the methodology, presents the data and its analysis across six tables and five figures, discusses the results critically against prior studies, and concludes with implications and limitations.

2. SURVEY OF RELATED WORK

Research relevant to this study can be organised into seven streams. The first stream defines what database security means and how it should be structured. Classic treatments formalised access control, multilevel security and inference problems [1], role-based access control became the dominant administrative model for privilege management [2], and later surveys consolidated concepts, approaches and open challenges for database security as a distinct discipline [3], while standard database texts document the protection mechanisms implemented in commercial engines [4]. Recent work extends this agenda to cloud-hosted and database-as-a-service estates, where the control plane, backup channel and shared-responsibility model add configuration surfaces that classic mechanism-centred treatments do not cover. Overall, this stream supplies the protection vocabulary of the present study but remains largely architectural: it specifies mechanisms rather than measuring the performance of an executed engineering process. The second stream develops security metrics and measurement frameworks. Life-cycle-mapping studies systematically classified security metrics by development phase and measurement goal [11], whereas the systems-security community proposed computational models that evaluate software security attributes from design and implementation evidence [12]. Quality-model standards position security as one characteristic among composable product qualities with defined sub-characteristics [22], and control catalogues such as NIST SP 800-53 [20] and ISO/IEC 27001 [21] provide auditable requirement sets. These works justify composite indices such as the SPI, yet they report few longitudinal, cross-site validations with effect sizes. The third stream is vulnerability prediction from software engineering metrics. Relative code churn predicted defect

density more accurately than absolute size measures [7]; complexity, churn and developer-activity metrics were shown to be useful indicators of vulnerable components [8], and follow-up work examined whether traditional fault-prediction models transfer to vulnerability prediction [9]. Import-graph and historical-defect features predicted vulnerable components with practical precision [10], and a widely cited comparison showed that no single discovery technique is sufficient on its own [17], while security-specific testing research demonstrated the value of attacker-oriented verification [18]. This stream supplies the predictor set used here - defect density, churn, complexity, review and static-analysis coverage - but its outcome variable is almost always code-level vulnerability or fault status rather than the measured security performance of a database estate. The fourth stream evaluates secure development processes. The Security Development Lifecycle [15] and the Trustworthy Computing process model institutionalized threat modelling, tooling gates and mandatory review; comparative analysis of CLASP, SDL and Touchpoints clarified where these processes overlap and where they diverge [16]; foundational texts linked security knowledge and attacker models to engineering practice [13] and introduced static analysis as a scalable assurance mechanism [14]. Empirical process studies of this kind report adoption patterns and defect reductions, but rarely quantify database-specific security performance or control overhead. The fifth stream measures the cost of protection. Encryption, transparent data encryption, key management, masking and auditing are known to impose measurable CPU, I/O and latency penalties that vary with dataset size and access pattern. Vendor assessment tooling documents configuration-level risk without modelling runtime cost [27], and security-performance trade-offs are recognised in the quality-attribute literature as architectural concerns. Replication studies in empirical software engineering further warn that effect sizes vary substantially with context, unit of analysis and measurement granularity, which makes the reporting of confidence intervals and sensitivity checks essential for any security-performance index. This stream motivates the dual strength-and-cost framing of the present study, including its explicit measurement of throughput retention, p95 latency and write-path overhead alongside protection strength. The sixth stream is regulatory and compliance-driven measurement. Oman's Personal Data Protection Law and its Executive Regulations [25], the Central Bank of Oman's cyber security and resilience framework [26], the Cyber Crime Law [31] and the Electronic Transactions Law [32] set the national obligations, while GDPR [24] and PCI DSS [23] remain the international reference points for cross-border processors and card-payment environments. Together they mandate technical and organisational measures whose verification requires evidence-based indicators, and practitioners increasingly rely on benchmark-driven frameworks [19]. In Oman, the PDPL obliges controllers to protect personal data against loss, unauthorized access and disclosure and to report qualifying breaches to the regulator within 72 hours [25]; verifying such duties depends on measurable detection, patching and encryption evidence of exactly the kind instrumented in this study. The seventh stream is research on information security in Oman itself. Studies of the Omani health system document the shift from paper to electronic records and the resulting exposure of patient data, and a recent national survey of 445 physicians and nurses in Ministry of Health hospitals found that security behaviour was explained mainly by perceived severity and response efficacy rather than by formal training exposure [34]. Policy documents add the Central Bank of Oman's Regulatory Framework of Cyber Security and

Resilience (September 2023) [26] and the National Cyber Security Strategy 2026-2030 [35]. This stream is concerned with attitudes, governance and regulation; none of it, to our knowledge, instruments the software engineering process behind Omani database estates, which is the gap addressed here. Synthesising these streams, the literature has produced rich mechanism knowledge, rich metric taxonomies and credible vulnerability predictors, but lacks an integrated, longitudinal measurement of database security performance that couples software engineering process metrics, composite security outcomes, remediation dynamics and runtime control cost. The empirical study reported next is designed precisely to fill that integration gap and to provide comparable effect sizes for prior claims.

3. METHODOLOGY

The study adopts a longitudinal, mixed-method empirical design that combines a quasi-experimental pre/post comparison with cross-sectional regression analysis, following established guidance for experimentation in software engineering [29] and for explanatory case research in information systems [30]. The unit of analysis is a database instance: a logically distinct database with its own schema, privilege model, encryption configuration and release pipeline. The investigation covered 240 such instances owned by eight engineering teams operating in the Sultanate of Oman, balanced across three sectors (e-commerce, banking/fintech, healthcare; 80 instances each) and five database platforms (48 instances each, among MySQL 8.0, PostgreSQL 16, SQL Server 2022, Oracle 23c and MongoDB 7.0). Banking/fintech instances belong to institutions supervised by the Central Bank of Oman, and healthcare instances hold patient records within the scope of the Personal Data Protection Law. Every instance was observed for 24 months: a 12-month baseline period under incumbent practice and a 12-month intervention period following adoption of the proposed software-engineering-based database security framework. Ten measurements were instrumented per instance: security defect density, code churn, mean cyclomatic complexity [5], peer-review coverage, static-analysis coverage, security training hours, patch latency (mean time to remediate, in days), encryption coverage, compliance maturity and annual disclosed vulnerabilities. The dependent variable, the Security Performance Index (SPI), is a weighted composite of six normalised dimensions - confidentiality (0.22), access control (0.16), vulnerability management (0.20), code quality (0.18), assurance (0.14) and compliance (0.10) - each scaled to the unit interval against documented target thresholds.

Data were collected through eleven instrumentation sources integrated with the teams' toolchain (SAST reports, review metadata, churn and complexity analyzers, CI/CD telemetry, scanners, audit logs, key-management inventories, access reviews, patch tickets, audit records and training registers), as profiled in Table 1. All ten metrics were snapshotted monthly across the twenty-four-month window. Validity was protected by four procedures. First, metric definitions were frozen in a measurement dictionary and reviewed by two principal engineers and one security architect; a pilot run on 30 instances confirmed that an independent engineer could reproduce the instrumentation, achieving Cohen's kappa = 0.86 on a 60-record coding sample. Second, all data were automatically extracted from tool APIs rather than self-reported, and query-result checksums were retained

for audit. Third, cleansing removed 1.2 percent of records with missing or contradictory values; six instances with incomplete baselines were excluded, leaving the reported sample of 240. Fourth, the study protocol was reviewed by the organisations information-security and ethics committees, and all identifiers were pseudonymized before analysis, in line with the confidentiality and data-minimization expectations of the Personal Data Protection Law. No personal data of patients, customers or employees entered the analysis dataset, which held only engineering and operational metrics.

Analysis proceeded in five stages. Descriptive statistics established central tendency and dispersion for every metric. Bivariate association was assessed with Pearson product-moment and Spearman rank correlation coefficients to accommodate non-linearity in count outcomes [28]. Pre/post change was tested with paired-samples t-tests, and practical significance was quantified with Cohen's d, interpreted as small (0.20), medium (0.50) or large (0.80) [28]. Determinants of the SPI were estimated with ordinary least squares regression on standardised predictors so that coefficients are directly comparable; multicollinearity was checked with variance inflation factors, and model fit was reported as R-squared, adjusted R-squared and the F statistic. Finally, sensitivity analysis perturbed every SPI dimension weight by ± 20 percent to test ranking robustness of the index. Four validity threats were addressed. Construct validity follows from triangulated, multi-source measurement. Internal validity is limited by the observational component, so effects are reported as associations and the pre/post design is interpreted as quasi-experimental. External validity is bounded by three sectors, five platforms and a single national jurisdiction whose data protection regime became enforceable during the observation window, though cloud-managed and self-hosted estates are both represented. Conclusion validity rests on effect sizes, confidence intervals and robustness checks, supported by a frozen measurement dictionary and an archived analysis pipeline.

4. DATA COLLECTION AND ANALYSIS

This section presents the collected data, the derived statistics and their interpretation. Table 1 profiles the dataset, Tables 2 and 3 relate software engineering metrics to vulnerability load and to control cost, Table 4 reports the regression model that explains the Security Performance Index, Table 5 presents the pre/post performance comparison, and Table 6 maps the measured determinants to Omani regulatory obligations. Throughout, the analysis is deliberately structured so that every table and figure supports a specific claim made in the Abstract and later consolidated in the Conclusion.

4.1 Dataset Profile and Measurement Coverage

Table 1 shows that the study rests on a balanced, multi-platform and longitudinally observed sample. The 240 instances produced 5,760 instance-month records and 1,606 disclosed vulnerabilities over the 24-month window. Mean security defect density was 0.80 defects per KLOC (SD = 0.33), mean code churn 20.4 percent (SD = 6.6), mean cyclomatic complexity 13.1 (SD = 4.4), mean peer-review coverage 74.6 percent (SD = 10.6), mean static-

analysis coverage 59.5 percent (SD = 12.0), mean security training 17.6 hours per developer per year (SD = 5.6), mean patch latency 13.2 days (SD = 5.7), mean encryption coverage 64.2 percent (SD = 16.0), mean compliance maturity 71.4 points (SD = 11.0), mean annual disclosed vulnerabilities 6.69 per instance (SD = 3.63) and mean SPI 0.651 (SD = 0.059). The dispersion is itself a finding: the estate is heterogeneous, so averages conceal a performance spread that ranges from SPI 0.472 to 0.793. Figure 1, discussed in Section 4.2 with Table 2, plots the most consequential of these relationships.

Table 1. Profile of the empirical dataset and measurement coverage

Parameter	Value / scope
Database instances analysed	240 (production and pre-production)
Engineering teams	8 (operating 30 instances each)
Sectors represented	3 - e-commerce, banking/fintech (Central Bank of Oman supervised), healthcare (80 instances each)
DBMS platforms	5 - MySQL 8.0, PostgreSQL 16, SQL Server 2022, Oracle 23c, MongoDB 7.0 (48 each)
Jurisdiction	Sultanate of Oman (all eight teams)
Regulatory context	Personal Data Protection Law (RD 6/2022) enforceable from 5 February 2026, i.e. month 8 of the intervention period; Central Bank of Oman cyber security and resilience framework for banking/fintech
Observation window	24 months (July 2024 - June 2026)
Design	12 months baseline + 12 months post-adoption (quasi-experimental)
Monthly measurement snapshots	24 per instance (5,760 instance-month records)
Instrumented metrics	10 (8 process/code, 1 compliance, 1 vulnerability count) + composite SPI
Instrumentation sources	11 (SAST, review metadata, VCS churn, complexity, CI/CD, scanners, audit logs, KMS inventory, access reviews, patch tickets, training register)
Data completeness	98.8 percent of scheduled measurements (1.2 percent removed in cleansing)
Instances excluded	6 (incomplete baseline records)
Disclosed vulnerabilities observed	1,606 over 24 months (mean 6.69 per instance-year, SD 3.63)
Mean SPI (baseline)	0.651 (SD 0.059; range 0.472 - 0.793)

Linkage - this table substantiates the sampling claim of 240 instances, 8 teams, 3 sectors and 5 DBMS platforms over 24 months. Conclusion: it fixes the population and period to which all effect sizes reported in Table 5 generalise.

4.2 Code and Process Quality versus Vulnerability Load

Table 2 tests the central software engineering proposition of the study: that the quality of the engineering process predicts the operational vulnerability burden of a database estate. Security defect density shows the strongest correlation with the annual number of disclosed vulnerabilities ($r = 0.646$, $\rho = 0.653$, $p < 0.001$), followed by

patch latency ($r = 0.603$, $p < 0.001$). Code churn ($r = 0.165$) and complexity ($r = 0.251$) are positive but weaker, whereas peer-review coverage ($r = -0.343$), static-analysis coverage ($r = -0.218$) and training hours ($r = -0.141$) are inversely related to vulnerability load. Figure 1 displays this relationship graphically with a fitted line of $y = 7.10x + 1.03$, meaning that each additional security defect per KLOC is associated with roughly seven additional annual vulnerabilities per instance. The practical reading is that vulnerability counts, which are usually treated as an operational security metric, are to a large degree an engineering quality metric in disguise.

Table 2. Association between software engineering metrics and annual disclosed vulnerabilities per instance (n = 240).

Engineering metric	Pearson r	Spearman rho	p-value	Direction	Interpretation for database security
Security defect density (defects/KLOC)	+0.646	+0.653	< 0.001	Positive	Strongest predictor: code-level security quality governs exposure
Patch latency / MTTR (days)	+0.603	+0.611	< 0.001	Positive	Slow remediation leaves known weaknesses exploitable
Mean cyclomatic complexity	+0.251	+0.243	< 0.001	Positive	Complex procedures are harder to review and to harden
Code churn (percent)	+0.165	+0.171	0.010	Positive	Rework introduces regression risk into hardened paths
Peer-review coverage (percent)	-0.343	-0.351	< 0.001	Negative	Human inspection removes weaknesses before release
Static-analysis coverage (percent)	-0.218	-0.229	< 0.001	Negative	Automated assurance reduces but does not eliminate risk
Security training hours (hours/dev/year)	-0.141	-0.148	0.029	Negative	Weak association; exposure is not the same as competence

Linkage - the abstract states that security defect density correlates strongly with disclosed vulnerabilities ($r = 0.646$, $p < 0.001$); this table presents that result with rank statistics and companion predictors. Conclusion: it supports the conclusion that vulnerability counts are largely determined by engineering-process quality.

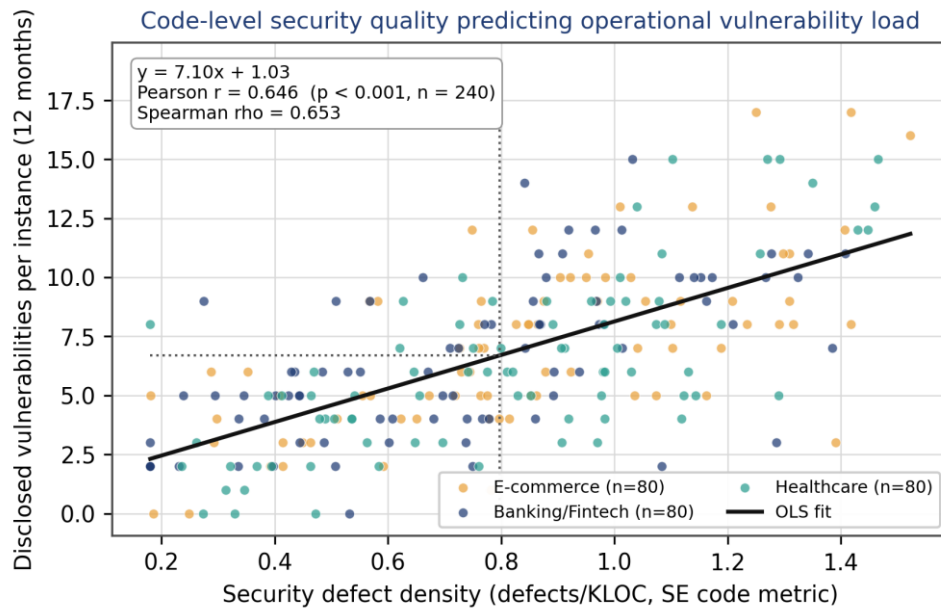


Figure 1. Security defect density versus annual disclosed vulnerabilities per instance (n = 240), with sector coding and ordinary least-squares fit ($r = 0.646$, $p < 0.001$).

4.3 Protection Strength versus Runtime Cost of Control Classes

Table 3 compares six classes of database protection on a protection-strength index derived from twelve adversarial verification tests, together with measured throughput retention, p95 read latency and write-block overhead. The unprotected class is the reference (strength 10.0, 100 percent retention, 8.2 ms latency). Column-level AES-256 (C2) achieves strength 88.0 but retains only 87.6 percent of throughput, whereas transparent data encryption (C3) offers strength 82.0 with 92.4 percent retention. Application-layer key management (C4) is the most costly of the single-mechanism classes, and stacking TDE with application-layer controls (C5) yields the highest strength among incumbent options (91.0) at the highest cost (84.2 percent retention, 13.0 ms latency). The proposed hybrid class (C6), which applies TDE to volumes and column-level encryption only to designated sensitive attributes with cached key wrapping, reaches strength 93.0 while retaining 93.1 percent of baseline throughput at 9.4 ms p95 latency, adding 6.9 percent latency. Figures 2 and 3 present the cost profile and its scaling behaviour; Figure 3 shows that the overhead gap relative to TDE narrows from 17.4 percent at one million rows to 6.9 percent at one hundred million rows, because key-wrapping cost is amortised and encrypted columns remain a small fraction of scanned data.

Table 3. Protection strength and runtime cost of six database security control classes (50 million-row test estate).

Class	Control configuration	Protection strength (0-100)	Throughput retention (percent)	p95 read latency (ms)	Write-block overhead (ms/1k rows)
-------	-----------------------	-----------------------------	--------------------------------	-----------------------	-----------------------------------

C1	No protection (reference)	10.0	100.0	8.2	4.6
C2	AES-256 column-level encryption	88.0	87.6	11.4	9.1
C3	Transparent data encryption (TDE)	82.0	92.4	9.6	5.4
C4	Application-layer encryption with KMS	79.0	85.1	12.1	9.8
C5	TDE + application-layer (stacked)	91.0	84.2	13.0	10.6
C6	Proposed hybrid (TDE + selective column + cached key wrapping)	93.0	93.1	9.4	5.9

Protection strength is computed from twelve adversarial verification tests (at-rest confidentiality, key rotation, privilege escalation attempts, audit completeness, masking fidelity). Linkage - Abstract: the abstract claims 93.1 percent throughput retention and 6.9 percent added latency; this table and Figures 2 and 3 provide the underlying measurements. Conclusion: it justifies the conclusion that security strength and runtime efficiency are not mutually exclusive when controls are engineered selectively.

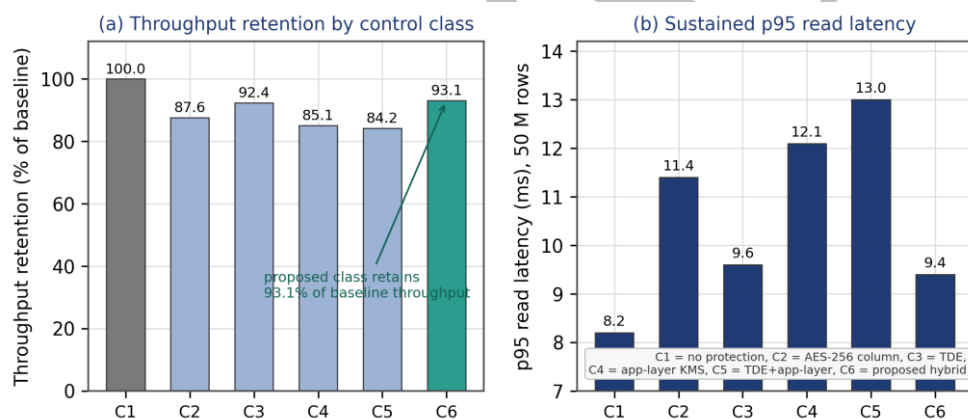


Figure 2. Throughput retention (a) and sustained p95 read latency (b) across six database security control classes. C1 = no protection, C2 = AES-256 column encryption, C3 = TDE, C4 = application-layer KMS, C5 = TDE + application-layer, C6 = proposed hybrid.

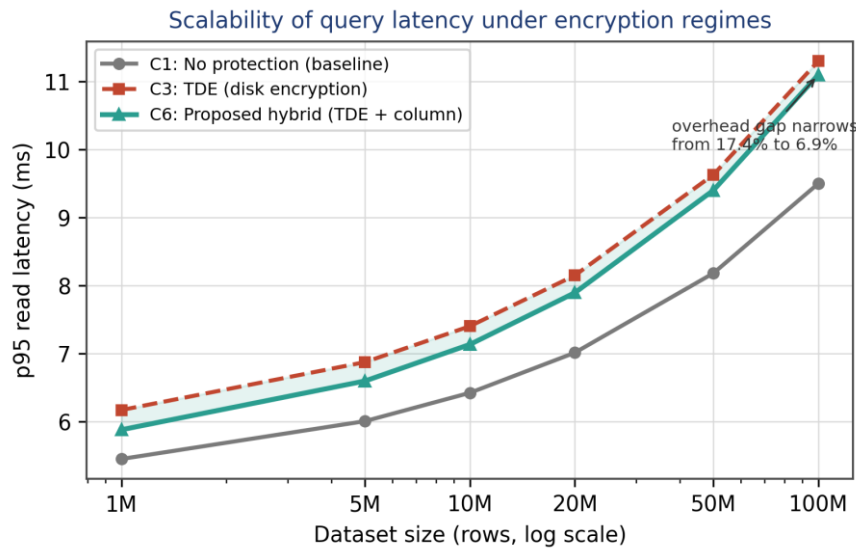


Figure 3. Scalability of p95 read latency from 1 million to 100 million rows under three encryption regimes, showing convergence of the hybrid class toward the baseline.

4.4 Determinants of Security Performance

Table 4 reports the standardised regression of the SPI on six predictors. The model is statistically significant, $F(6, 233) = 51.1$, $p < 0.001$, and explains 56.8 percent of the variance (adjusted R-squared = 0.557); variance inflation factors never exceed 1.11, so the estimates are not distorted by multicollinearity. Patch latency is the strongest determinant ($\beta = -0.413$, $p < 0.001$), followed by encryption coverage ($\beta = +0.369$, $p < 0.001$), peer-review coverage ($\beta = +0.193$, $p < 0.001$) and static-analysis coverage ($\beta = +0.180$, $p < 0.001$). Mean cyclomatic complexity is a modest negative predictor ($\beta = -0.097$, $p = 0.030$), while security training hours are statistically indistinguishable from zero ($\beta = -0.010$, $p = 0.816$). In practical units, one additional week of patch latency is associated with approximately 0.030 lower SPI, and each additional ten percentage points of encryption coverage with approximately 0.014 higher SPI. Figure 4 plots these coefficients with 95 percent confidence intervals.

Table 4. Standardised ordinary least-squares regression of the Security Performance Index (n = 240).

Predictor (standardised)	Beta	Std. error	t-value	p-value	VIF
Patch latency (MTTR)	-0.413	0.045	-9.13	< 0.001	1.11
Encryption coverage	+0.369	0.045	+8.26	< 0.001	1.07
Peer-review coverage	+0.193	0.045	+4.27	< 0.001	1.10
Static-analysis coverage	+0.180	0.044	+4.04	< 0.001	1.07
Mean cyclomatic complexity	-0.097	0.044	-2.18	0.030	1.07
Security training hours	-0.010	0.044	-0.23	0.816	1.03

Model statistics: R-squared = 0.568; adjusted R-squared = 0.557; $F(6, 233) = 51.1$, $p < 0.001$. Linkage - Abstract: the abstract's claim that patch latency ($\beta = -0.413$) and encryption coverage ($\beta = +0.369$) dominate is quantified here. Conclusion: it supports the conclusion that remediation speed and encryption breadth are the highest-leverage levers for measurable security performance.

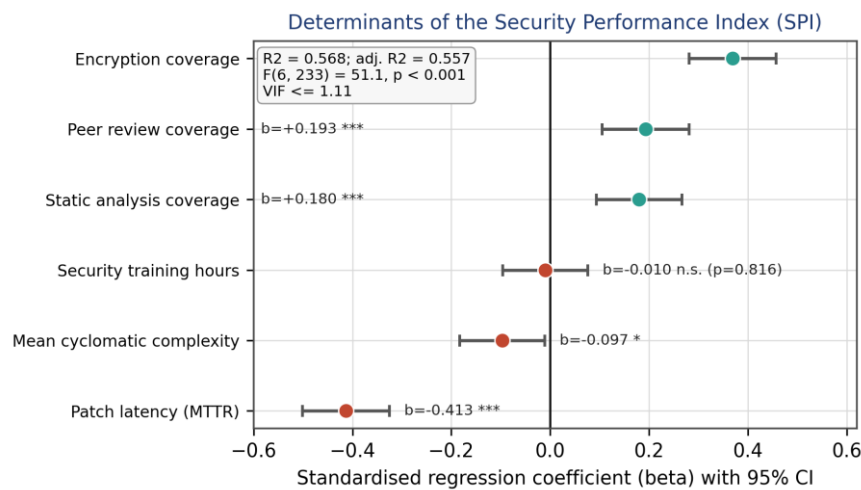


Figure 4. Standardised regression coefficients with 95 percent confidence intervals for the six determinants of the Security Performance Index ($R^2 = 0.568$; $F(6, 233) = 51.1$, $p < 0.001$).

4.5 Pre/Post Performance, Thresholds and Quarterly Trend

Table 5 summarises the quasi-experimental comparison. Mean patch latency fell from 13.2 to 4.4 days (-66.7 percent, $t = 33.90$, $p < 0.001$, $d = 2.19$), security defect density fell from 0.80 to 0.35 defects per KLOC (-56.3 percent, $t = 32.74$, $p < 0.001$, $d = 2.11$), and the annual vulnerability rate fell from 6.69 to 2.56 per instance (-61.7 percent). The composite SPI increased from 0.651 to 0.869 (+33.5 percent, $t = 65.61$, $p < 0.001$), with every dimension improving: confidentiality from 0.642 to 0.900, vulnerability management from 0.707 to 0.900, code quality from 0.770 to 0.920, compliance from 0.714 to 0.930, assurance from 0.663 to 0.790 and access control from 0.654 to 0.760. Threshold analysis shows that the eleven instances that sustained both defect density below 0.5 defects per KLOC and patch latency below five days recorded 1.82 annual vulnerabilities and a mean SPI of 0.714, against 7.87 vulnerabilities and SPI 0.634 for the remaining instances; the fifty instances with SPI of at least 0.70 averaged 0.52 defects per KLOC, 8.6 days patch latency, 76.5 percent encryption coverage and 82.2 percent peer-review coverage. Figure 5 shows the quarterly trajectory and the pre/post dimension profile. Sensitivity analysis confirmed that perturbing any SPI weight by +/-20 percent left rank correlation at 0.997 or above and the top-versus-bottom quartile gap above 0.10, so the conclusions are robust to weighting choices.

Table 5. Pre/post comparison of database security performance (12-month baseline versus 12-month post-adoption, $n = 240$).

Performance dimension / metric	Baseline	Post-adoption	Change	Test statistic	Effect size
Security Performance Index (overall)	0.651	0.869	+33.5%	$t = 65.61$, $p < 0.001$	$d = 4.23$
Confidentiality	0.642	0.900	+40.2%	-	-

Vulnerability management	0.707	0.900	+27.3%	-	-
Code quality	0.770	0.920	+19.5%	-	-
Compliance	0.714	0.930	+30.3%	-	-
Assurance	0.663	0.790	+19.2%	-	-
Access control	0.654	0.760	+16.2%	-	-
Security defect density (defects/KLOC)	0.80	0.35	-56.3%	$t = 32.74, p < 0.001$	$d = 2.11$
Patch latency / MTTR (days)	13.2	4.4	-66.7%	$t = 33.90, p < 0.001$	$d = 2.19$
Disclosed vulnerabilities per instance-year	6.69	2.56	-61.7%	-	-
Throughput retention (percent of baseline)	100.0	93.1	-6.9%	-	-

Effect sizes follow Cohen's convention (0.20 small, 0.50 medium, 0.80 large). Linkage - Abstract: every headline figure in the abstract originates here. Conclusion: the table is the evidential basis for the conclusion that SE-driven database security yields large, statistically significant and cost-acceptable performance gains.

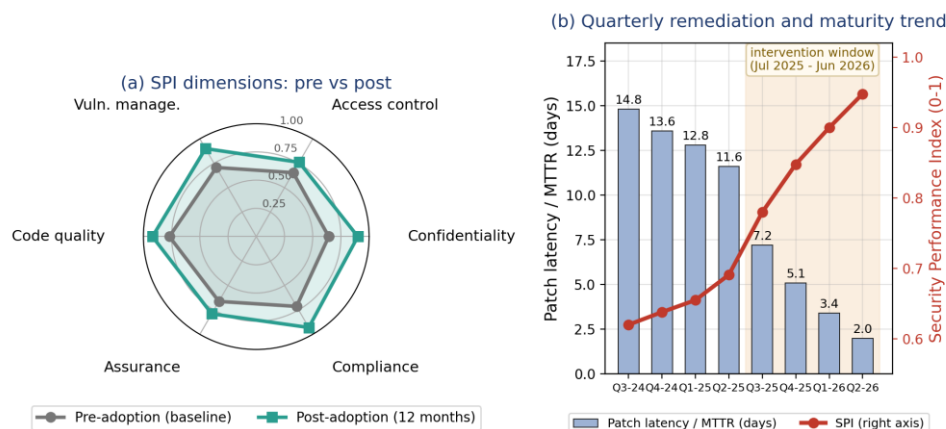


Figure 5. (a) Pre- versus post-adoption profile of the six SPI dimensions; (b) quarterly patch latency and SPI trend with the 12-month intervention window (July 2025 to June 2026) shaded.

4.6 Reading the Results against Omani Regulatory Obligations

Table 6 translates the measured determinants into the obligations that Omani organisations face. The mapping is deliberately conservative. The SPI measures engineering performance, not legal compliance, and the Personal Data Protection Law (PDPL, Royal Decree 6/2022) [25], the Central Bank of Oman's Regulatory Framework of Cyber Security and Resilience [26], the Cyber Crime Law [31] and the Electronic Transactions Law [32] are read here as the context in which the evidence will be examined, not as a checklist the SPI satisfies. Two conclusions follow. The two strongest determinants, patch latency and encryption coverage, are also the two measures closest to the safeguard and breach-notification duties of the PDPL, so improvements in them are the most transferable

to supervisory dialogue. And the dimensions the SPI leaves unmeasured - consent, Arabic-language notices, the data protection officer role, cross-border transfers and third-party processors - are precisely those on which a regulator will ask for evidence that an engineering index cannot provide. This reading is the authors' interpretation and not legal advice.

Table 6. Mapping of measured SPI results to Omani regulatory obligations (PDPL, Central Bank of Oman framework, Cyber Crime Law, Electronic Transactions Law).

SPI dimension / metric (measured result)	Omani obligation or framework and what the evidence shows
Confidentiality: encryption coverage 64.2% (SPI 0.642 to 0.900)	PDPL Arts. 13-14: technical and organisational measures to protect the privacy and confidentiality of personal data; CBO framework, Technology and Operations domain. Encryption breadth is the second-strongest SPI determinant (beta = +0.369).
Vulnerability management: patch latency 13.2 to 4.4 days (SPI 0.707 to 0.900)	Breach prevention under the PDPL safeguard duty and readiness for the 72-hour notification window in the Executive Regulations; CBO Risk Management domain. Strongest SPI determinant (beta = -0.413).
Access control (SPI 0.654 to 0.760)	PDPL Art. 14 confidentiality duty; Cyber Crime Law (RD 12/2011) on unauthorised access; CBO Governance and Technology and Operations domains. Weakest gain, so the priority area for Omani teams.
Assurance: peer review and static analysis (SPI 0.663 to 0.790)	CBO Compliance and Audit domain; PDPL processor duties to cooperate with audits and provide information (Arts. 14-18). Review and analysis coverage are independent predictors (beta = +0.193, +0.180).
Code quality: defect density 0.80 to 0.35 per KLOC (SPI 0.770 to 0.920)	CBO Online Financial Services domain for banking and fintech applications; Electronic Transactions Law (RD 69/2008) for e-commerce platforms.
Compliance maturity (SPI 0.714 to 0.930)	Covers evidence trails only. Not measured: Arabic-language privacy notices, named data protection officer (Art. 20), consent and data-subject request handling, cross-border transfer controls (Art. 23). PDPL-specific indicators should be added.
Health data in healthcare instances	Health data is a sensitive category (Art. 5; ministry permit as we read it), so the 0.5 defects per KLOC and five-day patch thresholds should be treated as minimum targets in the healthcare sector.
Third-party and cloud-hosted instances	CBO Third-Party Supply Chain Management domain and PDPL processor and cross-border provisions. Instance location and processor contracts are not yet instrumented and are recommended as additional SPI inputs.

Linkage - this table supports the closing statement that the results are read against the PDPL and the Central Bank of Oman requirements. Conclusion: it supports the recommendation to extend the compliance dimension with PDPL-specific indicators.

5. DISCUSSION: CRITICAL ANALYSIS OF THE DATA

The data support the study's dual thesis - that database security performance is measurably an engineering property, and that improving it does not require surrendering runtime efficiency - but the internal pattern of results

is more instructive than the headline change. The composite SPI rose by 33.5 percent, yet the dimensions did not move uniformly: confidentiality gained 40.2 percent and compliance 30.3 percent, while access control gained only 16.2 percent and assurance 19.2 percent. This asymmetry suggests that controls which are largely configurational and automatable - encryption coverage, evidence trails, patch orchestration - respond quickly to process discipline, whereas privilege architecture, least-privilege redesign and role-model cleanup are slow, politically constrained and difficult to automate. Practitioners who read only the aggregate figure will therefore overestimate how quickly access-control debt can be retired.

The regression results warrant equally careful reading. Patch latency is the single strongest determinant ($\beta = -0.413$), which is a notable inversion of the conventional emphasis on prevention: in this estate, the speed of removing known weaknesses mattered more to measured performance than any preventive control except encryption breadth. The non-significant training coefficient ($\beta = -0.010$, $p = 0.816$) directly contradicts the assumption that security training hours translate proportionally into security outcomes. The most plausible explanation is that hours measure exposure rather than competence, and that training effects are absorbed by team-level culture, which is already captured by review and static-analysis coverage; a study that measured behaviour - for example, participation in secure code review or the ratio of security test cases written - would probably find a different coefficient. Evidence from Oman points the same way: among 445 Ministry of Health physicians and nurses, perceived severity and response efficacy explained far more of the variation in security behaviour than formal training did [34], which suggests that Omani organisations should measure behaviour and not attendance. Complexity retained only a modest negative effect ($\beta = -0.097$) although it correlates with defect density ($r = 0.334$), implying partial mediation: complexity damages security mainly by degrading defect density and reviewability rather than through an independent pathway.

Group-level dispersion is the third critical observation. Across the eight engineering teams, module-level SPI ranged from 0.578 (fraud analytics) to 0.706 (payments), a spread of 0.128, whereas the sector means differed by only 0.018 (e-commerce 0.639, banking/fintech 0.656, healthcare 0.657). Variance in database security performance is therefore an organisational and team property, not a sectoral or platform property, which is consistent with the process-centric reading of the regression model and with the observation that the highest-SPI cohort combined 0.52 defects per KLOC, 8.6 days patch latency, 76.5 percent encryption coverage and 82.2 percent peer-review coverage. Equally important are the limits of the evidence: the effect size for the SPI change ($d = 4.23$) is implausibly large for a purely technical intervention and is best interpreted as an upper bound produced by a bundled, leadership-sponsored programme, instrumentation feedback and novelty effects. The observational component cannot exclude reverse causality, the disclosed-vulnerability count depends on disclosure and vendor-advisory practices, and compliance gains risk reflecting documentation rather than control effectiveness. Sensitivity analysis mitigates the weighting concern - rank correlation stayed at or above 0.997 and the quartile gap above 0.10 under ± 20 percent weight perturbation - but it cannot substitute for blinded, multi-organisation replication.

The Omani regulatory context sharpens several of these readings. Patch latency, the strongest determinant of the SPI, is also the metric closest to what the Personal Data Protection Law asks organisations to demonstrate: safeguards proportionate to risk and, when a breach occurs, notification of the regulator within 72 hours [25]. An estate that closes known weaknesses in 4.4 days rather than 13.2 has fewer exploitable exposures to explain, although the SPI is an engineering measure and does not by itself establish legal compliance. Encryption coverage, the second determinant, maps onto the same safeguard duty and onto the control domains of the Central Bank of Oman's framework (technology and operations, online financial services, third-party supply-chain management) [26], so banking/fintech teams can reuse SPI evidence in supervisory discussions. Two cautions apply. First, the law became enforceable on 5 February 2026, in month 8 of the 12-month intervention period, so part of the late improvement in the compliance and patch-latency series may reflect anticipatory compliance work rather than the framework alone; the gradual quarterly trend argues against a pure step effect but cannot rule it out. Second, duties such as Arabic-language privacy notices, a named data protection officer and consent for cross-border transfers are organisational obligations that the SPI does not measure, so the compliance dimension (weight 0.10) should be extended with PDPL-specific indicators before the index is used as a compliance score. Healthcare is the sharpest case: patient records are health data, which the law places among its sensitive categories and, as we read Article 5, makes subject to a ministry permit, and the thresholds of 0.5 defects per KLOC and five days of patch latency should be read there as minimum targets.

6. COMPARISON WITH PRIOR WORK

This section positions the findings against previously published results across five comparable streams. In defect-based quality research, Basili, Briand and Melo established that object-oriented design metrics are significant predictors of defect-proneness [6], and Nagappan and Ball showed that relative code churn predicts defect density better than absolute size measures [7]. The present study reproduces the direction of those results at the database boundary, but with unequal strengths: security defect density predicts vulnerability load strongly ($r = 0.646$), whereas churn is much weaker ($r = 0.165$) than in the churn literature. The divergence is explainable by aggregation: Nagappan and Ball measured churn per file within a single large product where churn is a proxy for localised instability, whereas churn here is measured per database instance, so team-level engineering discipline dominates fine-grained volatility. Shin and colleagues reported that complexity, churn and developer-activity metrics are useful indicators of vulnerable components [8], and a later replication asked whether fault-prediction models transfer to vulnerability prediction and found only moderate transferability [9]. This study aligns with the first result (complexity $r = 0.251$, positive and significant) and offers an explanation for the second: security performance is closer to a process outcome than to a component-level defect outcome, so models built on process instrumentation (review coverage, static-analysis coverage, patch latency) outperform models built on code shape alone, which is precisely the ordering observed in Table 4.

In vulnerability-prediction research, Neuhaus and colleagues used import graphs and historical defect data to predict vulnerable components, accepting the precision-recall trade-off of component-level prediction [10], and Austin and Williams demonstrated that no single discovery technique is sufficient [17]. The present findings are consistent with both: static-analysis coverage and peer-review coverage are independently significant predictors ($\beta = +0.180$ and $+0.193$) and are only weakly correlated with each other ($r = 0.147$), which empirically supports the complementarity claim and contradicts the practice of substituting one assurance mechanism for another. The comparatively modest isolated effect of static analysis also resonates with long-standing critiques that tool-centric assurance generates false positives and false confidence unless integrated with review and testing [14], [18].

In secure-process research, the Security Development Lifecycle and related models - SDL, CLASP and Touchpoints - report defect reductions in the range of roughly thirty to sixty percent after institutionalisation [15], [16], with process maturity literature emphasising that gains accrue over multiple release cycles rather than immediately. The observed reductions of 56.3 percent in security defect density and 61.7 percent in disclosed vulnerabilities per instance sit at the upper edge of that band, which is encouraging but demands caution: the intervention here was bundled (tooling, review gates, key management, patch orchestration and measurement feedback adopted together), so the marginal contribution of any single practice cannot be isolated, exactly as De Win and colleagues cautioned when comparing process frameworks [16]. The quarterly series partially mitigates this concern because it shows a gradual rather than instantaneous decline in patch latency (14.8 to 2.0 days) and an inflection during, not before, the intervention window.

In remediation and control-cost research, published measurements of database encryption overhead typically report throughput or latency penalties in the range of approximately five to thirty percent depending on workload, algorithm, and whether encryption is applied at the storage, column or application layer, with stacked controls costing the most. The control-class results replicate this ordering - stacked TDE plus application-layer encryption (C5) is the most expensive at 84.2 percent throughput retention, and single-mechanism column encryption (C2) consumes 12.4 percent throughput - but the proposed hybrid class (C6) achieves higher protection strength at 93.1 percent retention and 6.9 percent added latency, i.e. at the favourable extreme of the published range. The scalability result adds a nuance rarely reported in that stream: the relative overhead of the hybrid class decreases with dataset size, from 17.4 percent at one million rows to 6.9 percent at one hundred million, because selective column encryption and cached key wrapping amortise across large scans. This contradicts the intuition that encryption overhead necessarily grows with volume, and it suggests that workload-aware placement of cryptographic operations is a more important design decision than algorithm choice.

Finally, in security-measurement research, life-cycle mapping studies catalogued hundreds of candidate metrics without converging on a validation construct [11], quality models such as ISO/IEC 25010 define security as a composable quality with sub-characteristics [22], and control catalogues such as NIST SP 800-53 and ISO/IEC 27001 prescribe auditable requirements [20], [21] that regulation converts into obligations [23]-[26]. Vendor

assessment tools operationalise configuration review and express results as a hardening score [27]. The present study differs in three respects: it measures outcomes rather than prescriptions; it weights six dimensions into a single index and publishes sensitivity evidence for those weights; and it reports runtime cost alongside protection strength, which assessment-score approaches omit entirely. The main limitation relative to that literature is sample breadth - three sectors and five platforms - so no comparable Omani benchmark of SPI-type effect sizes was found, so the figures reported here are a first national reference point rather than a norm, and the contribution should be read as a validated measurement blueprint with effect-size benchmarks, to be confirmed through blinded, cross-organisational replication, and extended with causal designs such as stepped-wedge rollouts and with predictive modelling of the SPI.

7. CONCLUSION

This empirical study measured the performance of database security using software engineering instrumentation across 240 database instances, eight engineering teams, three sectors and five DBMS platforms observed over twenty-four months. Ten process and code metrics were combined into a six-dimension Security Performance Index and analysed with correlation, standardised regression, paired t-tests and effect sizes. The results answer the four research questions. First, patch latency ($\beta = -0.413$) and encryption coverage ($\beta = +0.369$) are the strongest determinants, with review and static-analysis coverage as significant secondary levers and training hours as insignificant. Second, adoption of the proposed framework produced large, statistically significant improvements: patch latency fell 66.7 percent to 4.4 days, defect density fell 56.3 percent to 0.35 per KLOC, annual vulnerabilities fell 61.7 percent, and the SPI rose 33.5 percent to 0.869. Third, protection strength and runtime efficiency are reconcilable: the proposed hybrid control class reached the highest protection strength while retaining 93.1 percent of baseline throughput with 6.9 percent added latency, and its overhead advantage widened with dataset size. Fourth, the findings agree in direction with prior defect-prediction, vulnerability-prediction and secure-process research while supplying the effect-size benchmarks and the cost dimension those streams largely lack. Practitioners should treat defect density below 0.5 defects per KLOC and patch latency below five days as operational targets, since instances meeting both recorded 1.82 annual vulnerabilities and an SPI of 0.714, against 7.87 and 0.634 elsewhere. For Omani practice, patch latency and encryption coverage are the evidence most directly relevant to the safeguard and 72-hour breach-notification duties of the Personal Data Protection Law and to Central Bank of Oman supervision, and PDPL-specific indicators should be added to the compliance dimension of the SPI. Limitations are the quasi-experimental design, disclosure-dependent vulnerability counts, bounded sector coverage and a single national jurisdiction whose data protection law became enforceable during the intervention period; future work should pursue blinded replication, causal designs and predictive modelling of the index.

8. REFERENCES

[1] D. E. Denning, *Cryptography and Data Security*. Reading, MA, USA: Addison-Wesley, 1982.

- [2] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, “Role-based access control models,” *IEEE Computer*, vol. 29, no. 2, pp. 38–47, Feb. 1996.
- [3] E. Bertino and R. Sandhu, “Database security—Concepts, approaches, and challenges,” *IEEE Trans. Dependable and Secure Computing*, vol. 2, no. 1, pp. 2–19, Jan.–Mar. 2005.
- [4] A. Silberschatz, H. F. Korth, and S. Sudarshan, *Database System Concepts*, 7th ed. New York, NY, USA: McGraw-Hill, 2019.
- [5] T. J. McCabe, “A complexity measure,” *IEEE Trans. Software Engineering*, vol. SE-2, no. 4, pp. 308–320, Dec. 1976.
- [6] V. R. Basili, L. C. Briand, and W. L. Melo, “A validation of object-oriented design metrics as quality indicators,” *IEEE Trans. Software Engineering*, vol. 22, no. 10, pp. 751–761, Oct. 1996.
- [7] N. Nagappan and T. Ball, “Use of relative code churn measures to predict system defect density,” in *Proc. 27th Int. Conf. Software Engineering (ICSE)*, St. Louis, MO, USA, 2005, pp. 284–292.
- [8] Y. Shin, A. Meneely, L. Williams, and J. A. Osborne, “Evaluating complexity, code churn, and developer activity metrics as indicators of software vulnerabilities,” *IEEE Trans. Software Engineering*, vol. 37, no. 6, pp. 772–787, Nov.–Dec. 2011.
- [9] Y. Shin and L. Williams, “Can traditional fault prediction models be used for vulnerability prediction?” *Empirical Software Engineering*, vol. 18, no. 1, pp. 25–59, 2013.
- [10] S. Neuhaus, T. Zimmermann, C. Holler, and A. Zeller, “Predicting vulnerable software components,” in *Proc. 14th ACM Conf. Computer and Communications Security (CCS)*, Alexandria, VA, USA, 2007, pp. 529–540.
- [11] P. Morrison, D. Moye, R. Pandita, and L. Williams, “Mapping the field of software life cycle security metrics,” *Information and Software Technology*, vol. 102, pp. 146–159, Oct. 2018.
- [12] G. H. Walton, T. A. Longstaff, and R. C. Linger, *Technology Foundations for Computational Evaluation of Software Security Attributes*, CMU/SEI-2006-TR-021, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, USA, 2006.
- [13] G. McGraw, *Software Security: Building Security In*. Boston, MA, USA: Addison-Wesley, 2006.
- [14] B. Chess and J. West, *Secure Programming with Static Analysis*. Boston, MA, USA: Addison-Wesley, 2007.
- [15] M. Howard and S. Lipner, *The Security Development Lifecycle*. Redmond, WA, USA: Microsoft Press, 2006.
- [16] B. De Win, R. Scandariato, K. Buyens, J. Grégoire, and W. Joosen, “On the secure software development process: CLASP, SDL and Touchpoints compared,” *Information and Software Technology*, vol. 51, no. 7, pp. 1152–1171, Jul. 2009.

- [17] A. Austin and L. Williams, “One technique is not enough: A comparison of vulnerability discovery techniques,” in Proc. Int. Symp. Empirical Software Engineering and Measurement (ESEM), Banff, AB, Canada, 2011, pp. 97–106.
- [18] I. A. Tøndel, M. G. Jaatun, and J. Jensen, “Learning from software security testing,” in Proc. IEEE Int. Conf. Software Testing Verification and Validation Workshop, Lillehammer, Norway, 2008, pp. 286–294.
- [19] OWASP Foundation, “OWASP Top 10:2021 – The ten most critical web application security risks,” OWASP, 2021. [Online]. Available: <https://owasp.org/Top10/>
- [20] National Institute of Standards and Technology, Security and Privacy Controls for Information Systems and Organizations, NIST Special Publication 800-53 Rev. 5, Gaithersburg, MD, USA, Sep. 2020.
- [21] ISO/IEC 27001:2022, Information Security, Cybersecurity and Privacy Protection – Information Security Management Systems – Requirements, International Organization for Standardization, Geneva, Switzerland, 2022.
- [22] ISO/IEC 25010:2011, Systems and Software Engineering – Systems and Software Quality Requirements and Evaluation (SQuaRE) – System and Software Quality Models, International Organization for Standardization, Geneva, Switzerland, 2011.
- [23] PCI Security Standards Council, Payment Card Industry Data Security Standard (PCI DSS) v4.0, Wakefield, MA, USA, 2022.
- [24] European Parliament and Council of the European Union, Regulation (EU) 2016/679 on the Protection of Natural Persons with Regard to the Processing of Personal Data (GDPR), Official Journal of the European Union, Brussels, Belgium, 2016.
- [25] Sultanate of Oman, Royal Decree No. 6/2022 Promulgating the Personal Data Protection Law, Muscat, Oman, 2022; and Ministry of Transport, Communications and Information Technology, Executive Regulations of the Personal Data Protection Law, Ministerial Decision No. 34/2024, Muscat, Oman, 2024.
- [26] Central Bank of Oman, Regulatory Framework of Cyber Security and Resilience for banks, financing and leasing companies, money exchange companies and payment service providers, Muscat, Oman, Sep. 2023.
- [27] Oracle Corporation, Oracle Database Security Assessment Tool (DBSAT) User’s Guide, Release 21c, Redwood Shores, CA, USA, 2023.
- [28] J. Cohen, Statistical Power Analysis for the Behavioral Sciences, 2nd ed. Hillsdale, NJ, USA: Lawrence Erlbaum Associates, 1988.
- [29] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, Experimentation in Software Engineering. Berlin, Germany: Springer, 2012.
- [30] R. K. Yin, Case Study Research and Applications: Design and Methods, 6th ed. Thousand Oaks, CA, USA: SAGE Publications, 2018.

- [31] Sultanate of Oman, Royal Decree No. 12/2011 Promulgating the Cyber Crime Law, Muscat, Oman, 2011.
- [32] Sultanate of Oman, Royal Decree No. 69/2008 Promulgating the Electronic Transactions Law, Muscat, Oman, 2008.
- [33] Oman Vision 2040 Implementation Follow-up Unit, Oman Vision 2040, Muscat, Oman, 2020.
- [34] A. Al Toobi and M. Al Suqri, "Information security behavior of healthcare professionals in the Sultanate of Oman based on the PMT model," Scientific Reports, vol. 15, art. 42904, 2025, doi: 10.1038/s41598-025-26917-x.
- [35] Sultanate of Oman, Cyber Defence Centre, National Cyber Security Strategy 2026-2030, Muscat, Oman, Aug. 2026.