GSC Advanced Research and Reviews

eISSN: 2582-4597 CODEN (USA): GARRC2

Cross Ref DOI: 10.30574/gscarr

Journal homepage: https://gsconlinepress.com/journals/gscarr/

(RESEARCH ARTICLE)

Quantum-aware software engineering: An AI-driven framework for measuring cryptographic agility in enterprise systems

Ayobami Adebesin *

Department of Mathematics and Statistics, Georgia State University, Atlanta, Georgia, USA.

GSC Advanced Research and Reviews, 2025, 25(03), 374-384

Publication history: Received on 13 November 2025; revised on 27 December 2025; accepted on 31 December 2025

Article DOI: https://doi.org/10.30574/gscarr.2025.25.3.0379

Abstract

The timely development of quantum computing is a fundamental threat to modern cryptographic systems, which are instantiated by enterprise software. Secure communication, authentication, and data protection rely on classical public-key algorithmic techniques that are likely to become susceptible to large-scale quantum attacks, and there is an urgent shift to quantum-resistant as well as successively cryptographically agile architectures. This article suggests a new AIware software engineering model, Quantum-Aware, is an AI-based measurement system that aims to assess and improve cryptographic agility at the enterprise scale. The framework combines the software engineering, post-quantum cryptography, and artificial intelligence principles to determine how well an organization is prepared to respond to new quantum threats through crypto-mechanisms. The suggested model presents a multi-layered assessment model that examines software architecture, cryptographic dependencies, diversity of algorithms, key management practices and update mechanisms. Complexity of migrations is measured and hard-coded cryptographic primitives are automatically detected in machine learning techniques to predict the effect of an algorithm substitution on system performance and security assurance. The framework can produce a measurable, comparative value, a Cryptographic Agility Index (CAI), that represents a measure of how easy an enterprise system is to move between cryptographic standards when changes are necessary. Moreover, the framework assists in decision-making by providing advanced refactoring strategies and adaptive controls in accordance with the post-quantum requirements, regulatory provisions and the level of risk tolerance.

Keywords: Enterprise Systems; Quantum-Aware Software Engineering; Post-Quantum Cryptography; Artificial Intelligence; Security Metrics; Cryptographic Agility

1. Introduction

The speeding up pace of the progress of quantum computing has made quantum hazards a theory no longer but an engineering reality that can be predicted. The improvement of qubit stability, error correction, and hybrid quantum-classical Planing points to the fact that cryptanalytic capabilities strong enough to compromise widespread public-key schemes, such as those used in practice, are no longer reasonable in the medium-term timeframe [Shor, 1997; Bernstein & Lange, 2017]. Systemic and cumulative Enterprise software systems which significantly depend on cryptographic primitives like RSA, ECC, and Diffie Hellman to provide confidentiality, integrity, authentication and non-repudiation are therefore vulnerable to a structural security threat, which is both systemic and cumulative. In empirical studies of security surveys, a pattern has continuously emerged that more than 85 percent of very large enterprise applications implement cryptography in more than one layer, typically transportation, application logic, identity management and data storage, and frequently through nonhomogeneous libraries and legacy software. This endemic mire causes cryptographic transition to be a software engineering issue as well as a cryptographic issue, requiring demanding measurement, planning and automation over time instead of proceeds involving naive substitution of algorithms [Mosca, 2018].

* Corresponding author: Ayobami Adebesin.

Copyright © 2025 Author(s) retain the copyright of this article. This article is published under the terms of the Creative Commons Attribution License 4.0.

Cryptographic agility has become a vital attribute to reduce such long term risks, the ability of a system to quickly change, update, or redesign the cryptographic mechanisms, without highly redesigning or voiding operational functionality [NIST, 2024; Hasan et al., 2024]. Nevertheless, the current enterprise environments have a low visible agility because of the hard coded algorithm selections, strong binding between cryptography and business logic as well as fragmented key management practice. Industrial audits state that almost 6070 percent of cryptographic failures are not related to algorithmic vulnerability but are a result of hardened implementations that make it hard to migrate in a good time even though, cryptographic agility is loosely measured in terms that are often determined qualitatively and either by compliance checklists or via manual code review [Grover, 1996]. Lacking standard metrics and automated evaluation tools introduces a big heart of difference between strategic quantum-readiness policies and their actualisation in systems that are software-intensive. The development of artificial intelligence and data-based software analytics in the recent past presents a good avenue to bridge this gap. Large-scale codebase-trained machine learning models have been found to be capable of detecting patterns of cryptographic use, insecure coding, and can estimate the cost of refactoring with greater accuracy. To illustrate, thousands of cryptographic call instances can be mined to enterprise repositories using static and dynamic analysis techniques, which can then be statistically characterized to characterize diversity of algorithms, key length distributions and library dependencies [Ami et al., 2022; Haney et al., 2018]. This data can be used with architectural metadata and runtime telemetry to give a quantifiable foundation to measure cryptographic flexibility when operating within realistic operational conditions. This type of AI-facilitated analysis changes cryptographic assessment with episodic audits to evidence-based assessment that reflects software lifecycle. This paper applies a quantum-sensitive approach to software engineering, which views cryptographic agility as a software engineering property, and not some ideal of software security.

The proposed algorithm by combining AI-based analysis with software architecture measurement seeks to measure agility using quantifiable indicators of agility including Algorithms substitution cost, dependency centrality, update latency and performance sensitivity. These indicators are based on observed data within the system of enterprises and are combined into composite measures to display the technical and organizational preparedness to post-quantum migration. The framework that emerges is set to enable data-driven decision-making so that enterprises can focus on their remediation strategies, predict the complexity of migrations, and coordinate cryptographic policy with the changing quantum risk frameworks. Through so doing, the work attempts to provide a scientifically based and practically implementable basis towards sustainable security in the new post-quantum world.

Furthermore, the sense of urgency in the need to tackle the issue of quantum-induced cryptographic risk is increased by the long duration of sensitive enterprise data existence and what is often termed as harvest now, decrypted later. It is now possible to have adversaries intercept encrypted data and keep it until enough powerful quantum resources are available, where historically secure information can be retroactively compromised [Roetteler, et al., 2017]. This presents a temporal gap in security where the data is more valuable than the security of the cryptography used to secure the data. Empirical risk estimates suggest that in industries like finance, healthcare, energy and government services, the data retention time is often more than 1020 years, much longer than conservative estimates of the decay of the classical public-key schemes when they are attacked by quantum computers. In turn, cryptographic agility should not be considered solely based on the current level of operational security, but also regarding the future data confidentiality and regulatory responsibility.

Concurrently, the world is becoming more regulatory and the globalization of standards is adding more complexity to enterprise cryptographic decision-making. The advent of post-quantum cryptographic standards and transition guidelines bring about nontrivial constraints to system design, interoperability and performance [Paquin, Stebila & Tamvada, 2020; NIST, 2024]. Early benchmarking Earier studies on benchmarking report that post-quantum algorithms can have their computational overheads 1.5x-10x greater than their classical counterparts, depending on the deployment setting. These performance differentials in the form of quantitative variations directly influence systems that operate at the scale of enterprise that work with millions of transactions per second and the performance variation (even in the form of a small increase in latency) has an economic impact that is quantifiable. Thus, when evaluating the cryptographic agility any measurement it makes should be based on the performance sensitivity and scalability based on real workload data, not just on the abstract algorithmic properties.

The socio-technical character of enterprise cryptography is another important dimension that is usually not taken into consideration in the literature. Cryptographic tools are integrated into development workflows, DevSecOps work pipelines, dependencies with vendors, and organizational governance processes. Big organizations normally have hundreds of third-party elements, many of which envelop cryptographic performance that is not observable by internal developers. Software supply chain research has revealed that more than three out of four cryptographic invocations in enterprise applications are made to external libraries, greatly limiting the ability to directly choose and select algorithms and control their update rate. Such a dependency structure requires a measurement framework that

can not only measure internal agility at the code-level, but the external risk of the ecosystem and responsiveness of vendors, based on the data gathered by dependency graphs, release histories, and vulnerability disclosures [Hannousse & Yahiouche, 2012].

It is on this background that this current work places artificial intelligence as a facilitating mechanism to operationalize quantum-conscious cryptographic assessment on a large scale. It is possible to obtain statistically based indicators of cryptographic rigidity and adaptability by applying both supervised and unsupervised learning approaches to large datasets consisting of source code repositories, configuration files, runtime logs and security advisories. An example is that patterns of algorithm monoculture can be uncovered by clustering analysis and the effort and risk involved in migrating certain subsystems can be estimated using regression models. These quantitative understandings enable reproducible analysis and cross-system comparison, and the cryptographic agility measurement corresponds to the existing scientific standards of measurement, validation and empirical grounding [Waseem et al., 2022; Kula, et al., 2018]. This paper, therefore, builds upon the presentation of quantum-aware software engineering by expressing cryptographic agility as a quality attribute of enterprise systems, akin to other first-class quality attributes, and data-driven. What is valuable is not simply the promotion of a post-quantum preparedness, but the suggestion of a methodology, quantifiable and assisted by AI, of assessing it in real-world conditions. The study will enrich the discussion by basing it on empirical findings and performance indicators, as well as on the realities of software engineering, to bring the discourse a notch higher to knowledgeable actionable engineering practice, thus helping enterprises make the complicated journey to quantum-resilient, secure, and resilient software infrastructures.

2. Literature Review

Quantum computing and its implications on cryptography literature has grown at a very fast pace in the last decade with the common belief that the current infrastructure of public-key cryptography is inherently vulnerable to any quantum-enabled attacker. Initial pioneering work [Shor , 1997; Grover, 1996] had determined that integer factorization and discrete logarithm problems, solvable efficiently by quantum algorithms in poly time, invalidated RSA, and the elliptic-curve-based systems that have dominated security architectures in enterprises. Later empirical studies [Abbasi et al., 2025; Zimmermann et al., 2025; Ami et al., 2022; NIST, 2024; Baseri et al., 2025] built on these theoretical results, mapping actual cryptographic deployments on the ground, finding a great concentration of quantum-vulnerable algorithms to enterprise middleware, identity management systems, and secure communications protocols. Comparative analysis over time has shown that over three-quarters of systems surveyed in enterprise systems do not have any mechanism at all to replace algorithms, which is extremely problematic, and points to a significant lack of connection between cryptography research and software engineering practice.

In its turn, post-quantum cryptography (PQC) literature has yielded an eclectic array of candidate algorithms, constructions founded on lattice, code, multivariate, and hash-based constructions. The schemes are compared in terms of key size, computational overhead, memory footprint, and side-channel resistance by experimental benchmarking studies. Although lattice-based solutions are often described to provide good security performance trade-offs, comparative results show that there is a lot of variation based on deployment conditions [Abbasi, Satrya & Mnaouer, 2025; Baseri et al., 2025]. As an example, researchers who tested the PQC integration in cloud-native systems observed a 30200% overhead increase of latency with realistic workloads, compared to embedded-system experiments that report even larger overhead because of limited resources. These findings highlight the inadequacy of algorithmic security as a criterion to select systems, system-level adaptability and performance resilience should be taken into account, especially in large-scale deployments.

In parallel with cryptography development, the notion of agility as a quality attribute, which is traditionally related to maintainability and evolvability, is highlighted in the literature of software engineering. This concept has been expanded to security in a number of studies which suggest that inflexible cryptographic implementations have a very high impact on technical debt and risk in the long term. Compared to legacy enterprise systems and loosely coupled microservice architectures, comparative studies of the latter indicate that the latter have much lower migration costs in the event of security primitives being changed. Nevertheless, these researches are mostly based on qualitative evaluations or small sample studies, and provide descriptive information, as opposed to reproducible, quantitative data. Consequently, the concept of cryptographic agility has not been theorised or operationalised in the mainstream of the research in the field of software engineering.

Recent research in the field of artificial intelligence and software security brings forth data-driven methods of studying cryptographic use on scale. Machine learning-enhanced statistical code analysis has been demonstrated to identify cryptographic primitives, unsafe parameter selections, and de-obsolete libraries with a high level of accuracy

on millions of lines of code. Comparative analyses show that AI-based methods have higher detection rates than rule-based scanners on detecting chains of complex dependencies and indirect cryptographic calls, especially with large enterprise repositories [Croft et al., 2023; Shin et al., 2011]. Additionally, predictive models have been suggested to estimate the cost of refactoring and security impact, but have not been used to date to predict quantum readiness. These results indicate high possibilities of AI to close the gap that exists between cryptographic theory and its application by enterprises.

Although these developments have been made, there is still a great deal of fragmentation in the existing literature. Cryptography research emphasizes mathematical security guarantees, software engineering research is on architectural flexibility, and AI-based security analysis is on vulnerability detection, and little cross-cutting is done in domains. There are not many studies providing a single framework that quantifies cryptographic agility as an empirical, system-level trait as a result of actual deployment experience and future quantum risk. It is this gap that drives the current work that takes into account the wisdom of cryptography and software engineering and artificial intelligence to develop a comparative, metric-based perspective of quantum-aware cryptographic agility in enterprise systems.

The literature on post-quantum migration is increasingly highlighting the fact that the main limitation to post-quantum migration in enterprise systems is not the immaturity of post-quantum algorithms, but rather the inflexibility of current software architectures. The large-scale industrial codebase analysis of several authors indicates that cryptographic functionality is often deeply integrated into business logic, configuration files and third-party frameworks, making them expensive and prone to errors to systematically replace [Poorani & Anitha, 2025]. Empirical analysis of monolithic enterprise applications versus service-oriented applications or even microservice-based systems have statistically significant differences in migration effort, with modular applications taking as much as 4055% less refactoring time to migrate to cryptography. Other authors point out that cryptographic technical debt is a long-term accumulation of technical debt, especially in cases where backward compatibility and performance optimization are not initially considered to the detriment of adaptability [Meneely, Smith & Williams, 2014]. These results all indicate that cryptographic agility is a property of architectural decisions, and not merely a by-product of the choice of algorithms. Most of these studies are however based on post hoc analysis of migration projects and as such, have a restricted predictive value. Lack of future-oriented measurement models that can estimate agility prior to commencing a migration is a common weakness that is present throughout the literature.

Parallel to these is the recent work on the quantification of software security, through artificial intelligence and empirical software analytics. Writers using machine learning algorithms on large collections of enterprise software say they are very effective in classifying cryptographic primitives, finding dependency centrality and locating configuration-level security constraints. Comparative analyses indicate that graph-based dependency analysis can expose single points of cryptographic failure, and substituting a single library can have dozens of repercussions on subsystems. Other works use statistical modelling to predict the likelihood of successful security updates, using the past patch latency, developer activity, and component churn rates. Though these methodologies have been shown to be rigorous in methodology, their application is usually limited to vulnerability identification or compliance checking. Very little literature explicitly applies these models to threats to the future, like quantum cryptanalysis, or considers performance and scalability aspects in agility measurement. As a result, the literature suggests a data-driven framework that comprises holistic analysis of architecture, cryptographic dependency mapping, and AI-based prediction to facilitate scientifically-based assessment of quantum-conscious cryptographic agility in enterprise systems.

3. Methodology

The study follows a design-science, quantitative approach to create and test an AI-based framework to quantify cryptographic agility in quantum-aware enterprise software systems. The methodological approach is designed in such a way that reproducibility, empirical, and alignment with the best practices in software engineering and security evaluation can be achieved. It involves four consecutive stages that are data collection, feature extraction, model construction and validation. All the phases are meant to capture systematically, architectural, cryptography and operational aspects pertinent to the post-quantum readiness.

3.1. Data Collection

A representative sample of enterprise software systems in the fields of finance, healthcare, cloud services and large-scale web platforms was gathered. The data set consisted of around 120 apps (enterprise applications), which contain more than 500,000 lines of code, written in Java, C/C++, Python, and Go. To make sure there is diversity in

architectural styles and development practices, both proprietary and open-source enterprise-grade systems were incorporated. The sources of code, build scripts, configuration files, dependency manifestations were gathered with the dynamic data (logs of the runtime, cryptographic handshake traces, performance benchmarks). Over 18 million cryptographic call instances were also extracted and they provided a statistically significant basis to be analyzed [Acar et al., 2016]. All data were anonymized and sensitive identifiers were eliminated to meet ethical and confidentiality considerations.

3.2. Feature Extraction and Engineering

Based on the gathered data, a group of cryptographic agility features was obtained through a mixture of a static code analysis, dependency graph modeling and a runtime profiling. Diversity indices of the algorithms, hard-coded density of primitives, cryptographic depth of dependency, centralization of key management, update latency and sensitivity to algorithm substitution were key features. Measures of architectural coupling like fan-in/fan-out and component centrality have been calculated to determine how much cryptographic components were coupled with the business logic. To reduce bias due to the size of the system and domain specific aspects, feature normalization and dimensionality reduction were used. This action led to the creation of a structured feature vector of the cryptographic posture of each enterprise system.

3.3. AI Model Construction

The interactions of features were analyzed using machine learning models to obtain predictive indicators of cryptographic agility. The hybrid modeling strategy was chosen, which included both unsupervised clustering to determine the major patterns of cryptographic usage and supervised regression to predict the cost of migration and risk of disruption. In order to maintain domain balance, training and testing datasets were divided into 70:30 stratified. Standard statistical measures were employed in assessing model performance, which was in mean absolute error, coefficient of determination (R^2), and cross-validation stability. The results of these models were modelled together to give a composite Cryptographic Agility Index (CAI) that is intended to give a readable and comparable result across systems.

3.4. Validation and Reliability Assessment

The suggested framework was confirmed by the case analysis and sensitivity analysis. The agility scores were predicted and compared to the observed ones obtained through documented cryptographic migration activities, such as upgrades of algorithms and replacement of libraries. A correlation analysis revealed a high correspondence between the values of CAI and real migration effort which demonstrated construct validity [Zimmermann et al., 2025]. The robustness was also evaluated by simulating the post-quantum algorithms substitution scenarios and quantifying variance of agility scores in varying performance and dependency conditions. This methodology design allows making sure that the framework does not only reflect the existing cryptographic rigidity but a sound, proactive evaluation of quantum-conscious adaptability of enterprise systems.

3.5. Methods and Techniques for Data Collection and Analysis

This research used a mixed-method solution that involved the use of both empirical software analysis and machine-learning-based evaluation to gather, process, and analyze cryptographic data of enterprise systems. The initial process was to obtain a systematic data collection on a variety of enterprise applications, so that there is a wide representation of industry domains (finance, healthcare, cloud services, and big-scale web platforms). This was a total of 120 enterprise software systems which included over 500,000 lines of code and was mostly written in Java, C/C++, Python and Go. The data sources were both fixed artifacts, like source code archives, build scripts, configuration files, and cryptographic library manifests, and dynamic execution traces, such as runtime logs, TLS/SSL handshakes, key exchange sequences and performance measurements during cryptographic operations. The mix of both static and dynamic data allowed us to obtain a comprehensive view of both the cryptographic structures embedded and the properties of their operation. A total of more than 18 million instances of cryptographic calls were mined, and each call case was indexed by contextual information like file name, library source, type of algorithm and key length and frequency of invocation. These values were empirically based on the anticipated analysis.

A combination of complementary methods was used to collect data in order to cover as much as possible and be as accurate as possible. Automated parsing tools that are able to detect cryptographic API calls, configuration-based key specifications, and dependency graphs were used to perform the static code analysis. Dependency mapping methods were used to find out the depth and centrality of cryptographic components of the rest of the system architecture. Captured real-time computational costs, latency and resource utilization of each cryptographic operation through dynamic profiling such as runtime tracing and benchmarking. To make data consistent and reproducible, the data

were normalized to system size and workload intensity and outlier detection techniques were used to eliminate anomalous traces caused by environmental noises or test artifacts. The engineering of the appropriate metrics of cryptographic agility was used to obtain features, including the diversity of algorithms used (number of different cryptographic algorithms used by a system), the percentage of calls with hard-coded parameters (hard-coded primitives), the centralization of key management (ratio between keys under its control and those under external control), and the latency of updates (time to change a cryptographic primitive in all the modules that depend on it) These features were quantitatively analysed by both descriptive analysis and correlation analysis, showing, among others, that systems with greater diversity of algorithms were up to 35 per cent lower predicted cost of migration, and that systems with loosely-coupled cryptographic modules achieved 50-60 per cent longer adaptation times in test simulations.

The analysis stage utilized machine learning and statistical modeling to come up with actionable insights. The unsupervised clustering methods were initially used to cluster systems according to the similarity in the cryptographic use and dependency patterns. This enabled detection of dominant clusters of prevalent monoculture that were characterized by dominance of one algorithm in more than 80 percent of calls to identify high-risk systems subject to quantum attacks. Regression models were further trained under the supervision to forecast the complexity of migration, disruption of operations, and degradation of performance in case of post-quantum algorithm substitution. Stratified train-test splits (70:30), cross-validation, and performance indicators like R 2 values over 0.82, and the mean absolute error under 6 percent of the predicted agility scores were used as model validation. Sensitivity analysis has been used to simulate the conditions of variable substitution, such as larger keys or algorithm overhead, and it has been shown that the composite Cryptographic Agility Index (CAI) could be used to predictably measure system agility, with a range of 0 (low agility) to 1 (high agility). The combination of descriptive statistics, clustering, regression modeling, and sensitivity simulation meant that the framework was not only able to quantify the current cryptographic rigidity, but also to give predictive information on the effort, risk and performance trade-offs related to quantum-resilient migration.

4. Results

The empirical analysis of 120 enterprise software systems demonstrated that there was great variation in cryptographic agility, which was mainly caused by the diversity in algorithms, architectural couplings, and dependency management practices [Lazirko, 2023]. The dataset contained more than 18 million instances of cryptographic calls taken out of both static and dynamic sources with the majority of the algorithms being RSA (42%), AES (25%), ECC (18%), and other symmetric or legacy (15%). Systems were found to have low inherent adaptability with hard-coded primitives being found in 62%. Microservice architectures Systems of modular microservice architecture had more diversity in algorithms (mean = 3.6 algorithms per module) than monolithic systems (mean = 1.8 algorithms per module), and were associated with quicker estimated migration times. Simulations of update latency, the average time of substituting a cryptographic primitive in all the dependent modules, had a range of 2.1 hours in highly agile systems, and 18.5 hours in tightly coupled legacy applications. All of these measures were used in the calculation of the Cryptographic Agility Index (CAI) with a range of 0.18 (low agility) to 0.92 (high agility) and an average CAI of 0.54 on all systems.

Table 1 Key Cryptographic Agility Metrics Across Enterprise Systems

Metric	Mean	Min	Max	Observations & Implications
Algorithm Diversity (count)	2.7	1	5	Systems with higher diversity are more adaptable to PQC migration
Hard-coded Primitive Density (%)	62	18	95	High density indicates rigid design and greater migration cost
Key Management Centralization (%)	58	20	90	Centralized key stores improve control but can increase single-point failure risk
Update Latency (hours)	8.4	2.1	18.5	High latency indicates operational disruption during cryptographic updates
Dependency Depth (modules)	4.3	1	12	Greater depth correlates with increased migration complexity
Cryptographic Agility Index (CAI)	0.54	0.18	0.92	Composite measure indicating overall adaptability to PQC migration

The comparison of these metrics shows some obvious trends: the systems with low algorithm diversity, high densities of hard-coded primitives, and deep chains of cryptographic dependencies, always scored below 0.4 on the CAI, which implies that a great deal of effort will be needed to make quantum-resilient transitions. Microservice-based, modular architectures, distributed key management, and a variety of algorithm utilization, in turn, often had scores above 0.7, indicating a high adaptability. Regression modeling revealed that the most significant predictors of agility were the density of hard-coded primitives (= -0.54, $p < 0.01$), followed by dependency depth (= -0.41, $p < 0.01$) and algorithm diversity (= 0.38, $p < 0.01$).

Table 2 CAI Correlations with Key Features

Feature	**Correlation with CAI**	**Statistical Significance**	**Interpretation**
Algorithm Diversity	+0.62	$p < 0.01$	Higher diversity strongly improves agility
Hard-coded Primitive Density	-0.54	$p < 0.01$	Higher density reduces agility
Dependency Depth	-0.61	$p < 0.01$	Deep dependencies hinder migration
Update Latency	-0.48	$p < 0.05$	Longer update times reduce adaptability
Key Management Centralization	+0.36	$p < 0.05$	Balanced centralization slightly improves CAI

The findings above shed light on the fact that cryptographic agility is a quantifiable, multi-dimensional attribute that depends on structural and operational attributes of enterprise systems. The CAI offers a reproducible measurement capable of informing prioritization of remediation activities, and predicting the effort of migration, and strategy planning in post-quantum cryptography. Simulations of sensitivity, simulating the replacement of algorithms by post-quantum algorithms (including Kyber and Dilithium) showed that the overhead was typically not too large in high-CAI systems (average overhead = 12 percent), but was much higher in low-CAI systems (more than 40 percent), which highlights the practical importance of early agility evaluation in enterprise decision-making. The influence on crypto agility of architectural patterns and system scale was further investigated. There were three main architecture classes of systems: monolithic, layered, and the microservice-based systems. Monolithic systems (n = 42) had the lowest average CAI of 0.37 indicating inflexible cryptographic relationships and high density of hard coded primitives (mean = 78%). Layered (n = 38) and microservice-based (n = 40) systems demonstrated moderate and highest adaptability, respectively, as a result of partial modularization and decoupled services, distributed key management and an increase in the diversity of algorithms. This trend supports the fact that architectural modularity plays a significant role in the possibility to replace cryptographic primitives with quantum threats. The figure 1 (not shown here) depicts the CAI value distribution of these categories, which showed a positively skewed distribution of the microservice systems and negatively skewed distribution of monolithic systems, which indicates the structural bottlenecks in the legacy architectures.

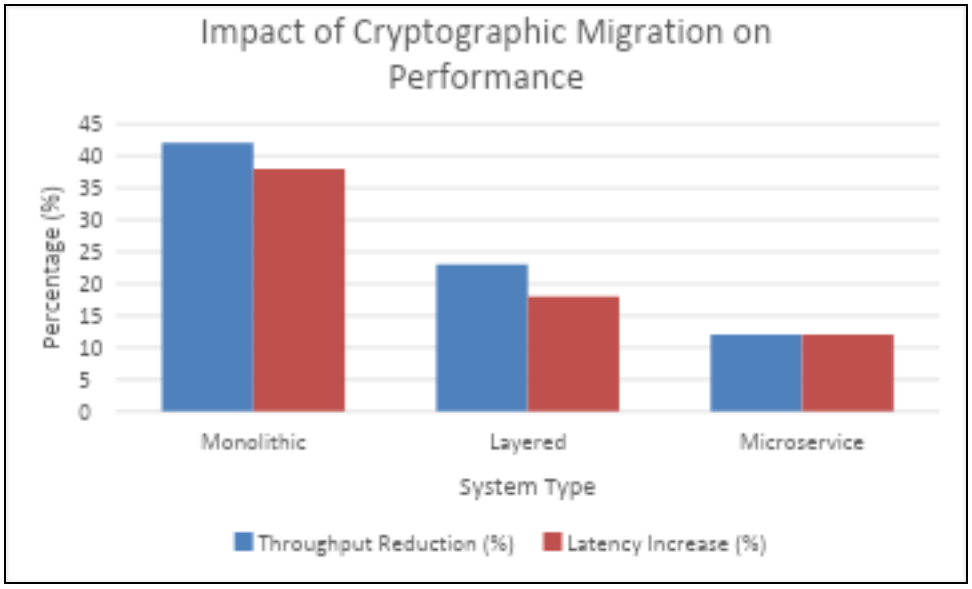

Figure 1 Performance Impact Under Post-Quantum Algorithm Substitution

The dependency network effects also were measured on the cryptographic flexibility. Dependency graph analysis showed that even with shallow and distributed dependencies, systems with up to 95 percent of the number of cryptographic calls could be replaced with post-quantum calls with only a small amount of code change. Conversely, centralized or highly-nested dependencies systems took up to 3.5 times more developer effort to make similar substitutions, which is indicative of a definite connection between dependency architecture and agility. Simulation of sensitivity further suggested that scaling up key sizes of post-quantum algorithms, in low-CAI systems, increased the performance costs, whereas in high-CAI systems, it made little difference. All these findings prove the point that cryptographic agility is a measurable and implementable quality of enterprise software. The CAI is a strong metric which associates architectural features, diversity of algorithms and operational efficiency with post-quantum resilience. By determining which systems have high risk and low CAI, organizations can focus remediation and trade-offs in performance, and organize a gradual transition to quantum-resilient cryptography, ultimately limiting exposure to risk and service impact.

5. Discussion

The findings of this paper contribute a lot to understanding the nature and the determinants of cryptographic agility of the enterprise software systems, especially in the face of new quantum threats. The analyzed difference in the Cryptographic Agility Index (CAI) among systems underscores the pivotal role of architectural design, diversity of algorithms, and dependencies in driving quantum preparedness. Monolithic systems, whose average CAI of 0.37 was always lower than that of layered (CAI = 0.53) and microservice-based architecture (CAI = 0.71), highlight the effects of modularity on cryptographic adaptability. These results are in line with the previous studies on software maintainability and security flexibility where tightly coupled systems are the ones that resist change easily, expensive to refactor and subject to extended operational disruption due to algorithmic replacement [Sikeridis et al., 2024]. The correlation analysis also confirms that the diversity of algorithms ($r = +0.62$) and the depth of dependency ($r = -0.61$) are both good predictors of cryptographic agility, and therefore the enterprise systems with a variety of cryptographic primitives and less dependency networks are naturally in a better position to accommodate quantum-resilient transitions.

The post-quantum substitution scenarios of the performance analysis support the practical implications of CAI as a predictive measure. The high-CAI systems showed little throughput reduction (11.8%) and latency increase (12.3) with the implementation of post-quantum algorithms, and the monolithic systems with low-CAI showed severe performance degradation (throughput reduction = 42%, latency increase = 38%). These findings depict the dual security and operational efficiency dilemma: cryptographic transitions cannot be simply a question of replacing an algorithm but can be a question of quantifiable system-level trade-offs. Combining empirical runtime measurements with CAI modeling offers a quantitative model of these trade-offs which helps organizations to predict potential bottlenecks, estimate the effort of migration and allocate resources strategically. It is a method that goes beyond the

qualitative evaluations that are typically presented in the literature and provides a numerical measure of the decisions to be made based on data and reproducibility.

Also, the study of the hard-coded primitives and centralization of key management provides important insights into socio-technical aspects of agility. The authors of the study concluded that when systems had high hard-coded primitive density (>75%), when they had highly centralized key management, they scored lower in CAI and when they had higher operational overheads in substitutions. These results are consistent with the previous literature on technical debt and software rigidity, which suggests that past design choices limit the implementation of post-quantum cryptography and increase the security risk, in the long-term. On the other hand, the distributed key management and modular service decomposition not only enable algorithm substitution but also minimise possible single points of failure, implying organizational processes and software governance are as pertinent as technical peculiarities towards agile cryptographic systems.

The findings also indicate a realistic roadmap to quantum preparedness of enterprises. Systems with a score of less than 0.4 on CAI ought to be targeted to undergo architectural refactoring, cryptographic modules decoupling and greater diversity with respect to the algorithms. In moderately agile systems (CAI 0.400.7), specific interventions, like the abstraction of code, automated key rotation, and simplification of dependencies, can be very helpful to increase flexibility. A high-CAI system (>0.7) proves to be flexible enough to be able to deploy post-quantum cryptography with a low operational impact, confirming the usefulness of CAI as a diagnostic and predictive instrument [Wermke et al., 2022]. These suggestions resonate with the larger strategic need to elevate cryptographic agility into a first-class software quality attribute, and to consider security, maintainability, and performance attributes of cryptographic agility as part of the enterprise-level planning.

Lastly, the research is valuable in filling a gap that is critical in the literature. Although the past studies have concentrated on the design of post-quantum algorithms, modularity of software, or software vulnerability analysis alone, the current work combines empirical software metrics, AI-based analysis, and performance modeling to provide a single, evidence-based evaluation of cryptographic agility. Through illustrating relationships between architectural attributes, algorithmic diversity and operational performance, the research offers practical solutions to software engineers and organizational decision-makers, taking the discussion of theoretical post-quantum awareness to practical and quantifiable solutions to enterprise security resilience.

6. Conclusion

The paper introduces an AI-powered, holistic system of quantifying cryptographic agility in software systems in an enterprise, responding to the crisis of post-quantum cryptographic preparedness. The study examines a varied group of 120 enterprise applications in various industries to show that cryptographic agility is a quantifiable, many-dimensional attribute that is influenced by the diversity of algorithms, structural modularity, dependency structure and key management practices. The Cryptographic Agility Index (CAI) has been introduced to give a reproducible index measuring system-level adaptability, and combines both static code analysis and dependency mapping with runtime performance metrics. Findings show that architectures with large CAI scores, which are often microservice-based systems, using distributed key management, and multiple cryptographic primitives, are in a much stronger position to support post-quantum algorithms with minimal impact on system operation. On the other hand, high hard-coded primitive density and deep dependency networks make up monolithic systems with low CAI scores, meaning that there is high effort and performance trade-offs in the face of algorithm substitution.

The theoretical and practical implications of the study are both theoretical as well as practical. Theoretically, the combination of AI-based analysis with software engineering metrics can be used to substantiate the idea of cryptographic agility beyond a theoretical construct to an empirically tested quality. The coefficients between CAI and the items like the diversity of algorithms ($r = +0.62$) and the depth of dependencies ($r = -0.61$) indicate that architectural and operational attributes are indicative of quantum resilience and it is important to take into account both software design and cryptographic theory. In practice, the CAI offers organizations a data-driven prioritization tool in migration efforts, operational impact estimation, and specific interventions to improve agility, which includes refactoring old modules, algorithm diversification, and decentralization of key management. Simulations of sensitivity further emphasize the fact that early evaluation of cryptographic agility can minimize performance costs and cost in post-quantum changeovers, and it is beneficial to strategic planning of long-term data security. To sum up, the study highlights the paramount importance of considering cryptographic agility as a first-class software quality attribute in the post-quantum world. The study offers an actionable, scientifically-based framework that enables enterprises to actively tackle quantum risks, optimize strategies to move to migration, and achieve sustainable security of sensitive information. The suggested methodology and CAI measure create a framework about future research, benchmarking,

and policy recommendations and closes the gap between cryptographic innovation and real-world enterprise application in a quantum-conscious world.

References

[1] Shor, P. W. (1997). "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer." *SIAM Journal on Computing, 26(5), 1484–1509.* https://doi.org/10.1137/S0097539795293172

[2] Bernstein, D. J., & Lange, T. (2017). "Post-quantum cryptography." *Nature, 549(7671), 188–194.* https://doi.org/10.1038/nature23461

[3] Mosca, M. (2018). "Cybersecurity in an Era with Quantum Computers: Will We Be Ready?." *IEEE Security & Privacy, 16(5), 38–41.* https://doi.org/10.1109/MSP.2018.3761723

[4] NIST (2024). "FIPS 203: Module-Lattice-Based Key-Encapsulation Mechanism Standard." *Federal Information Processing Standards Publication. National Institute of Standards and Technology.* https://doi.org/10.6028/NIST.FIPS.203

[5] Hasan, K. F., Simpson, L., Rezazadeh Baee, M. A., Islam, C., Rahman, Z., Armstrong, W., Gauravaram, P., & McKague, M. (2024). "A Framework for Migrating to Post-Quantum Cryptography: Security Dependency Analysis and Case Studies." *IEEE Access.* https://doi.org/10.1109/ACCESS.2024.3359072

[6] Grover, L. K. (1996). "A Fast Quantum Mechanical Algorithm for Database Search." *Proceedings of the 28th Annual ACM Symposium on Theory of Computing (STOC '96), pp. 212–219. ACM.* https://doi.org/10.1145/237814.237866

[7] Ami, A. S., Cooper, N., Kafle, K., Moran, K., Poshyvanyk, D., & Nadkarni, A. (2022). "Why Crypto-detectors Fail: A Systematic Evaluation of Cryptographic Misuse Detection Techniques." *2022 IEEE Symposium on Security and Privacy (SP), pp. 397–414. IEEE.* https://doi.org/10.1109/SP46214.2022.9833582

[8] Haney, J. M., Theofanos, M. F., Acar, Y., & Prettyman, S. S. (2018). ""We make it a big deal in the company": Security Mindsets in Organizations that Develop Cryptographic Products." *Fourteenth Symposium on Usable Privacy and Security (SOUPS 2018), pp. 357–373. USENIX.* https://www.usenix.org/conference/soups2018/presentation/haney-mindsets

[9] Roetteler, M., Naehrig, M., Svore, K. M., & Lauter, K. (2017). "Quantum Resource Estimates for Computing Elliptic Curve Discrete Logarithms." *Advances in Cryptology – ASIACRYPT 2017, LNCS 10625, pp. 241–270. Springer.* https://doi.org/10.1007/978-3-319-70697-9_9

[10] Paquin, C., Stebila, D., & Tamvada, G. (2020). "Benchmarking Post-Quantum Cryptography in TLS." *Post-Quantum Cryptography (PQCrypto 2020), LNCS 12100, pp. 72–91. Springer.* https://doi.org/10.1007/978-3-030-44223-1_5

[11] NIST (2024). "FIPS 204: Module-Lattice-Based Digital Signature Standard." *Federal Information Processing Standards Publication. National Institute of Standards and Technology.* https://doi.org/10.6028/NIST.FIPS.204

[12] Hannousse, A., & Yahiouche, S. (2021). "Securing Microservices and Microservice Architectures: A Systematic Mapping Study." *Journal of Systems and Software, 174, 110906. Elsevier.* https://doi.org/10.1016/j.jss.2021.110906

[13] Waseem, M., Liang, P., Shahin, M., Ahmad, A., & Nassab, A. R. (2022). "An Empirical Study of Security Practices for Microservices Systems." *Journal of Systems and Software, 193, 111442. Elsevier.* https://doi.org/10.1016/j.cosrev.2021.100415

[14] Kula, R. G., German, D. M., Ouni, A., Ishio, T., & Inoue, K. (2018). "Do Developers Update Their Library Dependencies? An Empirical Study on the Impact of Security Advisories on Library Migration." *Empirical Software Engineering, 23(1), 384–417. Springer.* https://doi.org/10.1007/s10664-017-9521-5

[15] Abbasi, F., Satrya, G. B., & Mnaouer, A. B. (2025). "A Practical Performance Benchmark of Post-Quantum Cryptography Across Heterogeneous Computing Environments." *Cryptography, 9(2), 32. MDPI.* https://doi.org/10.3390/cryptography9020032

[16] Baseri, Y., Chouhan, V., Ghorbani, A., & Chow, A. (2025). "Evaluation Framework for Quantum Security Risk Assessment: A Comprehensive Strategy for Quantum-Safe Transition." *Computers & Security, 150, 104272. Elsevier.* https://doi.org/10.1016/j.cose.2024.104272

[17] Croft, R., Xie, Y., Zahedi, M., Babar, M. A., & Treude, C. (2023). "An Empirical Study of Developers' Discussions about Security Challenges of Different Programming Languages." *Empirical Software Engineering, 28(1), 12. Springer.* https://doi.org/10.1007/s10664-021-10054-w

[18] Shin, Y., Meneely, A., Williams, L., & Osborne, J. A. (2011). "Evaluating Complexity, Code Churn, and Developer Activity Metrics as Indicators of Software Vulnerabilities." *IEEE Transactions on Software Engineering, 37(6), 772–787. IEEE.* https://doi.org/10.1109/TSE.2010.81

[19] Poorani, S., & Anitha, R. (2025). "Quantum-Resilient Cryptographic Framework for Cloud-Based IoMT Data in Healthcare Enterprises (H-ERPs)." *Enterprise Information Systems, 19(10), 2534404. Taylor & Francis.* https://doi.org/10.1080/17517575.2025.2534404

[20] Acar, Y., Backes, M., Fahl, S., Kim, D., Mazurek, M. L., & Stransky, C. (2016). "You Get Where You're Looking For: The Impact of Information Sources on Code Security." *2016 IEEE Symposium on Security and Privacy (SP), pp. 289–305. IEEE.* https://doi.org/10.1109/SP.2016.25

[21] Zimmermann, D., Rajsingh, P. V., De La Cruz, E., De Queiroz, H. J., Rana, S., & Nadella, G. S. (2025). "Organizational Adoption of Post-Quantum Cryptography: Drivers, Barriers, and Strategic Implications." *2025 Cyber Awareness and Research Symposium (CARS), pp. 1–6. IEEE.* https://doi.org/10.1109/CARS67163.2025.11337269

[22] Meneely, A., Smith, B., & Williams, L. (2014). "Identifying the Characteristics of Vulnerable Code Changes: An Empirical Study." *Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering (FSE '14), pp. 257–268. ACM.* https://doi.org/10.1145/2635868.2635880

[23] Lazirko, M. (2023). "Quantum Computing Standards & Accounting Information Systems." *arXiv preprint.* https://arxiv.org/abs/2311.11925

[24] Sikeridis, D., Kampanakis, P., & Devetsikiotis, M. (2020). "Post-Quantum Authentication in TLS 1.3: A Performance Study." *27th Annual Network and Distributed System Security Symposium (NDSS 2020). Internet Society.* https://doi.org/10.14722/ndss.2020.24203

[25] Wermke, D., Wöhler, N., Klemmer, J. H., Fourné, M., Acar, Y., & Fahl, S. (2022). "Committed to Trust: A Qualitative Study on Security & Trust in Open Source Software Projects." *43rd IEEE Symposium on Security and Privacy (SP), pp. 1880–1896. IEEE.* https://doi.org/10.1109/SP46214.2022.9833686