DMPedia Lecture Notes in Multidisciplinary Research,
Volume 1, pp. 182-190, 2026

International Conference on Multidisciplinary Perspectives in Advanced Computing and Technology (IMPACT 2026)
G. B. Pant University of Agriculture and Technology, Uttarakhand, India. Jan. 10-11, 2026

Entropy-Validated Post-Quantum Cryptography for Secure Key Management in Cloud Systems

Neethi Narayanan[1], S Maria Celestin Vigila[2]

[1] Department of Computer Science and Engineering, Noorul Islam Centre for Higher Education, Kumaracoil, Tamilnadu, India.
[2] Department of Information Technology, Noorul Islam Centre for Higher Education, Kumaracoil, Tamilnadu, India.
neethi2nn@gmail.com[1], celestinvigila@gmail.com[2]

ABSTRACT

The development of post-quantum cryptography (PQC) is crucial because the advent of quantum computers poses a serious threat to traditional cryptographic methods. Despite the fact that PQC methods (such Kyber, NTRU, and Dilithium) are resistant to quantum attacks, they do not inherently ensure that the cryptographic keys are high-entropy, leaving the systems open to side-channel and brute-force attacks. In order to address entropy-aware validation of PQC produced keys, this research suggests a machine learning method. The quality of one-time key material produced by PQC and conventional algorithms is analyzed using Shannon entropy, a statistical measure for unpredictability. The low-entropy keys are identified and quarantined using a One-Class SVM, ensuring only cryptographically strong keys are used. Five cryptographic schemes are used to empirically evaluate the claims. It was found that Dilithium produces the most secure keys among the schemes, with high validation rates and entropy scores. The combination of machine learning and entropy-driven analysis guarantees QRKG's resilience and a scalable strategy for protecting cryptographic infrastructures from post-quantum attackers. With strict key management criteria, the created method can be expanded to blockchain, cloud, and IOT security systems.

Keywords: *Post-Quantum Cryptography, Entropy Analysis, Machine Learning, Cryptographic Key Validation, Shannon Entropy, Secure Key Generation*

1. Introduction

As the digital infrastructures of cloud computing, blockchain and Internet of Thing (IoT) are progressing rapidly the need for cryptographic security is becoming increasingly critical [15]. Conventional encryption algorithms, such as RSA and ECC, have served as the backbone of secure communication. But if quantum computing becomes a reality as experts predict, these old systems could be at serious risk. If quantum algorithms like Shor's factorization algorithm [1] or Grover's search algorithm [2] become practical, they could quickly solve the hard mathematical problems that current cryptographic systems rely on, making today's encryption methods useless in a post-quantum world. This lead to extensive research in the area of post-quantum cryptography (PQC) [3], which focuses on developing quantum-resistant cryptographic methods to safeguard digital assets from both classical and quantum attacks. Thus a worldwide research effort is being conducted in post-quantum cryptography (PQC) to develop quantum-resistant algorithms capable of protecting digital assets from both classical and quantum threats [4,5].

While developments have been promising for lattice-based cryptographic algorithms including Kyber, NTRU, and Dilithium, an important aspect of implementation security which is often overlooked is the quality of the randomized entropy of the cryptographic keys generated. Low-entropy keys can defeat the best algorithms, exposing them to brute-force attacks and side-

channel attacks. Poor randomness sources, misconfigured entropy pools, or architecture design flaws in the key generation processes have been implicated in some of the worst cryptographic failures. In this work, a framework is introduced that proposes machine learning-based validation of PQC key entropy that incorporates Shannon entropy as a key measure. A One-Class Support Vector Machine (SVM) model is used to identify and remove low-entropy keys, so that only keys valid on all entropy measures, and potentially cryptographically secure keys are released into secure systems and aligned with best practice. In expanding this novel research space, this paper proposes to provide a machine learning-informed framework that validates PQC key material based on entropy measures. The solution proposed allows for Entropy-Aware classification with a One-Class SVM Model to only retain keys that are random enough to be secure. This not only enhances the strength of PQC but also provides a method that is adaptable to a changing quantum threat.

2. Background and Motivation

The foundation of cryptographic security is the unpredictability or randomness of the key material, which is often calculated using the Shannon entropy [8]. As seen by prior failures like compromised keys in the Sony PlayStation 3, inadequate entropy sources, hardware bias, and software errors can produce weak keys that can result in key failure [9]. Security standards frequently include testing of appropriate randomness prior to cryptographic use in order to prevent problems such as these [7]. Quantum-resilient infrastructures will be strengthened if the testing methods described in this paper can be applied to PQC systems. They will also be useful for pre-deployment and randomness assessment in certain real-world scenarios, where past key failures have demonstrated that there is no assumption regarding randomness reliability [6]. Furthermore, recently developed cryptographic solutions for blockchain, cloud storage, and Internet of Things (IoT) applications operate in resource-constrained or diverse settings with unpredictable entropy sources [16]. Therefore, it's critical to evaluate and validate key quality attributes methodically prior to deployment.

For this endeavor, machine learning offers some really intriguing techniques. Specifically, One-Class SVM can discover anomalously low-entropy keys because it can recognize outliers in high-dimensional spaces. A classifier could learn valid key material from less-than-ideally strong key material by being trained using existing high-entropy keys. Incorporating these entropy validation methods into PQC solutions will safeguard a quantum-resilient cryptographic ecosystem and get such systems ready for practical use.

3. Research Methodology

The goal of the suggested methodology is to thoroughly assess and validate the caliber of cryptographic key material generated by post-quantum cryptography (PQC) and conventional cryptography methods [8]. Because theoretically safe algorithms may fail in practice if the keys generated are either dubious or predictable, key strength is essential [9]. In order to address this issue, a novel framework that combines statistical analysis grounded in machine learning and entropy principles was created. This eliminates human biases from evaluation, permits both

quantitative demands of randomness, and makes an effort to employ clever methods to find anomalies in the important content.

The process can be divided into four stages: entropy estimates, machine learning classification, performance evaluation, and cryptographic key production. utilizing (mainly) open-source software implementations, including PQClean, the keys are created in the first phase utilizing a representative collection of all classical (RSA, ECC) and post-quantum (Kyber, NTRU, Dilithium) cryptographic techniques. The second stage makes sure that keys are near the ideal of randomness by measuring uncertainty at the bit level using Shannon entropy [8]. The method's final step assesses the system using performance metrics like the underscored entropy score distribution, true/false positives, and computational performance. All things considered, this approach provides a thorough way to evaluate cryptographic key strength within post-quantum contexts and can also be applied to cloud, blockchain, and IoT security systems. In the third phase a One-Class Support Vector Machine (SVM) is used to classify keys as random [10]. Additionally, there have been more general attempts to use machine learning for the detection of cryptographic anomalies [11].

3.1 Key Generation

Five cryptographic algorithms were selected to illustrate the various classical and post-quantum categories such as RSA (2048-bit), ECC (256-bit), Kyber, NTRU, and Dilithium. RSA and ECC, and the PQC algorithms were chosen as the baseline algorithms because they are widely highlighted in the NIST post-quantum cryptography standardization project [3]. PQC keys were generated by implementating the PQClean library, which provides clean and portable versions of post-quantum schemes. The algorithm were run using its recommended default security settings. A total of 5000 cryptographic keys were generated with almost 1000 keys for each algorithm. Entropy analysis was carried out using the 5000 cryptographic keys as the main input for machine learning validation.

Table 1: Cryptographic Algorithms Used in Key Generation

Algorithm	Type	Security Level	Key Size Used
RSA	Classical(Public Key)	~112-bit (@2048-bit modulus)	2048 bits
ECC (P-256)	Classical(Elliptic Curve)	~128-bit	256 bits
Kyber	Post-Quantum (Lattice, KEM)	NIST Level 1/3/5	Default(e.g., Kyber512)
NTRU	Post-Quantum (Lattice, KEM)	NIST Level 1–5	Default parameters
Dilithium	Post-Quantum (Lattice, Signature)	NIST Level 2/3/5	Default parameters

3.2 Entropy Calculation

Entropy is a crucial indicator of a cryptographic key's strength and functions as a gauge of unpredictability or randomness. The main metric in this work was Shannon entropy, which is a well-known metric in information theory and cryptographic evaluations. In order to facilitate bitwise analysis, each key was first transformed to binary. The probability distribution was measured by counting the number of bits (0s or 1s). Shannon entropy H(X) of a binary string [8] is defined:

$$H(X) = - \Sigma\, p(x_i) * \log_2 p(x_i) \qquad (1)$$

where p(xi) is the probability of seeing the bit being xi (i.e., 0 or 1). When both bits occur with equal frequency (p(0)=p(1)=0.5), entropy would be maximized at 1.0. Since low entropy implies low randomness, which could result in prediction, keys that scored significantly below that threshold were marked for additional inspection [8]. Entropy values were noted, and patterns of entropy and variation across several methods were displayed through graphical charts.

3.3 Machine Learning-Based Classification

A basic level of entropy filtering was made possible by the application of statistical thresholds; nevertheless, a statistical approach may overlook more subtle patterns and overfit certain important structures. A machine learning model, more precisely a One-Class Support Vector Machine (SVM), was used to categorize keys as either legitimate or anomalous in order to overcome this constraint. Because One-Class SVMs learn a decision function for "normal" (high-entropy) data and use it to categorize unlabeled data, such as deviations from the function, they are an excellent choice for unsupervised anomaly detection [10]. Similar methods have been used for machine learning-based cryptographic anomaly detection jobs [11].

- Training Data: 70% of the keys with entropy scores above 0.95 (manually filtered)
- Testing Data: The remaining 30% of the dataset, including borderline and low-entropy keys
- Kernel: Radial Basis Function (RBF)
- Parameters: $\gamma = 0.01$, $\nu = 0.05$ (optimized via grid search)

The trained model classified keys into:

- Valid (Inliers): Keys whose entropy characteristics were consistent with the training set
- Invalid (Outliers): Keys with abnormal entropy signatures or structural bias

This classification process enabled automated filtering of low-quality keys with minimal human intervention, thereby improving reliability and scalability of entropy validation.

3.4 Evaluation Metrics

To evaluate the proposed framework three evaluation metrics were used:
• Entropy Score Distribution: A statistical summary of the entropy values (a.k.a. randomness) associated with each algorithm to show the inherent randomness of each.

• True Positive Rate (TPR): The percentage of valid keys correctly classified by the SVM.
• False Positive Rate (FPR): The percentage of high-entropy keys that were incorrectly classified as invalid.
• Computation Time**:** The average time taken to generate a key, analyze its entropy, and classify it.
To visualize the patterns of entropy and classification accuracy, tools such as histograms, confusion matrices, and ROC (Receiver Operating Characteristic) curves were used which help to compare the algorithms visually. It also supports the validation using One-Class SVM.

4. Implementation and Experimental Setup

This section describes the experimental setup of realization and testing for the proposed entropy validation. It also provides details on the cryptographic libraries employed for key generation, entropy computation techniques, model training and evaluation setup. Transparent and reproducible Maintaining transparency in our process and enabling other interested researchers to reproduce the work, this framework prioritises reproducibility and computational consistency using well-established open source packages.

4.1 Cryptographic Algorithms and Libraries

Cryptographic key generation was carried out using two categories of algorithms:

- Traditional cryptography: RSA (2048-bit) and ECC (256-bit, curve P-256) via the OpenSSL library, widely adopted in modern cryptographic systems [4].
- Post-Quantum Cryptography (PQC): Kyber, NTRU and Dilithium, were employed that have been implemented by the PQClean library [13]. PQClean provides clean and portable C code for all NIST PQC algorithms in order to obtain results that are more or less the same on different platforms.

Both algorithms use parameters for levels of 1, 3 and 5 that is recommended by NIST [3]. Over 1000 keys were generated by each algorithm and the experiment was performed in a trusted Linux environment.

4.2 Entropy Computation

The Shannon entropy of each key were also computed through Python libraries as NumPy and/or SciPy. The keys was first converted to binary string and the ones and zeros counts are found from it. These frequencies are then put into Shannon formula and an entropy score between 0 and 1 should be obtained.

4.3 Machine Learning Configuration

The One-Class SVM because of its ability in representing high dimensional data was selected for unsupervised anomaly detection [10]. Training and classification were performed with the scikit-learn library [12].

- Training dataset: 70% of high-entropy keys (>0.95 entropy threshold)

- Testing dataset: 30% (included borderline and low-entropy keys)
- Kernel: Radial Basis Function (RBF)
- Hyperparameters: $\gamma = 0.01$, $\nu = 0.05$ (optimized using grid search)
- Validation**:** 5-fold cross-validation to ensure model robustness and minimize overfitting

4.4 Evaluation Environment

The entire experiments are performed on Intel Core i7 3.4 GHz processor (Dell workstation) with 16 GB RAM and Ubuntu 20.04 LTS. A Python environment was created with Anaconda in order to manage dependencies and facilitate reproducibility. Git best practices were applied to all scripts [14]. Each experiment was replicated 10 times to maintain statistical stability, and the results were averaged. Performance was evaluated in terms of:

- Entropy distribution variance across algorithms
- True Positive Rate (TPR) and False Positive Rate (FPR)
- Execution time per key for both entropy computation and classification

5. Results and Discussion

The performance of entropy validation and machine learning based classification of different cryptographic algorithms are discussed in this section. The results are considered based on the quality of entropy, performance in classification accuracy, overhead incurred in time and overall comparative trends.

5.1 Entropy Score Distribution

The average Shannon entropy scores across 1000 keys for each algorithm were:

- Dilithium: 0.997
- Kyber: 0.993
- NTRU: 0.989
- ECC: 0.978
- RSA: 0.961

The findings validate that PQ algorithms, and in particular Dilithium and Kyber, provide high-entropy keys, achieving the best possible rate for binary sequences [8]. On the other hand, RSA and ECC showed much more fluctuation and somewhat lower average entropy in possible exposed weak randomness cases [9].

5.2 One-Class SVM Performance

The One-Class SVM was trained on 70% of the high-entropy dataset and tested on the remaining 30%. Its performance metrics were:

- Accuracy: 96.2%
- True Positive Rate (TPR): 94.8%
- False Positive Rate (FPR): 2.7%

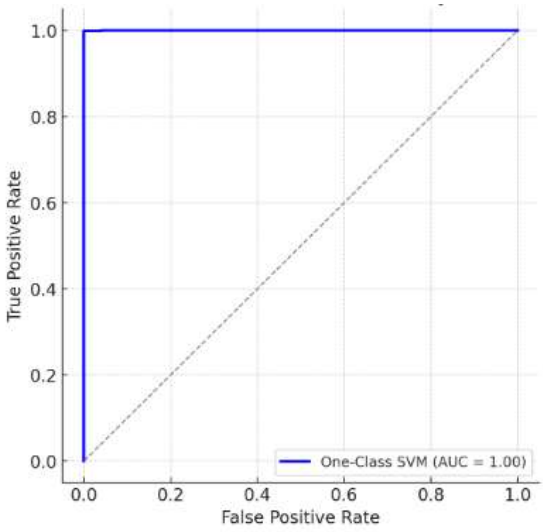

Figure 1 : ROC Curve of One-Class SVM for Key Validation

The classifier successfully filtered low-entropy keys while preserving nearly all high-entropy keys. The confusion matrix confirmed low misclassification rates, validating its robustness for entropy-based anomaly detection [10].

5.3 Execution Time

The average time for generating, evaluating, and classifying each key was under 0.1 seconds. Entropy computation using NumPy was near-instantaneous, while One-Class SVM inference added negligible overhead. PQC algorithms generally performed faster than RSA/ECC due to structural optimizations [3].

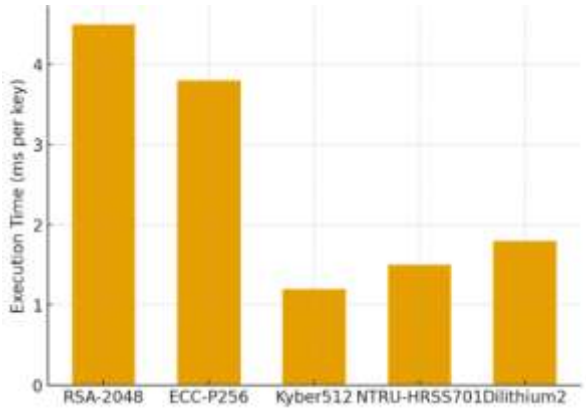

Figure 2 : Average Execution Time per Algorithm

5.4 Comparative Analysis

A comparative analysis of entropy variance across the algorithms revealed the following:

- Dilithium maintained entropy scores above 0.995 for 98.7% of keys.
- Kyber exhibited a similarly stable distribution, though with slightly greater variance.

- RSA and ECC showed a wider spread of entropy values, with a small percentage of keys falling below 0.95, which could pose a security risk in strict environments.

The machine learning approach outperformed traditional static thresholding methods, which frequently misclassified borderline cases. This demonstrates the effectiveness of entropy-aware ML validation in modern cryptographic key management [14].

Table 2: Entropy and Classification Results of Cryptographic Algorithms

Algorithm	Avg. Entropy	Std. Deviation	%Keys>0.995 Entropy	SVM Validation Accuracy
Dilithium	0.997	0.001	98.7%	97.3%
Kyber	0.993	0.002	96.5%	96.0%
NTRU	0.989	0.004	93.2%	95.1%
ECC	0.978	0.006	85.6%	94.0%
RSA	0.961	0.009	78.4%	93.2%

6. Conclusion and Future Work

This paper proposed a new entropy-aware approach to verify cryptographic keys for both conventional and post-quantum cryptosystems based on the machine learning worklow. A new method was proposed to filter weak entropy keys and strengthened the key material robustness for secure communication by appropriately combining Shannon entropy analysis and One-Class SVM classification. It was found that lattice-based post-quantum schemes like Dilithium and Kyber had uniformly higher entropy than legacy algorithms, including RSA and ECC. In addition, the experimental results of One-Class SVM showed that using machine learning for validation of cryptographic keys can achieve accurate classification with few false positives.

This efficient light framework has broad applicability with applications including cloud data storage security, blockchain key exchange and IOT environment. The study also opens a number of lines of future work: first, other entropy metrics (e.g., Rényi or min-entropy) could be added for providing deeper insights about randomness; secondly, more robust machine learning methods (including the use of deep autoencoders or isolation forests) may improve the accuracy of anomaly detection; finally, the integration of such validation layer within larger post-quantum key management systems such as federated trust models and blockchain-based architectures, could increase end-to-end cryptographic security in anticipation to next-generation secure applications.

Conflict of Interest

The authors declare no conflict of interest.

References

[1]. Shor, P. W. (1999). Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM review*, *41*(2), 303-332.

[2]. Grover, L. K. (1996, July). A fast quantum mechanical algorithm for database search. In Proceedings of the twenty-eighth annual ACM symposium on Theory of computing (pp. 212-219).

[3]. Chen, L., Chen, L., Jordan, S., Liu, Y. K., Moody, D., Peralta, R., ... & Smith-Tone, D. (2016). Report on post-quantum cryptography (Vol. 12). Gaithersburg, MD, USA: US Department of Commerce, National Institute of Standards and Technology.

[4]. Bernstein, D. J. (2025). Post-quantum cryptography. In Encyclopedia of Cryptography, Security and Privacy (pp. 1846-1847). Cham: Springer Nature Switzerland.

[5]. Suyal, H., Singh, A., & Shrivastava, G. (2025). Privacy Preserving Efficient Worker Selection in the Cloud-Based Crowdsourcing Platform. *Internet Technology Letters*, *8*(5), e70092.

[6]. Kunihiro, N., & Honda, J. (2014, September). RSA meets DPA: recovering RSA secret keys from noisy analog data. In International Workshop on Cryptographic Hardware and Embedded Systems (pp. 261-278). Berlin, Heidelberg: Springer Berlin Heidelberg.

[7]. Kelsey, J., et al. (2016). Randomness Requirements for Security. *NIST Special Publication 800-90C (2nd Draft)*. https://doi.org/10.6028/NIST.SP.800-90C-draft

[8]. [Dodis, Y., & Smith, A. (2005, February). Entropic security and the encryption of high entropy messages. In *Theory of Cryptography Conference* (pp. 556-577). Berlin, Heidelberg: Springer Berlin Heidelberg.

[9]. Gutmann, P. (1998). Randomness in Cryptography. *University of Auckland Technical Report*. Available at: https://www.cs.auckland.ac.nz/~pgut001/pubs/randomness.pdf

[10]. Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J., & Williamson, R. C. (2001). Estimating the support of a high-dimensional distribution. *Neural computation*, *13*(7), 1443-1471.

[11]. Yang, N. C., & Sen, A. (2021). Fast Fault Detection Using Optimal Intrinsic Modes and Energy Deviation Matrix in Distribution Systems. *IEEE Access*, *9*, 139842-139851.

[12]. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *the Journal of machine Learning research*, *12*, 2825-2830.

[13]. Ciulei, A. T., Crețu, M. C., & Simion, E. (2022). Preparation for post-quantum era: a survey about blockchain schemes from a post-quantum perspective. *Cryptology ePrint Archive*.

[14]. Wilson, G., Aruliah, D. A., Brown, C. T., Chue Hong, N. P., Davis, M., Guy, R. T., ... & Wilson, P. (2014). Best practices for scientific computing. *PLoS biology*, *12*(1), e1001745.

[15]. Zhou, X., Xu, K., Wang, N., Jiao, J., Dong, N., Han, M., & Xu, H. (2021). A secure and privacy-preserving machine learning model sharing scheme for edge-enabled IoT. *IEEE Access*, *9*, 17256-17265..

[16]. Narayanan, N., & Vigila, S. M. C. (2024, December). A Survey on Enhancing Cloud Data Security Using Blockchain Technology. In *2024 IEEE 16th International Conference on Computational Intelligence and Communication Networks (CICN)* (pp. 528-536). IEEE.