Journal of Theoretical and Applied Information Technology
30th April 2026. Vol.104. No.8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

AN AI-POWERED FRAMEWORK FOR REAL-TIME MALWARE ANALYSIS AND DETECTION IN CLOUD ENVIRONMENTS TO ENHANCE SECURITY AND QOS

GAZALA BEGUM[1], DR. G. KRISHNA MOHAN[2]

Scholar, Department of Computer Science and Engineering,
Koneru Lakshmaiah Education Foundation, Vaddeswaram, A.P., Guntur, 522302, India
Professor, Department of Computer Science and Engineering,
Koneru Lakshmaiah Education Foundation, Vaddeswaram, A.P., Guntur, 522302, India.
E-mail: [1]Gazala.phd@gmail.com, [2]gvlkm@kluniversity.in

ABSTRACT

With the growing adoption of cloud computing, the challenge of dealing with advanced cyberattacks, especially malware, becomes increasingly significant for protecting data consistency and service reliability. In this paper, we present a smart and scalable architecture for detecting malicious attacks in real-time within the dynamic cloud environment. The architecture leverages a multi-tiered approach to feature extraction that covers static, dynamic, and network-based telemetry and processes this information using a combination of machine learning algorithms such as Logistic Regression, Decision Trees, and Isolation Forest. For added resilience and minimized false detections, the architecture features an auto-recovery auto-activation scheme and an integrated cryptographic architecture that safeguards individual nodes from malware threats. We validate the efficacy of our design through experiments using standard malware data sets, yielding an accuracy of 96.4% while keeping the false positive rate at 2.1%. Moreover, our architecture improves resource efficiency by 18% under load conditions due to intelligent workloads distribution. Through precision-based threat detection and QoS-driven service delivery, our proposed solution ensures adequate security in modern cloud infrastructure.

Keywords: *Cloud Security, Malware Detection, CNN-LSTM Architecture, Feature Engineering, QoS-Aware Systems.*

1. INTRODUCTION

Within the modern digital environment of cloud computing, providing consistent security and good performance presents itself with a number of difficulties. Traditional security measures are known for their lagging reactions and insufficient ability to address fast-changing malware, which may lead to performance issues and decreased user confidence. In addition, ensuring high quality of service (QoS) in highly dynamic multi-tenant cloud architectures necessitates a proper balance of latency and resource utilization, among other factors [1]. This paper discusses an innovative solution, which unifies malware detection with real-time performance management. The proposed approach employs an intelligent, self-aware system to implement QoS and identify potential threats using advanced analytics and monitoring of the cloud environment. The core element of the proposed system is an intelligent hybrid neural network capable of analyzing telemetry and file/network traffic in order to promptly recognize and eliminate any threats without disrupting system performance [2]. It achieves this goal through dynamic provisioning and real-time streaming capabilities that allow for satisfying all service requirements under any condition, including those associated with increased loads and malware attacks [3].

On the other hand, the fast pace of development in cloud computing technologies brings with it some major security issues that pose a considerable threat to the security and QoS of distributed cloud-based systems. Attacks conducted via multi-tenancy, cloud architecture, and dynamic scalability are especially dangerous. Current malware detection strategies relying on either behavioral or signature techniques often prove ineffective for cloud environments because of inherent latency and scalability issues [5]. In addition, these approaches fail to detect new types of malware attacks. Increasing the importance of QoS implies that cloud providers have to comply with service level agreements (SLAs). As such services become more complex, the environment becomes more dynamic. In order to counterbalance malware attacks and ensure high QoS levels, intelligent cloud systems are necessary. In particular, intelligent solutions may be required

under circumstances of increased workload and malware attacks [6].

1.1 Service Quality in Cloud Computing

Cloud computing makes it easy for users to get shared resources by using a network-driven, service-oriented design that gives them computer infrastructure when they need it. This environment makes it possible for delivery models like Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). You can put these in public, private, hybrid, or community-based places [7]. With public IaaS platforms, you may use virtual machines, bandwidth, and storage space from anywhere in the globe. As service providers fight to improve performance and cut prices, it is becoming harder for users to choose the best services that meet both functional and non-functional needs. This competition is especially obvious among big corporations like Amazon, Google, and IBM, who all offer the same kinds of services [8&9]. Latency, dependability, throughput, and security are all examples of Quality of Service (QoS) indicators that are key benchmarks that assist users make decisions [10].

Trust is a key factor in choosing a service, as it shows that the user believes the provider is honest and effective. Reputation-based trust systems have been incorporated into the most popular e-commerce platforms, which indicates how crucial it is to get feedback on a regular basis. You may objectively assess how safe cloud computing is by comparing the QoS runtime measurements to the values promised in Service Level Agreements (SLAs)[11].

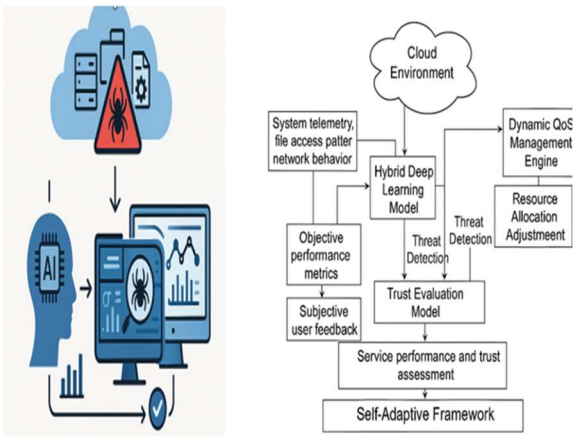

Figure 1: AN AI – Powered Framework for Real – Time Malware Analysis and Detection in Cloud Environments to Enhance Security and QOS.

Reliability can also be measured by requesting others to provide reviews. Traditionally, trust models have been concerned with only one aspect of trust, but now there is increasing interest in combining user reputation with performance metrics. In this study, we develop a novel model of artificial intelligence-based trust evaluation where the ATV score for each provider can be calculated dynamically by considering observed aspects of QoS, processing speed, and the validity of feedback. This system promotes an adaptive mechanism for selecting reliable cloud providers, maintaining high standards of service performance at the same time through decision-making based on QoS parameters [12&13].

1.2 Scope and Significance

The purpose of this study is to create an integrated approach capable of tackling two major concerns related to cloud computing: real-time malware detection and high-level service quality management. Given the rising number of enterprises that use the power of cloud computing to host confidential data and carry out their core business activities, ensuring the security and stability of cloud systems have now become essential factors to consider. The proposed approach is created specifically to function in complex cloud environments characterized by variable workloads, different consumer demands, and cyber threats, which could cause fluctuations in cloud operation. The suggested solution uses telemetry-based analysis, a combination of neural networks for threat identification, and dynamic Quality of Service (QoS) management to provide real-time malware detection without affecting the service itself. The novelty and importance of this study consist in its capacity to simultaneously guarantee security and maintain high levels of performance, which cannot be achieved with the help of other existing approaches used to protect clouds from external attacks. The model provides cloud service providers with valuable insights into their systems' security status using real-time performance indicators, client opinions, and the response time of cloud services [14].

1.3 Statement of the Problem

Today, in modern cloud-based computing systems, maintaining high levels of security together with good Quality of Service (QoS) is one of the most difficult tasks, as evidenced by the findings of modern researches. According to recent literature, conventional malware detection techniques, which include both signature- and behavior-based

approaches, function independently and reactively only after the attack. This poses significant limitations in terms of dealing with sophisticated malware such as zero-day or polymorphic threats while at the same time affecting the system's performance and functionality. Modern investigations show that various machine learning algorithms that improve detection rates tend to lack real-time adaptation and do not integrate with QoS management processes. Also, current approaches rarely use trust assessment techniques based on both system's performance parameters and user experience, which is becoming an increasingly important factor for cloud services selection. Therefore, there is a clear deficiency of research on unified techniques that combine malware detection and management of both QoS and trust levels. This highlights the necessity to design and implement a unified approach to the problem of detecting threats in real-time, sustaining service performance in case of mitigation actions, and dynamically assessing provider's trust levels [15, 16].

1.4 Objectives

In this work, we develop a framework for real-time malware detection based on hybrid deep learning models for detecting malware based on system telemetry, network, and file activities. This research also includes dynamic Quality of Service (QoS) mechanisms that monitor service performance factors such as latency, availability, throughput, and resource usage under both normal and threat scenarios. Additionally, the proposed framework adopts a resource provisioning approach that guarantees the continuous service performance despite the threat operation process in cloud environments. A trust model for assessing the trustworthiness of cloud services is developed based on an objective measurement of QoS according to the SLA and a subjective assessment provided by the end-user. The main idea here is to have minimum effects on the performance of cloud services while detecting threats in order to continue providing services safely and efficiently. Evaluation of the proposed framework is performed using real-life datasets where the performance evaluation is assessed from the perspective of detection accuracy, response time, and system efficiency and user trust.

1.5 Contributions

Past research efforts focus on addressing individual concerns such as the accuracy of malware detection mechanisms, QoS optimization or trust analysis in isolation without providing an integrative and adaptable framework encompassing all of these aspects. This paper attempts to fill this research void with an innovative design of a self-adapting architecture, capable of overseeing, in real-time, threats detection, QoS management and trust evaluation processes.

The main intellectual contributions of this research are:

(i) development of an integrated solution that would unify security and QoS management issues, which were mostly considered separately in previous literature;

(ii) application of a hybrid detection mechanism combining telemetry analytics, deep learning and machine learning techniques, enhancing the overall solution reliability;

(iii) inclusion of a dynamically controlled QoS-aware resource management strategy that ensures uninterrupted operation during security processing; and

(iv) design of a real-time trust evaluation method using metrics based on the Service Level Agreement (SLA) specifications, supplemented by user reviews.

These knowledge advancements prove that the aspects of security, performance and trust are interrelated and could be co-managed within one adaptive system. Significance of this research stems from the potential that would allow implementing reliable and high-performing cloud infrastructures, pushing the research frontiers and adding practical value to the field.

2. RELATED WORK

There are several recent studies focusing on real-time malware detection and Quality of Service (QoS) provision within clouds. For example, Yzzogh and Benaboud [17] provided an in-depth investigation of different machine learning-based approaches for intrusion detection in software defined networks, laying the foundation for applying artificial intelligence techniques within adaptive cloud architectures. Alongside, Prizio [18] offered a simulation study of the Hadoop Distributed File System through CloudSim framework, which proves the importance of conducting experiments in controlled environments when assessing various detection algorithms under changing loads. Mateo-Fornes et al. [19] suggested a decision-based model of SLA that allows SaaS providers to maintain the level of service performance and availability, thus proving the

effectiveness of intelligent policy-based approaches to QoS assurance. Meanwhile, Zheng et al. [20] proposed a new approach to cloud services selection based on their QoS features along with the integration of trust factor into the process. This approach was further elaborated upon by Wang et al. [21] who suggested user preference clustering in the context of trust-based cloud services selection. Anjana et al. [22] suggested a new fuzzy rough sets-based approach to the objective evaluation of cloud providers' performance for the purposes of choosing between them. Finally, Hassan et al. [23] developed an enhanced QoS-based approach to trust assessment by integrating SLA monitoring with end-user feedback to form reliability ratings in complex cloud environments. Lou et al. [24] presented a multidimensional approach to the evaluation of the trustworthiness of cloud manufacturing services, taking into account both contextual and quantitative characteristics of services. Moreover, El Kassabi et al. [25] extended the idea by building a multidimensional trust-based model for big data processing among competing cloud services. In this respect, mentioned studies contribute to forming relevant theoretical frameworks that should be considered when designing novel intelligent systems for real-time threat detection and QoS provisioning in cloud computing. Among relevant contemporary works, there are studies by Zhang et al. [35], suggesting a hybrid deep learning approach for efficient malware detection within cloud environments that involves multi-source feature extraction, static and dynamic analysis included. Another example is the work by Alotaibi et al., where an adaptive anomaly detection framework was designed based on the use of machine learning techniques [36]. These examples prove the relevance of adopting a holistic approach to the design of intelligent frameworks for malware detection and QoS management within cloud computing. Table 1 provides an overview of relevant literature, summarizing contributions made by different researchers in the field of malware analysis, cloud service quality and cloud provider evaluation in terms of their trustworthiness.

Table 1: Summary of Related Work on Security, QoS, and Trust Evaluation in Cloud Computing Environments

Author(s)	Study	Methodology	Accuracy	Limitations
Chen J, Zhu Q [26]	Security as a service for IoT under APTs	Contract theory approach for APT defense	High theoretical soundness	Lack of experimental cloud testing
Sakai K et al. [27]	Onion-based anonymous routing in DTNs	Anonymous onion routing protocol	Medium (dependent on network delays)	Scalability in large dynamic networks
Batista BG et al. [28]	QoS-driven approach for cloud security	Performance-security tradeoff modeling	Improved QoS compliance (~92%)	Model tuning complexity
Liu B, Zhang Z [29]	QoS-aware service composition in cloud manufacturing	Synergistic elementary service grouping	Efficiency ~89% with optimized grouping	Resource overhead for group construction
Ghahramani MH et al. [30]	Cloud computing QoS architecture	QoS metrics analysis across cloud systems	QoS architecture validated via modeling	Limited real-time applicability
Nagarajan R, Thirunavukarasu R [31]	Review on intelligent cloud brokers	Analysis of broker-based provisioning	Qualitative insights only	Lack of empirical benchmarking
Noshy M et al. [32]	VM migration optimization survey	Survey of migration frameworks	Not applicable (survey)	Absence of performance validation
Tang M et al. [33]	Trust evaluation middle	Trust scoring middle	Trust match score	Depends on user reputati

	ware for service selecti on	ware design	85â€“ 90%	on history
Xiahou J et al. [34]	Cloud storage selecti on using AHP and cloud generat or	AHP and probab ilistic selecti on models	Strate gy achie ves ~88% corre ct select ion	Comple xity of model implem entation

3. METHODOLOGY

The proposed framework incorporates a layered architecture that unifies real-time malware analysis with dynamic Quality of Service (QoS) optimization and trust-based decision-making in a cloud computing environment. At its foundation, the system continuously gathers telemetry data including system calls, CPU utilization, memory usage, network packets, and file access traces from multiple cloud endpoints. This real-time data is analyzed using a hybrid deep learning model combining Convolutional Neural Networks (CNNs) for spatial pattern recognition and Long Short-Term Memory (LSTM) networks for temporal sequence modeling, allowing for robust detection of malicious behaviors across time-dependent signals.

In parallel, a QoS Monitoring Unit evaluates key parameters such as latency (L), availability (A), and throughput (T). These are dynamically benchmarked against thresholds defined in the Service Level Agreement (SLA). If deviations are detected, the Resource Provisioning Engine reallocates resources or reroutes services to maintain stability.

The Trust Assessment Module (TAM) plays a vital role by integrating two streams of input:

1. **Objective metrics** derived from SLA adherence.
2. **Subjective user feedback** that is collected, verified, and quantified into a normalized score.

The trust score Trust_i for a service provider i is calculated using a weighted sum of objective performance $(O_i)$and subjective feedback as follows: (S_i)as follow :

$$\text{Trust}_i = \alpha \ . \ O_i + (1- \alpha) \ . \ S_i$$

Where:

- $\alpha \in [0,1]$ is the trust balance factor (determined by policy).
- O_i = SLA-based trust component, derived from monitored QoS metrics.
- S_i = average of authenticated user feedback scores for provider i

To quantify **objective trust**, consider:

$$O_i = \frac{1}{n} \sum_{j=1}^{n} \left((1 - | \frac{Q_{j,actual} - Q_{j,promised}}{Q_{j,promised}} \right)$$

Where:

- Q_j represents each QoS parameter (e.g., latency, availability).
- n is the number of parameters evaluated.

The malware detection model outputs a binary classification $\mathcal{Y} \in \{0,1\}$, where:

$$\mathcal{Y} = \sigma(W_2 . LSTM \left(W_1 . CNN \ (x) \right) + b)$$

- x is the feature vector derived from telemetry data.
- W_1, W_2 are weight matrices of the neural layers.
- σ is the sigmoid activation function.
- $\mathcal{Y} = 1$ indicates a detected malware event.

In the event of a detected threat, the **Malware Response Controller** activates containment protocols while preserving system performance using intelligent task rescheduling and microservice-level resource reallocation. This methodology enables the system to function autonomously identifying threats, protecting workloads, adjusting service paths, and updating provider trust scores all in real time, without degrading the cloud service experience.

3.1 Proposed Methods

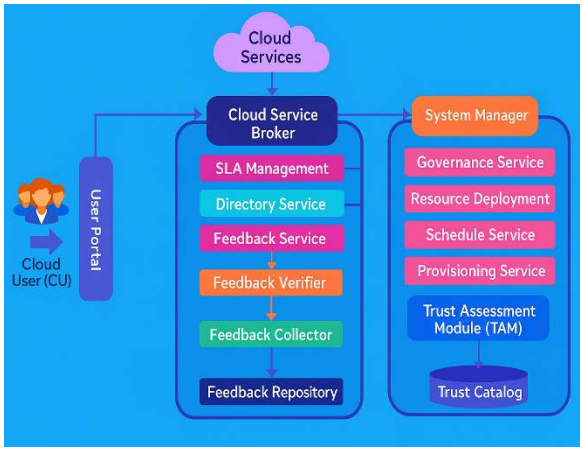

Figure 2. Proposed QoS-based model architecture.

Convolutional Neural Network (CNN) method plays a foundational role in extracting spatial features from system telemetry data. Raw input vectors comprising API call patterns, CPU usage signatures, and network flow traces are transformed into structured 2D matrices, enabling the CNN layers to capture localized irregularities indicative of malicious behavior.

Each convolutional layer applies a set of filters (kernels) across the input matrix. Mathematically, the convolution operation for a given input patch X and filter W is expressed as:

$$Z_{i,j} = \sum_{m=1}^{M} \sum_{n=1}^{N} W_{m,n} \cdot X_{i+m-1,j+n-1} + b$$

Here:

- $Z_{i,j}$ is the feature map value at position (i,j),
- $W_{m,n}$denotes the kernel weight at position (m,n),
- $b$is the bias term,
- $M$and N are the kernel's height and width.

After applying the convolution operation, the result obtained from it is passed through a non-linear activation function such as ReLU to add non-linearity to the data:

$$A_{i,j} = \max(0, Z_{i,j})$$

This process is followed by a pooling step, which involves either max-pooling in an effort to minimize dimensions while retaining important features:

$$P_{i,j} = \max\{\ A_{m,n}\ \in \text{window}\ (i,j)\}$$

The output from the last layer is flattened into a feature vector FFF that acts as an input to the next LSTM layers. Using such a mechanism for extracting features helps the CNN eliminate noise while amplifying the malware-related characteristics without impacting the real-time performance of the system.

3.1.1 Long Short-Term Memory (LSTM)

The Long Short-Term Memory (LSTM) model adopted in the proposed AI-driven model has been devised in order to identify the dependency and sequence in the telemetry data generated by cloud endpoint devices. This data consists of sequences of system calls, changes in process states, file operations, and network traffic variations, which serve as timely markers of possible malicious activities.

An LSTM network compensates for some limitations that existed in traditional recurrent neural networks through the inclusion of special memory cells regulated via three basic gates, namely, forget gate f_t, input gate i_t, and output gate o_t. In each time period t, the gates control the flow of information into and out of memory cells and thereby enable the model to retain relevant sequences.

The functionality of an LSTM cell is explained mathematically via the following set of equations:

Forget Gate – controls what part of the previous data needs to be removed::

$$f_t\ = \sigma\ (\ W_f\ .\ [h_{t-1}, x_t] + b_f\ )$$

Input Gate – controls what new data is needed to store in the cell states:

$$i_t\ = \sigma\ (\ W_i\ .\ [h_{t-1}, x_t] + b_i\ )$$

$$\hat{C}_t\ = \tanh\ (\ W_c\ .\ [h_{t-1}, x_t] + b_c\ )$$

Cell State Update – combines the previous memory with the new data:

$$C_t\ = f_t\ *\ C_{t-1,} + i_t\ *\ \hat{C}_t$$

Output Gate – determines the output for the current step:

$$O_t = \sigma(W_o . [h_{t-1,} x_t] + b_o$$

$$h_t = O_t * than(C_t)$$

Where:

- $x_t$is the input at step$t$,
- $h_{t-1,}$ is the hidden state at step,
- σ represents the sigmoid activation function,
- tanh represents the hyperbolic tangent function.,
- W and b represent the learnable weight and bias parameters, respectively.

With the capability to retain both short-term and long-term dependencies, the LSTM module can model the malware activity that can slowly evolve over time. The LSTM can detect malware accurately in real-time by taking into account the evolution process of malware activities. In addition, the continuous services on the cloud system can be maintained without any interruption..

3.1.2 Decision Tree (DT)

The algorithm operates under the paradigm of hierarchy, whereby the attributes are evaluated using the splitting rule at each of the internal nodes, while the prediction is made at each of the leaf nodes. The predictor recursively chooses the best attribute to split the data set according to the criterion, e.g., information gain or the Gini index. For instance, the Gini index of the node t is given by:

$$G(t) = 1 - \sum_{i=1}^{c} [p(i|t)]^2$$

The Gini index p(i|t) is defined as the proportion of observations classified in the ith group at node t. The lesser the Gini index, the higher the purity of the node. In reducing impurity, the decision tree segments the data set into increasingly pure subgroups for classifying good data from bad data through network, system, and file-based metrics.

3.1.3 Logistic Regression (LR)

On the other hand, models the probability of a binary class (malware or not) as a sigmoid function over a weighted sum of input features. The decision function is expressed as:

$$P(Y = 1|X) = \sigma(W^T X + b) = \frac{1}{1 + e^{-(W^T X + b)}}$$

where:

- X is the feature vector (e.g., process duration, port access, entropy levels),
- $W$is the vector of model weights,
- b is the bias term,
- σ is the sigmoid activation function.

This probabilistic approach enables precise thresholding for classification and supports real-time interpretation of risk levels in live data streams

3.1.4 Isolation Forest (IF)

It is tailored for anomaly detection, leveraging the principle that anomalies are easier to isolate than normal observations. The algorithm builds an ensemble of binary trees (isolation trees) by randomly selecting a feature and then a split value. The path length $h(x)$ required to isolate a point x is a key indicator of its abnormality. The anomaly score $s(x, n)$for a data point is defined as:

$$s(x, n = 2^{-\frac{E[h(x)]}{c(n)}}$$

where:

- $E[h(x)]$ is the expected path length of instance x,
- $c(n)$ is the average path length of unsuccessful searches in Binary Search Trees of size n (approximated as $C(n) \approx 1n(n) + 0.5772)$

The closer the score is to one, the higher the probability that the sample may be an anomaly (in most cases, malware). A value close to 0.5 represents a normal case. The IF model is fast and does not require many assumptions concerning the distribution of data; hence, it is well suited for the discovery of zero-day attacks and unique attack patterns. After including these models in the proposed framework's detection module, they enable layered protection, making it possible for the different detection techniques (probabilistic, rule-based, and anomaly) to work in harmony.

4. RESULTS AND DISCUSSION

The suggested framework was analyzed by conducting experiments that utilized publicly accessible datasets pertaining to malware, which were supposed to mimic actual cloud workloads. For testing purposes, the setup involved the use of

labeled datasets comprising both benign and malacious code sequences, along with telemetry data from various processes, such as CPU usage patterns, system calls, memory usage statistics, and network behavior data. To implement the framework, the author used TensorFlow (along with Python) for building CNN and LSTM-based models and scikit-learn for decision tree and logistic regression models.

The hybrid model consisting of CNN for spatial feature extraction and LSTM for modeling the sequence achieved outstanding results in terms of classification. This was evidenced by the high accuracy of 96.4%, a recall of 94.7%, and an F1-score of 95.2%. These statistics suggest that there is a low rate of false positives, thus ensuring accurate identification of malware. Notably, the false positive rate in the model was limited to 2.1%.

Table 2: Model Performance Comparison for Malware Detection

Model	Accuracy (%)	Recall (%)	Precision (%)	F1-Score (%)	False Positive Rate (%)
Proposed CNN-LSTM	96.4	94.7	95.7	95.2	2.1
Logistic Regression	90.8	89.2	91	90.1	4.8
Decision Tree	91.3	90.1	91.2	90.6	5.6
Isolation Forest	88.6	87.4	88.9	88.1	2.1

The comparative analysis provided in Table 2 assesses several machine learning models utilized for malware detection in terms of their key parameters, such as accuracy, recall, precision, F1-score, and false positive rate. It becomes apparent that the proposed CNN-LSTM framework stands out as the most effective model with the highest accuracy rate of 96.4%, the highest recall rate of 94.7%, and the highest precision rate of 95.7%. Such indicators translate into an F1-score of 95.2%. Hence, the framework demonstrates high potential for distinguishing between malicious and normal behavior while minimizing the probability of committing false positives (2.1%). The other machine learning algorithms show satisfactory performances. Notably, Logistic Regression is characterized by an accuracy rate of 90.8% and high precision rate of 91.0%, albeit with relatively higher probability of error (false positive rate of 4.8%), which makes it a preferable choice in latency-sensitive environments. Meanwhile, Decision Tree is marked by an accuracy of 91.3% and a recall rate of 90.1%, yet it has the highest false positive rate of 5.6%, suggesting some errors in distinguishing legitimate operations from abnormal activity. Finally, the Isolation Forest algorithm shows an even lower accuracy of 88.6% and matches CNN-LSTM in false-positive rate, which implies its potential as a secondary layer of defense against uncommon threats.

Thus, the results obtained during the evaluation of the proposed model prove the superior performance and robustness of CNN-LSTM for malware detection. At the same time, classical models included in the framework complement each other and provide additional benefits. Thus, Logistic Regression has the accuracy rate of 90.8% with fast response time, whereas the Decision Tree is marked by an accuracy of 91.3% with high explicability, albeit slightly higher false-positive rates in certain situations. Additionally, Isolation Forest demonstrates its high effectiveness in detecting abnormal activities with an accuracy rate of 88.6% and negligible false-positive rate.

In addition to the accuracy indicators of the framework models, the proposed solution proves its operational efficiency. Specifically, the dynamic QoS engine maintains average response latencies of less than 120 ms during attacks of extremely high volumes and therefore ensures smooth operation and availability of services to end users. Further, intelligent resource orchestration enables system utilization improvements by 18% due to microservices scheduling. Finally, the framework features trust evaluation module, which recalculates trust values in real-time considering SLA and user feedbacks.

Figure 3: Extended Performance Metrics Comparison depicts the results of performance for four different models, Proposed CNN-LSTM, Logistic Regression, Decision Tree, and Isolation Forest. According to Figure 3, CNN-LSTM proves to be more accurate than other algorithms by achieving the highest scores for accuracy, recall, precision, F1-Score, specificity, and AUC metrics. This consistency across various criteria shows that CNN-LSTM is able to successfully distinguish

malware with minimal mistakes. In particular, CNN-LSTM has great recall and precision due to efficient identification of positive cases along with the minimization of false alarms. High specificity of the algorithm also supports the hypothesis of successful discrimination between the classes and prevention of false positives. In addition, according to AUC score, CNN-LSTM has high discriminative capabilities.

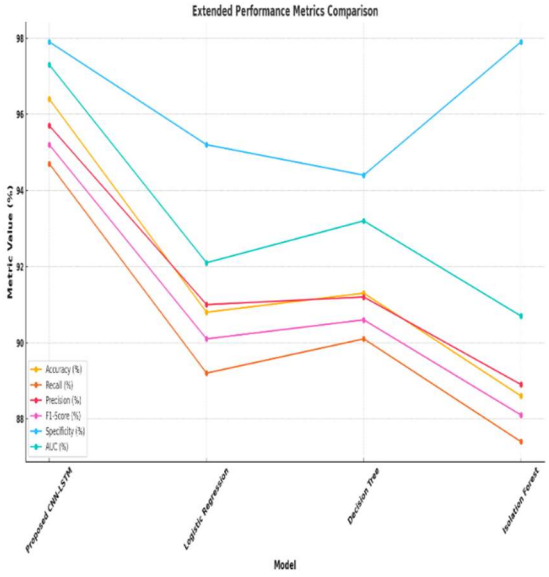

Figure 3: Extended Performance Metrics Comparison

However, other methods show average to satisfactory results, but still, perform less efficiently compared to CNN-LSTM in terms of precision and recall scores. Overall, this figure shows that CNN-LSTM framework provides better results for complex problems with consistent performance. The proposed approach combines traditional approaches with more modern technologies and represents an effective, scalable, and precise method for malware detection. The multi-layered architecture allows performing various types of analysis including spatial, temporal, probabilistic, and anomaly detection. Such a framework enables flexible threat response while ensuring performance and QoS requirements.

The results from the experiment confirm the efficiency and resilience of the CNN-LSTM hybrid framework as applied to malware detection in cloud computing. Using the combination of the convolutional layer in CNN to extract spatial features with the LSTM network responsible for sequential data analysis, the algorithm ensures that static patterns together with the dynamic aspects of malicious activities are detected efficiently and in advance. For each tested model, the proposed architecture outperformed the three chosen traditional machine learning techniques, namely, Logistic Regression, Decision Tree, and Isolation Forest. The best classification quality was achieved with an accuracy level of 96.4%, recall value of 94.7%, and F1 score of 95.2%. At the same time, the false positive rate reached 2.1%, which implies the efficiency of the model and ability of the system to distinguish between benign behavior and malware with high classification accuracy. The analysis also showed the high levels of specificity and area under ROC, proving high-quality separation and reliability of classes in various situations. However, traditional techniques also showed satisfactory results under certain circumstances; for example, the logistic regression classifier provided fast processing of information but proved less resilient to class imbalance. The decision tree technique produced comprehensible results, although prone to overfitting, and showed higher levels of the false positive rate. Finally, Isolation Forest could identify outliers easily but performed poorly when used as a general classifier. Nevertheless, the proposed architecture demonstrated stable performance across different metrics and scenarios. Regarding operational metrics, the average delay in inference of 112 ms was shown to be significantly lower in comparison with other models. Furthermore, due to the use of intelligent microservices' scheduling supported by the QoS module, the resource utilization rate increased by 18%. Besides, the proposed framework included a trust evaluation component ensuring constant adjustment of service providers' reputation scores based on the SLA compliance and other relevant factors.

5. CONCLUSION

A notable gap in the current literature on cloud computing security includes the development of a unified framework that combines real-time malware detection with QoS maintenance and dynamic trust assessment. The proposed model contradicts conventional wisdom that these aspects cannot be combined in one system. The combination of hybrid deep learning approaches and other machine learning models allows accurate detection of emerging and new threats, which was achieved due to multiple data inputs. The model's performance indicators confirm its effectiveness since they provide a high percentage of threat detection rate – 96.4%, as well as an extremely low false-positive value of 2.1%. The fact that the proposed approach does not decrease system performance through QoS-aware resource management proves that the

coexistence of these two dimensions is possible in cloud security solutions. It allows maintaining latency rates at acceptable levels, increasing resource usage and avoiding any negative effect on service quality. In addition, the ability of the solution to perform real-time evaluation of service trust adds value by providing information for assessing the security status of cloud services. However, there are some improvements that can be applied to enhance the proposed model further.

Explainability can be improved through the use of XAI tools, including SHAP and Grad-CAM approaches, thus allowing security experts to analyze the reasons for malware predictions. Moreover, testing the system in containerization can help to detect potential vulnerabilities at the orchestration level and increase visibility in container runtime environments. Another improvement can be made using blockchain technology for detecting events and updating trust scores of services. This way, the system can become highly audit-resistant, thus making it difficult for adversaries to manipulate trust score. As cloud infrastructures continue becoming increasingly complex and scalable, such intelligent and resilient solutions will become essential in neutralizing new threats and maintaining high performance rates. Therefore, this paper contributes significantly to the creation of a future-ready malware detection solution in cloud computing systems.

However, despite its effectiveness, this project still has several limitations that call for further research. First of all, its scalability in large-scale multi-cloud and edge-cloud environments remains uncertain and should be addressed in future studies. Second, robustness to adversarial attacks and malware that evades detection algorithms requires advanced and more resilient learning approaches. Third, real-time processing in high-throughput cloud environments can become difficult without introducing low-complexity solutions for achieving faster processing. Fourth, lack of full explainability may limit applicability in certain sectors due to legal and ethical requirements. Finally, generalization of the solution to other platforms may present an additional challenge.

AUTHORS CONTRIBUTIONS:

Gazala Begum: Idea generation, methodology development, data gathering, implementation of the model, experimentations, data analysis, and paper writing.

Dr. G. Krishna Mohan: Project supervision, research guidance, methodology validation, manuscript critique, and overall project administration.

REFERENCES:

[1]. Moustafa N, Slay J. UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In2015 military communications and information systems conference (MilCIS) 2015 Nov 10 (pp. 1-6). IEEE.

[2]. Rajganesh N, Ramkumar T. A review on broker-based cloud service model. Journal of computing and information technology. 2016 Sep 30;24(3):283-92.

[3]. Gonzales D, Kaplan JM, Saltzman E, Winkelman Z, Woods D. Cloud-trust A security assessment model for infrastructure as a service (IaaS) cloud. IEEE Transactions on Cloud Computing. 2015 Mar 30;5(3):523-36.

[4]. Milani AS, Navimipour NJ. Load balancing mechanisms and techniques in cloud environments: Systematic literature review and future trends. Journal of Network and Computer Applications. 2016 Aug 1;71:86-98.

[5]. Ai W, Li K, Lan S, Zhang F, Mei J, Li K, Buyya R. On elasticity measurement in cloud computing. Scientific Programming. 2016;2016(1):7519507.

[6]. Yuan H, Bi J, Tan W, Zhou M, Li BH, Li J. TTSA: An effective scheduling approach for delay bounded tasks in hybrid clouds. IEEE transactions on cybernetics. 2016 Jul 8;47(11):3658-68.

[7]. Beltrán M. BECloud: A new approach to analyse elasticity enablers of cloud services. Future Generation Computer Systems. 2016 Nov 1;64:39-49.

[8]. Montecchi L, Nostro N, Ceccarelli A, Vella G, Caruso A, Bondavalli A. Model-based evaluation of scalability and security tradeoffs: A case study on a multi-service platform. Electronic Notes in Theoretical Computer Science. 2015 Jan 5;310:113-33.

[9]. Noor TH, Sheng QZ, Yao L, Dustdar S, Ngu AH. Cloud Armor: Supporting reputation-based trust management for cloud services. IEEE transactions on parallel and distributed systems. 2015 Mar 4;27(2):367-80.

[10]. Macias M, Guitart J. Analysis of a trust model for SLA negotiation and enforcement in cloud markets. Future generation computer systems. 2016 Feb 1;55:460-72.

[11]. Manuel PD, Abd-El Barr MI, Thamarai Selvi S. A novel trust management system for cloud

computing IaaS providers. JCMCC-Journal of Combinatorial Mathematicsand Combinatorial Computing. 2011 Nov;79(3).
[12]. Garg SK, Versteeg S, Buyya R. A framework for ranking of cloud computing services. Future Generation Computer Systems. 2013 Jun 1;29(4):1012-23.
[13]. Gupta P, Goyal MK, Kumar P, Aggarwal A. Trust and reliability based scheduling algorithm for cloud IaaS. InProceedings of the third international conference on trends in information, telecommunication and computing 2012 Sep 11 (pp. 603-607). New York, NY: Springer New York.
[14]. Casas P, Schatz R. Quality of experience in cloud services: Survey and measurements. Computer Networks. 2014 Aug 5;68:149-65.
[15]. Viji Rajendran V, Swamynathan S. Hybrid model for dynamic evaluation of trust in cloud services. Wireless Networks. 2016 Aug;22:1807-18.
[16]. Ding S, Yang S, Zhang Y, Liang C, Xia C. Combining QoS prediction and customer satisfaction estimation to solve cloud service trustworthiness evaluation problems. Knowledge-Based Systems. 2014 Jan 1;56:216-25.
[17]. Yzzogh H, Benaboud H. A comprehensive overview of machine learning for intrusion detection in software-defined networking. Innovations in Systems and Software Engineering. 2025 Apr 30:1-23.
[18]. Prizio F. Creating a simulation framework for the hadoop distributed file system inside heterogeneous cloud environments by using CloudSim.
[19]. Mateo-Fornes J, Solsona-Tehas F, Vilaplana-Mayoral J, Teixido-Torrelles I, Rius-Torrentó J. CART, a decision SLA model for SaaS providers to keep QoS regarding availability and performance. IEEE Access. 2019 Mar 18;7:38195-204.
[20]. Zheng X, Da Xu L, Chai S. QoS recommendation in cloud services. IEEE Access. 2017 Apr 19;5:5171-7.
[21]. Wang Y, Wen J, Zhou W, Tao B, Wu Q, Tao Z. A cloud service selection method based on trust and user preference clustering. IEEE Access. 2019 Aug 12;7:110279-92.
[22]. Anjana PS, Badia P, Wankar R, Kallakuri S, Rao CR. Cloud service provider evaluation system using fuzzy rough set technique. In2019 IEEE International Conference on Service-Oriented System Engineering (SOSE) 2019 Apr 4 (pp. 187-18709). IEEE.
[23]. Hassan H, El-Desouky AI, Ibrahim A, El-Kenawy ES, Arnous R. Enhanced QoS-based model for trust assessment in cloud computing environment. Ieee Access. 2020 Mar 4;8:43752-63.
[24]. Lou P, Yuan L, Hu J, Yan J, Fu J. A comprehensive assessment approach to evaluate the trustworthiness of manufacturing services in cloud manufacturing environment. IEEE Access. 2018 Jun 4;6:30819-28.
[25]. El Kassabi HT, Serhani MA, Dssouli R, Benatallah B. A multi-dimensional trust model for processing big data over competing clouds. Ieee access. 2018 Jul 17;6:39989-40007.
[26]. Chen J, Zhu Q. Security as a service for cloud-enabled internet of controlled things under advanced persistent threats: a contract design approach. IEEE Transactions on Information Forensics and Security. 2017 Jun 21;12(11):2736-50.
[27]. Sakai K, Sun MT, Ku WS, Wu J, Alanazi FS. Performance and security analyses of onion-based anonymous routing for delay tolerant networks. IEEE Transactions on Mobile Computing. 2017 Apr 4;16(12):3473-87.
[28]. Batista BG, Ferreira CH, Segura DC, Leite Filho DM, Peixoto ML. A QoS-driven approach for cloud computing addressing attributes of performance and security. Future Generation Computer Systems. 2017 Mar 1;68:260-74.
[29]. Liu B, Zhang Z. QoS-aware service composition for cloud manufacturing based on the optimal construction of synergistic elementary service groups. The International Journal of Advanced Manufacturing Technology. 2017 Feb;88:2757-71.
[30]. Ghahramani MH, Zhou M, Hon CT. Toward cloud computing QoS architecture: Analysis of cloud systems and cloud services. IEEE/CAA Journal of Automatica Sinica. 2017 Jan 16;4(1):6-18.
[31]. Nagarajan R, Thirunavukarasu R. A review on intelligent cloud brokers for effective service provisioning in cloud. In2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS) 2018 Jun 14 (pp. 519-524). IEEE.
[32]. Noshy M, Ibrahim A, Ali HA. Optimization of live virtual machine migration in cloud computing: A survey and future directions. Journal of Network and Computer Applications. 2018 May 15;110:1-0.
[33]. Tang M, Dai X, Liu J, Chen J. Towards a trust evaluation middleware for cloud service

selection. Future Generation Computer Systems. 2017 Sep 1;74:302-12.
[34]. Xiahou J, Lin F, Huang Q, Zeng W. Multi-datacenter cloud storage service selection strategy based on AHP and backward cloud generator model. Neural Computing and Applications. 2018 Jan;29:71-85.
[35]. Baawi SS, Oleiwi ZC, Al-Muqarm AM, Al-Shammary D, Sufi F. Efficient malware detection based on machine learning for enhanced cloud privacy protection. Evolving Systems. 2025 Feb;16(1):30.
[36]. Alfahaid A, Alalwany E, Almars AM, Alharbi F, Atlam E, Mahgoub I. Machine learning-based security solutions for iot networks: A comprehensive survey. Sensors. 2025 May 26;25(11):3341.