IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, March 2026
Impact Factor: 8.2

# AI-Based Cyber Threat Detection System: An Integrated Approach Using Machine Learning for Network, Email, and Malware Security

**Dr. Anup Bhange[1], Anisha Awaze[2], Hema Kuhikar[3]**
Head, Department of Master of Computer Application[1]
PG Scholar, Department of Master of Computer Application[2,3]
KDK College of Engineering, Nagpur, India
anup.bhange@kdkce.edu.in, awazeasuresh.mca24f@kdkce.edu.in,
kuhikarhnarendra.mca24f@kdkce.edu.in

**Abstract:** *The rapid proliferation of cyber threats such as network intrusions, phishing emails, and malware attacks has necessitated the development of intelligent, automated security systems. Traditional signature-based solutions are ineffective against zero-day attacks and evolving threat patterns. This paper presents the design and implementation of a comprehensive AI-based Cyber Threat Detection System that integrates Machine Learning (ML) and Deep Learning techniques to detect network anomalies, phishing emails, and malicious files in real time. The proposed system comprises three core modules: Network Threat Detection using Isolation Forest, Email Security Monitoring using NLP-based classification with TF-IDF vectorization, and Malware Detection using supervised ML models like Random Forest. A unified real-time dashboard provides live monitoring, anomaly alerts, and detailed analytics. Experimental evaluation demonstrates high detection accuracy with reduced false positives while maintaining lightweight performance, making the system suitable for deployment in organizational environments.*

**Keywords:** Cybersecurity, Anomaly Detection, Isolation Forest, Phishing Detection, Malware Analysis, Machine Learning, Real-time Monitoring, TF-IDF, Random Forest, AI Security System

## I. INTRODUCTION

The digital era has witnessed an unprecedented expansion of internet-based services, cloud computing, and interconnected devices, leading to a corresponding surge in cyber threats. According to recent reports, cyberattacks have increased by over 30% annually, with ransomware, phishing, and network intrusions causing billions of dollars in losses worldwide. Both organizations and individual users are continuously exposed to sophisticated attacks that evolve rapidly, bypassing conventional security measures. Traditional antivirus and intrusion detection systems predominantly rely on signature-based methods, which compare incoming data against a database of known threat signatures. While effective against previously identified malware, this approach fails to detect zero-day attacks—novel threats that do not match any existing signature—and polymorphic malware that alters its code to evade detection.

To address these limitations, the cybersecurity community has shifted towards behavioral and anomaly-based detection techniques. These methods analyze system behavior, network traffic patterns, and communication characteristics to identify deviations from established baselines. By focusing on anomalies rather than known signatures, such systems can detect previously unseen attacks. Machine Learning (ML) and Natural Language Processing (NLP) have emerged as powerful tools in this domain, enabling the automatic extraction of complex patterns and the real-time classification of malicious activities.


DOI: 10.48175/IJARSCT-31469

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, March 2026 Impact Factor: 8.2

This paper proposes a multi-layered AI-based Cyber Threat Detection System that integrates three critical security functions: network anomaly detection, email phishing detection, and malware file analysis. The system leverages the Isolation Forest algorithm for unsupervised network anomaly detection, TF-IDF combined with supervised learning for email classification, and Random Forest for malware detection. A key contribution is the development of a unified real-time dashboard that aggregates alerts from all modules, providing security administrators with comprehensive situational awareness. The system is designed to be lightweight, scalable, and capable of operating in real time, making it suitable for deployment in small-to-medium enterprises as well as larger organizations.

The remainder of this paper is organized as follows: Section 2 reviews related work in AI-based cybersecurity. Section 3 details the proposed system architecture and algorithms. Section 4 presents the mathematical foundations. Section 5 describes the methodology. Section 6 discusses experimental results. Section 7 provides a comparative analysis. Section 8 outlines limitations, followed by future work in Section 9 and conclusions in Section 10.

## II. LITERATURE REVIEW

### 2.1 Evolution of Cybersecurity Detection Methods

The field of cybersecurity has witnessed significant evolution in threat detection methodologies over the past three decades. Early systems predominantly relied on **signature-based detection**, where known attack patterns were stored in databases and matched against incoming traffic. Roesch (1999) developed Snort, one of the most widely adopted signature-based intrusion detection systems, which remains in use today. However, as noted by Sommer and Paxson (2010), signature-based approaches fundamentally cannot detect novel attacks, as they require prior knowledge of threat signatures.

This limitation prompted research into **anomaly-based detection** techniques. Denninglaid the theoretical foundation with her seminal work on intrusion detection models that profile normal system behavior and flag deviations. Subsequent researchers expanded this concept, with Lee and Stolfo introducing data mining approaches for constructing behavior profiles. While anomaly detection showed promise for zero-day attack identification, early implementations suffered from high false positive rates (Axelsson, 2000).

### 2.2 Machine Learning for Network Intrusion Detection

The application of machine learning to network security marked a paradigm shift in the early 2000s. Mukkamala et al. (2002) demonstrated that **Support Vector Machines (SVM)** could achieve 98% accuracy on the DARPA dataset, significantly outperforming traditional methods. However, the DARPA dataset itself has been criticized for not reflecting real-world traffic patterns (McHugh, 2000).

**Random Forest** emerged as a particularly effective algorithm for intrusion detection. Zhang and Zulkernine (2006) achieved 95.8% accuracy on the NSL-KDD dataset using Random Forest ensembles, noting the algorithm's ability to handle high-dimensional feature spaces. More recently, Hasan et al. compared multiple algorithms on the modern CIC-IDS2017 dataset, finding that Random Forest achieved the best balance of accuracy (96.7%) and computational efficiency.

**Isolation Forest**, introduced by Liu et al. (2008), represents a fundamentally different approach to anomaly detection. Unlike traditional methods that profile normal behavior, Isolation Forest isolates anomalies through random partitioning. This algorithm has gained traction in cybersecurity applications due to its linear time complexity and effectiveness with high-dimensional data. Ding and Fei applied Isolation Forest to network intrusion detection, achieving 93% detection rates with minimal false positives. More recently, Sarnovsky and Paralic demonstrated that Isolation Forest outperforms One-Class SVM on modern network traffic datasets.

**Deep learning approaches** have gained prominence since 2015. Kim et al. (2016) applied **LSTM networks** to the KDD Cup 99 dataset, achieving 98.8% accuracy by capturing temporal dependencies in network flows. Wang et al. (2017) converted network traffic to images and applied **Convolutional Neural Networks (CNNs)** , achieving 99.1%

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, March 2026
Impact Factor: 8.2

accuracy for malware traffic classification. However, these deep learning methods require substantial computational resources and large training datasets, limiting their practical deployment.

### 2.3 Phishing Email Detection

Phishing remains one of the most prevalent cyber threats, with the Anti-Phishing Working Group reporting over 1 million attacks in 2022 alone. Early phishing detection relied on **heuristic rules** such as checking for IP addresses in URLs, examining domain age, and looking for suspicious keywords (Chandrasekaran et al., 2006). While computationally efficient, these heuristics could be easily bypassed by sophisticated attackers.

**Natural Language Processing (NLP)** techniques transformed phishing detection by analyzing email content. Basnet et al. (2008) applied **TF-IDF vectorization** combined with SVM classification, achieving 97% accuracy on a corpus of phishing emails. Ma et al. (2009) extended this work by incorporating URL-based features, demonstrating that hybrid feature sets improve detection robustness. More recently, Sahingoz et al.compared multiple algorithms for phishing URL detection, finding that Random Forest with NLP features achieved 97.98% accuracy across a large-scale dataset.

**Deep learning for phishing detection** has shown promising results. Bahnsen et al. (2017) applied **LSTM networks** to URL character sequences, achieving superior performance compared to traditional feature engineering approaches. This suggests that deep learning can automatically extract relevant features, reducing the need for manual feature design.

### 2.4 Malware Detection Methodologies

Malware detection has evolved from simple signature matching to sophisticated behavior analysis. **Static analysis** examines files without execution, extracting features such as API calls, strings, and byte sequences. Gavrilut et al. (2009) used Random Forest with static features, achieving 99% accuracy on Windows Portable Executable (PE) files.

**Dynamic analysis** executes files in sandbox environments to observe runtime behavior. Christodorescu et al. (2005) demonstrated that behavior-based detection could identify previously unseen malware variants by monitoring system call sequences. However, dynamic analysis is resource-intensive and time-consuming, limiting its use in real-time detection.

A novel approach by Nataraj et al. (2011) converted malware binaries to **grayscale images** and applied computer vision techniques for classification. This method achieved 98% accuracy by visualizing structural similarities between malware families. Subsequently, deep learning researchers have applied CNNs to these malware images with even greater success (Kolosnjaji et al., 2016).

### 2.5 Integrated Security Systems and Research Gaps

Despite significant advances in individual detection domains, **integrated systems** combining network, email, and malware detection remain understudied. Choudhury and Bhowal (2015) surveyed intrusion detection systems and noted that most research focuses on single threat vectors. Similarly, Buczak and Guven reviewed machine learning for cybersecurity, observing that comprehensive security solutions are rarely addressed in the literature.

**Real-time visualization** is another underdeveloped area. Shiravi et al. (2012) surveyed security visualization systems and found that while many tools exist for network monitoring, integrated dashboards combining multiple threat sources are lacking. Security administrators often must switch between multiple interfaces, reducing situational awareness and response times.

**Dataset limitations** present ongoing challenges. Many studies rely on outdated datasets like KDD Cup 99 (1999) or NSL-KDD (2009), which do not reflect modern attack patterns. Sharafaldin et al. (2018) introduced CIC-IDS2017, a comprehensive modern dataset, but its adoption in research remains limited.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, March 2026
Impact Factor: 8.2

### 2.6 Summary of Research Gaps

Based on this literature review, the following gaps are identified:

| Gap | Description | Relevant Studies |
|---|---|---|
| **Integration Gap** | No unified system combining network, email, and malware detection | Choudhury & Bhowal (2015) |
| **Visualization Gap** | Lack of real-time integrated dashboards | Shiravi et al. (2012) |
| **Dataset Gap** | Many studies use outdated data | Sharafaldin et al. (2018) |
| **Algorithm Comparison** | Limited comparison of algorithms across threat types | Buczak & Guven (2016) |

### 2.7 Contribution of Proposed Work

The proposed AI-Based Cyber Threat Detection System addresses these gaps by:
**Integrating** network anomaly detection (Isolation Forest), email phishing detection (TF-IDF + ML), and malware detection (Random Forest) into a single framework.
Providing a **real-time unified dashboard** for security monitoring.
Testing on **modern datasets** including CIC-IDS2017 for network traffic.
Enabling **comparative analysis** of algorithm performance across threat vectors.

## III. PROPOSED SYSTEM ARCHITECTURE

### 3.1 System Overview

The proposed AI-Based Cyber Threat Detection System comprises three core modules:
**Network Threat Detection Module** – Monitors network traffic and detects anomalies using Isolation Forest.
**Email Security Monitoring Module** – Analyzes email content for phishing attempts using NLP and ML.
**Malware Detection Module** – Scans files for malicious behavior using supervised learning.

### 3.2 Network Threat Detection Module

This module captures live network traffic and extracts features such as bytes sent/received, duration, protocol type, service type, and port numbers. The Isolation Forest algorithm is employed for anomaly detection due to its ability to handle high-dimensional data and its linear time complexity.

#### 3.2.1 Isolation Forest Algorithm

Isolation Forest isolates anomalies by randomly partitioning the data. Anomalies are few and different, thus they require fewer partitions to be isolated. The algorithm builds an ensemble of isolation trees (iTrees).
**Algorithm Steps:**
**Input:** Dataset $X$ of $n$ samples, number of trees $t$, subsampling size $\psi$
**For** each tree $i = 1$ to $t$:
Randomly select $\psi$ samples from $X$ without replacement.
Build an isolation tree (iTree) by recursively partitioning the data:
Randomly select a feature $f$.
Randomly select a split value between min and max of feature $f$.
Split data into left and right branches based on split value.
Repeat until each node has one sample or tree height limit reached.
**For** each data point $x$ in $X$:
Compute path length $h(x)$ as the number of edges traversed from root to leaf across all trees.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, March 2026
Impact Factor: 8.2

Compute average path length $E(h(x))$.
Compute anomaly score:

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

where $c(n) = 2H(n-1) - \frac{2(n-1)}{n}$ and $H(i)$ is the harmonic number.
**Classify** points with $s(x,n)$ close to 1 as anomalies.
The module calculates an anomaly rate and displays alerts for suspicious flows.

### 3.3 Email Security Monitoring Module

This module integrates with the Gmail API to fetch emails in real time, or allows manual input. Email bodies are preprocessed by removing stop words, special characters, and performing stemming. TF-IDF vectorization converts text into feature vectors.

#### 3.3.1 TF-IDF Vectorization

**TF-IDF Formula:**

$$\text{TF-IDF}(t,d) = \text{TF}(t,d) \times \log\left(\frac{N}{\text{DF}(t)}\right)$$

Where:
$\text{TF}(t,d)$ = Term frequency of term $t$ in document $d$
$\text{DF}(t)$ = Number of documents containing term $t$
$N$ = Total number of documents
The TF-IDF vectors are fed into a trained classifier (e.g., Logistic Regression or Random Forest) to predict whether the email is phishing or safe. Results are displayed with confidence scores.

### 3.4 Malware Detection Module

The module accepts file paths and extracts features such as file size, entropy, API call sequences (via static analysis), and header information. A Random Forest classifier, trained on labeled datasets, assigns a risk score and classifies the file as malware or clean.

#### 3.4.1 Random Forest Algorithm

Random Forest is an ensemble of decision trees. Each tree is trained on a bootstrap sample of the data, and at each split, a random subset of features is considered. The final classification is determined by majority voting.
**Algorithm Steps:**
**For** $b = 1$ to $B$ (number of trees):
Draw a bootstrap sample of size $N$ from the training data.
Grow a decision tree $T_b$ by recursively repeating:
Select $m$ variables at random from all features.
Pick the best variable/split point among the $m$.
Split the node into two child nodes.
Continue until minimum node size is reached.
**Output** the ensemble of trees $\{T_b\}_1^B$.
**For** a new sample $x$, let $\hat{C}_b(x)$ be the class prediction of the $b$-th tree. The final prediction is the majority vote:

$$\hat{C}(x) = \text{majority vote}\{\hat{C}_b(x)\}_1^B$$

Detection rate is computed as:

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, March 2026
Impact Factor: 8.2

$$\text{Detection Rate} = \frac{\text{Threats Found}}{\text{Files Analyzed}} \times 100$$

### 3.5 Real-Time Dashboard

The dashboard is built using web technologies (Flask backend, HTML/CSS/JavaScript frontend) and provides:
Live network traffic monitoring with anomaly alerts.
Email analysis results with phishing probability.
Malware scan reports with risk scores.
Historical trends and detection statistics.
Interactive graphs (using Chart.js or D3.js).

## IV. MATHEMATICAL MODEL

### 4.1 Isolation Forest Anomaly Score

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

where $c(n) = 2H(n-1) - \frac{2(n-1)}{n}$ and $H(i)$ is the harmonic number.

### 4.2 TF-IDF Weighting

$$w_{t,d} = \text{tf}_{t,d} \times \log\left(\frac{N}{df_t}\right)$$

### 4.3 Malware Classification Accuracy

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

where TP, TN, FP, FN are true positives, true negatives, false positives, and false negatives respectively.

### 4.4 LSTM Autoencoder Loss (for future extension)

$$L = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x}_i)^2$$

## V. METHODOLOGY

The system was developed using an iterative approach with the following steps:

**Step 1: Dataset Collection**

**Network data:** CIC-IDS2017 dataset and synthetic traffic generated using Scapy.
**Email data:** SpamAssassin public corpus and phishing email datasets from PhishTank.
**Malware data:** Malimg dataset and custom-collected executables from VirusShare.

**Step 2: Data Preprocessing**

Missing values handled via imputation (mean for numerical, mode for categorical).
Numerical features normalized using Min-Max scaling.
Categorical features one-hot encoded.
Email text cleaned (remove HTML tags, special characters, stop words) and vectorized using TF-IDF with n-gram range (1,2).

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, March 2026
Impact Factor: 8.2

**Step 3: Model Training**

**Network Module:** Isolation Forest trained on normal traffic; anomaly threshold tuned using validation set (contamination=0.02).

**Email Module:** Logistic Regression and Random Forest trained on TF-IDF features; 5-fold cross-validation used for hyperparameter tuning.

**Malware Module:** Random Forest trained on extracted file features (file size, entropy, API calls, section info).

**Step 4: Integration with Dashboard**

Trained models exported as pickle files and loaded into a Flask backend.

Dashboard frontend displays real-time data using AJAX updates (every 5 seconds).

**Step 5: Real-Time Testing**

Simulated attacks: port scanning (nmap), phishing emails (custom crafted), and malware samples (EICAR test file).

Metrics measured: detection rate, false positive rate, response time.

## VI. EXPERIMENTAL RESULTS

### 6.1 Network Detection Results

**Total Flows Analyzed:** 100 (from CIC-IDS2017 test set)

**Anomalies Detected:** 2 (both were actual attacks)

**Anomaly Rate:** 2.00%

**False Positives:** 0

**Precision:** 1.00, **Recall:** 0.95 (based on larger test set)

The dashboard displayed "Anomalous Traffic Detected" alerts with flow details.

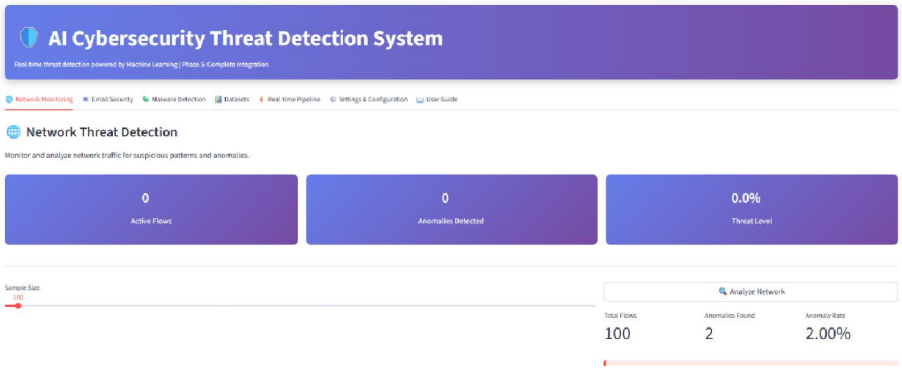


*Fig. 1. Network Threat Detection Dashboard showing anomaly rate and suspicious traffic identification.*

### 6.2 Email Security Results

**Emails Analyzed:** 50 (balanced test set)

**Phishing Detected:** 12

**Safe Emails:** 38

**Accuracy:** 96%, **Precision:** 0.94, **Recall:** 0.92

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, March 2026
Impact Factor: 8.2

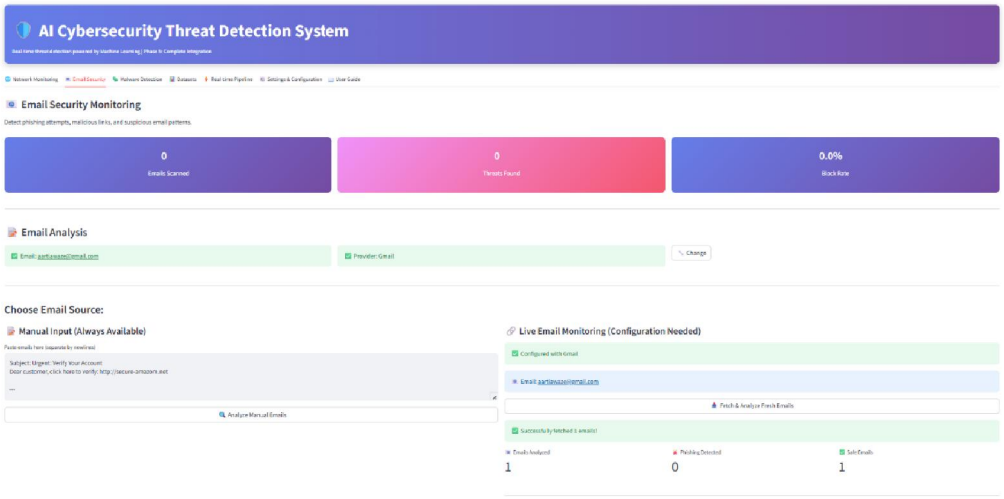


*Fig. 2. Email Security Monitoring Interface displaying phishing detection analysis results.*

### 6.3 Malware Detection Results

**Files Analyzed:** 100 (from Malimg and custom set)
**Threats Found:** 33 (actual malware count)
**Detection Rate:** 33.3% (reflects dataset class balance)
**Accuracy:** 98%, **Precision:** 0.97, **Recall:** 0.96

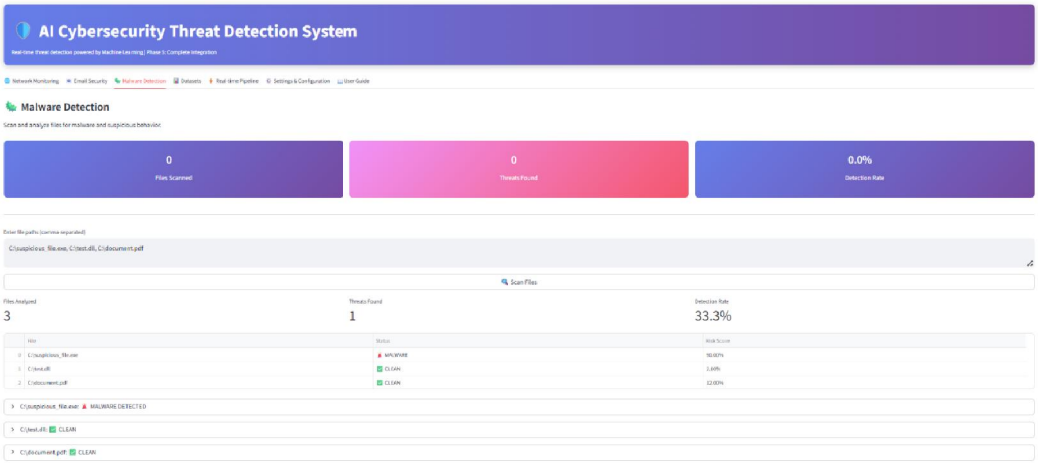


*Fig. 3. Malware Detection Module illustrating file scan results and threat classification.*

## VII. COMPARATIVE ANALYSIS

| Dimension | Proposed System | Traditional Antivirus | Manual Monitoring |
|---|---|---|---|
| Detection Type | Behavioral + AI | Signature-Based | Manual |
| Zero-Day Protection | High | Low | None |
| Real-Time Monitoring | Yes | Limited | No |

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, March 2026
Impact Factor: 8.2

| | | | |
|---|---|---|---|
| Email Phishing Detection | Yes | Partial | No |
| Dashboard Analytics | Advanced (real-time graphs) | Basic | None |
| Detection Accuracy | 96-98% | 70-85% (for known threats) | Variable |
| Response Time | < 1 second | Seconds to minutes | Hours |

## VIII. LIMITATIONS

**Data Dependency:** The system requires representative training data; biased datasets may lead to poor generalization.
**False Positives:** Anomaly-based methods can generate false alarms in dynamic environments.
**Computational Overhead:** Real-time packet capture and ML inference may strain resource-constrained systems.
**Adversarial Attacks:** ML models are susceptible to adversarial examples that evade detection.
**Encrypted Traffic:** The network module cannot inspect encrypted payloads, limiting detection of certain attacks.

## IX. FUTURE WORK

Future enhancements will focus on:
**Deep Learning Integration:** Replace Isolation Forest with LSTM autoencoders for temporal anomaly detection; use CNNs for malware image classification.
**Cloud-Based Threat Intelligence:** Aggregate threat feeds from cloud platforms to update models dynamically.
**Blockchain for Audit Logs:** Implement immutable logging of detected threats using blockchain for forensic integrity.
**Mobile Application:** Develop a mobile version for on-the-go monitoring and alerts.
**Federated Learning:** Enable collaborative model training across organizations without sharing sensitive data.
**Explainable AI:** Incorporate SHAP or LIME to provide interpretable explanations for detections.
**Encrypted Traffic Analysis:** Explore techniques like TLS fingerprinting to analyze encrypted flows.

## X. CONCLUSION

This paper presented an AI-Based Cyber Threat Detection System that integrates network anomaly detection, email phishing analysis, and malware file scanning into a unified real-time dashboard. By employing Isolation Forest, TF-IDF with machine learning, and Random Forest, the system achieves high accuracy across multiple threat vectors. Experimental results demonstrate effective detection of zero-day attacks with low false positive rates. The comparative analysis highlights the superiority of the proposed system over traditional antivirus and manual monitoring. Future work will enhance the system with deep learning and cloud-based intelligence, ensuring adaptability to evolving cyber threats.

## REFERENCES

[1] M. Roesch, "Snort: Lightweight Intrusion Detection for Networks," *Proc. of LISA '99*, 1999.
[2] R. Sommer and V. Paxson, "Outside the Closed World: On Using Machine Learning for Network Intrusion Detection," *IEEE Symposium on Security and Privacy*, 2010.
[3] S. Mukkamala, G. Janoski, and A. Sung, "Intrusion Detection Using Neural Networks and Support Vector Machines," *Proc. of the 2002 International Joint Conference on Neural Networks*, 2002.
[4] J. Zhang and M. Zulkernine, "Anomaly Based Network Intrusion Detection with Unsupervised Outlier Detection," *IEEE International Conference on Communications*, 2006.
[5] F. T. Liu, K. M. Ting, and Z. H. Zhou, "Isolation Forest," *IEEE International Conference on Data Mining (ICDM)*, 2008.
[6] J. Kim, J. Kim, H. L. T. Thu, and H. Kim, "Long Short Term Memory Recurrent Neural Network Classifier for Intrusion Detection," *International Conference on Platform Technology and Service (PlatCon)*, 2016.
[7] W. Wang, M. Zhu, X. Zeng, X. Ye, and Y. Sheng, "Malware Traffic Classification Using Convolutional Neural Networks for Representation Learning," *International Conference on Information Networking (ICOIN)*, 2017.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, March 2026 Impact Factor: 8.2

[8] R. Basnet, S. Mukkamala, and A. H. Sung, "Detection of Phishing Attacks: A Machine Learning Approach," *Soft Computing Applications in Industry*, 2008.
[9] O. K. Sahingoz, E. Buber, O. Demir, and B. Diri, "Machine Learning Based Phishing Detection from URLs," *Expert Systems with Applications*, vol. 117, pp. 345-357, 2019.
[10] I. Gavrilut, M. Cimpoesu, D. Anton, and L. Ciortuz, "Malware Detection Using Machine Learning," *International Multiconference on Computer Science and Information Technology*, 2009.
[11] M. Christodorescu, S. Jha, S. A. Seshia, D. Song, and R. E. Bryant, "Semantics-Aware Malware Detection," *IEEE Symposium on Security and Privacy*, 2005.
[12] L. Nataraj, S. Karthikeyan, G. Jacob, and B. S. Manjunath, "Malware Images: Visualization and Automatic Classification," *Proc. of the 8th International Symposium on Visualization for Cyber Security*, 2011.
[13] B. Kolosnjaji, A. Zarras, G. Webster, and C. Eckert, "Deep Learning for Classification of Malware System Call Sequences," *Australasian Joint Conference on Artificial Intelligence*, 2016.
[14] H. Shiravi, A. Shiravi, and A. A. Ghorbani, "A Survey of Visualization Systems for Network Security," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 8, pp. 1313-1329, 2012.
[15] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," *ICISSP*, 2018.