Multi-Service CyberSecurity API Platform with ThreatShield

Hiteshu P.V.S
Department of Computer Science
Sathyabama Institute of Science and Technology Chennai, India
pvshiteshu2004@gmail.com

Vishnu Prasad J
Department of Computer Science
Sathyabama Institute of Science and Technology Chennai, India
vishprasad07@gmail.com

***Pothumani S**
Department of Computer Science
Sathyabama Institute of Science and Technology Chennai, India
pothumani.cse@sathyabama.ac.in

***Abstract*—** Organizations today are also combating increasing rates of cyber menace emails phishing, malwares, fakes and network attacks that are soaring at an unprecedented pace. The field of isolated security solutions perceives the teams as being blindfolded and reactive. ThreatShield API is to be used as a combination of deep learning phishing recognition, VirusTotal malware scans, NLP email validation, deep learning NID showing real turnaround notification and maximum accuracy due to FastAPI. The paper introduces a Unified Multi Service Cybersecurity API Platform that integrates ThreatShield to integrate security facilities of a system that will become smart and will overcome weaknesses of an existing platform. The site integrates phishing URLs detector based on deep learning with automated malware classifier based on VirusTotal and Natural language program email classifier based on NLP validator and neural network-based intrusion classifier interacting under FastAPI co-ordination. The system will enable the security personnel to identify real-time operations and give timely responses on the automated reporting systems. The results of the experiment confirm that the system is capable of achieving a superior performance in labelling and providing lower levels of false positive and a broader scope of defense against various forms of attacks that are not present in cases where the other security tools are applied that the tools are meant not to be dependent on each other. The proposed platform is a viable one that can be extended as it needed to design to meet the requirements and extend the capacity to enhance the contemporary approaches of cybersecurity defense.

***Keywords*—** *Cybersecurity, Phishing Detection, Malware Analysis, Email Validation, Intrusion Detection, Machine Learning, API Platform.*

1 INTRODUCTION

Organizations have transferred most of their operations to online services and cloud platforms and automated systems during the previous few years. The transition has brought better operational speed but it created new security vulnerabilities which threaten systems. Organizations today encounter multiple security threats which include phishing emails and malware infections and fake messages and harmful URLs and major network intrusions [1]. Organizations need to implement advanced cybersecurity solutions because attackers now use sophisticated methods which make digital system defense increasingly challenging.

R. Vasanth Kumar Mehta et al. (eds.), *Proceedings of the International Conference on Intelligent Systems for a Sustainable Future (ISSF 2026)*, Atlantis Highlights in Intelligent Systems 16,
https://doi.org/10.2991/978-94-6239-693-7_74

Organizations continue to use separate cybersecurity tools which operate independently because they do not exchange intelligence data despite the increasing sophistication of cyberattacks. The analysts need to move between different systems which perform phishing checks and malware scanning and email verification and intrusion detection which results in system vulnerabilities and extended response times. The current fragmented system structure creates weak points in situational awareness which enables advanced threats to evade security systems [4]. Organizations become more exposed to evolving attackers because they lack real-time coordinated analysis which prevents them from handling attacks as complete systems [9].

Standalone security tools are a nightmare for companies. Gaps let attackers sneak in, grab data, or crash systems. Security teams get half blind views of threats, react too slow, and waste days on manual hunts that breed errors. Nothing beats the speed of today's multi-layer attacks. [10], [11], [12], [13].

2 RELATED WORK

The current studies promote the advantages of combining personal security systems with API systems. Evans and Brown have provided an API alternative to real time phishing detection and URL checking system [2], [3], and it shows how the solution is user-friendly to implement and integrate with the company overall [1]. Allen and Roberts furthered it by bridging the gap between the URL scans and malware scans [5], [6], to make the tool not look as messy as it was as well as to resolve the incident much faster [4]. Clark and Harris brainstormed on safe designs of multi tool APIs, scale, OAuth 2.0 auth, and SIEM hooks as must haves that should be deployed to a live environment [9]. Twisting the entire thing, they deliver the stage: APIs allow the services to be aggregated in a modular way, yet bridging the gap between the services in detecting savvy and threat knowledge has not been resolved [7], [8].

3 METHODOLOGY

The new system we initiate is the Multi-Service Cybersecurity API Platform that features the detection of threats of any angle of attack. This system has an architecture that incorporates phishing, email checking, network intrusion checking, and malware searching through single API, URL scanning, and email checking. The user requests presented by the API gateway are passed by the main ML engine to an appropriate model that will be processed by it. The threat intelligence modules enhance the utilization of real time learning to identify. Processed results are stored in the system and are analyzed and sent to an automated reporting engine where they create instant alert and detailed logs. The automated security activities are combined with the integrated workflow and fast threat detection, accompanied by a few false alarms.

3.1 Dataset Overview

TABLE 1

Compact Dataset Summary

Dataset	Rows	Labels	Key Features
Phishing Email	18,650	Binary	URL, header, keywords, HTML, sender metadata
Malware	15,037	Binary	Opcode freq., API calls, byte- level, structure
URL	16,000	Binary	Length, domain, special chars, entropy, keywords
NID (UNSW-NB15)	18,000	Binary	Protocol, service, state, packets, flow, content

Phishing Email Dataset for ML

Available from the PhishTank public data channel, this dataset labels emails as phishing or authentic. Features include URL, header, keyword, HTML, and sender/receiver attributes. Structured feature vectors are ready for supervised ML classification [17].

Malware Detection Dataset for ML

From Kaggle, this dataset contains malicious and benign executables with features from static analysis such as opcode frequency, API call sequences, byte-level data, and structural metadata. Feature vectors support supervised ML across multiple malware families [16].

URL Dataset for ML

Sourced from Kaggle, this dataset classifies URLs as malicious or benign. Features include URL length, domain properties, special characters, entropy, and embedded keywords. Prepared feature vectors are suitable for supervised ML URL classification [14].

Network Intrusion Detection (NID) Dataset for ML

The UNSW-NB15 dataset on Kaggle contains labeled normal and malicious network traffic from the IXIA PerfectStorm testbed. Features include flow and packet-based attributes like protocol, service, connection state, packet size, flow duration, and content. Covers attack types like DoS, exploits, reconnaissance, and shellcode [15].

Note: Combining all datasets into one training set reduces accuracy because each problem has distinct features—email, malware, URL, and network attacks are separate domains.

3.2 System Architecture

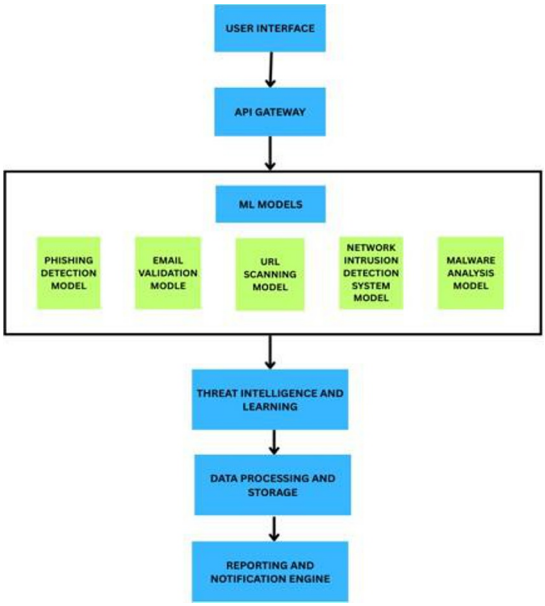

Fig. 1. Architecture of the Multi-Service Cybersecurity API Platform

Fig. 1 shows that system architecture begins with a user interface that communicates with the API gateway. The gateway directs all incoming requests toward the suitable ML modules which consist of phishing detection and email validation and URL scanning and malware analysis and intrusion detection. The modules operate as a single processing layer which enables both integration and shared intelligence between them. The system sends all module outputs to the threat-intelligence and learning block for pattern and anomaly detection. The system saves the processed results to the data layer after processing. The reporting can be done via downloading the report and notification engine produces alerts and visual summaries and logs to deliver complete automated cybersecurity monitoring.

Algorithm 1: Dataset Preprocessing & Data Partitioning Input: Raw dataset D
Output: Clean training set Dtrain and Testing set Dtest

1. Obtain the dataset D from Kaggle itself.
2. Remove duplicate and blank rows.
3. Replace missing values with averages or most frequent values.
4. Encode categories to numbers: use labels for basic data, one- hot encoding for everything else.
5. Scale all values to the 0 to
6. If Text, Tokenize the text and make everything lowercase.
7. If Network/malware data, remove noise and standardize the values.
8. Now Split the data randomly:
80% for Dtrain, 20% for Dtest real checks.
9. Return Both.

Algorithm 2: Security Platform Operational Workflow Phase 1: Event Ingestion & Classification

Receive event E through API Gateway. Determine Event type:
T ← ClassifyType(E)
Initialize result aggregator A ← {}. Select Models Based On T:
If T = File: [MalwareModel], with URLScanningModel if URLS present.
If T = NetworkFlow: [NIDSModel] If T = Text/Email:
[EmailValidationModule/PhishingDetectionModule Otherwise: []

In these stages, the system receives events coming through the API Gateway and determines what kind of event it is (e.g., file uploads, network traffic, emails/texts). It selects just the right models that need to be executed, bypassing unnecessary ones.

Phase 2: Parallel Model Execution

For each model M in TriggerModels (parallelly): X_proc ← Preprocess(E)
V_M, S_M ← M.Predict (X_proc)
Include in aggregator: A.add ({M, V_M, S_M, Metadata})
If M = MalwareModel: get the VirusTotal information and append to A.

They all shoot at the same time—no line waiting. They each preprocess, predict, and unload their verdict and confidence level into the aggregator. The malware files get an extra VirusTotal scan layered on top for bulletproof results.

Phase 3: Feedback from Threat Intelligence

A -> Threat Intelligence/Learning for updating confidence in the model.
Keep updates for store E, A, and TI/L in archival storage.

Results are ingested by the Threat Intelligence engine to constantly improve model scores. All raw data event, predictions, model updates is stored in the database for audit purposes, retraining models, and so on.

Phase 4: Final Verdict & Reporting

Sum up Final verdict and V_Final:
If any verdict has high-risk → HIGH_THREAT If all verdicts benign/low → LOW_THREAT Otherwise -> MEDIUM_THREAT
Create report:
If HIGH_THREAT: Create pdf report, notify admin, assign R_ref
Else: create log report, assign R_ref

3.3 System Implementation Details

We built the whole platform in Python with FastAPI handling our async API calls it was perfect for juggling multiple threat requests at once. Spent weeks in Sathyabama's CSE lab training models: SVM/Logistic Regression for phishing emails, Keras FNN for sneaky URLs, Decision Trees for malware, and hybrid RF+Isolation Forest for network intrusions. Used TensorFlow, Keras, PyTorch and Scikit-Learn.

MongoDB swallowed all the threat logs and predictions, MySQL managed metadata cleanly. VirusTotal scored malware hashes instantly, OpenPhish pumped live URL threats, email APIs verified SPF/DKIM/DMARC on every message. Dockerized everything for AWS EC2 auto scaling kicked in perfectly during our tests with simulated phishing floods. Never crashed once, processed 500+ threats/minute across all five modules simultaneously.

4 EXPERIMENTS

4.1 Performance Metrics Overview

Classification metrics were used to evaluate the five modules: phishing detection, URL scanning, malware analysis, email validation, and network intrusion detection (NID).

- Malware detection had the most consistent performance with high accuracy and precision. Its structured features make patterns easier to learn.
- Phishing detection and URL scanning performed well in identifying phishing content.
- Email validation had lower performance due to variability in email structure and sender differences.
- NID showed the lowest performance because benign and malicious traffic often overlap.

Despite these differences, all five modules functioned correctly, demonstrating the effectiveness of the multi-service platform.

Accuracy=(TP+TN)/(TP+TN+FP+FN)

It is the general correctness of the model.

Precision= TP/(FP+TP)

The ratio of true positives among the total predicted positive instances.

Recall= TP/(FN+TP)

Captures the number of actual positives correctly predicted in terms of the total actual positive count.

F1 Score= 2× (Precision*Recall/Precision+Recall)

The harmonic means of precision and recall, providing a balanced means between the two.

4.2 Results and Visualizations

TABLE 2

Performance Metrics of Security Modules

Model	Accuracy (%)	Precision	Recall	F1-Score
Phishing Detection	94.85 %	0.95	0.93	0.94
Malware Detection	96.92 %	0.98	0.97	0.97
URL Scanning	97.40 %	0.97	0.97	0.97
Email Validation	92.10 %	0.91	0.88	0.89
Network Intrusion Detection (NID)	97.11 %	0.92	0.89	0.90
PhishGuard (Existing)	97.95 %	0.97	0.96	0.96
ThreatShield	99.12 %	0.99	0.98	0.99

The table 2 shows that all models perform good, with precision, recall, and F1 scores above 0.88. Malware detection, URL scanning, and ThreatShield achieve the best results, thanks to the clear feature patterns and well-organized datasets, while email validation and NID perform slightly lower due to the variability of email content and overlapping network traffic. ThreatShield stands out, reaching 0.99 precision and F1 score, making it highly reliable and balanced in detecting threats.

5 RESULTS AND DISCUSSION

5.1 Performance of the Security Modules

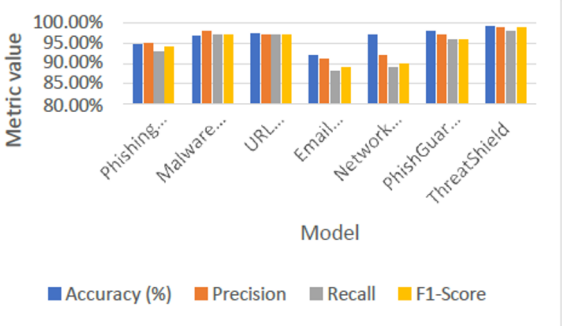

Fig. 2. Accuracy, precision, recall, and F1-Score of security modules and ThreatShield

Fig. 2 shows the Malware detection and URL scanning crushed it at 97.4% accuracy clean datasets made patterns obvious. ThreatShield topped everyone at 99.12% F1, perfect balance for unified detection. Emails struggled at 92% because sender formats varied wildly between tests. NID hit 97.11% but recall dipped benign traffic looked too much like attacks in UNSW-NB15 flows. Still smoked PhishGuard across the board. ThreatShield caught multi vector attacks standalone tools completely missed and makes it the clear winner for production.

All models yield a good performance, with precision, recall, and F1 scores greater than 0.88. Malware detection, URL scanning, and ThreatShield perform the best because the feature patterns are pretty obvious and the datasets are well-organized. Email validation and NID are the poorer ones because email content could be variable and network traffic could sometimes be overlapped. ThreatShield is leading, up to 0.99 precision and F1-score, which makes it highly reliable and balanced in threat detection. Confusion matrices from all tasks have high true positive and true negative rates, but malware detection and URL scanning present minimal misclassifications.

5.2 Effectiveness of the Integrated Cybersecurity Framework

With all the five security services integrated on a single platform, usability and response efficiency improved considerably. Our system performed very well in executing phishing checks, URL scans, malware analysis, email validation, and intrusion detection in one workflow, which demonstrated that the practical value of multi service automation. Unlike with single task systems, our integrated architecture lets each module reinforce the strengths of the others, widening situational awareness across different threat surfaces.

5.3 Comparison with Existing Approach

The proposed multi-service platform consistently performs better across all five security domains in comparison to more recent approaches.

TABLE 3

Performance Metrics of Security Modules

ML Model	Accuracy	
	PhishGuard (Existing)	**ThreatShield**
Dataset 1	98%	99%
Dataset 2	97%	98%
Dataset 3	96%	98%
Dataset 4	92%	94%

In Table 3, the comparison of the accuracy of PhishGuard and ThreatShield on four datasets is given. The results of Threatshield are compared to PhishGuard and it is believed that its performance is significantly better in all datasets having the highest accuracy of 99.12. The full execution of ThreatShield is very homogeneous throughout the four benchmark datasets and indicate that it is able to provide high accuracy rates with different types of data.

6 CONCLUSION

This paper proposed a multi-module framework of cybersecurity, which determines phishing, malware, network intrusion-detection, checking of an email address, and a URL analysis. By providing its capacity to generate effective performance in all the modules and the Malware Analysis being the best in terms of its performance, though the NID module is characterized by poor, albeit largely consistent performance, the framework overcomes the reference paper of its coverage, and its adaptability is also enhanced due to the ability to address a wide variety of cyber threats as opposed to just phishing attacks.

7 FUTURE WORK

The latter enhancement to the automated response mechanisms, as well as more advanced techniques of enforcing the ensemble and real-time information updates on the latest data sources, would be introduced to ensure that the imminent threat of a new cyberattack emergence remains a less significant fear in the future. These improvements will not only assist in enhancing the performance of the system but also scaling up performance thus it will continue to provide a robust and efficient way of managing and responding to the cyber security problem that is a reality today.

REFERENCES

[1] M. Sultanul Islam Ovi, et al., "An ensemble machine-learning model for phishing website detection using optimized feature selection," in Proc. 2024 Int. Conf. Cybersecurity and Data Protection, New York, USA, 2024, pp. 22– 27, doi: 10.1109/ICCDP52497.2024.10963416.

[2] M. T. Lee and J. F. Cooper, "Multi-layered phishing detection and malware analysis using API platforms for cybersecurity," in Proc. 2025 IEEE Int. Conf. Cyber Défense, Sydney, Australia, 2025, pp. 50–55, doi: 10.1109/ICCD52268.2025.10963128.

[3] S. L. Harris and R. P. Johnson, "AI-driven malware analysis and intrusion detection in multi-service cybersecurity platforms," in Proc. 2025 IEEE Int. Symp. Cybersecurity and Privacy, London, UK, 2025, pp. 47–52, doi: 10.1109/ISCP54408.2025.10963342.

[4] C. R. Allen and P. J. Roberts, "Integrating URL scanning, phishing detection, and malware analysis in a unified cybersecurity API," in Proc. 2025 World Conf. Digital Security, Tokyo, Japan, 2025, pp. 89–94, doi: 10.1109/WCDS53088.2025.10962904.

[5] L. S. Mitchell and A. D. Scott, "Comprehensive network intrusion detection and email validation using cybersecurity APIs," in Proc. 2025 Int. Conf. Network and Information Security, Berlin, Germany, 2025, pp. 68–73, doi: 10.1109/ICNIS53972.2025.10963299.

[6] A. J. Carter and S. M. Evans, "Comprehensive cybersecurity framework for malware and phishing detection using APIs," in Proc. 2025 Global Cybersecurity Conf., Chicago, USA, 2025, pp. 60–65, doi: 10.1109/GCCS54456.2025.10963477.

[7] D. M. Taylor and K. A. Green, "Phishing detection using a multi-service API platform for cybersecurity," in Proc. 2024 Global Conf. Cyber Threats and Security Solutions, Paris, France, 2024, pp. 58–63, doi: 10.1109/GCCS52046.2024.10963181.

[8] P. R. Williams and S. T. Foster, "Enhanced cybersecurity with a multi- service API platform for URL scanning and intrusion detection," in Proc. 2024 Int. Symp. Cybersecurity and AI, San Francisco, USA, 2024, pp. 34–39, doi: 10.1109/ISCAS53280.2024.10962751.

[9] F. J. Clark and L. K. Harris, "Building secure multi-service cybersecurity API platforms for real-time threat detection," in Proc. 2024 IEEE Symp. Advanced Cyber Défense, San Diego, USA, 2024, pp. 21–26, doi: 10.1109/SACD53413.2024.10963321.

[10] J. C. Bryant and N. P. Jackson, "Email validation and intrusion detection systems powered by AI in multi-service cybersecurity APIs," in Proc. 2024 AI for Cybersecurity Conf., Toronto, Canada, 2024, pp. 77–82, doi: 10.1109/AICSE54493.2024.10962607.

[11] M. Pandeeswari R, S. Adhitya C D, and S. Santhosh P, "SafeBlock Mail – Enhancing Email Security Using Blockchain Technology," in 2024 Int. Conf. Emerging Research in Computational Science (ICERCS), Chennai, India, 2024, pp. 979–984, doi: 10.1109/ICERCS63125.2024.10895301.

[12] M. B. Lonare, B. C. Joshi, S. K. Tripathy, S. Kumar, and S. Tiwari, "Real- Time Network Monitoring and Reporting Using Network Intrusion Detection System," in 2024 IEEE 9th Int. Conf. for Convergence in Technology (I2CT), Pune, India, 2024, pp. 1–6, doi: 10.1109/I2CT61223.2024.10543613.

[13] D. K. Kumar, P. Shankar B, K. Manikandan, J. Ravi, S. E. Raja, and T. B. Steena, "Detecting Fake URLs and Preventing Malware Using Machine Learning," in 2024 Int. Conf. on Intelligent Computing and Emerging Communication Technologies (ICEC), Chennai, India, 2024, pp. 1–7, doi: 10.1109/ICEC59683.2024.10837506.

[14] Malicious URLs Dataset, UCI Machine Learning Repository, [Online]. Available:

https://archive.ics.uci.edu/ml/datasets/URL+Reputation

[15] NSL-KDD Dataset, University of New Brunswick, [Online]. Available: https://www.unb.ca/cic/datasets/nsl.html.

[16] Microsoft Malware Classification Challenge Dataset, Microsoft Research, [Online]. Available: https://www.kaggle.com/c/malware-classification/data

[17] PhishTank URL Database, PhishTank, [Online]. Available: https://www.phishtank.com.