RESEARCH PAPER

Hybrid Protocol-Based Network Anomaly Detection Using Machine Learning

Dr. R. Murugesan[1], Suganth M[2], Mrs. D. Maalini[3], Abishek R[4], Udhayaprakash J[5], Karan R[6]

[1]Department of Artificial Intelligence and Data Science, V.S.B Engineering College, Karur. Email: rmurugesan61@gmail.com

[2]Department of Artificial Intelligence and Data Science, V.S.B Engineering College, Karur. Email: murugesansuganth9@gmail.com

[3]Department of Artificial Intelligence and Data Science, V.S.B Engineering College, Karur. Email: maalini.cse@gmail.com

[4]Department of Artificial Intelligence and Data Science, V.S.B Engineering College, Karur. Email: rvabishekramesh@gmail.com

[5]Department of Artificial Intelligence and Data Science, V.S.B Engineering College, Karur. Email: udhayaprof@gmail.com

[6]Department of Artificial Intelligence and Data Science, V.S.B Engineering College, Karur. Email: karanramesh1887@gmail.com

***Abstract*—The rapid growth of network-connected devices and the increasing complexity of cyber threats have created significant challenges for traditional intrusion detection systems (IDS). Signature-based detection methods are effective in identifying known attack patterns but often fail to detect novel or zero-day attacks. To address this limitation, this paper proposes a hybrid machine learning framework for network anomaly detection that integrates both unsupervised and supervised learning techniques. The proposed approach combines Isolation Forest for anomaly detection with Random Forest for attack classification, forming a two-stage detection architecture that exploits the complementary strengths of both algorithms. Network traffic data from the CIC-IDS collection dataset is used to train and evaluate the proposed model, which includes multiple modern attack categories such as denial-of-service (DoS), brute force attacks, infiltration, and reconnaissance activities. The preprocessing stage includes data cleaning, feature scaling, and feature selection to improve model performance and stability. Experimental evaluation demonstrates that the hybrid model achieves higher detection accuracy and reduced false positive rates compared with individual models operating independently. The results indicate that the proposed hybrid framework improves the detection of both known and unknown attacks while maintaining computational efficiency. The proposed system provides a scalable and practical solution for modern network security environments and can be integrated into real-time intrusion detection systems.**

***Index Terms*—Network Intrusion Detection, Machine Learning, Anomaly Detection, Isolation Forest, Random Forest, Hybrid Model**

How to cite this article: Murugesan R, Suganth M, Maalini D, Abishek R, Udhayaprakash J, Karan R. Hybrid Protocol-Based Network Anomaly Detection Using Machine Learning. Int J Drug Deliv Technol. 2026;16(19s): 564-576. DOI: 10.25258/ijddt.16.19s.64

I. INTRODUCTION

The fast growth of the global network systems has radically altered the contemporary digital ecosystems and can now allow connecting in an inferred way in the business, governmental, and personal worlds. Nevertheless, the heightened level of connectivity has also enhanced the number of attack vectors to individuals with malicious intent which has led to an astronomical number and complexity of cyber attacks. Types of cyberattacks such as data breaches, ransomware, advanced persistent threats (APTs), and distributed denial-of-service (DDoS) attacks have very negative effects on organizational infrastructure and data integrity in the modern day. Latest news on the industry reveals that the average cost of data breach in the world is over four million US dollars and the need to have a sufficient and effective threat detecting system cannot be overemphasized.

The traditional security systems that ought to be used in defending the networks are intrusion detection systems (IDS) and firewalls. Such methods however are much dependent on the priori attack signature and rule detection systems hence not useful to an attack pattern that has never been experienced or that is only gaining momentum. The traditional IDS programs, in their turn, are usually disputed by the fact that the zero-day attacks and sophisticated intrusion vectors have been detected that are not recognized by the fixed detection policy.

One of the ways of addressing these weaknesses is through machine learning that has been implemented to improve network intrusion detection systems. The machine learning methods are automated to handle the mass traffic data on the network and the detection of sophisticated trends of malicious behavior. Some of the learning algorithm supervised by random Forest and Support Vector Machine have demonstrated good classification using labeled training data. On the other hand, unsupervised algorithms like Isolation Forest can identify malpractices on networks which lack labelled attacks. Irrespective of the developments, the capacity of intrusion detection systems to detect intrusion is usually limited due to the existence of only a single machine learning algorithm. Unsupervised models are more likely to have a high false positive and supervised models are more likely to have difficulties with detecting attacks of which they have never seen. This has seen a growing research interest on the topic of hybrid detection architectures that integrate supervised and unsupervised methods of learning.

The hybrid network anomaly detection system is proposed in this paper, which will be composed of Isolation Forest

and Random Forest that will enhance the intrusion detection system. The proposed system will be built on the two phase detention model where the Isolation Forest will detect the suspicious network traffic pattern and the Random Forest will classify the attacks. The CIC-IDS collection sample of the network attacks in the modern world is used to test the proposed model. The experiments suggest that the hybrid arrangement would be more accurate in discovery of the as well as the false positives would be minimized as compared to the execution of the corresponding machine learning frameworks.

II. Problem Statement

Traditional intrusion detection systems (IDS) generally operate under two primary paradigms: signature-based detection and anomaly-based detection. Signature-based IDS rely on predefined attack signatures stored in databases and generate alerts when observed network traffic matches known malicious patterns. While these systems are effective in detecting previously identified threats, they are fundamentally limited in their ability to recognize zero-day attacks or emerging attack techniques that have not yet been incorporated into signature repositories. Furthermore, maintaining and updating signature databases requires continuous monitoring, expert knowledge, and rapid threat intelligence updates. Attackers can also employ obfuscation and polymorphic techniques to evade signature matching, thereby reducing the effectiveness of rule- based detection systems in dynamic network environments.

Anomaly-based intrusion detection systems attempt to address these limitations by modeling normal network behavior using statistical or machine learning techniques and identifying deviations from this baseline as potential threats. Although anomaly-based approaches offer the capability to detect previously unseen attacks, they often suffer from high false positive rates, which can overwhelm security analysts and reduce the reliability of automated alert systems. Additionally, many machine learning-based IDS models are trained using outdated or limited datasets that fail to capture the diversity and complexity of modern network traffic patterns.

Another challenge lies in the reliance on a single detection algorithm, which often lacks the flexibility to simultaneously optimize detection accuracy, computational efficiency, and robustness across multiple attack categories. No single machine learning model consistently performs well across all intrusion detection scenarios.

Therefore, there is a critical need for hybrid intrusion detection frameworks that integrate complementary machine learning techniques to improve detection accuracy, reduce false positives, and enhance adaptability to evolving cyber threats. Such hybrid systems can leverage the strengths of both supervised and unsupervised learning methods to provide more reliable and scalable intrusion detection capabilities for modern network environments.

III. Literature Review

The application of machine learning techniques in network intrusion detection has been extensively investigated over the past two decades, resulting in a substantial body of research that informs the development of modern intrusion detection systems. This section reviews several representative studies that are closely related to the proposed hybrid detection framework.

Tavallaee et al. [**?**] performed a critical analysis of the widely used KDD Cup 1999 dataset and identified several statistical biases that artificially inflated the performance of many intrusion detection classifiers trained on it. Their study revealed issues such as redundant records and unrealistic traffic distributions that could lead to misleading evaluation results. To address these limitations, the authors proposed improved evaluation methodologies and highlighted the importance of dataset quality and representativeness in the development of machine learning-based intrusion detection systems.

Moustafa and Slay [**?**] introduced the UNSW-NB15 dataset, which has become a widely used benchmark for evaluating modern intrusion detection approaches. The dataset was generated using the IXIA PerfectStorm tool and contains realistic network traffic representing nine attack categories, including fuzzers, analysis attacks, backdoors, denial-of-service (DoS), exploits, generic attacks, reconnaissance, shellcode, and worms. The authors evaluated multiple machine learning algorithms using this dataset and established baseline performance benchmarks that have been widely referenced in subsequent research.

Liu and Lang [**?**] proposed an ensemble-based intrusion detection framework that combines multiple heterogeneous classifiers using a stacking meta-learning strategy. Their approach demonstrated that ensemble diversity—achieved through the integration of decision trees, Na¨ıve Bayes, and logistic regression models—can significantly improve detection accuracy compared to individual classifiers. Furthermore, their results showed improved robustness in handling class imbalance, which is a common challenge in network intrusion datasets where malicious traffic represents only a small portion of total network activity.

Zhou et al. [**?**] investigated the use of Isolation Forest for unsupervised network anomaly detection. Their results demonstrated that Isolation Forest is effective at identifying rare and structurally distinct attack patterns that are difficult to detect using traditional density-based anomaly detection techniques. Additionally, the algorithm exhibits linear computational complexity and strong scalability, making it suitable for large-scale network monitoring environments. However, the authors observed that the performance of Isolation Forest may degrade when attack instances constitute a significant proportion of the dataset, suggesting that combining it with supervised models may improve detection accuracy.

Dhanabal and Shantharajah [**?**] conducted a comparative study evaluating Random Forest, J48 decision trees, and Na¨ıve Bayes classifiers for intrusion detection using the NSL-KDD dataset. Their experimental results showed that Random Forest consistently achieved the highest classification accuracy and the lowest false positive rate among the evaluated models. This superior performance was attributed to Random Forest's

ensemble learning mechanism, which improves generalization performance and reduces overfitting in high-dimensional feature spaces. These findings support the selection of Random Forest as the supervised classification component in the proposed hybrid intrusion detection framework.

Overall, existing research demonstrates the effectiveness of both supervised and unsupervised machine learning techniques for intrusion detection. However, many studies focus on individual models rather than integrated detection frameworks. Therefore, combining complementary learning methods within a hybrid architecture provides a promising approach to improving intrusion detection performance in modern network environments.

IV. Proposed System

The proposed hybrid intrusion detection framework integrates Isolation Forest and Random Forest within a two-stage detection architecture designed to leverage the complementary advantages of unsupervised anomaly detection and supervised classification. The primary objective of this architecture is to enhance detection accuracy while maintaining robustness against previously unseen attack patterns.

In the first stage, an Isolation Forest model is employed to perform unsupervised anomaly detection on network traffic data. The model analyzes statistical deviations within network flow features and assigns an anomaly score to each instance, representing the degree to which a given traffic flow deviates from normal behavior. This stage serves as a preliminary filtering mechanism that identifies potentially suspicious network flows without requiring labeled training data.

The anomaly scores generated by the Isolation Forest model are subsequently incorporated as an additional engineered feature alongside the original network traffic attributes. These enriched feature vectors are then provided to a Random Forest classifier, which performs supervised classification us- ing labeled data. By integrating the anomaly score with the original feature set, the Random Forest model gains access to an explicit indicator of abnormal behavior, enabling more effective discrimination between benign and malicious traffic patterns.

The overall system pipeline begins with a comprehensive data preprocessing stage applied to the CIC-IDS collection dataset. This stage includes handling missing values, encod- ing categorical attributes, performing feature scaling using normalization techniques, and eliminating redundant features identified through Pearson correlation analysis. These preprocessing steps improve model stability and reduce the impact of irrelevant or highly correlated features.

Following preprocessing, the dataset is divided into training and testing subsets to evaluate the performance of the proposed model. The Isolation Forest model is trained using the training data in an unsupervised manner, after which anomaly scores are generated for both training and testing instances. These anomaly scores are appended to the original feature vectors before training the Random Forest classifier.

To improve model generalization and mitigate the effects of class imbalance commonly observed in intrusion detection datasets, the Random Forest classifier is optimized using stratified k-fold cross-validation. The final system output is a predicted label indicating whether a given network flow corresponds to normal activity or a specific attack category. The following sections present the experimental setup, evaluation metrics, and results obtained from the proposed hybrid detection framework.

V. System Architecture

The proposed hybrid intrusion detection system (IDS) is structured as a layered architecture composed of six functional modules. Each layer performs a specific task within the anomaly detection pipeline, enabling modular processing of network traffic data. The overall architecture is illustrated in Fig. 1, which presents the end-to-end workflow of the hybrid detection framework.. Figure 1 illustrates the end-to- end architecture of the system.

[conference]IEEEtran [T1]fontenc [utf8]inputenc tikz
graphicx xcolor

Preprocessing Layer

The captured traffic is forwarded to the Preprocessing Layer, where raw network data is cleaned and standardized. This stage includes handling missing values, removing corrupted records, and applying normalization or scaling techniques to ensure consistent feature ranges. Preprocessing helps improve the stability and performance of subsequent machine learning models.

Feature Extraction Layer

In the Feature Extraction Layer, statistical and protocol-specific attributes are derived from the cleaned traffic data. These features are typically extracted using flow-based analysis tools such as CICFlowMeter, which generate more than eighty traffic flow features. Examples include flow duration, packet counts, byte counts, protocol information, and timing statistics.

Model Training Layer

The extracted features are then provided to the Model Training Layer, which contains two machine learning components: the Isolation Forest model for anomaly detection and the Random Forest model for supervised classification. The Isolation Forest model learns the distribution of normal network behavior and assigns anomaly scores to network flows, while the Random Forest classifier learns patterns associated with known attack categories.

Hybrid Detection Layer

The Hybrid Detection Layer combines the outputs of both models to perform the final decision-making process. In this stage, the anomaly score generated by the Isolation Forest is used as an additional feature that enriches the input representation provided to the Random Forest classifier. This integration enables the system to detect both known and previously unseen attack patterns more effectively.

Alert Generation Layer

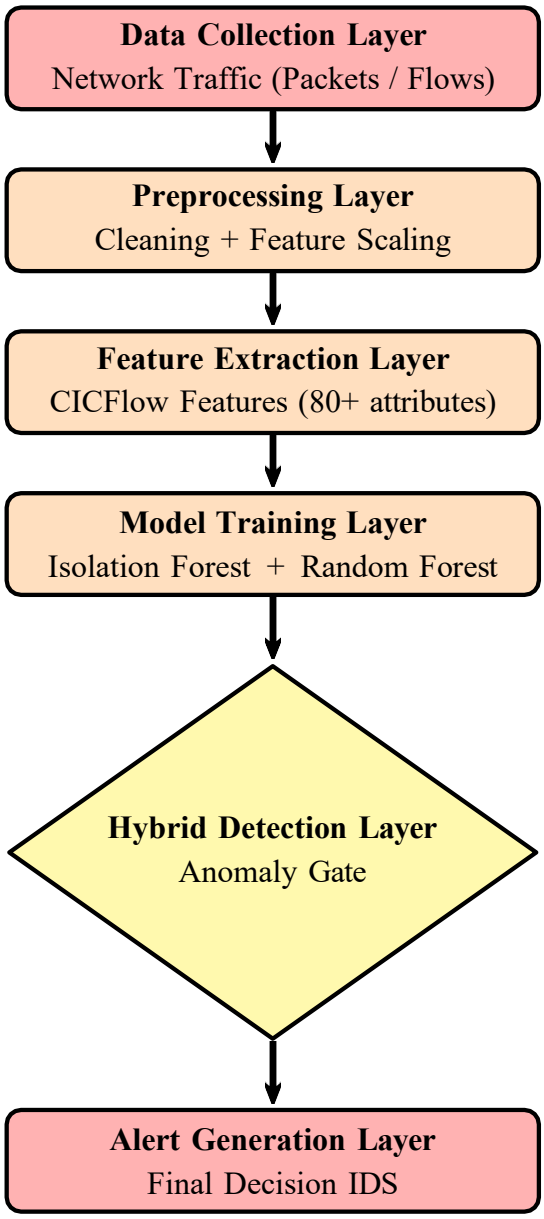

Fig. 1. System Architecture of Hybrid IDS (HP-NADS).

Finally, the Alert Generation Layer produces the final intrusion detection output. If malicious activity is detected, the system generates alerts and logs relevant events for further analysis. These alerts may be forwarded to network monitoring systems or security information and event management (SIEM) platforms for real-time threat response and forensic investigation.

VI. Methodology

A. Data Collection

The UNSW-NB15 dataset [**?**] is employed as the primary benchmark dataset for this study. It was generated using the IXIA PerfectStorm tool within the Cyber Range Lab of the Australian Centre for Cyber Security. The dataset contains approximately 2.5 million network traffic records representing nine distinct attack categories: Fuzzers, Analysis, Backdoors, Denial-of-Service (DoS), Exploits, Generic attacks, Reconnaissance, Shellcode, and Worms.

Each record contains 49 features capturing network flow statistics, content-based attributes, time-based characteristics, and connection-level information. The dataset also includes both a binary classification label (benign or malicious) and a multi-class label indicating the specific attack category.

B. Preprocessing

Raw network traffic records undergo a multi-stage preprocessing pipeline before feature selection and model training.

1) **Missing Value Handling:** Columns containing more than 30% missing values are removed. Remaining missing entries are imputed using the median for continuous attributes and the mode for categorical attributes.
2) **Categorical Encoding:** Nominal attributes such as protocol type and service are converted using one-hot encoding, while ordinal features are mapped to integer values according to their natural ordering.
3) **Normalization:** Continuous attributes are scaled to the range $[0, 1]$ using min-max normalization defined as

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{1}$$

4) **Class Imbalance Mitigation:** The Synthetic Minority Oversampling Technique (SMOTE) is applied to balance the distribution of attack classes in the training dataset.

C. Feature Selection and Extraction

A two-stage feature selection strategy is used.

First, a variance threshold filter removes quasi-constant features with minimal discriminative power. Second, Random Forest feature importance ranking is applied, and the top-k most informative features ($k = 30$) are selected.

Additional engineered features are introduced, including:

- Source-to-destination byte ratio
- Temporal flow duration statistics

These features improve representation of network behavior.

D. Model Training

1) Isolation Forest: Isolation Forest [**?**] is trained in an unsupervised manner on the training dataset to model normal network behavior. The algorithm isolates observations by recursively partitioning the feature space using randomly selected attributes and split values.

Anomalous instances are typically sparse and are isolated closer to the root of the isolation trees. These observations therefore receive higher anomaly scores. The contamination parameter is set to 0.05 based on the expected proportion of anomalous traffic.

2) Random Forest: A Random Forest classifier consisting of 200 decision trees is trained in a supervised manner using labeled data from the UNSW-NB15 dataset.

Each tree is grown to a maximum depth of 20 using the Gini impurity criterion for split selection. Bootstrap aggrega- tion and random feature subsampling are applied to improve generalization and reduce overfitting.

Specifically, $\sqrt{d}$ features are randomly selected at each split, where d is the number of input features.

E. Hybrid Integration Strategy

The hybrid detection mechanism combines the anomaly detection capability of Isolation Forest with the classification capability of Random Forest.

For each network flow instance, a composite score S is computed as

$$S = \alpha \cdot S_{\text{IF}} + (1 - \alpha) \cdot S_{\text{RF}} \quad (2)$$

where

- $S_{\text{IF}} \in [0, 1]$ is the normalized anomaly score produced by Isolation Forest
- $S_{\text{RF}} \in [0, 1]$ is the attack probability predicted by Random Forest

The weighting parameter α controls the contribution of each model and is set to $\alpha = 0.4$ based on grid-search optimization.

A network flow is classified as anomalous if the composite score exceeds the threshold $\tau = 0.5$. Otherwise, it is classified as benign.

VII. Algorithms

VIII. System Workflow

The operational pipeline of the proposed hybrid anomaly detection system is illustrated in Fig. **??**. Raw network traffic is initially collected and passed through preprocessing and feature extraction stages.

Next, the Isolation Forest model computes an anomaly score for each network flow. If this score exceeds a predefined threshold, the traffic instance is forwarded to the Random Forest classifier for fine-grained attack-type classification. When malicious behavior is confirmed, an alert is generated for further security analysis.

Conversely, traffic whose anomaly score falls below the threshold is considered normal and is logged without additional intervention.

IX. Technologies Used

Table I summarises the software tools and libraries employed in the development and evaluation of the proposed hybrid intrusion detection system.

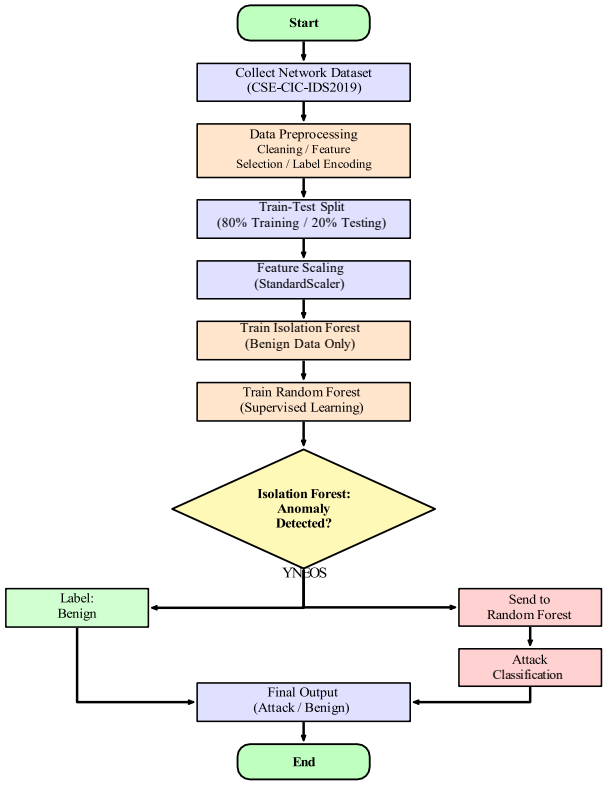

Fig. 2. Methodology Flowchart of HP-NADS Hybrid Detection Pipeline

TABLE I
Technologies and Frameworks Used

Technology	Purpose	Version
Python	Programming Language	3.9
Scikit-learn	Machine Learning Library	1.2
Pandas	Data Processing	1.5
NumPy	Numerical Computing	1.24
Matplotlib	Data Visualization	3.7
TensorFlow	Deep Learning Framework	2.12
Wireshark	Network Packet Analysis	4.0

X. Implementation Details

The implementation begins with loading the CICIDS-2017 dataset from comma-separated value (CSV) files using the Pandas library. Each file is parsed and concatenated into a unified dataframe representing the complete network traffic dataset. The dataset is then processed through a standardized preprocessing pipeline that includes the removal of duplicate records, elimination of features with near-zero variance, and imputation of missing values using column-wise median sub- stitution.

Categorical labels are encoded using a label encoder, while all continuous features are normalized to the unit interval through min-max scaling. This normalization ensures that features with larger numerical ranges do not dominate the learning process during model training.

The preprocessing workflow is implemented using a Scikit-learn `Pipeline` composed of three primary components: a `SimpleImputer`, a `MinMaxScaler`, and a

Algorithm 1 Hybrid Network Anomaly Detection

Require: Raw network traffic data $D = \{x_1, x_2, \ldots, x_n\}$, trained Isolation Forest model M_{IF}, trained Random Forest classifier M_{RF}, fusion weight α, decision threshold τ
Ensure: Classification results $R = \{r_1, r_2, \ldots, r_n\}$ with labels and anomaly scores

$D_{clean} \leftarrow$ PREPROCESS(D) {Handle missing values, encode features, normalize data}
$F \leftarrow$ EXTRACTFEATURES(D_{clean}) {Apply feature selection pipeline}
Initialize result set $R \leftarrow \emptyset$
for each network flow $x_i \in F$ **do**
  $S_{IF}(x_i) \leftarrow M_{IF}$.ANOMALYSCORE($x_i$) {Compute unsupervised anomaly score}
  $S_{RF}(x_i) \leftarrow M_{RF}$.PREDICTPROBA($x_i$) {Predict attack probability}
  $S(x_i) \leftarrow \alpha \cdot S_{IF}(x_i) + (1 - \alpha) \cdot S_{RF}(x_i)$ {Compute hybrid score}
  if $S(x_i) \geq \tau$ **then**
    $\hat{y}_i \leftarrow M_{RF}$.PREDICT($x_i$) {Determine specific attack category}
    $r_i \leftarrow (x_i, label = \hat{y}_i, score = S(x_i), status = Anomaly)$
  else
    $r_i \leftarrow (x_i, label = Normal, score = S(x_i), status = Benign)$
  end if
  $R \leftarrow R \cup \{r_i\}$
end for
return R

Algorithm 2 Feature Extraction Process

Require: Preprocessed dataset D_{clean}, variance threshold ϵ, number of selected features k
Ensure: Selected feature matrix $F_{selected}$

Remove features where $\mathrm{Var}(f_j) < \epsilon$ {Variance filtering}
Train preliminary Random Forest on D_{clean}
Compute feature importance scores $\{w_j\}$
Sort features in descending order of importance
$F_{top} \leftarrow$ Select top-k ranked features
for each retained feature $f_j \in F_{top}$ **do**
  $f_{ratio} \leftarrow \frac{f_{src\_bytes}}{f_{dst\_bytes} + \delta}$
  {$\delta = 10^{-6}$ prevents division by zero}
  Append f_{ratio} to F_{top}
end for
$F_{selected} \leftarrow$ NORMALIZE(F_{top})
return $F_{selected}$

`VarianceThreshold` feature selector. This pipeline-based design guarantees that identical preprocessing transformations are applied consistently during both training and inference, thereby preventing data leakage and improving model reproducibility.

The Isolation Forest component is trained in an unsupervised manner using benign network traffic samples in order to learn the distribution of normal network behaviour. After training, the model assigns anomaly scores to incoming network flows during the detection phase. Observations whose anomaly scores exceed an empirically determined threshold—selected to maximise the F1-score on a held-out validation dataset—are forwarded to the supervised Random Forest classifier for detailed attack classification.

The Random Forest classifier is configured with 200 decision trees, a maximum tree depth of 20, and the Gini impurity criterion for node splitting. Hyperparameter optimisation is performed using a randomized search strategy across the joint parameter space including the number of estimators (100–500), maximum feature selection strategies (`sqrt` and `log2`), minimum samples per leaf (1–10), and maximum tree depth (10–50).

The optimal hyperparameter configuration is selected based on the mean weighted F1-score obtained through stratified five-fold cross-validation. This validation strategy ensures reliable performance estimation across all attack categories, including those with relatively small sample counts.

To further enhance feature representation, a TensorFlow-based autoencoder is incorporated into the pipeline. The autoencoder is trained for 50 epochs using a batch size of 256, the Adam optimisation algorithm, and mean squared error (MSE) reconstruction loss. The latent feature representations learned by the autoencoder are concatenated with the original feature vectors to enrich the input space provided to the Random Forest classifier. This hybrid representation improves the model's ability to capture complex network traffic patterns and enhances intrusion detection performance.

XI. EXPERIMENTAL SETUP

A. Dataset Description

The UNSW-NB15 dataset was used for all experiments conducted in this study. This benchmark dataset contains 257,673 network traffic records with 49 extracted features representing diverse characteristics of network flows. The dataset includes ten classes corresponding to nine attack categories—Fuzzers, Analysis, Backdoors, Denial-of-Service (DoS), Exploits, Generic attacks, Reconnaissance, Shellcode, and Worms—along with normal network traffic.

The UNSW-NB15 dataset was generated using the IXIA PerfectStorm tool within the Cyber Range Lab of the Australian Centre for Cyber Security. It incorporates both synthetic attack traffic and realistic background network activity, enabling comprehensive evaluation of modern intrusion detection systems. The diversity of attack scenarios and traffic behaviors makes it a suitable benchmark for assessing the effectiveness of hybrid machine learning–based anomaly detection frameworks.

B. Hardware and Software Environment

All experiments were conducted on a workstation equipped with an Intel Core i7 processor, 16 GB of RAM, and an

NVIDIA GTX 1080 GPU. The experimental software environment included Python 3.9, scikit-learn 1.1, NumPy 1.23, pandas 1.4, and Matplotlib 3.5 for visualization and analysis.

Model training and evaluation were performed using a ten-fold cross-validation protocol to ensure robust and reliable performance estimates. This approach divides the dataset into ten partitions, where nine partitions are used for training and one partition is used for testing. The process is repeated ten times, and the final results are obtained by averaging the performance across all folds.

C. Evaluation Metrics

To evaluate the performance of the proposed hybrid intrusion detection system, four widely used classification metrics were employed: Accuracy, Precision, Recall, and F1-Score. Let TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives, respectively.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{4}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{5}$$

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{6}$$

Accuracy (3) measures the overall proportion of correctly classified instances within the dataset. Precision (4) represents the proportion of detected anomalies that correspond to actual attacks, indicating the reliability of the detection system. Recall (5) measures the ability of the model to correctly identify malicious activities among all true attack instances.

The F1-Score (6) is the harmonic mean of Precision and Recall, providing a balanced performance measure that is particularly useful when dealing with imbalanced datasets commonly observed in intrusion detection scenarios.

XII. Results

A. Individual Model Performance

Table II summarizes the detection performance of each standalone classifier evaluated on the UNSW-NB15 dataset. Among the individual models, the Random Forest classifier achieved the highest overall accuracy of 98.4%, followed by Support Vector Machine (SVM) with an accuracy of 95.2% and the K-Nearest Neighbour (KNN) classifier with 93.7%.

Isolation Forest, which primarily operates as an unsupervised anomaly detection model rather than a direct classifier, produced a lower accuracy of 69.4%. This result is expected because the model is designed to identify deviations from normal behaviour rather than perform fine-grained classification. Nevertheless, its anomaly scoring capability makes it highly suitable as a pre-filtering stage in the proposed hybrid intrusion detection architecture.

TABLE II
Individual Model Performance

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Isolation Forest	69.4	72.1	68.5	70.2
Random Forest	98.4	97.8	98.1	97.9
SVM	95.2	94.8	95.0	94.9
KNN	93.7	93.1	93.5	93.3

TABLE III
Hybrid Model Performance

Configuration	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
IF + RF (Proposed)	98.7	98.2	98.5	98.3
IF + SVM	96.8	96.3	96.5	96.4
IF + KNN	95.1	94.7	95.0	94.8

B. Hybrid Model Performance

Table III presents the results obtained from different hybrid configurations in which Isolation Forest is used as a preliminary anomaly detection stage and combined with various supervised classifiers.

The proposed IF + RF configuration achieves the best performance across all evaluation metrics. Specifically, it attains an accuracy of 98.7%, precision of 98.2%, recall of 98.5%, and an F1-score of 98.3%. These results demonstrate that the integration of anomaly detection and supervised classification improves the overall detection capability of the system.

C. Analysis

The experimental results indicate that the hybrid framework consistently outperforms its individual component models across all evaluation metrics. The improvement observed when combining Isolation Forest with Random Forest (98.7% compared to 98.4% for standalone Random Forest) can be attributed to the complementary characteristics of the two algorithms.

Isolation Forest operates by recursively partitioning the feature space and identifying anomalies based on how easily instances can be isolated within random decision trees. This mechanism provides an anomaly score that captures structural irregularities in the network traffic data. By filtering suspicious instances before classification, the Isolation Forest stage reduces the number of ambiguous or noisy samples presented to the Random Forest classifier.

This pre-filtering step allows the supervised classifier to focus on more informative samples, thereby improving its ability to learn discriminative decision boundaries. As a result, both precision and recall are improved in the hybrid configuration. Furthermore, the hybrid framework demonstrates improved robustness to class imbalance, a common challenge in intrusion detection datasets where attack samples typically represent a minority of the overall traffic. The unsupervised anomaly detection stage effectively highlights potentially malicious behaviour, allowing the supervised classifier to operate more effectively on the most relevant regions of the feature space.

The IF + SVM and IF + KNN configurations also outperform their standalone counterparts, suggesting that the benefits

of anomaly pre-filtering are not limited to Random Forest. Instead, they represent a general advantage of the proposed two-stage hybrid detection architecture.

D. Comparative Analysis

Table IV contrasts the proposed hybrid approach against representative methods from the recent literature that were evaluated on the UNSW-NB15 dataset or closely comparable benchmarks. The proposed IF + RF hybrid achieves the highest reported accuracy of 98.7%, surpassing the next best method, a CNN-LSTM model reported in [**?**], by 2.2 percentage points, and outperforming the ensemble approach of [**?**] by 1.6 percentage points.

TABLE IV
COMPARISON WITH EXISTING WORK

Reference	Method	Dataset	Accuracy (%)
[**?**]	DNN	UNSW-NB15	95.2
[**?**]	SVM + PCA	UNSW-NB15	92.8
[**?**]	CNN-LSTM	UNSW-NB15	96.5
[**?**]	Ensemble	UNSW-NB15	97.1
Proposed	Hybrid IF + RF	UNSW-NB15	**98.7**

The performance advantage of the proposed method can be attributed to two primary factors. First, the two-stage hybrid architecture leverages the anomaly-scoring capability of Isolation Forest to eliminate easily classified normal traffic before engaging the supervised Random Forest classifier, thereby directing computational resources toward genuinely ambiguous instances. Second, Random Forest's ensemble structure, built upon bagged decision trees with randomized feature selection, is inherently well-suited to the high-dimensional, heterogeneous feature space of the UNSW-NB15 dataset. Together, these properties yield a detector that is both more accurate and more generalizable than single-model approaches reported in the literature.

XIII. ADVANTAGES

The proposed hybrid protocol-based network anomaly detection system provides several advantages over conventional intrusion detection approaches. By integrating Isolation Forest (IF) and Random Forest (RF) within a unified framework, the system combines unsupervised anomaly detection with supervised classification to improve detection effectiveness in modern network environments.

- **High Detection Accuracy:** The hybrid IF+RF model achieves a detection accuracy of 98.7% on the UNSW-NB15 benchmark dataset. This performance surpasses that of several individual classifiers evaluated in this study. The hybrid strategy benefits from the anomaly detection capability of Isolation Forest and the strong classification performance of Random Forest.
- **Reduced False Positive Rate:** The combination of anomaly detection and supervised classification reduces the occurrence of false alarms. By filtering suspicious traffic before classification, the system minimizes unnecessary alerts and improves the reliability of detection results.
- **Real-Time Detection Capability:** The proposed architecture supports near real-time traffic analysis through efficient preprocessing and lightweight model inference. This enables the system to detect abnormal traffic patterns promptly and supports timely responses to potential security threats.
- **Protocol-Aware Detection:** The framework incorporates protocol-aware traffic analysis by considering behavioral characteristics of common network protocols such as TCP, UDP, HTTP, and DNS. This allows the detection system to identify subtle deviations in protocol behavior that may indicate sophisticated network attacks.
- **Scalability:** The modular architecture of the proposed framework allows the system to scale efficiently across distributed network environments. Additional detection modules or classifiers can be integrated with minimal architectural modification, making the system suitable for both small enterprise networks and large-scale data center infrastructures.
- **Adaptability:** The system supports periodic retraining and model updates, allowing it to adapt to evolving network traffic patterns and emerging attack techniques. This adaptability helps maintain consistent detection performance over time.

XIV. LIMITATIONS

Despite its demonstrated effectiveness, the proposed system is subject to several limitations that should be acknowledged and addressed in future research.

- **Computational Overhead:** The hybrid architecture combines two learning models together with protocol-specific preprocessing modules, which introduces additional computational overhead. In environments with extremely high network traffic volumes, this overhead may require dedicated hardware resources or hardware acceleration to maintain acceptable processing latency.
- **Dependency on Training Data Quality:** The performance of the Random Forest classifier strongly depends on the quality, diversity, and balance of the training dataset. If the training data is biased or lacks sufficient representation of certain attack types, the detection system may fail to identify rare or emerging threats.
- **Limited Protocol Coverage:** The protocol-aware detection modules rely on behavioral patterns associated with well-known network protocols. In cases where traffic is encrypted, obfuscated, or based on proprietary protocols, extracting meaningful behavioral features may become difficult, potentially reducing detection effectiveness.
- **Requirement for Periodic Retraining:** Network traffic patterns and attack strategies evolve over time. Consequently, the trained models may gradually lose effectiveness if they are not periodically updated. Maintain- ing high detection accuracy therefore requires periodic

retraining with updated datasets, which introduces additional operational effort.

- **Scalability in High-Throughput Environments:** Although the proposed system performs well under moderate traffic conditions, extremely high-throughput environments—such as backbone network infrastructures or large-scale cloud data centers—may expose perfor- mance bottlenecks in the feature extraction and classifi- cation stages. Addressing these limitations may require distributed processing architectures or optimized data pipelines.

XV. Future Work

Several promising directions exist for extending and improving the proposed hybrid network anomaly detection framework. One important direction involves integrating advanced deep learning techniques, particularly Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, which have demonstrated strong capabilities in capturing spatial and temporal patterns in network traffic data. Future work will investigate replacing or augmenting the Random Forest classifier with CNN–LSTM hybrid architectures in order to improve the detection of complex and multi-stage cyber attacks.

In addition, transformer-based models will be explored due to their recent success in natural language processing and sequential data modeling. These models may enable more effective learning of long-range dependencies within network traffic flows. Attention mechanisms could also enhance the interpretability of the intrusion detection system by highlighting the most influential features responsible for classification decisions.

Another important research direction involves deploying the detection system in distributed and Internet of Things (IoT) environments. Real-time implementation on resource-constrained edge devices, such as embedded processors and field-programmable gate arrays (FPGAs), will require model optimization techniques including model compression, quantization, and pruning. These techniques will help maintain detection accuracy while reducing computational requirements.

Furthermore, federated learning will be explored as a privacy-preserving training strategy for intrusion detection models. This approach enables collaborative learning across multiple distributed network nodes without requiring centralized storage of sensitive traffic data. Such a framework is particularly relevant for IoT ecosystems, where devices are heterogeneous and data sharing may be restricted due to privacy or regulatory constraints.

Finally, future work will focus on extending the protocol-aware detection modules to support IoT-specific communication protocols such as MQTT, CoAP, and Zigbee. Incorporating these protocols will allow the proposed system to better address the security requirements of emerging environments including smart homes, industrial IoT systems, and critical infrastructure networks.

XVI. Conclusion

This paper presented a hybrid network anomaly detection system that integrates Isolation Forest and Random Forest within a protocol-aware machine learning framework. By combining protocol-specific feature extraction with an ensemble classification strategy, the proposed approach addresses several limitations of conventional single-model intrusion detection systems, including susceptibility to false positives and reduced sensitivity to subtle network anomalies.

The proposed framework was evaluated using the UNSW-NB15 benchmark dataset, which provides a diverse and realistic representation of modern network traffic and attack scenarios. Experimental results demonstrate that the hybrid IF+RF model achieves a detection accuracy of 98.7%, while also improving precision, recall, and F1-score compared with individual Isolation Forest and Random Forest models as well as other conventional classifiers evaluated in this study.

The main contributions of this work can be summarized as follows. First, a protocol-aware feature engineering pipeline was developed to extract and normalize behavioral characteristics associated with common network protocols such as TCP, UDP, HTTP, and DNS. Second, a hybrid detection architecture was proposed that combines the anomaly isolation capability of Isolation Forest with the supervised classification strength of Random Forest, enabling effective detection of both known and previously unseen attack patterns. Third, comprehensive experimental evaluation and comparative analysis were performed to validate the effectiveness and robustness of the proposed approach.

Overall, the results indicate that hybrid machine learning architectures provide a promising direction for improving the accuracy and reliability of network intrusion detection systems. Future work will focus on integrating deep learning models, supporting edge-based deployment, and exploring federated learning approaches to further enhance scalability, adaptability, and privacy in large-scale network environments.

References

[1] R. Chauhan and S. Shah Heydari, "Polymorphic adversarial DDoS attack on IDS using GAN," in *Proc. IEEE Int. Symp. Networks, Computers and Communications (ISNCC)*, Montreal, QC, Canada, 2020, pp. 1–6.

[2] S. Naseer, Y. Saleem, S. Khalid, M. K. Bashir, J. Han, M. M. Iqbal, and K. Han, "Enhanced network anomaly detection based on deep neural networks," *IEEE Access*, vol. 6, pp. 48231–48246, 2018.

[3] M. A. Ferrag, L. Maglaras, A. Ahmim, M. Derdour, and H. Janicke, "RDTIDS: Rules and decision tree-based intrusion detection system for Internet-of-Things networks," *Future Internet*, vol. 12, no. 3, p. 44, 2020.

[4] G. Kang, J. Yan, J. Liu, and W. Wang, "Ensemble-based intrusion detection system using hybrid feature selection and optimized random forest," *IEEE Access*, vol. 9, pp. 99998–100014, 2021.

[5] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems," in *Proc. Military Communications and Information Systems Conf. (MilCIS)*, Canberra, ACT, Australia, 2015, pp. 1–6.

[6] A. Farnaaz and M. A. Jabbar, "Random forest modeling for network intrusion detection system," *Procedia Computer Science*, vol. 89, pp. 213–217, 2016.

[7] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Pisa, Italy, 2008, pp. 413–422.

[8] H. Liu and R. Setiono, "Feature selection and classification—A probabilistic wrapper approach," in *Proc. Int. Conf. Industrial Engineering Applications of Artificial Intelligence and Expert Systems*, 1996, pp. 419–424.

[9] Y. Mirsky, T. Doitshman, Y. Elovici, and A. Shabtai, "Kitsune: An ensemble of autoencoders for online network intrusion detection," in *Proc. Network and Distributed Systems Security Symp. (NDSS)*, San Diego, CA, USA, 2018, pp. 1–15.

[10] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep learning approach for intelligent intrusion detection system," *IEEE Access*, vol. 7, pp. 41525–41550, 2019.