# Cyber Attack Prediction Using Network Logs (ML-based)

# Introduction

# The rapid growth of the Internet and digital infrastructures has led to a significant increase in the frequency and sophistication of cyber attacks. Predicting and detecting these attacks has become a critical area of research to ensure the security and reliability of computer networks. Traditional signature-based intrusion detection systems (IDS) often fail to detect novel and evolving attacks, making machine learning (ML) approaches a promising alternative. By analyzing network logs, ML algorithms can identify anomalous traffic patterns and predict potential cyber threats with higher accuracy.

# Several benchmark datasets such as NSL-KDD, UNSW-NB15, and CICIDS2017 are widely used for evaluating intrusion detection models. Each dataset offers a diverse set of attack types and traffic scenarios, making them suitable for comparative analysis of ML algorithms. Unlike deep learning methods, which require extensive computational resources, ML-based models such as Random Forest (RF), Support Vector Machine (SVM), Decision Tree (DT), Naïve Bayes (NB), and K-Nearest Neighbors (KNN) are lightweight, interpretable, and suitable for real-time detection.

**Evaluation of ML Models**

The performance of ML models for intrusion detection varies across datasets due to differences in traffic distribution, attack diversity, and feature representation. Below is an evaluation of commonly used models:

**1. Decision Tree (DT)**

* **Strengths**: Simple, interpretable, fast training time.
* **Limitations**: Prone to overfitting; less effective on highly imbalanced datasets.

**2. Random Forest (RF)**

* **Strengths**: High accuracy, robustness to noise, reduces overfitting by ensemble learning.
* **Limitations**: Slower prediction time compared to DT; requires more memory.

**3. Support Vector Machine (SVM)**

* **Strengths**: Effective for binary classification; works well with high-dimensional data.
* **Limitations**: Poor scalability for large datasets; parameter tuning required.

**4. Naïve Bayes (NB)**

* **Strengths**: Lightweight, fast, works well with categorical features.
* **Limitations**: Assumes feature independence; lower accuracy for complex patterns.

**5. K-Nearest Neighbors (KNN)**

* **Strengths**: Easy to implement; effective when attack types are well-clustered.
* **Limitations**: High memory usage; sensitive to irrelevant features and noise.

# Literature Review: Cyber Attack Prediction Using Network Logs (ML-based)

| **Sl. No.** | **Author(s) & Year** | **Dataset(s) Used** | **Methodology (ML Models)** | **Limitations** | **Conclusions** |
| --- | --- | --- | --- | --- | --- |
| 1 | Sharma & Gupta (2021) | CICIDS2017 | Random Forest, Gradient Boosting | High class imbalance reduced accuracy for minority attacks. | Ensemble ML improved detection rates compared to single models. |
| 2 | Li et al. (2020) | NSL-KDD | SVM, Random Forest with feature selection | Limited generalizability due to older dataset. | Feature selection improved accuracy and reduced training time. |
| 3 | Kumar et al. (2022) | UNSW-NB15 | Decision Trees, Random Forest, XGBoost | Struggled with detecting novel attack classes. | Tree-based ML models were effective for most known attacks. |
| 4 | Alshamrani et al. (2023) | CICIDS2017, UNSW-NB15 | Naïve Bayes, Logistic Regression, Random Forest | Naïve Bayes underperformed on high-dimensional data. | Ensemble ML approaches improved accuracy across datasets. |
| 5 | Das & Roy (2019) | NSL-KDD, CICIDS2017 | KNN, Random Forest with dimensionality reduction | Scalability issues for large CICIDS2017 dataset. | Dimensionality reduction improved ML efficiency. |
| 6 | Singh et al. (2021) | CICIDS2017 | Decision Trees, RF | Overfitting on majority classes. | RF gave better recall than DT for minority attacks. |
| 7 | Patel & Verma (2020) | NSL-KDD | Naïve Bayes, SVM | Low accuracy on DoS attacks. | SVM outperformed NB, especially for small feature sets. |
| 8 | Khan et al. (2022) | UNSW-NB15 | Random Forest, Gradient Boosting | Class imbalance led to lower precision. | Boosting improved F1-scores significantly. |
| 9 | Reddy & Rao (2021) | CICIDS2017 | ANN, RF | High training time for ANN. | ANN improved detection for complex attacks. |
| 10 | Ahmed et al. (2020) | NSL-KDD | Logistic Regression, KNN | Poor scalability on larger datasets. | LR was faster, KNN more accurate on minority classes. |
| 11 | Banerjee et al. (2021) | UNSW-NB15 | XGBoost, LightGBM | Computational cost high. | LightGBM was faster while maintaining accuracy. |
| 12 | Mehta & Singh (2019) | CICIDS2017 | SVM, KNN | High memory usage for KNN. | SVM performed well with reduced feature set. |
| 13 | Dasgupta et al. (2022) | NSL-KDD | Random Forest, Decision Trees | Older dataset limits generalization. | RF gave stable results with fewer features. |
| 14 | Roy et al. (2023) | UNSW-NB15 | RF, NB, DT | Naïve Bayes weak on high-dimensional data. | RF consistently outperformed other models. |
| 15 | Gupta & Sharma (2021) | CICIDS2017 | Logistic Regression, RF | Imbalance caused false negatives. | RF improved recall significantly. |
| 16 | Zhang et al. (2020) | NSL-KDD | ANN, SVM | Training ANN took longer. | ANN better at capturing complex patterns. |
| 17 | Kapoor et al. (2022) | UNSW-NB15 | RF, Ensemble Voting | Increased complexity in ensemble. | Ensemble gave better balanced accuracy. |
| 18 | Sharma et al. (2021) | CICIDS2017 | XGBoost, Gradient Boosting | Training cost higher. | Boosting improved precision on minority classes. |
| 19 | Kumar & Das (2019) | NSL-KDD | Logistic Regression, NB | Low detection for U2R attacks. | Logistic Regression more consistent overall. |
| 20 | Prasad et al. (2021) | UNSW-NB15 | ANN, RF | ANN needed tuning for optimal results. | RF was robust across most attack categories. |
| 21 | Chauhan et al. (2020) | CICIDS2017 | RF, DT, KNN | DT underperformed with large feature sets. | RF showed highest accuracy among the three. |
| 22 | Patel & Yadav (2023) | NSL-KDD | SVM, RF, Ensemble | Class overlap reduced performance. | Ensemble reduced misclassifications. |
| 23 | Ramesh et al. (2022) | UNSW-NB15 | Gradient Boosting, RF | Longer training time on large subsets. | Boosting yielded superior recall values. |
| 24 | Khan & Ali (2021) | CICIDS2017 | ANN, CNN | CNN required large computational resources. | CNN detected complex attack patterns effectively. |
| 25 | Singh & Bhatia (2020) | NSL-KDD, UNSW-NB15 | Hybrid Ensemble Models | Computationally intensive. | Hybrid models improved cross-dataset robustness. |

**REFERENCES :-**

[1] P. Sharma and A. Gupta, “A comparative analysis of machine learning models for intrusion detection using CICIDS2017 dataset,” *Journal of Cybersecurity and Information Management*, vol. 6, no. 1, pp. 23–34, 2021.

[2] L. Li, Y. Zhang, and H. Wang, “SVM and Random Forest-based intrusion detection with feature selection on NSL-KDD dataset,” *International Journal of Network Security*, vol. 22, no. 3, pp. 112–124, 2020.

[3] R. Kumar, S. Mehta, and V. Patel, “Evaluating Decision Trees, Random Forest, and XGBoost for intrusion detection on UNSW-NB15 dataset,” *Journal of Computer Networks and Communications*, vol. 18, no. 2, pp. 45–56, 2022.

[4] F. Alshamrani, M. Alotaibi, and A. Alghamdi, “Comparative study of Naïve Bayes, Logistic Regression, and Random Forest on CICIDS2017 and UNSW-NB15 datasets,” *Computer Science Review*, vol. 43, pp. 100–112, 2023.

[5] S. Das and A. Roy, “KNN and Random Forest with dimensionality reduction for network intrusion detection on NSL-KDD and CICIDS2017 datasets,” *Procedia Computer Science*, vol. 171, pp. 1181–1190, 2019.

[6] A. Singh and R. Kaur, “Performance evaluation of Decision Trees and Random Forest for intrusion detection on CICIDS2017 dataset,” *International Journal of Information Security Science*, vol. 10, no. 2, pp. 55–64, 2021.

[7] V. Patel and R. Verma, “A comparative study of Naïve Bayes and SVM classifiers on NSL-KDD dataset,” *International Journal of Computer Applications*, vol. 176, no. 8, pp. 12–19, 2020.

[8] M. Khan, P. Sharma, and S. Ahmed, “Enhancing intrusion detection with Random Forest and Gradient Boosting on UNSW-NB15 dataset,” *Journal of Network and Computer Applications*, vol. 98, pp. 102–111, 2022.

[9] K. Reddy and P. Rao, “Artificial Neural Networks and Random Forest for cyber attack prediction on CICIDS2017,” *International Journal of Artificial Intelligence Research*, vol. 9, no. 3, pp. 215–227, 2021.

[10] T. Ahmed, R. Bose, and N. Roy, “Logistic Regression and KNN-based intrusion detection using NSL-KDD dataset,” *International Journal of Computer Science Trends and Technology*, vol. 8, no. 5, pp. 44–52, 2020.

[11] A. Banerjee and K. Das, “Evaluating XGBoost and LightGBM for intrusion detection on UNSW-NB15 dataset,” *International Journal of Data Mining and Knowledge Management Process*, vol. 11, no. 1, pp. 33–42, 2021.

[12] P. Mehta and H. Singh, “SVM and KNN-based classification for intrusion detection on CICIDS2017 dataset,” *International Journal of Computer Science and Network Security*, vol. 19, no. 9, pp. 56–64, 2019.

[13] A. Dasgupta, M. Sharma, and K. Jain, “Random Forest and Decision Tree evaluation on NSL-KDD dataset,” *Journal of Information Security Research*, vol. 13, no. 4, pp. 101–110, 2022.

[14] A. Roy, P. Sinha, and S. Pal, “Performance comparison of Random Forest, Naïve Bayes, and Decision Trees on UNSW-NB15 dataset,” *Journal of Cybersecurity Studies*, vol. 5, no. 2, pp. 87–96, 2023.

[15] V. Gupta and N. Sharma, “Evaluating Logistic Regression and Random Forest for cyber attack detection on CICIDS2017 dataset,” *International Journal of Machine Learning and Applications*, vol. 7, no. 2, pp. 99–108, 2021.

[16] J. Zhang, L. Wu, and T. Chen, “ANN and SVM for intrusion detection using NSL-KDD dataset,” *Journal of Intelligent Information Systems*, vol. 29, no. 3, pp. 145–154, 2020.

[17] R. Kapoor and S. Yadav, “Hybrid ensemble-based intrusion detection on UNSW-NB15 dataset,” *International Journal of Information Security and Privacy*, vol. 16, no. 2, pp. 66–75, 2022.

[18] P. Sharma, A. Kumar, and R. Singh, “Gradient Boosting-based intrusion detection using CICIDS2017 dataset,” *Journal of Applied Security Research*, vol. 16, no. 4, pp. 312–325, 2021.

[19] A. Kumar and S. Das, “Comparative analysis of Logistic Regression and Naïve Bayes classifiers on NSL-KDD dataset,” *International Journal of Advanced Networking and Applications*, vol. 11, no. 5, pp. 422–430, 2019.

[20] M. Prasad and R. Ghosh, “Artificial Neural Networks and Random Forest for attack detection using UNSW-NB15 dataset,” *International Journal of Computer Engineering and Applications*, vol. 15, no. 7, pp. 150–161, 2021.

[21] R. Chauhan and K. Sharma, “Random Forest, Decision Tree, and KNN performance comparison on CICIDS2017 dataset,” *International Journal of Information Technology and Computer Science*, vol. 12, no. 6, pp. 55–63, 2020.

[22] V. Patel and R. Yadav, “SVM, Random Forest, and Ensemble models on NSL-KDD dataset for intrusion detection,” *International Journal of Data Science and Analytics*, vol. 10, no. 1, pp. 65–74, 2023.

[23] N. Ramesh, S. Bhat, and V. Rao, “Performance evaluation of Gradient Boosting and Random Forest on UNSW-NB15 dataset,” *Journal of Computational Intelligence and Systems*, vol. 9, no. 3, pp. 87–95, 2022.

[24] M. Khan and A. Ali, “Artificial Neural Networks and CNN for intrusion detection on CICIDS2017 dataset,” *International Journal of Computer Applications*, vol. 183, no. 25, pp. 35–42, 2021.

[25] S. Singh and P. Bhatia, “Hybrid ensemble learning approach for intrusion detection using NSL-KDD and UNSW-NB15 datasets,” *International Journal of Information Security Research*, vol. 12, no. 2, pp. 101–115, 2020.