Towards a Lightweight and Efficient Gaussian Mixture Model for Detecting Mirai Botnet Attacks in IoT Environments

Boutra Brahim
LASTIC Laboratory
University of Batna 2
Batna, Algeria
b.boutra@univ-batna2.dz

Khaled Hamouid
LIGM, ESIEE Paris
Université Gustave Eiffel
93160 Noisy-le-Grand, France
khaled.hamouid@esiee.fr

Mawloud Omar
IRISA Laboratory
Université Bretagne Sud
56000 Vannes, France
mawloud.omar@univ-ubs.fr

Mohamed Rahouti
Fordham University
113 W 60th Street
New York, NY, USA
mrahouti@fordham.edu

Hamza Drid
LAMIE Laboratory
University of Batna 2
05000 Batna, Algeria
h.drid@univ-batna2.dz

***Abstract*—Internet of Things (IoT) devices are increasingly susceptible to botnet threats, with Mirai-based attacks posing significant security challenges. To address these concerns, we present a lightweight anomaly detection model based on a Gaussian Mixture Model (GMM), tailored for detecting Mirai botnet traffic. By leveraging anomaly detection techniques, our approach identifies malicious activity with optimized latency and computational efficiency, making it suitable for resource-constrained IoT environments. Trained specifically on Mirai-related traffic, the proposed model achieves comparable detection accuracy while significantly improving latency compared to traditional decision tree-based models. These results underscore the effectiveness of the GMM approach in balancing detection performance and real-time responsiveness for IoT applications.**

***Index Terms*—Mirai Botnet Attacks, Attack Detection, Gaussian Mixture Model, Anomaly Detection**

I. Introduction

The Internet of Things (IoT) has revolutionized our interaction with technology, facilitating connectivity and automation across various domains such as smart homes, healthcare, transportation, and industrial automation. However, this rapid expansion of IoT devices has also exposed them to significant security threats. Many devices are deployed with default configurations and weak credentials, making them vulnerable to attacks.

Among these threats, the Mirai botnet stands out for its formidable effectiveness. Emerging in 2016 [1], this threat demonstrated how poorly protected IoT devices can be hijacked to launch large-scale Distributed Denial of Service (DDoS) attacks. The Mirai botnet exploits device weaknesses by using default credentials, generating massive traffic that can cripple critical online services and cause significant disruptions to Internet infrastructure. This attack not only highlights the vulnerability of IoT devices but also the growing difficulty of detecting such threats due to the immense volume of traffic generated and the complexity of their behavior.

Many studies focus on using specific datasets to detect botnet attacks, such as IoTID20 [2] and Kitsune [3] datasets. These datasets help analyze traffic patterns and identify abnormal behaviors associated with attacks like the Mirai botnet. Concurrently, other research explores applying machine learning models to develop detection schemes tailored to the specifics of botnet attacks [4]. However, despite the promising advances, machine learning-based solutions must still overcome significant challenges related to the limited capabilities of IoT devices, such as restricted computing power and resource constraints, which can impact their overall effectiveness. Therefore, it is crucial to develop solutions that improve prediction times to meet the specific needs of IoT networks while maintaining a high level of attack detection accuracy.

In this paper, we propose a Mirai attack detection solution based on a Gaussian Mixture Model (GMM) using the IoT2021 dataset. This solution is specifically designed to enhance response speed and optimize prediction times in IoT environments. Our lightweight and efficient model aims to provide rapid detection of Mirai attacks while maintaining acceptable accuracy. The goal is to ensure optimal performance for real-time applications, thereby addressing the critical needs of IoT networks.

II. Related Work

Several works focus on designing machine learning approaches for detecting Mirai botnets in IoT environments. This section presents some related studies on Mirai botnet detection in IoT contexts. It is important to note that our work explicitly aims to improve prediction time while

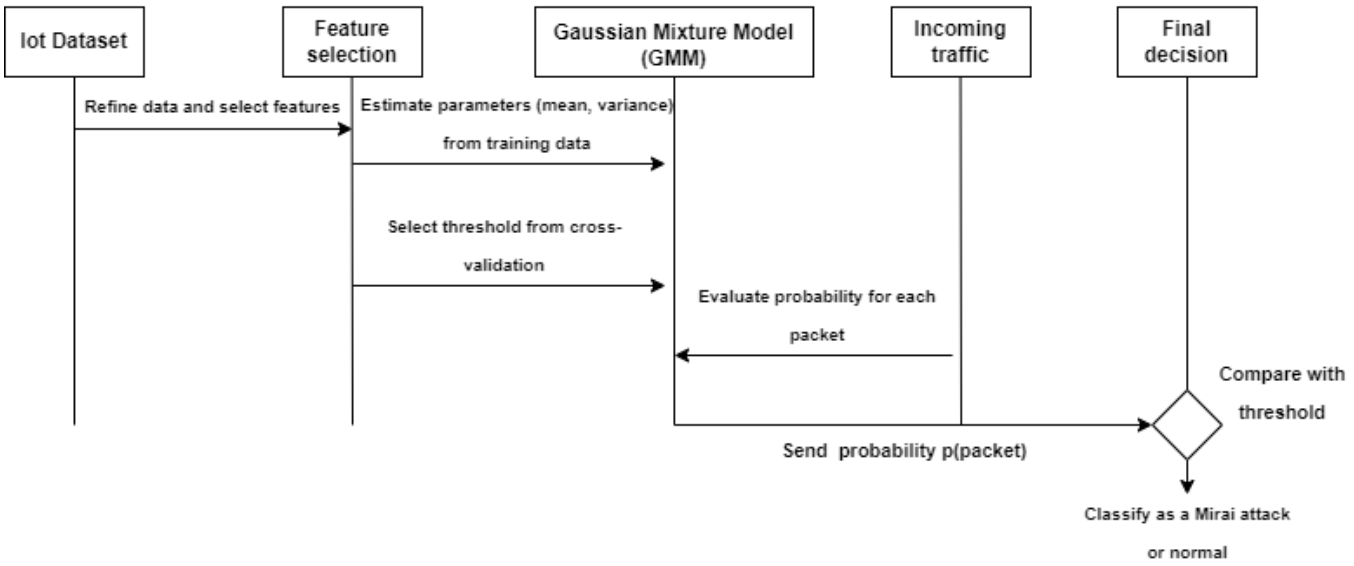

Fig. 1. GMM model design

maintaining an acceptable detection rate. In [5], the authors introduce an attack detection scheme for IoT devices using the Auto-Associative Dense Random Neural Network (AA-Dense RNN). This approach stands out for its training with normal traffic to detect Mirai attacks. Experimental results demonstrate a high accuracy of 99.84%, surpassing Lasso and KNN techniques. The AA-Dense RNN provides acceptable computation times. Suggestions for future improvements are also proposed. However, the study does not evaluate the prediction time in different data sizes.

In [6], the authors present an innovative deep learning-based solution for detecting botnet activities in consumer IoT devices. Employing a BLSTM-RNN model with Word Embedding, they compare its performance against a unidirectional LSTM-RNN in detecting Mirai botnet attacks. Despite a slight increase in processing time, the bidirectional approach demonstrates progressively superior results. The paper underscores the relevance of this approach for enhancing IoT botnet detection and outlines future research directions while making the Mirai botnet dataset publicly available. However, the study does not provide any measurement of prediction time, which limits its assessment for use in real-time scenarios.

Finally, in [7], Omar *et al.* compare seven supervised machine learning models, namely KNeighbors, Decision Tree, Random Forest, Support Vector Machine (SVM), Linear Discriminant Analysis (LDA), Stochastic Gradient Descent (SGD), and Multi-Layer Perceptron (MLP). They focus on the IoTID20 dataset, evaluating these models in terms of detection rate, model size, and prediction delay, with an emphasis on the efficiency of each model on these specific metrics. The results highlight the high performance of the decision tree model.

Recent studies have also explored advanced techniques for detecting DDoS and botnet attacks in IoT environments. In [8], the authors propose a hybrid CNN-SVM approach for real-time detection of DoS and DDoS attacks, focusing on improving detection accuracy while addressing latency constraints. Similarly, the study in [9] employs a snake optimizer with ensemble learning to enhance DDoS attack detection in IoT systems. Both approaches highlight the significance of lightweight yet accurate models for resource-constrained IoT networks, aligning with the objectives of our work.

In addition, anomaly detection methods have been widely applied for detecting malicious traffic in IoT networks. For instance, [10] presents an anomaly detection framework tailored to prevent DDoS attacks in fog-enabled IoT networks, leveraging machine learning for early detection. Another study [11] focuses on detecting IoT-generated DDoS traffic using a novel approach that integrates clustering and feature selection techniques, achieving high accuracy with reduced computational overhead.

These studies collectively demonstrate the growing emphasis on balancing detection performance and computational efficiency, particularly for IoT environments characterized by limited resources and real-time demands. Our work extends this research by employing a GMM model, optimizing prediction time, and maintaining robust detection accuracy tailored to the specific characteristics of Mirai botnet traffic. Moreover, our model was compared to the one selected in the study [7], which evaluated several algorithms and identified the Decision Tree model as the optimal solution. The results show that our GMM-based approach achieves comparable detection performance while significantly reducing prediction time.

Feature Name	Description
Flow_Duration	Duration of the connection in seconds
Fwd_IAT_Std	Standard deviation of inter-arrival times between packets sent
Fwd_IAT_Mean	Mean inter-arrival time between packets sent
Fwd_IAT_Tot	Total inter-arrival time between packets sent
Flow_IAT_Mean	Mean inter-arrival time of packets
Flow_IAT_Std	Standard deviation of inter-arrival times of packets
Bwd_IAT_Mean	Mean inter-arrival time between packets received
Bwd_IAT_Std	Standard deviation of inter-arrival times between packets received
Fwd_Header_Len	Length of the packet headers

TABLE I
FEATURE SELECTION.

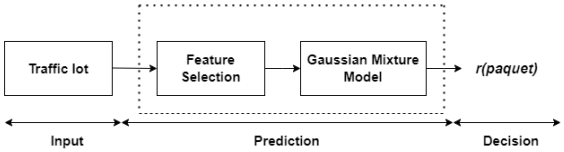

Fig. 2. Architecture of the proposed Mirai botnet attack detection approach

III. Proposed Mirai Botnet Attack Detection Approach

In this section, we describe the model used to detect Mirai attacks in an IoT environment. Our approach relies on the use of the Gaussian Mixture Model (GMM), a probabilistic model that represents the underlying distribution of the data as a combination of several Gaussian distributions. The effectiveness of using GMM for anomaly detection has been demonstrated in various works, such as in [12], where GMM was successfully applied to identify anomalies in network traffic. Figure 2 illustrates the detection process and architecture of our proposed approach.

In the next subsections, we will explain the approach used to design the GMM-based detection model (illustrated in Figure 1), detailing the steps involved in constructing and applying the Gaussian Mixture Model for detecting Mirai traffic. This approach is illustrated in Algorithm 1.

A. GMM model parameters

For each feature f_i (illustrated in Table I), we estimate the parameters of the normal distribution, namely the mean μ_f and the variance σ_f^2. These parameters are computed using the values of the features extracted from the training dataset. In addition, we consider the *Threshold* parameter used for attack detection.

- **Mean** μ_{f_i} :

$$\mu_{f_i} = \frac{1}{n}\sum_{j=1}^{n} x_j \tag{1}$$

where x_j is the j-th value of the feature f_i.

- **Variance** $\sigma_{f_i}^2$:

$$\sigma_{f_i}^2 = \frac{1}{n}\sum_{j=1}^{n}(x_j - \mu_{f_i})^2 \tag{2}$$

- **Threshold** ϵ : The threshold value is used to determine whether a given traffic is classified as a Mirai attack based on whether the derived probability from the GMM parameters is below or above this threshold. In the subsequent sections, we detail how to determine this threshold value.

B. Data description and preprocessing

We used the IoTID20 dataset, which had been refined by reducing it to 23 features [7] while focusing on normal and Mirai attack records. This refinement aimed to maximize the performance of machine learning models by enabling them to achieve peak effectiveness.

In our approach, we applied two additional preprocessing steps:

1) **Mirai-Focused Training:** We retained only Mirai attack instances for the training data, ensuring that the Gaussian model is specifically tuned to recognize attack patterns.
2) **Feature Selection:** We performed a feature selection process to improve model performance, ensuring that only the most relevant features are used. The final features included in the model are listed in Table 01, with further explanation provided in Section IV-A.

C. Mirai traffic detection model

Once the GMM model is trained, it is used to evaluate the probability of each data packet. For each incoming traffic, features are extracted, and then the probability of each feature f_i is computed according to the estimated normal distribution as follows:

$$p(f_i; \mu_{f_i}, \sigma_{f_i}^2) = \frac{1}{\sqrt{2\pi\sigma_{f_i}^2}} \exp\left(-\frac{(x_i - \mu_{f_i})^2}{2\sigma_{f_i}^2}\right) \tag{3}$$

The total probability of the packet is then determined by combining the probabilities of all its features:

Algorithm 1 Gaussian Mixture Model (GMM) for Mirai botnet attack detection.

Input: IoTID20 dataset D, features $F = \{f_1, f_2, ..., f_k\}$.
Output: Trained GMM model and detection threshold ϵ.
procedure TRAINING PHASE
    Step 1: Preprocess dataset D.
        // Remove irrelevant or noisy data.
    Step 2: Perform feature selection on D to retain F.
        // Select features based on their F1-score.
    Step 3: Split D into training (D_{train}) and validation (D_{val}) sets.
    Step 4: Filter D_{train} to include only Mirai attack data.
    Step 5: Compute Gaussian parameters for each feature $f_i \in F$:
        Mean: $\mu_{f_i} = \frac{1}{|D_{\text{train}}|} \sum_{x \in D_{\text{train}}} x_{f_i}$
        Variance: $\sigma^2_{f_i} = \frac{1}{|D_{\text{train}}|} \sum_{x \in D_{\text{train}}} (x_{f_i} - \mu_{f_i})^2$
end procedure
procedure VALIDATION PHASE
    Step 6: Use D_{val} to determine optimal detection threshold ϵ.
        Compute F1-scores for various ϵ values.
        Select ϵ that maximizes F1-score.
end procedure
procedure DETECTION PHASE
    Input: Incoming packet p, trained GMM parameters $\{\mu_{f_i}, \sigma^2_{f_i}\}$, and threshold ϵ.
    Step 7: Compute the probability of each feature f_i:

$$p(f_i; \mu_{f_i}, \sigma^2_{f_i}) = \frac{1}{\sqrt{2\pi\sigma^2_{f_i}}} \exp\left(-\frac{(x_{f_i} - \mu_{f_i})^2}{2\sigma^2_{f_i}}\right)$$

    Step 8: Compute the total probability of p:

$$p(p) = \prod_{i=1}^{k} p(f_i; \mu_{f_i}, \sigma^2_{f_i})$$

    Step 9: Classify p as:

$$\text{Mirai Attack, if } p(p) \geq \epsilon$$

$$\text{Normal trafic, if } p(p) < \epsilon$$

end procedure

$$p(\text{packet}) = \prod_{i=1}^{k} p(f_i; \mu_{f_i}, \sigma^2_{f_i}) \tag{4}$$

where k is the total number of features in the packet.

Finally, based on a predefined detection threshold, we flag the packet as either a Mirai attack or normal:

$$r(\text{packet}) = \begin{cases} \text{Mirai attack}, & \text{if } p(\text{packet}) \geq \epsilon \\ \text{Normal}, & \text{if } p(\text{packet}) < \epsilon \end{cases} \tag{5}$$

where ϵ is the detection threshold. The result $r(\text{packet})$ represents the final prediction about the packet, enabling the detection of Mirai attacks in the IoT environment.

IV. EXPERIMENT RESULTS

To evaluate the performance of our model, we used a dataset (IoTID20) reduced to 09 features, described in Table I, which includes both normal traffic and Mirai attack traffic. The dataset consists of 228,683 packet transmissions and was divided into 60% for training, 20% for cross-validation, and 20% for testing.

To optimize our model and select the prediction threshold ϵ, we employed cross-validation. This method ensured that the model generalizes well to unseen data and allowed us to fine-tune the prediction threshold based on validation set performance.

A. Impact of the feature selection on the model performance

We evaluated the impact of feature selection on the performance of our Mirai attack detection model, which is based on the Gaussian Mixture Model. Initially, we tested the model using the full set of 23 features on the test dataset. We then applied feature selection, retaining only features with an F1-score equal to 91% or higher. These features are illustrated in Table I.

The results show a notable enhancement in the model's performance with feature selection. The prediction time improved to 0.03 seconds compared to 0.04 seconds for the model without feature selection, indicating better efficiency and reduced latency. Additionally, feature selection led to better performance metrics in terms of precision, recall, and F1 Score. These improvements demonstrate that feature selection optimizes both detection and latency, as summarized in Table II.

Metric	With Feature Selection)	Without Feature Selection
Prediction Time (s)	0.0301	0.0446
Precision	0.9199	0.9156
Recall	0.9817	0.9807
F1-Score	0.9498	0.9470
Accuracy	0.9145	0.9097

TABLE II
COMPARISON OF MODEL PERFORMANCE WITH AND WITHOUT FEATURE SELECTION.

B. Determining of the Optimal Threshold

We conducted experiments to determine the optimal threshold value ϵ for our GMM-based Mirai detection model. The method used a validation dataset to assess various ϵ values and then select the threshold that provided the best F1-score for anomaly detection.

Figure 3 illustrates the F1-score for different ϵ values. We found that the optimal threshold was $\epsilon = 1.1125 \times 10^{-28}$, which yielded the highest F1-score of 94.95%. This threshold reflects the best accuracy in attack detection.

These results demonstrate that adjusting the ϵ threshold allows for achieving maximum model performance in terms of F1-score, highlighting the effectiveness of the chosen threshold for optimizing anomaly detection.

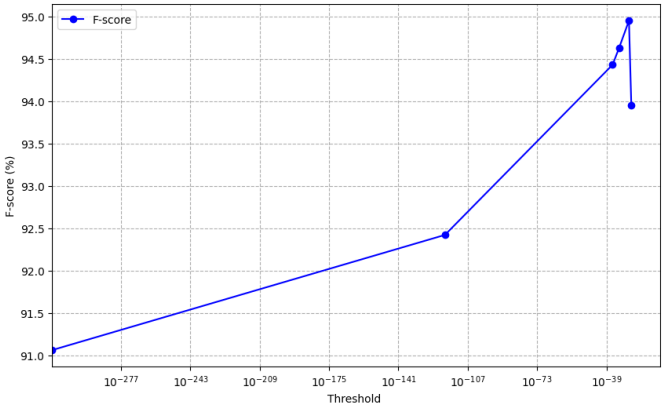

Fig. 3. F1-score for different ϵ values.

C. Comparison with Decision Tree Model

We compared our GMM-based Mirai attack detection model with a traditional machine learning model, namely the decision tree model, which had demonstrated strong performance in previous studies [7].

The comparison highlights prediction time as the primary metric while maintaining close precision. We evaluated the prediction times for data sizes ranging from 1 packet to 2000 packets. The average prediction times were measured on a machine equipped with an Intel Core i3 processor running at 3.6 GHz and 8 GB of RAM. For each model, the average prediction time was calculated by executing the prediction 1000 times for each data size and then averaging the measured times for these executions.

As shown in Figure 4, our GMM model consistently exhibited significantly lower average prediction times than the decision tree model across varying data sizes, from 1 to 2000 packets. This indicates that our GMM model is more efficient in terms of response time, even with increasing data volumes. These results highlight the effectiveness of the GMM for IoT applications with real-time and resource constraints, as it not only maintains comparable accuracy but also offers superior latency performance.

V. Conclusion

This paper introduced a Mirai attack detection approach based on the Gaussian Mixture Model. It is designed to enhance efficiency, especially the prediction time, while maintaining high detection accuracy. Our GMM model stands out for its ability to optimize response times, crucial for real-time detection in IoT environments.

Experimental results demonstrate that our GMM model, through precise feature selection and rigorous parameter estimation, significantly reduces prediction time compared to the traditional decision tree model. Although decision trees provide exemplary performance in terms of accuracy, they suffer from longer prediction times and computation costs. In contrast, our GMM approach reduces response times while maintaining comparable detection performance.

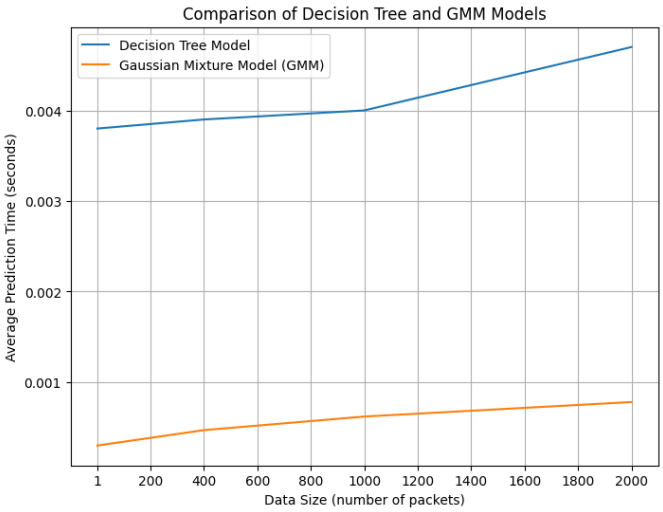

Fig. 4. Average prediction time for each Mirai attack detection method.

This improvement in speed and efficiency highlights the GMM's suitability for attack detection in real-time and resource-constrained applications. The GMM balances rapidity and accuracy, making it well-suited for demands in dynamic IoT environments.

Future work should involve extending our study to other types of attacks to validate our approach further. Exploring other optimization techniques could also improve prediction time and overall system performance.

References

[1] R. H. Hsu, J. Lee, T. Q. Quek, and J. C. Chen, "Reconfigurable security: Edge-computing-based framework for iot," *IEEE Network*, vol. 32, no. 5, pp. 92–99, 2018.

[2] I. Ullah and Q. H. Mahmoud, "A scheme for generating a dataset for anomalous activity detection in iot networks," in *Canadian Conference on Artificial Intelligence*. Cham: Springer International Publishing, May 2020, pp. 508–520.

[3] Y. Mirsky, T. Doitshman, Y. Elovici, and A. Shabtai, "Kitsune: An ensemble of autoencoders for online network intrusion detection," in *The Network and Distributed System Security Symposium (NDSS)*, 2018.

[4] A. Nazir, J. He, N. Zhu, A. Wajahat, X. Ma, F. Ullah, S. Qureshi, and M. S. Pathan, "Advancing iot security: A systematic review of machine learning approaches for the detection of iot botnets," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 35, no. 10, p. 101820, 2023.

[5] M. Nakip and E. Gelenbe, "Mirai botnet attack detection with auto-associative dense random neural network," in *2021 IEEE Global Communications Conference (GLOBECOM)*. IEEE, December 2021, pp. 01–06.

[6] C. D. McDermott, F. Majdani, and A. V. Petrovski, "Botnet detection in the internet of things using deep learning approaches," in *2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE, July 2018, pp. 1–8.

[7] M. Omar and L. George, "Toward a lightweight machine learning based solution against cyber-intrusions for iot," in *2021 IEEE 46th Conference on Local Computer Networks (LCN)*. IEEE, October 2021, pp. 519–524.

[8] Q. Al-Na'amneh, M. Aljaidi, A. Nasayreh, H. Gharaibeh, R. E. Al Mamlook, A. S. Jaradat, A. Alsarhan, and G. Samara, "Enhancing iot device security: Cnn-svm hybrid approach for real-time detection of dos and ddos attacks," *Journal of Intelligent Systems*, vol. 33, no. 1, p. 20230150, 2024.

[9] M. Aljebreen, H. A. Mengash, M. A. Arasi, S. S. Aljameel, A. S. Salama, and M. A. Hamza, "Enhancing ddos attack detection using snake optimizer with ensemble learning on internet of things environment," *IEEE Access*, 2023.
[10] D. K. Sharma, T. Dhankhar, G. Agrawal, S. K. Singh, D. Gupta, J. Nebhen, and I. Razzak, "Anomaly detection framework to prevent ddos attack in fog empowered iot networks," *Ad Hoc Networks*, vol. 121, p. 102603, 2021.
[11] I. Cvitić, D. Peraković, M. Periša, and M. Botica, "Novel approach for detection of iot generated ddos traffic," *Wireless Networks*, vol. 27, no. 3, pp. 1573–1586, 2021.
[12] D. Zamouche, S. Aissani, M. Omar, and M. Mohammedi, "Highly efficient approach for discordant bsms detection in connected vehicles environment," *Wireless Networks*, vol. 29, no. 1, pp. 189–207, 2023.