# APPLICATION LAYER DISTRIBUTED DENIAL OF SERVICES ATTACK DETECTION USING DEEP LEARNING MODELS

**Group-11 Team Members**

Akash Reddy Jammula Chaitanya Swarna

# Abstract:

The resilience of application layer services against Distributed Denial of Service (DDoS) attacks is of paramount importance in maintaining the integrity and availability of online services. This study introduces a deep learning-based detection framework that leverages the strengths of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks to identify application layer DDoS attacks. The framework utilizes the CICIDS2017 dataset, which includes a wide array of sophisticated and contemporary attack vectors, serving as a rigorous platform for model training and validation. The chosen deep learning models—CNN and LSTM—are adept at capturing spatial and temporal patterns within data, making them particularly suited for the nuanced task of DDoS detection. Our research aims to seamlessly augment existing network defenses by implementing this advanced analytical capability. Through rigorous testing and validation, the proposed models demonstrate a strong potential in effectively classifying network traffic and distinguishing between benign and malicious activities. The findings of this study contribute to the ongoing efforts to enhance cyber defense mechanisms using the power of deep learning.

# Introduction:

The cybersecurity landscape is increasingly threatened by sophisticated Distributed Denial of Service (DDoS) attacks that target multiple layers of the OSI model, exploiting vulnerabilities from the network level right up to the application layer. These attacks disrupt service availability, especially impacting the application layer due to its direct engagement with users. Our study is dedicated to developing effective detection mechanisms for these complex patterns of attacks using the CICIDS2017 dataset. We employ advanced deep learning techniques, specifically Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks, renowned for their capabilities in recognizing complex patterns and analyzing temporal sequences. Our models are designed to identify the subtle and complex behaviors characteristic of application layer DDoS attacks—behaviors that often elude traditional security measures and can closely mimic legitimate traffic, making detection particularly challenging. One of the main challenges we tackle is distinguishing these malicious activities from legitimate traffic, a task complicated by the constantly evolving and increasingly stealthy nature of DDoS strategies. By concentrating on the application layer and leveraging a deep learning-based approach, our research aims to enhance defenses and uphold the integrity of critical services against the dynamic threat posed by DDoS attacks.

## Confronting the Challenges and Limitations in DDoS Detection

Addressing application layer DDoS attacks with deep learning models, while promising, presents several challenges and limitations:

**Data Quality and Availability:** High-quality, diverse, and representative datasets like CICIDS2017 are crucial for training effective models. However, many datasets may not fully capture the latest attack patterns or might lack the variability needed to train models that can generalize well across different network environments.

**Model Complexity:** CNNs and LSTMs are computationally intensive, requiring significant resources for training and inference. This can be a limitation in environments where computational resources are constrained or where real-time response is critical.

**Adaptability:** DDoS attack vectors are continuously evolving, which means models trained on current datasets might quickly become outdated. Continuous retraining and model updating are necessary, which can be resource intensive.

**False Positives and Negatives:** Balancing the sensitivity of the model to detect attacks without generating too many false alarms is a critical challenge. High rates of false positives can disrupt normal operations, while high rates of false negatives can allow damaging attacks to proceed undetected.

**Scalability:** Scaling these models to operate efficiently across large and diverse networks can be challenging. Ensuring that the detection system can handle large volumes of traffic without degradation in performance or speed is crucial for practical deployment.

**Integration with Existing Systems**: Integrating advanced ML/DL models into existing cybersecurity infrastructures without causing disruptions can be complex. Compatibility with existing protocols and systems must be ensured to achieve seamless operation.

**Interpretability:** Deep learning models, especially those like CNNs and LSTMs, are often considered black boxes, meaning their decision-making process is not easily interpretable. This lack of transparency can be a barrier in security settings where understanding the reason behind a detection is as important as the detection itself.

## The Significance of Mitigating DDoS Threats

Addressing the problem of DDoS attacks, particularly at the application layer, is crucial for several reasons:

**Diversion Tactics:** DDoS attacks can serve as a distraction, allowing attackers to breach data systems unnoticed, leading to theft or corruption of sensitive data.

**Data Ransom and Extortion:** Attackers might leverage DDoS attacks to extort ransom by threatening to escalate or prolong the attack, endangering data integrity and access.

**Compliance Risks:** Many industries face stringent regulations on data protection. DDoS- induced breaches could lead to non-compliance, resulting in significant fines and legal issues.

**Monitoring and Encryption:** Implementing advanced monitoring to detect unusual data patterns and encrypting data are essential. Encryption ensures that, even if data is accessed, it remains unreadable without the proper keys.

**Secure Backup and Recovery:** Maintaining secure and readily accessible backups helps to quickly restore data compromised during a DDoS attack, minimizing downtime and data loss.

# Related Work

## Existing Solutions:

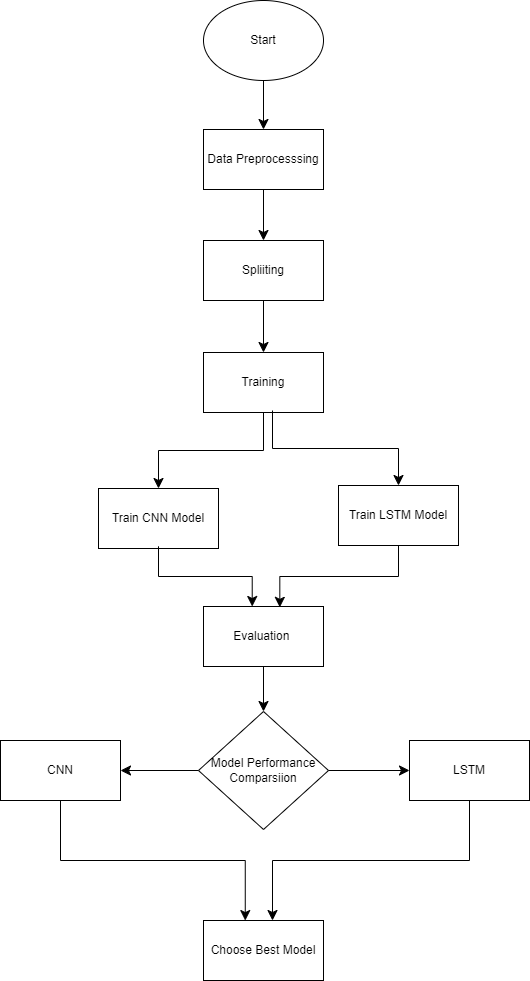
* Solutions using Long Short-Term Memory (LSTM) and fuzzy logic have been developed for detection and mitigation in SDN environments, showing high accuracy but focusing only on high-volume attacks.
* Combining supervised and unsupervised ML techniques has been proposed for detecting DDoS attacks, but these did not use current datasets and were limited in the range of methods and attack types explored.
* A framework using LSTM for slow-rate DDoS detection achieved high performance with specific datasets but had reduced efficacy when applied to different datasets.
* Multilayer Perceptron (MLP) neural networks have been used to detect HTTP-based slow-rate attacks, achieving significant detection rates, although they were not as effective in distinguishing specific types of attacks.

## Limitations of Existing Solutions:

* Many existing works have not utilized up-to-date datasets, which limits their effectiveness against new threats.
* Several studies have been conducted separately on transport or application layer attacks and not on both concurrently.
* Most existing approaches performed an offline analysis of their proposals using traffic captured from testbeds, which may not accurately reflect real-world network complexities and constraints.
* The accuracy of some ML/DL models decreases when the network topology changes, suggesting a need for retraining or adaptive models that can maintain high performance in varying network environments.

## System Design:

Our system design integrates Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) models, tailored for the specific challenge of detecting Application Layer Distributed Denial of Service (DDoS) attacks. The flowchart outlines a structured approach from data preprocessing to model evaluation, facilitating efficient detection and mitigation of DDoS threats. Each phase of the process is meticulously designed to maintain data fidelity, model robustness, and adaptability to evolving attack patterns. Continuous refinement mechanisms ensure that the deployed models remain effective in safeguarding against the dynamic nature of DDoS attacks, enhancing network security resilience over time.



## How is the system working? Data Preprocessing:

In the preprocessing phase, several steps are undertaken to prepare the data for model training. Feature selection involves choosing relevant attributes from the dataset that capture characteristics indicative of DDoS attacks. Finally, the dataset is split into training, validation, and testing sets for model development and evaluation.

## Model Training:

Two deep learning architectures, Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM), are designed and trained using the preprocessed data. The CNN model is tailored to learn spatial patterns and features from the network traffic data, while the LSTM model is optimized to capture temporal dependencies and sequences within the data.

## Model Evaluation:

Post-training, the models undergo rigorous evaluation in the Model Evaluation Module. Here, their performance is assessed using relevant evaluation metrics such as accuracy, precision, recall, and F1-score. This step serves to quantify the effectiveness of each model in accurately identifying DDoS attacks within the application layer.

## Model Comparison:

A comparative analysis is conducted to determine which deep learning model, CNN or LSTM, performs better in detecting DDoS attacks at the application layer. Based on the evaluation results, the model with superior performance is selected for further deployment and integration.

## Detection of Attack:

It focusses on alerting system administrators or security teams for further investigation and response. Once the attack is detected, the system sends an alert to the system administrator, enabling them to investigate the issue promptly and prevent potential data loss or service disruption.

In conclusion, the project demonstrates the effectiveness of utilizing deep learning models for detecting application layer DDoS attacks. By following the systematic approach outlined in this system design, the project contributes to enhancing network security and resilience against cyber threats, safeguarding critical infrastructure and services. Opportunities for future enhancements and research directions are also discussed to further improve the detection system's capabilities.

## Contributions by the Team for the Project:

Our team is currently engaged in the development of an application-layer distributed denial of service (DDoS) attack detection system. Akash Reddy spearheads the Data Preprocessing Module, focusing on optimizing the Convolutional Neural Network (CNN) architecture. Concurrently, Chaitanya is deeply involved in refining data preprocessing techniques and fine- tuning the Long Short-Term Memory (LSTM) architecture. Together, we are collaboratively establishing model selection criteria and gearing up for the execution phase.

**Dataset Description**

The CICIDS2017 dataset, developed by the Canadian Institute for Cybersecurity, is structured to resemble a week's worth of network traffic, capturing a variety of attack vectors and benign flows to facilitate the development and validation of intrusion detection systems. For our study, the dataset from Wednesday has been specifically utilized, which features various types of HTTP attack scenarios.

Wednesday Data Focus:

On this day, the dataset includes multiple attack types such as Slowloris, SlowHTTPTest, Hulk, and GoldenEye. Our analysis, however, selectively processes only Slowloris and SlowHTTPTest attack data while excluding Hulk and GoldenEye. This focused approach is aligned with the project's objective (slow rate attack) to develop deep learning models capable of detecting subtle and sophisticated HTTP-based DDoS attacks.

Slowloris and SlowHTTPTest are attacks designed to exhaust server resources by holding connections open or slow, thereby denying service to legitimate users. These attacks do not overwhelm the server with high traffic volumes, making them difficult to detect as they closely mimic normal traffic behaviors.

Data Processing and Usage:

The data from Wednesday encapsulates various attack signatures along with normal traffic, providing a robust set for training our models. The data processing steps included:

Filtering: Isolating records related to Slowloris and SlowHTTPTest, ensuring that the models train on relevant attack vectors.

Feature Selection: Optimizing the dataset to include features crucial for identifying characteristics of HTTP-based DDoS attacks, such as the duration of the flow, packet sizes, and intervals between packets.

Data Cleaning: Ensuring the quality of the dataset by removing corrupted or irrelevant entries, enhancing the accuracy of the model training phase.

Evaluation Metrics:

To assess the effectiveness of the convolutional neural networks (CNNs) and long short-term memory (LSTM) networks in detecting these attacks, several evaluation metrics have been employed:

Accuracy: Measures the overall effectiveness of the model in correctly classifying both attack and benign instances.

Precision: Important for determining how many of the instances classified as attacks were actually attacks, minimizing the risk of false positives which are crucial in a security context.

Recall: Indicates the model’s ability to identify all actual attacks, which is essential for ensuring that no attacks are missed.

F1 Score: Combines precision and recall into a single metric by taking their harmonic mean, useful for comparing model performance where a balance between precision and recall is required.

ROC-AUC: The area under the receiver operating characteristic curve (AUC) and the curve itself are used to evaluate the trade-offs between true positive rate and false positive rate at various threshold settings.

These metrics provide a comprehensive view of model performance, addressing both the effectiveness and efficiency of the detection system in a real-world application scenario.

By focusing specifically on the characteristics of slow rate HTTP-based attacks and excluding other types of DDoS attacks like Hulk and GoldenEye, our project aims to tailor the deep learning models for high precision and reliability in scenarios where the attack methods are subtle and sophisticated. This approach not only enhances the specificity of the models but also contributes to the broader field of cybersecurity by addressing the nuanced challenges of application-layer attack detection.

**Discussion on Results:**  
In our project, we applied convolutional neural networks (CNN) and long short-term memory (LSTM) models to detect HTTP-based DDoS attacks using the CICIDS2017 dataset, yielding insightful results on their respective capabilities under various metrics. Here we discuss the analysis of the model performance through epochs, evaluation metrics, and the implications of our findings.

**Graph Analysis**

Training and Validation Accuracy Trends

A graph showing the results of a training

Description automatically generated

CNN Model: The CNN model's accuracy both in training and validation shows a mostly consistent upward trajectory across the epochs. The gradual increase in training accuracy from around 86% to nearly 94% indicates effective learning, with validation accuracy closely mirroring these improvements. This suggests that the model generalizes well to new data, a desirable attribute in practical applications where models encounter varied and unpredictable inputs.

A graph showing the performance of training

Description automatically generated

LSTM Model: The LSTM model displays a more erratic pattern in training and validation accuracies. Initially, both metrics improve in tandem until around the sixth epoch, where validation accuracy suddenly drops, suggesting potential overfitting at this point. However, a notable recovery in later epochs—particularly a sharp increase in validation accuracy—demonstrates the model's capability to rebound and adapt, which could be attributed to its internal gating mechanisms that manage information flow.

The implementation of early stopping after 12 epochs, despite setting up for 100 epochs, is an essential step to avoid overfitting and unnecessary computational expense. The early stopping mechanism triggered due to the absence of significant improvement in validation loss, suggesting that continuing training beyond 12 epochs would not have contributed meaningful enhancements to model performance.

Evaluation Metrics Comparison

The comparative analysis of CNN and LSTM using precision, recall, F1-score, and accuracy reveals specific strengths and areas for improvement:

A graph showing different colored lines

Description automatically generated

Precision and Accuracy: The CNN model generally maintains higher precision and accuracy across the evaluations. This higher precision indicates fewer false positives, which is crucial for deploying in environments where falsely flagged traffic could lead to unnecessary or disruptive actions.

Recall and F1-Score: LSTM shows slightly better recall, suggesting it is more adept at identifying all actual attacks, a critical feature for security systems where failing to detect an attack can have severe consequences. However, the F1-score, which balances recall and precision, is moderately higher in CNNs, indicating a better overall balance between identifying attacks correctly and not mislabeling benign activities as attacks.

A graph of a comparison of performance metrics

Description automatically generated

Implications for Deployment in Cybersecurity Systems

The distinct attributes of CNN and LSTM models suggest specific deployment strategies based on the operational requirements of different network environments:

CNNs are particularly suited for stable, high-stakes environments where precision is paramount, and the cost of false alarms is high. Their ability to consistently perform well across various metrics makes them reliable for critical infrastructure protection.

LSTMs are advantageous in dynamic environments where attack patterns evolve rapidly and the ability to adapt to new threats is essential. Their proficiency in recall makes them indispensable for systems where the priority is the detection of all potential threats to mitigate risk comprehensively.

Future Directions and Hybrid Approaches

Given the strengths and weaknesses identified in each model, a promising area for future research involves developing hybrid systems that integrate the spatial processing capabilities of CNNs with the temporal processing strengths of LSTMs. Such systems could potentially harness the advantages of both models to create a more robust defense mechanism against a wide array of cybersecurity threats. This approach could lead to improved detection rates, better generalization across different types of network behaviors, and reduced false positives and negatives.

**Conclusion:**

The nuanced performance differences between CNN and LSTM models in our project underscore the complexity of choosing an appropriate model for security applications. CNNs, with their robustness in spatial pattern recognition and stable learning performance, seem particularly well-suited for detecting HTTP-based DDoS attacks in scenarios where maintaining a low rate of false alarms is critical. Conversely, LSTMs, with their ability to understand temporal sequences, might be preferred in environments where the cost of missing an attack is exceedingly high, despite potential fluctuations in model training performance.

In practical terms, our findings suggest a tailored approach to model selection based on specific operational requirements and threat profiles. For environments where rapid response to DDoS attacks is required, and false positives must be minimized, CNNs offer a valuable tool. Meanwhile, for systems where the absolute detection of every possible attack is paramount, LSTMs provide a crucial line of defense.

Further research could explore combining CNN and LSTM models in an ensemble approach to leverage the strengths of both: the spatial pattern recognition of CNNs and the temporal sensitivity of LSTMs. Such hybrid models could potentially offer a more robust solution, optimizing both the detection accuracy and the ability to generalize across different types of network traffic and attack vectors.

Ultimately, this project not only advances our understanding of applying deep learning techniques to cybersecurity challenges but also opens avenues for future innovations in developing more adaptive, resilient, and effective defense mechanisms against sophisticated network threats.

**References**:

* Dataset: <https://www.kaggle.com/code/kooaslansefat/cicids2017-safeml/input>
* <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9502698>
* <https://hal.science/hal-04262657/document>