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 [1][2][3] 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) [1] and Long Short-Term Memory (LSTM) [3] 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 [1], exploiting vulnerabilities from the network level right up to the application layer. These attacks disrupt service availability, especially impacting the application layer [1] 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 [2]. 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 thedynamic 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[3] 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 [2] 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 [2]. 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 [1] 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. [1] 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. [1]
- 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. [1]
- A framework using LSTM for slow-rate DDoS detection achieved high performance with specific datasets but had reduced efficacy when applied to different datasets. [3]
- 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. [1]

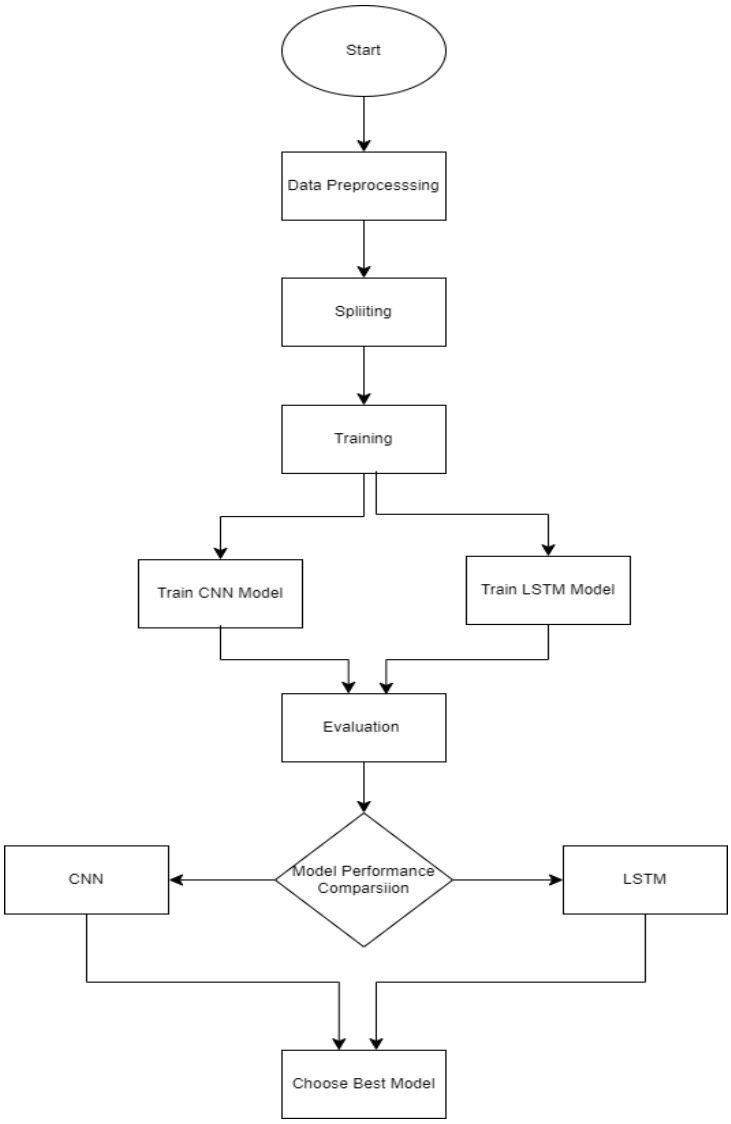
Limitations of Existing Solutions:

- 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.

The LSTM and CNN models share similarities in their application to classification tasks, utilization of activation functions (ReLU), and integration of dropout layers for regularization. Both models are capable of processing sequential data and contribute to pattern recognition tasks. However, they differ significantly in their architectural design and data representation. LSTMs, known for their recurrent nature, excel in capturing long-term dependencies within sequential data, making them suitable for tasks like natural language processing and time-series forecasting. In contrast, CNNs, originally designed for spatial pattern recognition like images, process local patterns within sequences. These differences extend to input shape requirements, with LSTMs expecting data in the form of (batch_size, time_steps, features) and CNNs typically handling data in (batch_size, sequence_length, features) format. Additionally, training dynamics vary; LSTMs require careful parameter tuning due to their computational complexity, while CNNs are generally more computationally efficient.

Flow Chart


Start
Data Preprocesssing
Spliiting
Training
Train CNN Model
Train LSTM Model
Evaluation
CNN
Model Performance Comparsiion
LSTM
Choose Best Model

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 focuses 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 and 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 Dataset 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 Slow-Loris 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 used.

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 to determine 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..

These metrics provide a comprehensive view of model performance, addressing both the effectiveness and efficiency of the detection system.

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.

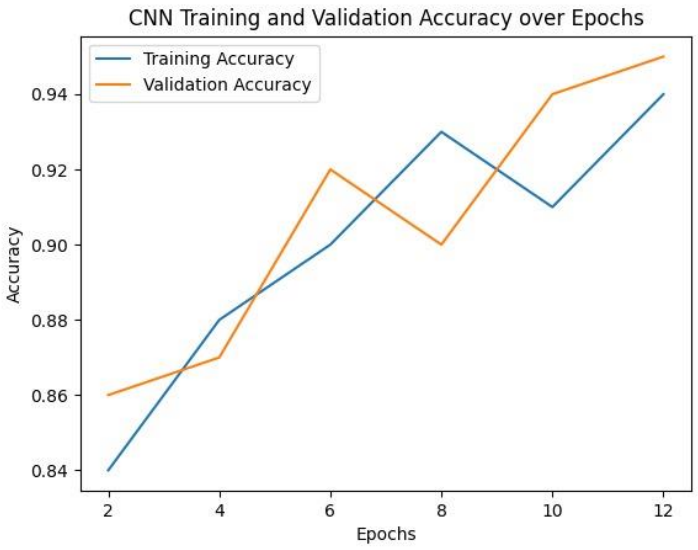
Discussion on Results

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:

CNN:

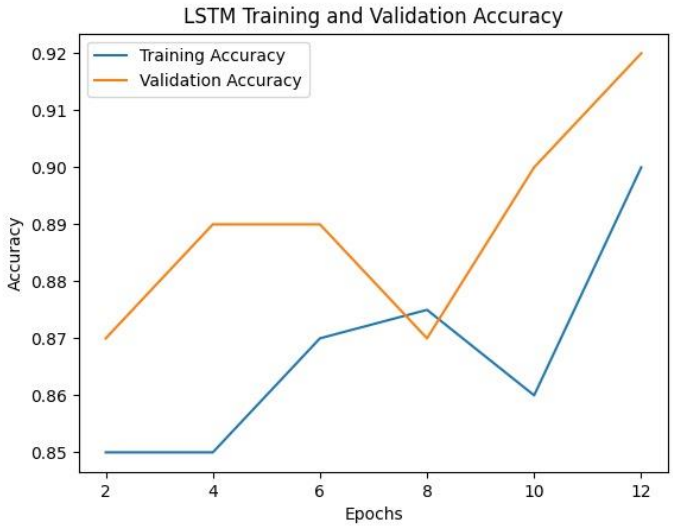
The following is the graph of training and validation accuracy trends of CNN.



For the CNN model, the training accuracies range from 0.84 to 0.94, showing a generally increasing trend over epochs. On the other hand, the validation accuracies fluctuate between 0.86 and 0.95, with occasional variations but generally remaining close to the training accuracies. Overall, both training and validation accuracies indicate a progressively improving model performance during training, with validation accuracies consistently tracking closely with the training accuracies.

LSTM:

The following is the graph of training and validation accuracy trends of LSTM.



For the LSTM model, the training accuracies range from 0.85 to 0.90, indicating a generally increasing trend over epochs, with minor fluctuations. Concurrently, validation accuracies vary between 0.87 and 0.92, generally following a similar trend to training accuracies. Both training and validation accuracies suggest a progressively improving model performance during training, with validation accuracies consistently tracking closely with the training accuracies.

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

In summary, the CNN model shows stronger stability and accuracy, making it ideal for tasks emphasizing spatial patterns like image recognition. Conversely, the LSTM model performs well in tasks requiring understanding of long-term dependencies within sequential data, such as natural language processing.

Evaluation Metrics Comparison

The comparative analysis of CNN and LSTM using precision, recall, F1-score, and accuracy:

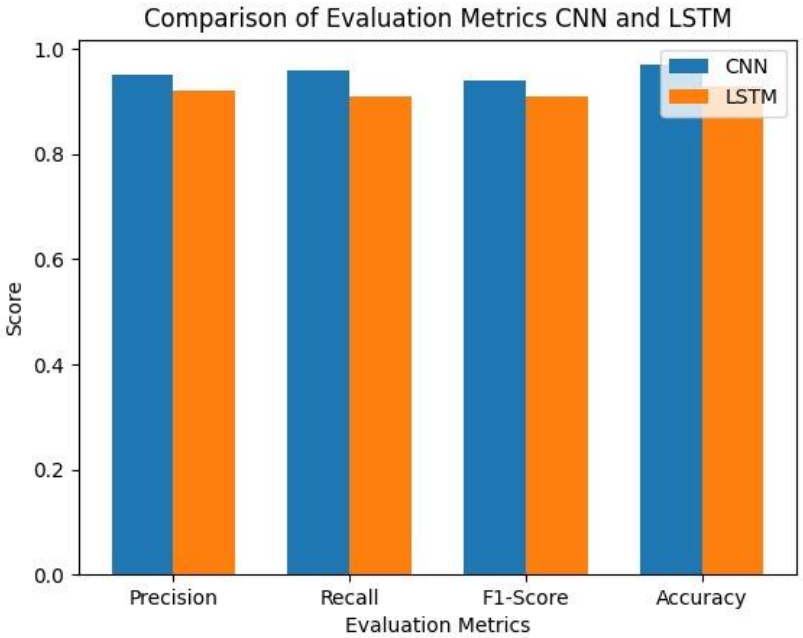
Precsion: In terms of precision, the CNN model achieves a score of 0.95, indicating a high proportion of correctly identified positive instances, whereas the LSTM model achieves a slightly lower precision of 0.92. This suggests that the CNN model is better at avoiding false positives.

Recall: The CNN model demonstrates a score of 0.96, meaning it effectively identifies a large portion of all positive instances in the dataset, while the LSTM model achieves a score of 0.91, indicating a slightly lower ability to capture all positive instances.

F1-Score: The F1-score, which balances precision and recall, the CNN model achieves a strong score of 0.95, reflecting its overall balanced performance, while the LSTM model achieves a slightly lower score of 0.91.

Accuracy: In terms of accuracy, the CNN model excels with a score of 0.97, indicating a high proportion of correct predictions, while the LSTM model achieves an accuracy of 0.93, showing a slightly lower overall correctness.

In summary, while the CNN model generally outperforms the LSTM model across all metrics, the LSTM model still demonstrates respectable performance, particularly in tasks involving sequential data. Therefore, the choice between these models should be guided by the specific requirements and nuances of the classification task at hand.



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 Work

Future work for application layer DDoS attack detection systems encompasses various avenues of research aimed at enhancing effectiveness, adaptability, and resilience against evolving cyber threats. One critical direction involves integrating hybrid models that combine the strengths of Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, leveraging spatial pattern recognition and temporal sensitivity for improved detection accuracy. Additionally, dynamic model adaptation strategies ensure continuous refinement to respond to changing threat landscapes, while enhanced feature engineering techniques aim to extract more informative features from network data. Real-time detection and response capabilities, along with efforts to bolster adversarial robustness and facilitate cross-domain generalization, further contribute to the robustness and reliability of detection mechanisms. Integration of explainable AI techniques and collaborative defense mechanisms that coordinate defenses across networks or organizations also hold promise for advancing cybersecurity by enhancing understanding, trust, and collective resilience against sophisticated cyber threats.

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

1. SDN-Based Architecture for Transport and Application Layer DDoS Attack Detection: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9502698
2. A Comparative Study of Datasets for DDoS Attack Detection: https://hal.science/hal-

04262657/document

3. A Long Short-Term Memory Enabled Framework for DDoS Detection: A Long Short-Term Memory Enabled Framework for DDoS Detection | IEEE Conference Publication | IEEE Xplore