National College of Ireland

Improving Cloud Security with Real-Time Detection of APT Attacks Using Advanced Deep Learning Algorithms

MSc Research Project
Cloud Computing

Abhiram Adluru
Student ID: x22187898

School of Computing
National College of Ireland

Supervisor: Vikas Sahni

National College of Ireland
Project Submission Sheet
School of Computing

National College of Ireland

Student Name:	Abhiram Adluru
Student ID:	x22187898
Programme:	M.S.C Cloud Computing
Year:	2023
Module:	MSc Research Project
Supervisor:	Vikas Sahni
Submission Due Date:	12/12/2024
Project Title:	Improving Cloud Security with Real-Time Detection of APT Attacks Using Advanced Deep Learning Algorithms
Word Count:	7060
Page Count:	19

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Abhiram Adluru
Date:	11th December 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Improving Cloud Security with Real-Time Detection of APT Attacks Using Advanced Deep Learning Algorithms

Abhiram Adluru
x22187898

Abstract

Cloud environments are becoming increasingly susceptible to Advanced Persistent Threats (APTs), threatening both data security and systems reliability. Conventional detection methods such as signature- or rule-based systems are generally incapable of identifying behaviours of an evolved attack mechanism due to their static nature. a detection system has been proposed which handles APT attack with real-time capabilities using deep learning models-CNN, LSTM, and ConvLSTM-to identify malicious activities in network traffic. Our approach, unlike prior ones, recognizes spatiotemporal patterns through advanced feature engineering and balance of the datasets. The system is implemented as a scalable web application on AWS Elastic Beanstalk, enabling real-time monitoring with instant feedback with the help of a user-friendly interface. Evaluation results have highlighted ConvLSTM as being the most efficient among all models under consideration, having exhibited higher values of precision and recall, as well as greater stability.

1 Introduction

Advance Persistent Threats (APT) is a potential security challenge that organizations are facing today. Over the years, technological advances have become an identifiable phenomenon that increases data integration and digitization of this information to facilitate various applications. However, experienced attackers are targeting confidential information from private organizations and corporate businesses leading to potential security threats. The most significant network attack by APT increases exfiltration of information to hosts, such as data thefts. Some common examples of related circumstances include the 9GB password-encrypted data leakage from Adobe in 2013 and the 40GB "Ashley Madison's Database" theft that occurred in 2015 are incidents registered under advance persistent threats (APT).

For every organization, it is therefore become mandatory to predict the occurrence of cyber events to overcome financial losses and data breaches. As per the information by IBM, in 2022, the data breach cost a staggering value of 9.44 million dollars in the US, where 50% of the breaches occurred in the cloud Bunting (2023). The cloud environment is an identifiable advancement in digital technology that has provided distinct advantages with highly integrated infrastructure and virtual resource utilization. Although the advance to a cloud environment has presented a proactive approach to managing virtual resources and advanced network infrastructure; cybersecurity threats are prominent

and persistent. Amid this focus, the above-recognized APT attacks have raised significant concerns since it target specific organizations and carry out attacks across different stages. The detection of these cyber-attacks has been considered a pressing issue, especially when cloud workloads are strongly affected by the attack.

Since the intrusion has been escalating continuously, cloud service providers, security developers, and researchers have been continuously investigating ways to monitor and detect network anomalies. In this situation, conventional detection methods such as Firewalls, Antivirus Software, fail to detect APT attacks. Moreover, it has raised significant dilemma due to experienced attackers and their strategy to organize the attack in different phases. Thus, advanced methods such as artificial intelligence (AI) and machine learning (ML) are found to be promising analytical technologies. These methods use datasets containing information on APT attacks and classify them according to phases, thus protecting the network system and preventing data exfiltration. This current research study aims to enhance APT detection for the cloud environment where real-time data analysis has been carried out using ML algorithms. In this regard, a Kaggle dataset has been used and the entire architecture has been set with real-time APT attack detection attributes using Amazon Web Service (AWS) based cloud services.

1.1 Research Objectives

- To develop a scalable and robust system for real-time detection of Advanced Persistent Threats (APTs) in cloud environments using deep learning models.
- To evaluate the performance of three advanced deep learning models—CNN, LSTM, and ConvLSTM—for accurately identifying spatiotemporal patterns in network traffic associated with APT attacks.
- To determine the most effective deep learning model for APT attack detection by analyzing metrics such as accuracy, precision, recall, F1-score, and confusion matrices.
- To implement a functional web application that provides real-time visualization of APT attack detection for monitoring and administrative purposes.

1.2 Research Question

How can advanced deep learning models be leveraged to accurately detect APT attacks in cloud environments, enhance cloud security through real-time identification of spatiotemporal patterns in network traffic, and determine which of the three deep learning models—CNN, LSTM, or ConvLSTM performs best for apt attack detection?

1.3 Report Structure

the next part of the report is as follows, Section 2 consists of related work on machine learning algorithms on advance persistent threats,Section 3 consists of the methodology used for implementing the project,Section 4 provides the design specifications of the aws services and tools used,Section 5 consists of implementation procedure and section 6,7 shows the evaluation results and conclusion.

2 Related Work

In this Section out key insights has been outlined into how security challenges have become a threat to technological advances. Over the years, the cloud system has received signific-

ant attention due to its efficient network enhancement and virtual resource optimization for user-friendly applications. This has increasingly helped organizations to perform different activities and meet business purposes in a digital environment. However, network security has become a rising concern for users due to increasing cyber-attacks. One such concern is the Adaptive Persistent Threats (APT) - a security challenge leading to data exfiltration and information leakage. This section has presented evidence-based information from previous studies and further implemented ideas on how the current study can address this challenge with a scope for reviewing the gaps in existing approaches.

2.1 Security Challenges Faced with Cloud Services

Cloud computing has been considered a flexible network architecture through which both data and resources are distributed across various locations. According to the information provided by Ahmad et al. (2022) cloud computing in the business context has put forward significant changes that promote a new advent for usage, storage, and sharing of virtual resources. This has helped in performing potent industrial operations. Amid the understanding of the wide applicability of cloud services, the transition has raised security concerns in organizations across multiple sectors. A view of the banking sector, for example, informs that cloud operations in banking and financial services indeed bring a revolutionary approach with high scalability, cost-effectiveness, and high elasticity in digital operations; however, security dangers have caused financial havoc to this industry in particular. As stated by Vinoth et al. (2022) the sensitivity of cloud architecture has raised a higher level of security problems due to which vulnerabilities are detected within cloud networks. The privacy and safety issues of customers across organizations are issue that corporate holders are facing in recent times.

According to the information provided by Tabrizchi and Rafsanjani (2020), the challenges with security and data privacy in the cloud computing environment have increased the demand for addressing the challenges with potential solutions. In the cloud computing environment, users generally lack knowledge of the geographic location where the sensitive data is stored. This is because data centers providing cloud services maintain a distance across multiple locations from users, which in turn creates security challenges as well as threats. Indeed traditional security solutions such as firewalls and antivirus software have been used to detect the intrusion Subramanian and Jeyaraj (2018). Unfortunately, the security measures are not effective in supporting the advanced virtualized system where rapid threats are emerging with enhanced features. Further information provided by Ahmad et al. (2022) demonstrated that traditional security measures have no direct applicability and are often ineffective in detecting the threat. Thus, improved solutions are needed to address the security challenges, which have become increasingly complex in terms of configurations.

2.2 APT Detection Enhancement Using Machine Learning Models

While understanding the rising security concern, advanced persistent threats (APT) have been identified as a serious issue for contemporary organizations. Evidence-based information provided by Kumar et al. (2024) stated that the rising dilemma of APT has insisted developers and researchers explore multifaceted defense mechanisms that can provide better intrusion detection solutions with fast responses. Research advances in APT detection have been promoted with enhanced tactics that emphasize thwarting these advanced attacks while offering the best solutions for execution. Chen et al. (2022) explained that

amid distinguished technological benefits, IoT and cloud environments have been facing cyber weaknesses that in turn increase their vulnerabilities within the wireless medium. In this security concern, the implementation of the machine learning (ML) method to detect intrusion in cloud and IoT environments has served promising solutions. It is evident that organizations in the current business landscape are facing immense security issues with APT attacks, thus necessitating the development of a "multifaceted defense strategy". It is aware that for years many traditional methods have increasingly monitoring such incident responses. However, the concerning part is their ineffective responses to complex or modified cyber threats. The mitigation of this concern has introduced advanced solutions where machine learning models play a vital role in detecting the intrusion based on behavior analysis of the APT attack and further evaluation.
According to the information provided by the study, Xuan et al. (2021) explained that the proposed "multi-layered analysis" technique - a machine learning approach has been introduced for APT threat detection where the model analyses various events on network traffic. The model has enabled the detection and synthesis of abnormal behavior in network traffic, thus providing insights into APT's existence in the user's system. Indicating the efficiency of the model,Xuan et al. (2021) Xuan et al., (2021) explained that the multi-layered analytical solution performs two essential tasks; first analyzing the network traffic components for any kind of abnormal signs and further building as well as classifying the behavioral profile depending on the component. Based on the experimental outcome, it has been determined that the model was indeed effective in APT detection and also provided insights into the novelty of detecting related incidents with higher complexities. In another study presented by Stojanović et al. (2020) focus has been given on automated intrusion detection methods to target cyber attacks. In this regard, a suitable APT dataset is essential to make efficient use of the data in creating the model's relevance. Understandably, the detection of APT attacks can be very challenging given the complexity of the existence of benchmark datasets for zero-day attacks. One of the potential requirements for machine learning-based network intrusion detection is the utilization of a benchmark dataset to gather adequate information on the attack behavior and further classify them with higher accuracy.

The above information denotes the evolving nature of the cybersecurity environment where "Advanced Persistent Threat" (APT) has been considered a formidable dilemma. Amid this challenge, conventional methods fail to detect the ever-advancing noise, thus requiring the introduction of a ground-breaking technique leveraging advanced machine learning techniques, thus outperforming traditional classifiers. Understandably, the information presented by Arefin et al. (2024) has introduced a novel ML technique - the MLP model and Gradient Booting (GB) method which is compared to the traditional KNN model. The meticulous engineering of features essential to detect the proficiency and accuracy levels of the state-of-the-art models has outperformed the baseline KNN method, whose accuracy level is identified to be 76.6%. Contrastingly, the comparative outcomes of both classifiers show that the MLP has achieved an accuracy of 94.5% and GB achieved 92.3% accuracy. Thus, the study provided significant insights into the effectiveness of ML algorithms introduced and marked as a revolutionizing approach in safeguarding network infrastructures. Contrasting to this evidence, Saini et al. (2023) have introduced an ensemble ML model that has combined the random forest (RF) classifier with the XGBoost classifier. The approach is exclusive, considering the specificity of the model in terms of mixed features, thus utilized in detecting APT attacks through accessible datasets. The progression of the study has demonstrated the model's signific-

ance in obtaining maximum accuracy in the prediction process using different datasets - CSE-CIC-IDS2018, CICIDS2017, NSLKDD, and UNSW-NB15. The experimental result denoted by the classification using each dataset has presented an accuracy of 98.92%, 99.91%, 99.24% 97.11% respectively. Further, a comparison of the result for each dataset shows that even though the model has performed efficiently using different datasets; CSE-CIC-IDS2018 provides a better outcome compared to others.

The malicious APT attacks have been identified as a clear intentional approach of advanced attackers. This has been identified as a rising dilemma for corporate sectors, the government, and businesses in leveraging Internet solutions. The mitigation of such a concern is important to ensure the swift functioning of operations across the sectors. Previously, it has been identified that machine-learning models have provided promising solutions in analyzing abnormal and malicious behavior of network traffic, which is exclusive for APT detection in contemporary times. In this focus, evidence-based information presented by Xuan and Dao (2021) stated that behavior analysis, as well as evaluation with advancement in APT attacks, have created difficulties for ineffective data representation from various attack campaigns. Thus, handling such attacks with efficient detection methods and benchmarked datasets is important. Understandably, hybrid modeling of different methods has gained significant recognition to develop a robust approach to the detection process. Understanding the priority, Xuan and Dao (2021) introduced a combined model based on "multilayer perceptron" (MLP), "convolutional neural network" (CNN) and "long-short-term-memory" (LSTM) models. The model's effectiveness has been determined by investigating its ability from the experimental performance, which shows that the model has achieved an accuracy level of 93-98%. This has indicated an outcome that is suitable for APT threat detection in current and upcoming times. The overall information therefore indicates the importance of machine-learning models in real-time data analysis using benchmarked datasets in the cloud environment.

2.3 APT Detection Enhancement Using Deep Learning Models

The industrial applications of the cloud and IoT are using huge data volumes to perform troubleshooting, identification of performance bottlenecks, and detection of malicious behavior to achieve efficient control over the physical world. In this approach, a concern has emerged with both traditional attacks in network traffic and persistent threats such as API. According to the evidence presented by Yu et al. (2021) APT is a prolonged concern that has targeted confidential data hacking through advanced cyber knowledge of attackers. The intruder gained access to the critical infrastructure system and deliberately affected the organizational operations at every level. Upon understanding the seriousness of the situation, researchers have introduced potential machine learning and deep learning-based network intrusion detection solutions, which are indeed promising. A study presented by aznulqalid et al. (2024) has introduced deep-learning models, such as neural architecture, and compared them with XAI state-of-the-art methods such as "Shaply Additive Explanations" (SHAP) and "Local Interpretable Model-agnostic Explanations" (LIME). The implication of the study has presented insights into how the XAI method can increase the transparency and interpretability of black-box models. Although a comparative analysis has been presented by the study between both methods, it has explained that the detection of APT cyber attacks leverages advanced DL methods for continuous monitoring of the threat. At the same time, experts also integrated XAI methods that ensure transparency and trust in the real-time intrusion detection process.

Providing security for the network and data has been identified as a rising dilemma

for business organizations and corporate holders. Since its identification in 2006, Advanced Persistent Threat has become a sophisticated novel attack that is carried out by well-funded attackers to gain accessibility to confidential data. The detection of these APT attacks has presented greater disadvantages when using traditional detection methods. Therefore, various advancements have been identified in recent decades that have presented remarkable results. For instance, Joloudari et al. (2020) introduced a Deep-Belief Network model ensembled with a Support Vector Machine (SVM) that has been effective in detecting network intrusion using the NSL-KDD dataset. The study implied that the ensemble model has provided an accuracy of 92.84% accuracy, which is better than the individual performance of the DBN model and SVM classifier. In another study presented by Singh et al. (2019), it was noted that organizations using the cloud system for advanced network infrastructure are facing vulnerabilities due to sophisticated APT attacks. However, very few solutions have provided accurate detection results due to the complications of the attack. The above study has introduced a comprehensive review of "semantic-aware work" through which potential contributions of advanced methods are analyzed.

The contribution of deep learning methods in detecting network intrusion and system vulnerabilities has directed research on identifying appropriate APT defense mechanisms. However, it has been noted that with the novel configuration of the attack, a single machine learning or deep learning method is ineffective and provides less accurate results. Understandably, some studies have introduced ensemble models composed of a deep model and other state-of-the-art methods that provide higher accuracy in real-time data analysis and automated response. However, more research is required to explore the prompt action introduced by these models in APT attack detection.

2.4 Security Challenges and APT Detection Using Other Approaches

Apart from understanding the potential contribution of machine learning and deep learning models in the detection of network security attacks, the scope has been identified for other detection methods. Evidence-based information provided by Milajerdi et al. (2019) introduced HOLMES - a system specifying the detection of APTs. The relevance of the model has been articulated through the real-world detection of APT attacks and system vulnerabilities in the cloud system. As per the evaluation of the outcome, it has been identified that the model is capable enough to generate high-level graphs, which efficiently summarise attackers; actions. Moreover, the experimental result obtained through the HOLMES APT campaign detection provides higher precision and low false-positive alarm rates. In contrast to this evidence, a study presented by Salim et al. (2023) demonstrates the growing adverse impact of APT attacks on computing technologies. At present, data transmission has been increased at a significant rate, which further shows the vulnerable condition. Understandably, the above study introduced a conceptual model "Effective Cyber Situational Awareness Model" for the detection and prediction of Mobile APTs - ECSA-tDP-MAPT. The model has been designed for effective detection of the attack and shows an improved performance from the experimental conduct.

The basis of exploring various models and methods is to understand their capability in APT attack detection in advanced network environments. For years, many solutions have been designed and introduced, which are compared and contrasted with existing ones to determine the accuracy, precision, and false-positive alarm rates. Determining these parameters for a model has informed of the high relevance of the model in predict-

ing advanced threats in network security, which is continuously evolving and becoming sophisticated. Evidence presented by Lu et al. (2024) has acknowledged the increased penetration of "renewable distributed energy sources" (DERS) in smart grids that heavily rely on ICT technologies. Based on the study's implication, this approach influences social behavior on the system's operations and the management level. Therefore, the study represents the importance of analyzing "High-DER-penetrated SGs" features to determine the degree of vulnerabilities caused by APT attacks. Therefore, the intelligent detection of APT attacks introduces an enhanced security architecture that has been explored to identify the relevance. Understandably, Hu et al. (2024) introduced another model - the "Data-Enhanced Meta-Learning" (DEML) model that detects APT traffic in the IoT network. Indicating the experimental outcome, the model has been identified to outperform many existing models with an accuracy level of 99.35%, thus presenting a novice yet improved detection method than various baseline models.

3 Methodology

APTs are one of the severe threats to cloud security because they are undetected and evolving, hence conventional detection methods are often unable to detect them. As depicted, our methodology comprises multiple processes of data preprocessing, feature engineering, model training, and evaluation for accurate APT attack detection. Each of them plays a crucial role in addressing the problem and will be positively detailed in the following subsections. The methodology diagram is shown in Figure 1.

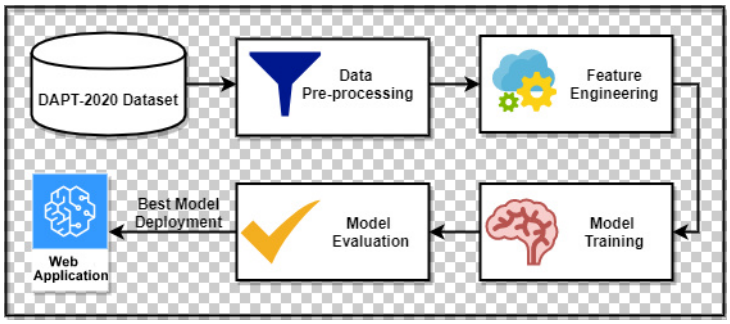

Figure 1: Methodology Framework for APT attack detection

3.1 Data Pre-processing

In the preprocessing phase,three separate CSV files containing network traffic data has been loaded into three separate DataFrames, corresponding to different days of work with a network. Next,three individual DataFrames has been merged into a single unified DataFrame for easier comprehensive analysis. While doing so,the columns of all three DataFrames have been analysed to understand the structure of data they contained and checked for uniformity. Descriptive statistics on the combined dataset provided a view of the datasets together and has been checked for any missing values across each column and found columns with null entries that may possibly be subjected to imputation or deletion as part of the subsequent processing.

Next, the data were scanned for duplicates since they can distort analysis or modeling outputs, and their numbers were counted. Furthermore, column names were standardized by the removal of excess spaces, by conversion to lower case, and finally by changing any

special characters into underscores to obtain uniformity of data and compatibility with many tools and scripts. Such data preprocessing techniques resulted in cleaning up the dataset and positioning it well for analysis and modeling users in this subsequent phase.

3.2 Exploratory Data Analysis

The diagram in Figure 2 illustrates the hierarchy and sophistication of diversity across various activities belonging to specific network stages, with benign activities colored blue, sharply distinguished from attack-associated activities in green and red. Each stage has its specific activities which include Directory Bruteforce, SQL Injection, and Malware Download and are classified under them, as Reconnaissance and Establish Foothold. This includes the variety of possible actions within each stage and further implies the isolation of normal activities, stressing the importance of precise detection mechanisms that could differentiate between benign and malicious behaviors.

Figure 2: Hierarchy of Activities Categorized Under Network Stages

3.3 Feature Engineering

The feature engineering process began with temporal extraction from the timestamp column, such as hour, week of the day, month, day of the year, and whether the occurrence was a weekend. These were extracted to capture any time-dependent patterns present within the dataset. New variations of useful features such as traffic intensity, average packet size, forward-to-backward packet ratio, packet length ratio, SYN-to-FIN ratio, flow rate, and packet rate were also computed and provided depth into network behavior. Internal and external traffic were differentiated on the basis of predefined IP address patterns while categorical features like activity and day_of_week were transformed using one-hot encoding. The stage column was label encoded. SMOTE was performed to tackle the class imbalance, balancing the dataset for model training efforts. The analysis of before and after SMOTE is shown in Figure 3. After applying SMOTE, the distribution became balanced, ensuring all classes had an equal number of samples. This helps improve model performance by reducing bias toward the majority class as shown in Figure 3.

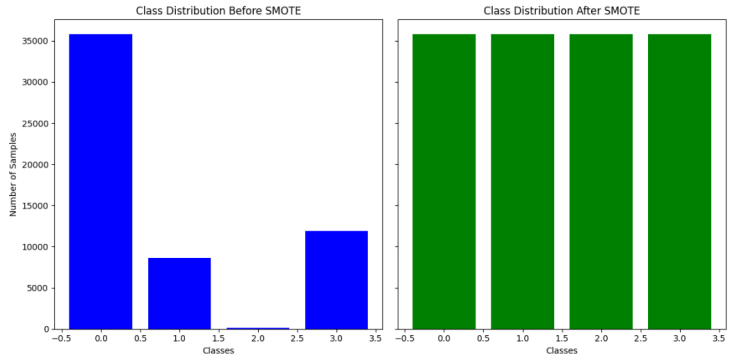

Figure 3: Class Distribution Before and After SMOTE

skewed numerical features were handeld through log transformation, and mustering high variance features, including some flag counts, was necessary. Small feature collection through the Random Forest classifier indicated the 20 most important features which were analyzed. Features such as idle_std, dst_port, and subflow_bwd_packets were the most influential in predicting the target class as shown in below Figure.

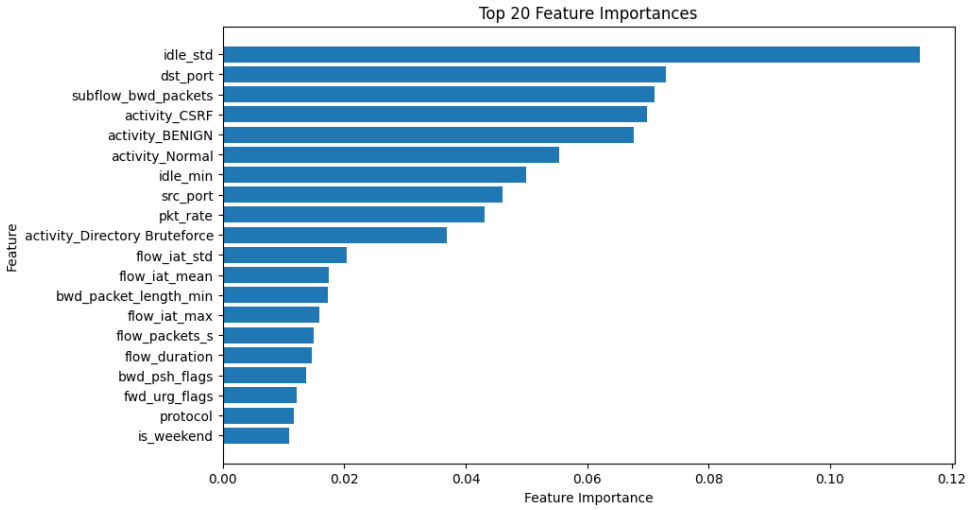

Figure 4: Top 20 Feature Importances Identified by Random Forest

Data normalization was done using StandardScaler to standardize them for model training, thereby allowing the features to reside within a certain limit.

3.4 Model Training

In this research,three deep learning models-CNN (Convolutional Neural Network), LSTM (Long Short Term Memory), and ConvLSTM (Convolutional LSTM) has been trained on the preprocessed and feature-engineered dataset. All models were designed to exploit the working characteristics of the data. The convolutional neural network (CNN) was specifically designed to realize the spatial patterns of features, while LSTM was created more for the investigation of temporal dependencies. Lastly, since the ConvLSTM combines the strengths of convolutional and recurrent layers, it performs excellently when extracting the spatiotemporal statistics. The data were reshaped accordingly for individual model

input requirements, such as adding channel dimensions for CNN and temporal dimensions for the LSTM and ConvLSTM.Categorical cross-entropy was utilised as the loss function and Adam as the optimizer for all three models, with learning rates set at 0.001 for each, all of which had the effect of aiding convergence in an effective manner.

3.5 Model Evaluation

To evaluate the models, the perfomance of CNN, LSTM, and ConvLSTM models has been accessed using accuracy, precision, recall, and F1-score as comprehensive metrics to evaluate their ability to detect APT attacks. Outputs from confusion matrices and classification reports were generated to investigate class-wise performance and identify areas for misclassifications. These competitions were carried out to find the best model to be deployed into a web application for APT attack detection in real time.

4 Design Specification

Our system architecture makes use of a client-server model hosted in a cloud environment to enable real-time analysis and scalability. At application launch, the client sends network packet data to cloud server which is currently deployed in AWS Elastic Beanstalk. Using Flask, the server processes incoming packets and computes predictions through a pre-trained deep learning model called ConvLSTM, classifying the network activity into stages that include benign and possible APT attack. The predictions are then sent back to the frontend through SocketIO to communicate real-time updates on the front end.

Further to ensure seamless deployment and continuous integration/continuous deployment (CI/CD), GitHub Actions is employed to automate the testing, building, and deployment of changes into the AWS environment. This architecture provides the defined qualities of a scalable, efficient, and capable APT attack detection solution delivered in real-time with high availability and low-latency user experience inside cloud infrastructure. The design architecture diagram of application is shown in Figure 5.

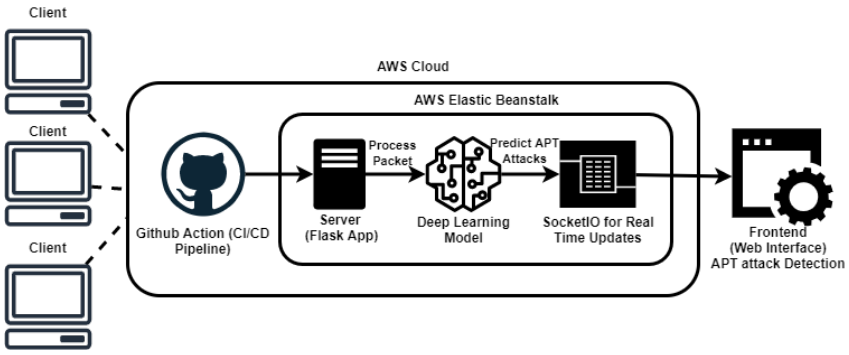

Figure 5: Design Architecture of APT Attack Detection APP in Cloud Environment

5 Implementation

The APT attack detection system was built with multiple libraries and tools bringing about their special functionalities and contribution to different stages of application building. Specifically,Pandas was used as its structure supports quick data handling and manipulation, enabling faster preprocessing and feature engineering of massive datasets of network traffic analyzed. Other visualization libraries, including Matplotlib, Seaborn,

and Plotly, were used to efficiently maneuver in analytic representations of distributions across classes, plot the importance of distinct features in the dataset, and finally correlate the different features of that dataset, all of which proved insightful. The hierarchical visualization of network activities was handled through NetworkX for determining relationships between various phases of attacks.

TensorFlow and Keras were utilized to built sophisticated models for machine learning and deep learning tasks like CNN, LSTM, and ConvLSTM. These frameworks were chosen because of their flexibility and capability of constructing elaborate model architecture, enabling us to run spatio-temporal features present within the data easily. Adam Optimizer, EarlyStopping, and ReduceLROnPlateau were functionalized for model training to ensure convergence and minimize overfitting. Scikit-learn support for label encoding, feature scaling, and evaluation metrics-accuracy, precision, recall, and F1 score-offer a strong performance evaluation framework for the model.

With Flask at the backend server for real-time predictions, SocketLive for emits to the frontend to show real-time updates, and Requests for connecting the client to the server. The server is also running on AWS, Elastic Beanstalk for scalability and availability, and CI/CD was handled by Github Actions under the hood to automate the entire update deployment process smoothly. The developed web application allows for real-time insight into ongoing network activities by applying advanced deep learning modeling techniques, thereby predicting with high accuracy detection and classification of advanced persistent threat (APT) attacks. The application constantly receives from clients network packets, which are processed by the backend server and classified in any of the three following categories, namely: benign, reconnaissance, establish foothold, and lateral movement. The predictions of the individual packets are displayed on the scrolling screen in real-time in iterations that show classification with clear icons and colors enhancing readability as shown in Figure 6. The application has a dynamic bar chart displaying the aggregated distribution of predictions, which assists users in monitoring trends and potential threats over time. This user-friendly interface allows for timely identification of APT attacks and response by administrators.

Figure 6: Real-Time APT Attack Detection with Web Application

The bar chart reveals that the benign behavior predominates the dataset with a highly significant class imbalance, at over 35,000 samples. Among the attack phases, Reconnaissance and Establish Foothold are the most frequently carried out phases, while a rather limited representation coincides with Lateral Movement.

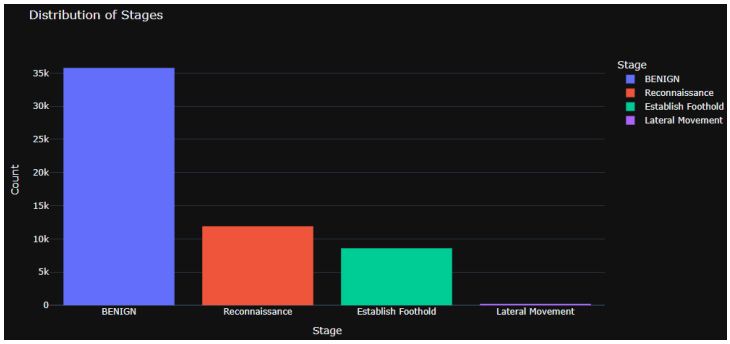

Figure 7: Distribution of Network Activity Stages Highlighting Class Imbalance

The bar chart in Figure 8 displays the frequency of various activities associated with certain stages, namely BENIGN, Reconnaissance, Establish Foothold, and Lateral Movement on the network. The maximum number of activities have been labelled under the BENIGN stage, Lateral Movement stages scored highest. Amongst the malicious acts within the Establish Foothold stage, the frequency of Directory Bruteforce was greatest, followed by Frequency of Network Scan under the Reconnaissance stage. CSRF, Backdoor, and SQL Injection take very little space, thus indicating an attention towards the need for the detection of rarer activities. This visualization represents the comparative imbalance noticed across the activity types, with a focus on the leading forms of attacks.

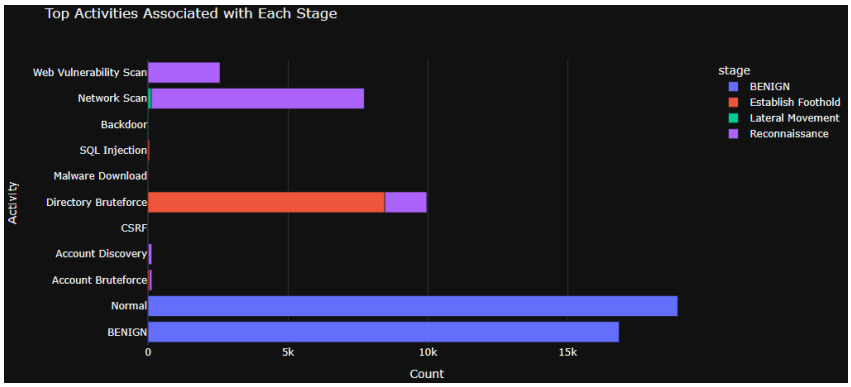

Figure 8: Top Activities Associated with Each Network Stage

The chart in Figure 9 describes the hourly distribution of various stages of network activity, pointing out fluctuations in relative activity levels during the day. BENIGN activity is always on top during all hours but reaches its peak in the evening. Reconnaissance activity works best around midday, while Establish Foothold shows a significant high early in the afternoon. Throughout the hours, Lateral Movement ranges low; all activities across all hours are on the low end. It shows that there are time-dependent patterns of malicious activities, therefore warranting more vigilance during peak hours for certain stages of attacks.

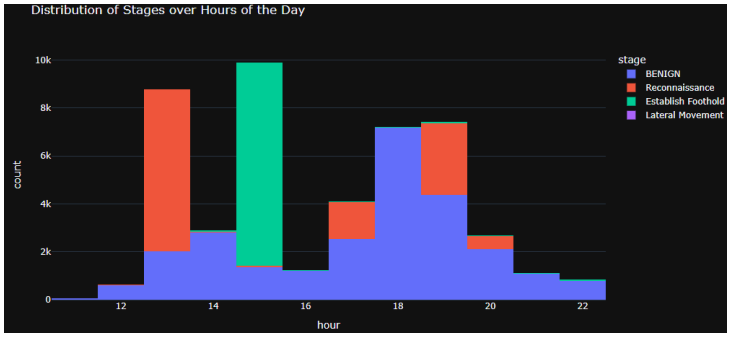

Figure 9: Distribution of Stages Across Hours of the Day

The box plot in Figure 10 displays the distribution of the mean packet length across the different stages of the network flow: BENIGN, Reconnaissance, Establish Foothold, and Lateral Movement. The activity in BENIGN shows a wide spread with many outliers; thus the packets that are being sent show manifest variability. On the other hand, attack stages like Reconnaissance, Establish Foothold, and Lateral Movement have much tighter distributions with few outliers, indicating packed behavior that tends to be more patterned. From this discussion, it can be deduced that the variability of the packet lengths is a good distinguishing feature that may help separate benign activity from malicious ones.

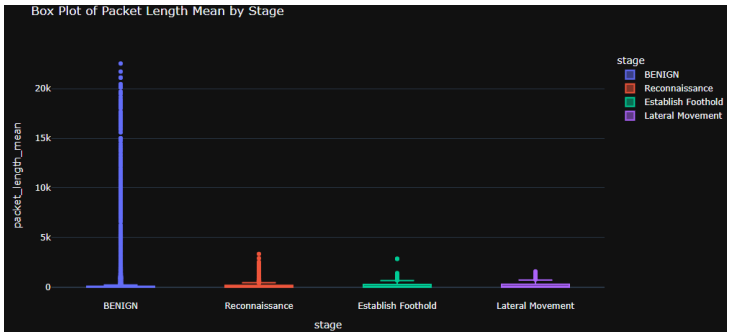

Figure 10: Box Plot of Packet Length Mean Across Different Stages

Radial plots in Figure 11 summarize mean packet lengths for various phases of the network-measured-forward, backward, and total. BENIGN activities have the greatest figures for all packet length measures, indicating a considerable increase in packet sizes compared to malicious incidents. In the attack stages, Establish Foothold and Reconnaissance values are moderate, whereas Lateral Movement is marked with the fewest packet lengths. This visualization demonstrates how packets behave differently across phases, helping to distinguish between normal and malicious traffic according to packet length properties.

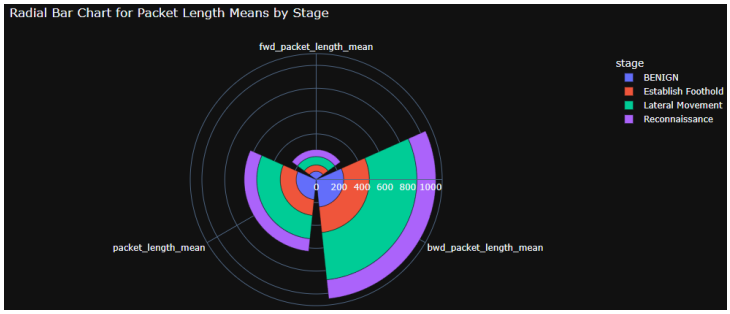

Figure 11: Radial Bar Chart Representing Packet Length Means Across Stages

6 Evaluation

Evaluation is an important step in establishing the performance and efficacy of our APT attack detection system. It assures the effectiveness of the deep learning models that have been set up in the application, including CNN, LSTM, and ConvLSTM, in determining more attack phases with high accuracy. In this evaluation, metrics such as accuracy, precision, recall, F1-score, and confusion matrices will be analyzed to measure the effectiveness of the models and thereby expose some of their strengths and weaknesses. This process involves validating the models on unseen data to confirm their robustness and capability of generalization. this section,dwells into the discussion of the evaluation results and the implications which provides an insight into the system's working and consequently isolating the best model for real-time deployment.

6.1 Experiment-1 / Evaluation based on Accuracy

The comparison graph in Figure 12 provides insight into the accuracy of the three models CNN, LSTM, and ConvLSTM. Each of the models operates very efficiently with nearly perfect accuracy; however, ConvLSTM outshined the others, with a 0.9996 accuracy rating compared to 0.9995 for CNN and LSTM. Based on this numerical assertion, ConvLSTM stands out as the superior model for identifying APT attacks because of its capability to recognize spatial as well as temporal features.

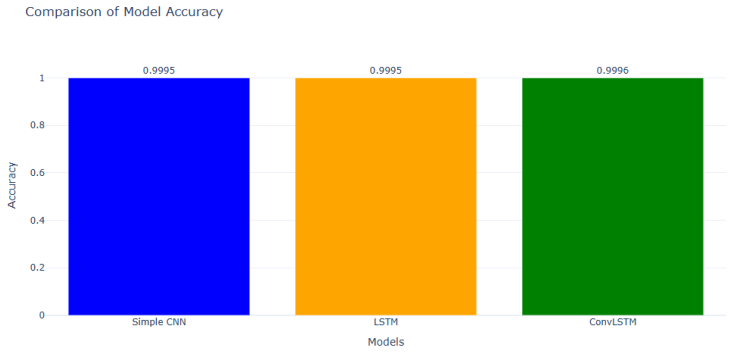

Figure 12: Comparative Analysis based on Accuracy

The graph in Figure 13 shows the training and validation accuracy of CNN, LSTM, and ConvLSTM models over 10 epochs. Early in modern training, they maintained high and consistent accuracy levels. No overfitting was seen in that all kept validation accuracy

close to training accuracy throughout. CNN and LSTM exhibited light fluctuations in validation performance while ConvLSTM showed the most stable and trustworthy behavior, bearing powerful generalization on across the datasets.

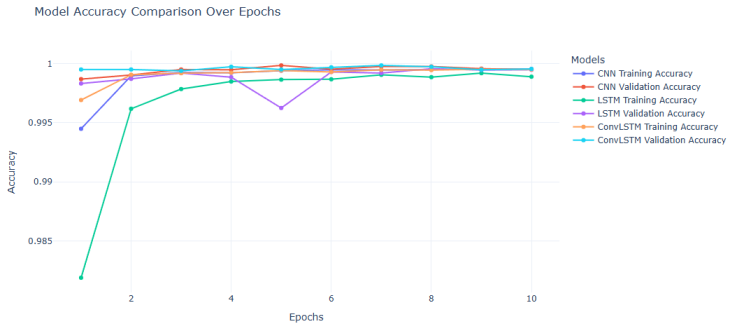

Figure 13: Model Accuracy Comparison Over Epochs

6.2 Experiment-2 / Evaluation based on Precision

Precision score is the proportion of true positive predictions out of all positive predictions made by the model, indicate its ability to avoid false positives. A comparison of precision scores of CNN, LSTM, and ConvLSTM models is shown in Figure 14. The ConvLSTM exhibited the highest precision of 0.9996, followed by LSTM 0.9995 and CNN 0.9995. The slight differences arise from the captured spatial and temporal patterns in the data, effectively achieving much better precision for ConvLSTM. Its combined architecture of convolutional and recurrent layers renders it particularly fitted for finding complex APT attack patterns. Therefore, ConvLSTM appears to be the most elegant solution for APT attack identification when compared with precision.

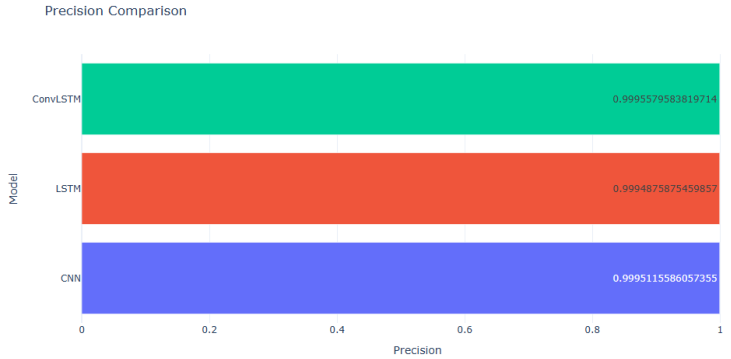

Figure 14: Comparative Analysis based on Precision

6.3 Experiment-3 / Evaluation based on Recall

Recall is the ratio of true positive predictions to the actual positive cases, representing how much the model may retrieve all relevant instances. The graph in Figure 15 shows the comparison of recall across the CNN, LSTM, and ConvLSTM models. ConvLSTM scored the highest recall at 0.99956, surpassing the CNN (0.99951) and LSTM (0.99949) models. The result underlines the fact that ConvLSTM accomplishes the greatest pattern matching to an APT attack from the rest with a small chance of generating false negatives.

It utilizes a very effective combination of convolution and recurrent layers, which allows it to model spatiotemporal dependencies effectively and to model minor details around attack behaviors very correctly. Thus, ConvLSTM performs best with recall being the most optimal.

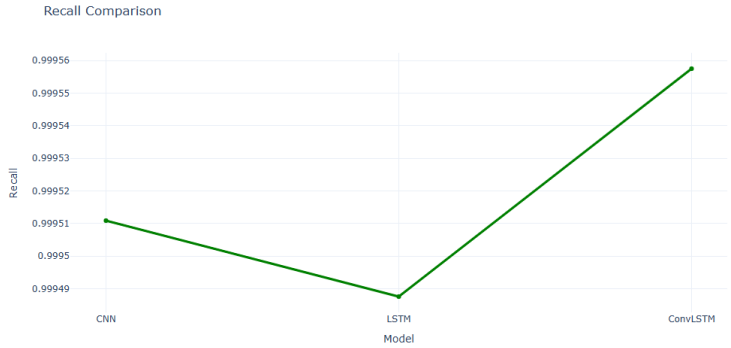

Figure 15: Comparative Analysis based on Recall

6.4 Experiment-4 / Evaluation based on F1-Score

Recall is the ratio of true positive predictions to the actual positive cases, representing how much the model may retrieve all relevant instances. The graph in Figure 16 shows the comparison of recall across the CNN, LSTM, and ConvLSTM models. ConvLSTM scored the highest recall at 0.99956, surpassing the CNN (0.99951) and LSTM (0.99949) models. The result underlines the fact that ConvLSTM accomplishes the greatest pattern matching to an APT attack from the rest with a small chance of generating false negatives. It utilizes a very effective combination of convolution and recurrent layers, which allows it to model spatiotemporal dependencies effectively and to model minor details around attack behaviors very correctly. Thus, ConvLSTM performs best with recall being the most optimal, whilst ensuring maximum detection of all relevant attack instances.

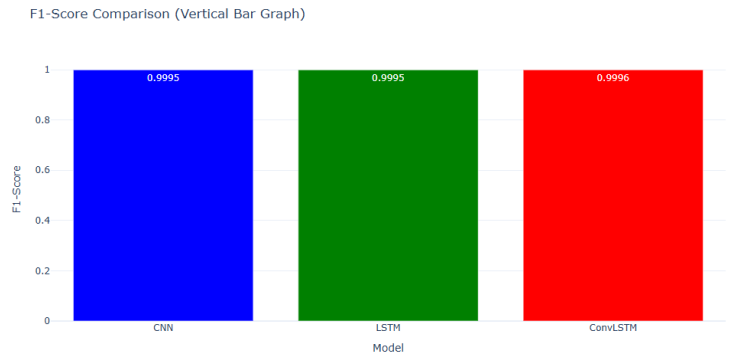

Figure 16: Comparative Analysis based on F1-Score

6.5 Discussion

All three models show almost identical accuracy, precision, recall, and F1-score, indicating the efficiency of detecting APT attacks. This may be due to powerful preprocessing and feature engineering mixed with the balancing of datasets that took place when applying SMOTE. This technique ensured that all the models were fed with a good quality

dataset characterized by significant patterns and minimal noise. Also, the spatiotemporal properties in this dataset are derived from different features of the packets and some sequential dependencies. Although there are slight variations in metrics, ConvLSTM consistently outperformed the other models. It delivers superior performance due to its unique architecture that integrates convolutional layers for spatial feature extraction with LSTM layers that model temporal dependencies. This enables ConvLSTM to model the interplay of sequential and spatial relationships present in APT attack stages like lateral movement and reconnaissance. While CNN caters mostly to spatial pattern detection and LSTM strongly focuses on temporal dependencies, ConvLSTM unites the two, giving a broader picture of the behavior of the network. In APT attack detection, this is greatly crucial since subtle packet sequence changes could indicate a possibly malicious activity.

The fundamental reason that all of the models performed similarly is that they learned from a highly structured dataset populated with clear patterns. However, ConvLSTM can generalize a bit better, and this was reflected in enhanced stability in performance metrics. In real-world scenarios of cloud security, where there is dynamic and complicated networking data, ConvLSTM presents itself as the optimal choice to fit this criteria as it can go through both spatial and temporal changes. Overall, ConvLSTM is a more robust model for APT attack detection. By using AWS Elastic Beanstalk, the model will eventually be able to flexibly scale up in order to allow for larger cycles in traffic and allow for real-time predictions without compromising precision metrics. ConvLSTM deployment fits well into cloud security goals of early detection of advanced persistent threats in order to enhance the robust health of the cloud infrastructure.

7 Conclusion and Future Work

In this project, a real-time APT attack detection system has been built and supported by the useful web application that could detect and show APT attacks in real-time. three advaned deep learning models-CNN, LSTM, and ConvLSTM were evaluated to identify the best model for accurately detecting APT attack stages. The performance of each model was analyzed thoroughly using accuracy, precision, recall, F1-score, and confusion matrix. Due to its capability of capturing both spatial and temporal dependencies that is critical for identifying complex behaviours of APT attacks, ConvLSTM became the best model. Its integration with web application ensures real-time monitoring that is robust and scalable to be deployed on AWS Elastic Beanstalk with a CI/CD pipeline that allows automated updates and scalability. In the future work live data streams will be integrated for enhanced realism, expand into multi-cloud environments for higher availability, and explore modern transformer-based architectures to improve detection. In addition, AI techniques can be in cooperated that can effectively promote the transparency of predictions for cloud administrators.

References

Ahmad, W., Rasool, A., Javed, A. R., Baker, T. and Jalil, Z. (2022). Cyber security in iot-based cloud computing: A comprehensive survey, *Electronics (Switzerland)* **11**(1): 16.

Arefin, S., Chowdhury, M., Parvez, R., Ahmed, T., Abrar, A. F. M. S. and Sumaiya, F. (2024). Understanding apt detection using machine learning algorithms: Is superior

accuracy a thing?, *IEEE International Conference on Electro Information Technology*, pp. 532–537.

aznulqalid, A. Q. M. S., Mutalib, N. H. A., Sabri, A. Q. M., Wahab, A. W. A., Abdullah, E. R. M. F., AlDahoul, N. and Mutalib, N. H. A. (2024). Explainable deep learning approach for advanced persistent threats (apts) detection in cybersecurity: a review, *Artificial Intelligence Review* **57**(11): 1–47.

Bunting, D. (2023). How to discover advanced persistent threats in aws.
URL: *https://www.chaossearch.io/blog/how-to-find-advanced-persistent-threats-aws*

Chen, Z., Liu, J., Shen, Y., Simsek, M., Kantarci, B., Mouftah, H. T. and Djukic, P. (2022). Machine learning-enabled iot security: Open issues and challenges under advanced persistent threats, *ACM Computing Surveys* **55**(5).

Hu, J., Niu, W., Yuan, Q., Yao, L., He, J., Zhang, Y. and Zhang, X. (2024). Deml: Data-enhanced meta-learning method for iot apt traffic detection, *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering*, Vol. 570, pp. 212–226.

Joloudari, J. H., Haderbadi, M., Mashmool, A., Ghasemigol, M., Band, S. S. and Mosavi, A. (2020). Early detection of the advanced persistent threat attack using performance analysis of deep learning, *IEEE Access* **8**: 186125–186137.

Kumar, A., Fahad, M., Arif, H. and Hussain, H. K. (2024). Advancements in detection and mitigation: Fortifying against apts - a comprehensive review, *BULLET: Jurnal Multidisiplin Ilmu* **3**(1): 141–150.
URL: *https://www.journal.mediapublikasi.id/index.php/bullet/article/view/4121*

Lu, Q., Li, J., Peng, Z., Wu, L., Ni, M. and Luo, J. (2024). Detecting the cyber-physical-social cooperated apts in high-der-penetrated smart grids: Threats, current work and challenges, *Computer Networks* **254**: 110776.

Milajerdi, S. M., Gjomemo, R., Eshete, B., Sekar, R. and Venkatakrishnan, V. N. (2019). Holmes: Real-time apt detection through correlation of suspicious information flows, *Proceedings - IEEE Symposium on Security and Privacy*, pp. 1137–1152.

Saini, N., Kasaragod, V. B., Prakasha, K. and Das, A. K. (2023). A hybrid ensemble machine learning model for detecting apt attacks based on network behavior anomaly detection, *Concurrency and Computation: Practice and Experience* **35**(28): e7865.

Salim, D. T., Singh, M. M. and Keikhosrokiani, P. (2023). A systematic literature review for apt detection and effective cyber situational awareness (ecsa) conceptual model, *Heliyon* **9**(7): e17156.

Singh, S., Sharma, P. K., Moon, S. Y., Moon, D. and Park, J. H. (2019). A comprehensive study on apt attacks and countermeasures for future networks and communications: challenges and solutions, *Journal of Supercomputing* **75**(8): 4543–4574.

Stojanović, B., Hofer-Schmitz, K. and Kleb, U. (2020). Apt datasets and attack modeling for automated detection methods: A review, *Computers and Security* **92**: 101734.

Subramanian, N. and Jeyaraj, A. (2018). Recent security challenges in cloud computing, *Computers and Electrical Engineering* **71**: 28–42.

Tabrizchi, H. and Rafsanjani, M. K. (2020). A survey on security challenges in cloud computing: issues, threats, and solutions, *Journal of Supercomputing* **76**(12): 9493–9532.

Vinoth, S., Vemula, H. L., Haralayya, B., Mamgain, P., Hasan, M. F. and Naved, M. (2022). Application of cloud computing in banking and e-commerce and related security threats, *Materials Today: Proceedings* **51**: 2172–2175.

Xuan, C. D. and Dao, M. H. (2021). A novel approach for apt attack detection based on a combined deep learning model, *Neural Computing and Applications* **33**(20): 13251–13264.

Xuan, C. D., Duong, D. and Dau, H. X. (2021). A multi-layer approach for advanced persistent threat detection using machine learning based on network traffic, *Journal of Intelligent and Fuzzy Systems* **40**(6): 11311–11329.

Yu, K., Tan, L., Mumtaz, S., Al-Rubaye, S., Al-Dulaimi, A., Bashir, A. K. and Khan, F. A. (2021). Securing critical infrastructures: Deep-learning-based threat detection in iiot, *IEEE Communications Magazine* **59**(10): 76–82.