Project Report

Malicious and Malformed Packet Detection

By
Aryan Deshmukh

INDEX

Introduction and overview

1. Introduction and Overview:

In an era marked by relentless digital advancements, the security of network systems stands as a paramount concern for individuals, organizations, and governments alike. The ubiquity of cyber threats, ranging from sophisticated malware to targeted cyberattacks, underscores the critical need for robust cybersecurity measures. Amidst this backdrop, the project embarks on a journey to explore the intricate realm of network packet transmission data, leveraging the power of supervised machine learning techniques.

At its core, the project seeks to address a fundamental question: Can we effectively differentiate between normal and malicious network activities based on packet transmission metadata? By delving into the rich tapestry of network data, the project endeavors to unravel patterns, anomalies, and signatures indicative of potential cyber threats. This exploration isn't merely theoretical; it carries profound implications for the realm of cybersecurity, with the potential to redefine the landscape of intrusion detection and prevention.

The chosen dataset, a subset of the UNSW-NB 15 dataset meticulously curated by the Australian Centre for Cyber Security, serves as the cornerstone of this endeavor. This dataset, comprises a comprehensive array of attributes.

2. Project Background and Scope:

In today's interconnected digital landscape, cybersecurity stands as a paramount concern for individuals, businesses, and governments alike. With the proliferation of sophisticated cyber threats ranging from malware to phishing attacks, the need for robust cybersecurity measures has never been more pressing. Network-Based Intrusion Detection Systems (NIDSs) play a pivotal role in safeguarding network infrastructure by identifying and thwarting potential cyber threats in real-time.

The project stems from the backdrop of escalating cyber threats and the growing necessity to fortify existing cybersecurity mechanisms. Traditional rule-based NIDSs often struggle to keep pace with the dynamic nature of modern cyber threats, necessitating the exploration of more advanced, adaptive approaches. By leveraging machine learning techniques, particularly supervised learning algorithms, the project seeks to augment the capabilities of NIDSs in discerning between benign and malicious network activities.

The scope of the project encompasses a thorough analysis of network packet transmission data, focusing on extracting meaningful insights from packet metadata. By scrutinizing attributes such as packet flow, protocol, and service information, the project endeavors to uncover patterns indicative of malicious behavior. Through meticulous data preprocessing, feature selection, and model training, the project aims to develop predictive models capable of accurately classifying network activities as either normal or malicious.

The chosen dataset, a subset of the UNSW-NB 15 dataset, offers a rich repository of network packet data aggregated from real-life network traffic and simulated cyber-attacks. With 49 attributes describing packet flow dynamics, the dataset provides a fertile ground for in-depth analysis and model development. By harnessing the wealth of information encapsulated within the dataset, the project aspires to contribute to the advancement of cybersecurity research and the development of more effective intrusion detection mechanisms.

Ultimately, the project endeavors to bridge the gap between cutting-edge machine learning techniques and cybersecurity challenges, paving the way for enhanced network security measures. By elucidating the intricate interplay between network packet data and cyber threats, the project seeks to empower organizations and individuals with the tools and insights needed to bolster their defenses against evolving cyber threats.

3. Project Rationale and Goals:

In today's interconnected digital landscape, cybersecurity stands as a paramount concern for individuals, businesses, and governments alike. The escalating sophistication of cyber threats necessitates continuous innovation and adaptation in defensive measures. The rationale behind this project stems from the urgent need to fortify cybersecurity infrastructure against evolving cyber threats.

The overarching goal of the project is twofold:

- Enhanced Cybersecurity: The primary objective is to develop robust models capable of accurately distinguishing between normal network activities and malicious intrusions. By leveraging supervised machine learning techniques, the project endeavors to identify subtle patterns and anomalies in network packet transmissions indicative of potential cyber threats. This proactive approach to threat detection aims to fortify cybersecurity defenses, mitigating the risk of data breaches, system compromises, and other cyberattacks.
- Improvement of Intrusion Detection Systems (IDSs): The project further aims to contribute to the refinement and advancement of Network-Based Intrusion Detection Systems (NIDSs). By uncovering key features and indicators of malicious activity through data analysis, the project seeks to inform the development of more effective and efficient IDSs. These systems play a crucial role in monitoring network traffic, detecting anomalies, and alerting administrators to potential security breaches. By enhancing the capabilities of IDSs, the project aims to empower organizations with enhanced situational awareness and proactive threat mitigation strategies.

In essence, the project aligns with broader efforts to strengthen cybersecurity posture and resilience in the face of evolving cyber threats. By leveraging data-driven insights and machine learning algorithms, the project endeavors to pave the way for more robust and adaptive cybersecurity solutions, ultimately safeguarding critical assets, information, and infrastructure against malicious cyber activities.

4. List of Tasks Completed in Internship:

Throughout the internship, I completed a series of tasks aimed at achieving the project goals. Here is a comprehensive list of tasks completed:

I. Dataset Selection and Exploration:
- Identification of suitable datasets for the project, focusing on the UNSW-NB 15 dataset (2015) due to its relevance to network security.
- Exploration of the dataset to understand its structure, features, and potential insights it offers regarding network packet transmissions.

II. Data Preprocessing:
- Initial data cleaning: Removal of redundant or irrelevant features such as 'id' and 'attack_cat'.
- Log transformation: Transformation of numerical features to address skewed distributions and bring them closer to Gaussian distributions, enhancing the suitability for machine learning algorithms.
- Outlier detection: Examination of numerical features for outliers and decision-making regarding their treatment, such as removal or retention.

III. Feature Selection:
- Mutual Information (MI) analysis: Calculation of MI scores between all features and the target variable ('label') to identify informative features.
- Dropping low-information features: Features with MI scores below a predetermined threshold (e.g., 0.2) were dropped to reduce dimensionality and mitigate overfitting.

IV. Implementation of Supervised Machine Learning Algorithms:
- Application of k-nearest neighbors (kNN) algorithm: Utilization of kNN for classification tasks, leveraging its ability to classify instances based on the similarity of their features.
- Decision tree modeling: Development of decision tree models to create intuitive, interpretable rules for classification, aiding in understanding feature importance and classification criteria.

V. Model Evaluation:
- Cross-validation: Partitioning of the dataset into training and testing sets, followed by k-fold cross-validation to assess model performance on multiple subsets of the data.
- Evaluation metrics: Calculation of accuracy, recall, precision, and F1-score to quantify the performance of the models in classifying normal and malicious network activities.

VI. Comparison and Selection of Best-Performing Model:
- Comparative analysis: Comparison of performance metrics (e.g., accuracy, recall)

between the kNN and decision tree models.

- Selection of optimal model: Identification of the model demonstrating superior performance based on evaluation metrics and suitability for the project's objectives.

VII. Testing on Unseen Data:

- Utilization of held-out test data: Application of the selected model to unseen data to assess its generalization ability and performance in real-world scenarios.
- Evaluation of model performance: Examination of accuracy, recall, precision, and other metrics on the test set to validate the effectiveness of the model in detecting malicious network activities.

5. Tools and Software Learned:

i. Data Preprocessing Techniques:

- Understanding and implementing techniques to clean and prepare data for analysis, such as handling missing values, outlier detection, and feature scaling.
- Familiarity with data transformation methods like log transformation to address skewed distributions and normalize data.

ii. Machine Learning Algorithms:

- k-Nearest Neighbors (kNN): Learning the fundamentals of the kNN algorithm, including its implementation for classification tasks and the importance of choosing the appropriate value of k.
- Decision Trees: Understanding decision tree algorithms, including their construction, feature importance, and visualization techniques to interpret model results.

iii. Data Visualization Tools:

- Matplotlib: Mastery of Matplotlib library for creating static, interactive, and publication-quality visualizations to explore data distributions, trends, and relationships.
- Seaborn: Utilization of Seaborn for enhanced data visualization, including statistical plots, heatmaps, and distribution plots.

iv. Evaluation Metrics for Classification Models:

- Comprehensive understanding of evaluation metrics such as accuracy, recall, precision, and F1-score to assess the performance of classification models.
- Interpretation of confusion matrices to analyze model predictions and understand true positives, true negatives, false positives, and false negatives.

v. Cross-Validation Techniques:

- Implementation of k-fold cross-validation to assess model performance and mitigate overfitting by evaluating models on multiple subsets of the data.
- Utilization of train-test splits to partition data into training and testing sets, ensuring robust model evaluation on unseen data.

6. System Implementation:

i. Data Preprocessing:

- Feature Selection: The initial step involved selecting relevant features for analysis while discarding irrelevant ones such as 'id' and 'attack_cat'. This ensured that the analysis focused on pertinent packet transmission metadata.
- Log Transformation: Certain numerical features underwent log transformation to normalize their distributions, making them more amenable to analysis. This step aimed to mitigate the effects of heavily positively skewed distributions, thereby improving the performance of machine learning algorithms.
- Outlier Detection: An inspection for outliers was conducted across all numeric features to identify any anomalies. However, since no clear indications of outliers were found, the decision was made to retain all data points for analysis.

ii. Model Training:

- kNN Algorithm Implementation: The k-nearest neighbors (kNN) algorithm was applied to the preprocessed dataset to classify network activities as normal or malicious. Multiple values of k were tested, and the algorithm's performance was evaluated using cross-validation techniques.
- Decision Tree Implementation: In addition to kNN, a decision tree algorithm was also employed to create a predictive model. Decision trees offer an intuitive representation of the data's classification patterns and provide insights into the most influential features for distinguishing between normal and malicious activities.

iii. Model Evaluation:

- Cross-Validation: To ensure robustness and generalizability of the models, k-fold cross-validation was employed. This technique involves splitting the dataset into multiple subsets, training the models on different combinations of these subsets, and evaluating their performance across various folds. By averaging the performance metrics across folds, a more accurate assessment of the models' effectiveness was obtained.

iv. Comparison and Selection of Best Model:

- Performance Metrics: The performance of both the kNN and decision tree models was evaluated using metrics such as accuracy, recall, precision, and F1-score. These metrics provided insights into the models' ability to correctly classify network activities and distinguish between true positives, true negatives, false positives, and false negatives.
- Selection Criteria: Based on the performance metrics obtained from cross-validation, the best-performing model was selected for further analysis. Factors such as accuracy, recall, and precision were considered in determining the model's suitability for real-world deployment.

7. System Testing:

System testing is a critical phase in the development of any machine learning-based system, especially in the context of cybersecurity where the accuracy and reliability of the model are paramount. In this project, system testing involved several key steps to ensure the effectiveness and robustness of the developed models:

i. Test Set Preparation: Before initiating testing, a portion of the dataset was set aside as the test set, distinct from the data used for model training and validation. This separation ensures that the model is evaluated on unseen data, providing a more accurate assessment of its real-world performance.
ii. Model Evaluation Metrics: Various evaluation metrics were employed to assess the performance of the models. These metrics typically include accuracy, recall, precision, and F1-score. Accuracy measures the overall correctness of the model's predictions, while recall quantifies the model's ability to correctly identify all relevant instances. Precision evaluates the accuracy of positive predictions, and the F1-score provides a balance between precision and recall.
iii. Confusion Matrix Analysis: The confusion matrix provides a detailed breakdown of the model's predictions compared to the actual labels. It consists of four quadrants: true positives (correctly predicted malicious packets), true negatives (correctly predicted normal packets), false positives (incorrectly predicted as malicious), and false negatives (incorrectly predicted as normal). Analyzing the confusion matrix helps identify any patterns or discrepancies in the model's performance.
iv. Model Performance Comparison: The performance of different models, such as k-nearest neighbors (kNN) and decision trees, was compared using the evaluation metrics mentioned above. This comparison aids in selecting the best-performing model for deployment in real-world scenarios.
v. Generalization Testing: Generalization testing ensures that the model performs well on data outside the training and validation sets. By evaluating the model on the test set, which comprises unseen data, one can assess its ability to generalize to new instances effectively.
vi. Overfitting Detection: Overfitting occurs when a model learns the training data too well, leading to poor performance on unseen data. Techniques such as cross-validation and regularization were employed to mitigate overfitting and ensure that the model's performance remains robust across different datasets.

8. Learning Outcome of Internship:

i. Deep Understanding of Network Packet Data Analysis:
- Through hands-on experience with real-world datasets, the internship provided a comprehensive understanding of network packet data analysis.
- Learned techniques to preprocess and manipulate raw data, including feature selection, log transformation, and outlier detection.

ii. Proficiency in Machine Learning Techniques for Cybersecurity:
- Acquired proficiency in applying supervised machine learning techniques to cybersecurity problems, particularly in the context of intrusion detection.
- Gained insights into the application of algorithms such as k-nearest neighbors (kNN) and decision trees for classifying network activities.

iii. Data Preprocessing and Model Training Skills:
- Developed skills in data preprocessing, including handling categorical features, normalizing numerical data, and addressing data skewness.
- Implemented model training procedures, including parameter tuning and optimization, to achieve optimal performance.

iv. Evaluation and Performance Metrics Mastery:
- Learned to evaluate model performance using a variety of metrics such as accuracy, recall, precision, and F1-score.
- Gain proficiency in interpreting confusion matrices and understanding the implications of different evaluation metrics on model performance.

v. Cross-Validation Techniques and Model Comparison:
- Acquired knowledge and skills in cross-validation techniques to assess model generalization performance.
- Learned how to compare and select the most suitable model based on performance metrics and validation results.

vi. Problem-Solving and Critical Thinking:
- Developed problem-solving skills by tackling real-world cybersecurity challenges and devising appropriate solutions.
- Cultivated critical thinking abilities by analyzing model results, identifying areas for improvement, and iteratively refining methodologies.

vii. Application of Knowledge in Practical Scenarios:
- Applied theoretical knowledge gained from coursework to real-world scenarios, enhancing practical problem-solving abilities.
- Explored the practical implications of machine learning in cybersecurity and gained insights into the challenges and opportunities in the field.

Overall, the internship provided a holistic learning experience, combining theoretical knowledge with practical application in the domain of cybersecurity. The skills and expertise acquired during the internship are invaluable assets for addressing complex cybersecurity challenges and contributing to the advancement of intrusion detection systems and network security practices.

9. Conclusion:

This internship culminates in a significant advancement towards fortifying cybersecurity measures through the application of supervised machine learning techniques. By leveraging a subset of the UNSW-NB 15 dataset and employing k-nearest neighbors (kNN) and decision tree algorithms, we've made strides in identifying patterns indicative of malicious network activities.

The project's success lies in achieving a test set accuracy exceeding 90%, demonstrating the effectiveness of the developed models in distinguishing between normal and malicious network behavior. This high level of accuracy underscores the potential of machine learning in bolstering cybersecurity defenses, offering a proactive approach to identifying and mitigating cyber threats.

Furthermore, the project highlights the importance of ongoing research and innovation in cybersecurity to keep pace with the evolving threat landscape. As cyber threats continue to evolve in sophistication, it's imperative to continually refine and enhance detection mechanisms to stay ahead of malicious actors.

Moving forward, the insights gained from this project can serve as a foundation for further research and development in cybersecurity. By continuing to explore and refine machine learning models, we can further enhance our ability to detect and mitigate cyber threats, ultimately contributing to a more secure digital environment for individuals, businesses, and governments alike.

10. References:

- https://research.unsw.edu.au/projects/unsw-nb15-dataset
- GitHub - pevma/mrp: Malformed random packet pcaps for some QA testing.
- ML-Malicious-Packet-Detection/project_group069.pdf at main · j-g-robbins/ML-Malicious-Packet-Detection · GitHub
- Defense Against Malformed Packet Attacks - S1720, S2700, S5700, and S6720 V200R011C10 Configuration Guide - Security - Huawei
- c# - "Malformed Packet: TNS" retriving data from Oracle database - Stack Overflow
- Cyber Attacks Explained: Packet Crafting - open source for you (opensourceforu.com)
- Single-packet attacks (hpe.com)