Network Intrusion Detection System

A Project Report

Submitted By

HARISH B - 2019202016

A report for the project submitted to the faculty of

INFORMATION SCIENCE AND TECHNOLOGY

In partial fulfillment for the degree of

MASTER OF COMPUTER APPLICATION

DEPARTMENT OF INFORMATION SCIENCE AND TECHNOLOGY

COLLEGE OF ENGINEERING, GUINDY

ANNA UNIVERSITY

CHENNAI-600025

APRIL 2022

Abstract:

A machine learning-based NIDS (ML-NIDS) that detects anomalies through ML algorithms by analyzing behaviors of packets. However, the ML-NIDS learns the characteristics of attack traffic based on training data, so it, too, is inevitably vulnerable to attacks that have not been learned, just like pattern-matching machine learning.The proposed approach can provide more robust and more accurate classification with the same classification datasets compared to existing approaches, so we expect it will be used as one of feasible solutions to overcome weakness and limitation of existing ML-NIDS..

Introduction (Network Security) :

Network Security is an emerging field in the IT-sector. As more devices are connected to the internet, the attack surface for hackers is steadily increasing. Network-based Intrusion Detection Systems (NIDS) can be used to detect malicious traffic in networks and Machine Learning is an up and coming approach for improving the detection rate. In this thesis the NIDS Zeek is used to extract features based on time and data size from network traffic. The features are then analyzed with Machine Learning in Scikit-learn in order to detect malicious traffic. A 98.58% Bayesian detection rate was achieved for the CICIDS2017 which is about the same level as the results from previous works on CICIDS2017 (without Zeek). The best performing algorithms were K-Nearest Neighbors, Random Forest and Decision Tree.

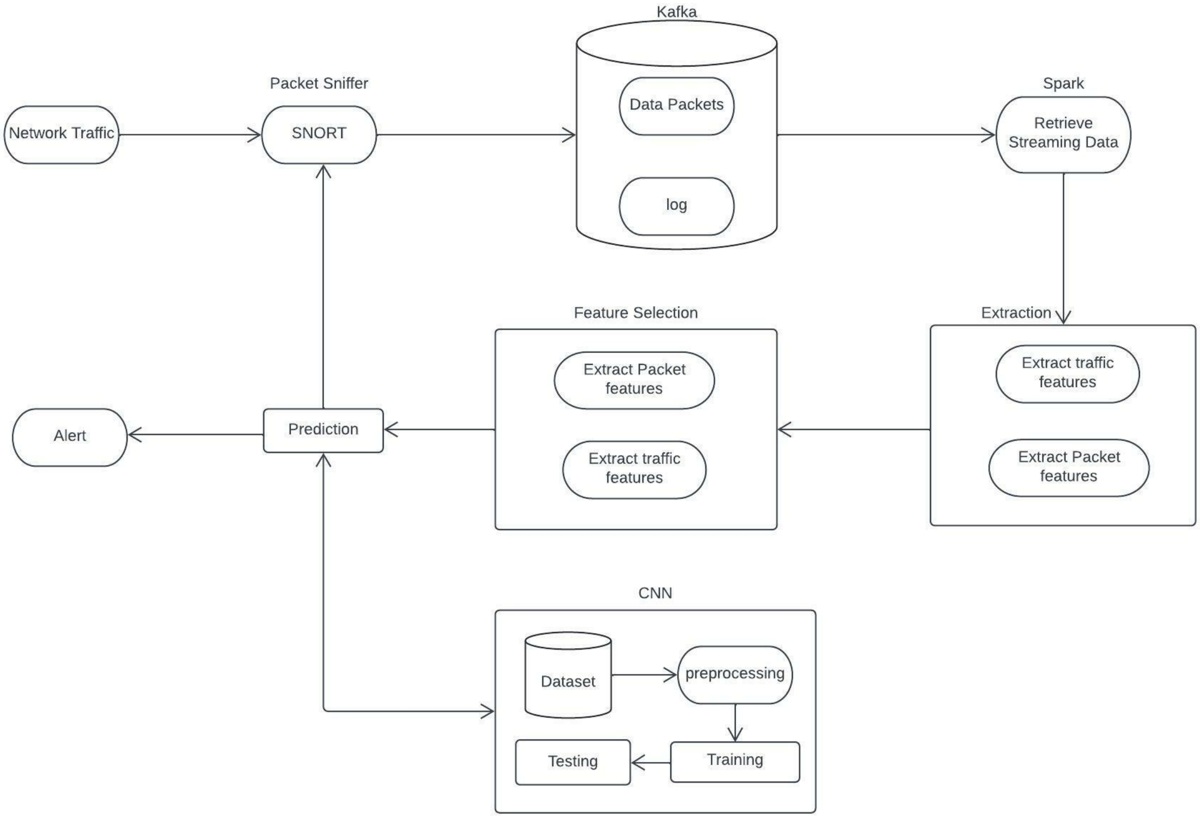
Problem Statement:

Network Intrusion Detection Systems (NIDSs) using pattern matching have a fatal weakness in that they cannot detect new attacks because they only learn existing patterns and use them to detect those attacks.

Objective

The Objective of this thesis project is to perform classification to improve detection of malicious traffic. Research which traffic features and machine learning algorithms that are suitable for detecting different kinds of malicious traffic. Write scripts to extract those features. Train machine learning models from a labeled dataset with malicious traffic. Evaluate the performance of the different models and scripts.

Architecture Diagram:



Architecture Explanation:

- ➢ Packet Sniffer will capture the packets from Network Traffic
- ➢ Data Packets and Log of Network Traffic will be streamed by kafka
- ➢ Spark will get the streaming data
- ➢ Extraction of features from packets and traffic
- ➢ Selection of features from extracted features
- ➢ CNN is trained and tested using Dataset
- ➢ Prediction is done using CNN
- ➢ Alert is created if there is malicious in Network

List of Modules:

- ❖ Packet sniffer
- ❖ Streaming
- ❖ Data retrieve
- ❖ Extraction and Selection
- ❖ CNN

Brief Description of Modules:

- Packet Sniffer:

  Snort will be configured to monitor network traffic. Snort is based on library packet capture (libpcap). Libpcap is a tool that is widely used in Transmission Control Protocol/Internet Protocol address traffic sniffers, content searching and analyzers for packet logging, real-time traffic analysis, protocol analysis and content matching.

- Streaming:

  Kafka is configured to stream the captured data. This will stream data packets and traffic logs. Kafka will run as a cluster of one or more servers that can span multiple datacenters. Some of these servers are from the storage layer, called the brokers. Other servers run Kafka Connect to continuously import and export data as event streams to integrate Kafka with your existing systems such as relational databases as well as other Kafka clusters.

- Data Retrieve:

  Spark is configured to retrieve the streaming data packets and traffic log. The incoming live data is converted into small batches and processed by the processing engine.

- Extraction and Selection:

  ZAT(Zeek Analysis Tool) is used to extract the features from data. From the extracted features is converted into dataset format. Dataset is used for features selection and those features are used for classification.

- CNN:

  CNN is constructed from three layers of convolution, pooling, and fully connected layer [17]. In the convolutional layer, a filter or kernel passes through the input image and forms a conclusion of an array of numbers. Multiplying the kernel across the portion of the input produces a single number, by moving the filter through the whole image. A metric of multiple numbers is generated, which refers to a feature map. Using multiple kernels produces a number of feature maps, which represent different characteristics of the input tensors.

References:

- Robust Network Intrusion Detection System Based on Machine-Learning With Early Classification | IEEE Journals & Magazine | IEEE Xplore

- Evaluating and Improving Adversarial Robustness of Machine Learning-Based Network Intrusion Detectors | IEEE Journals & Magazine | IEEE Xplore

- R. Zhao et al., "An Efficient Intrusion Detection Method Based on Dynamic Autoencoder," in IEEE Wireless Communications Letters, vol. 10, no. 8, pp. 1707-1711, Aug. 2021, doi: 10.1109/LWC.2021.3077946.