NETWORK TRAFFIC DATA STREAMING FOR INTRUSION DETECTION SYSTEMS : SUBJECT PROPOSAL

Sotiris Evangelou
AEM: 2159
sevangelou@uth.gr

Konstantinos Kalais
AEM: 2146
kkalais@uth.gr

Leuteris Chatziefraimidis
AEM: 2209
echatzief@uth.gr

January 16, 2020

1 Introduction

Cybersecurity is the practice of protecting systems, networks, and programs from digital attacks, and the subprocess of the identification of these attacks is called Intrusion Detection. Intrusion Detection Systems (IDS) that monitor network traffic and anomalies are called NIDS (network intrusion detection systems) and often use an approach called anomaly-based detection which detects deviations from normal traffic behaviours. An IDS is a system that needs to predict on continuous flows of data, and thus, offer real time security. The training is a process that can be done statically with pre-collected datasets, but the process of classifications and predictions should be on near real time (NRT) collections of data, so that the security mitigations are taking place as soon as possible.

2 Motivation

During the course of Machine Learning our team created a Network IDS that classifies network traffic data to decide whether an attack is occuring in the network, and what type of attack it is. The problem is that the data we used for the training and testing phase were static and pre-collected. Our motivation for this project, is to expand the initial system to collect, process and make continuous predictions on streaming data. We believe that this will make our system more useful in real cases where the data are continuously generated and need to be processed in a quick and efficient fashion.

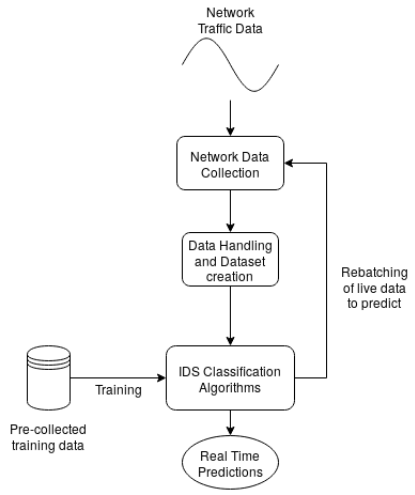

Figure 1: Structure of the Collecting, Processing and Predicting

3 Methodology

The procedure we are going to follow consists of four levels:

- In the first level, we need to listen to a network interface for network traffic.
- In the second level, we need to collect the data from the network traffic, and export them into a predefined format.
- In the third level, we need to parse the formatted file, extract the fields we need to create a dataset we will use for testing and predicting.
- In the fourth and final level, we need to pass the data to the IDS we have already created, and post-process the findings of the real time data predictions.

We can observe the conceived structure in figure 1.

4 Technical Implementation

4.1 Collection

The listening and collection of network traffic data will be achieved using Wireshark and/or Tcpdump. These are the most popular tools for collection of network packet data and the data can be exported to a pcap file that can later be parsed and processed.

4.2 Parsing and Dataset creation

Parsing can be easily done with a high-level language such as python, or, if we decide that the performance is essential we might be guided to a lower language such as C, C++ or Go. The dataset creation can be done using the same languages.

4.3 Data Feeding and Automation

The dataset feeding will be conducted using a scripting language such as Python or Bash, which will be the same way the whole ensemble process will be automated.

5 Notes

The data-streaming project described will be an extension to our Network IDS. The repository is located at this link: https://github.com/EvangelouSotiris/NIDS_Project_CE417. This project will include it as a "layer" of an ensemble model.