Network Intrusion Detection System with Machine Learning

Evangelou Sotirios, Kalais Konstantinos, Chatziefremidis Eleftherios

[1]Department of Electrical and Computer Engineering, University of Thessaly, Greece

Summary: A Network IDS system, that analyzes network packets and their metadata, in order to decide on the normality of the traffic, and the existence of an attack as well as its type. The data are being collected, processed into a dataframe and fed into the model in real time to produce real time intrusion detection.

https://github.com/EvangelouSotiris/NIDS_Project_CE417

Motivation

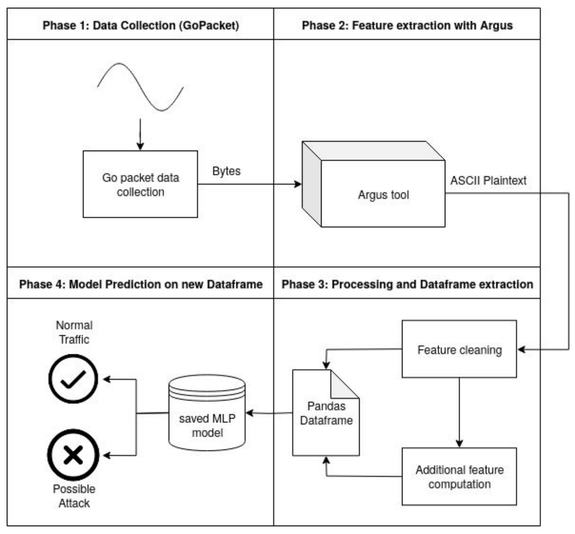
- Billions of computers, Servers,smartphones, IOT devices are connected each day on external or internal networks.
- The attack surface on network security widens by the day, and there is great need for well-performing Network Security Systems such as Firewalls, IDS, IPS etc.
- The packets of (possibly malicious) data flowing through the network take the form of Big Data and can be effectively handled and inspected leveraging the power of Data Streaming, Machine Learning and Neural Networks.

Data

- The dataset we used belongs to the Cyber Range Lab of the Australian Centre for Cyber Security (ACCS)[3], and was created artificially with the IXIA PerfectStorm tool and a custom testbed.
- The dataset is pre-split into a training set containing 175341 instances and a testing set containing 82332 instances.
- The dataset contains 42 features from which we remove 14 highly correlated (>85%) features during preprocessing phase using FeatureSelector[4] library.
- The dataset also provides 2 target columns, one according to the normality of the traffic (2 classes) and one according to the attack type (10 classes).

Data Streaming Pipeline

In order to collect data, export features from them, create dataframes and predict with our model in real time, we introduce this pipeline model with 4 phases.



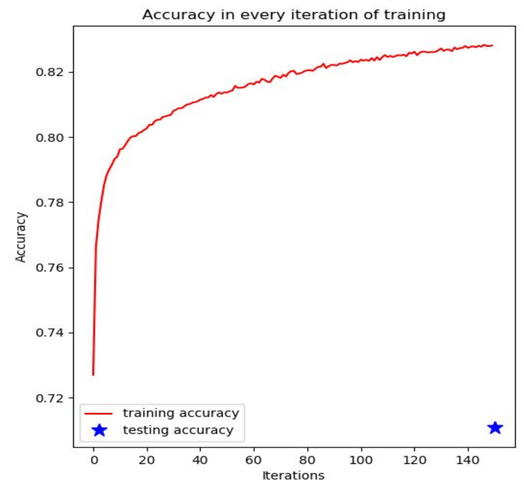
1. **Data collection**: Gopacket library is being used in order to read packets from a network interface and forward them as bytes into the next phase.
2. **Feature extraction**: Argus tool reads the stream input, produces reports on it, and the ra (read argus) tool is used to extract 23 of the 28 features we need for our dataframe from these reports.
3. **Additional feature computation:** Custom python scripts are used to compute the last five features based on information produced by the other features, and flow information.
4. **Real time model predictions**: A dataframe is created using the pipeline and passed into our saved high-accuracy MLP model for real time predictions on each instance.

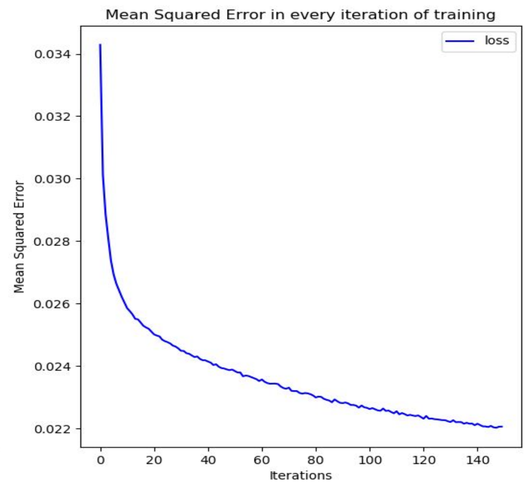
Machine Learning Model

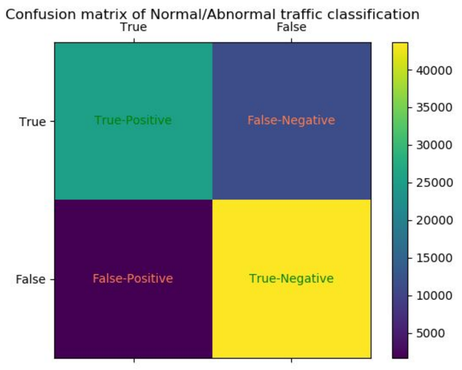
- Based on research papers[1].[2] three methods were approached to analyse the problem, namely *SVM*, *Random Forest* and *Multilayer Perceptron*, leveraging the Keras framework and Scikit learn library.
- The **MLP** we use in the project is a deep neural network with three Fully Connected layers of 250-100-10 neurons.
- We found that the best approach was to classify by attack type and aggregate the attacks to a class of abnormal behaviour to decide on the normality of the traffic.
- We are using Adam optimizer, a categorical crossentropy loss function, and fit the model on scaled data to a range of (0,1).
- The **SVM** we use in the project uses a C value of 100 and a gamma value of 0.00001. This method proved to be very slow for the size of the dataset and problem.
- The **Random Forest** uses 1000 subtrees-estimators and a max-depth of 5. The results it produces are slightly better than the SVM's performance but still not as good as MLP's.

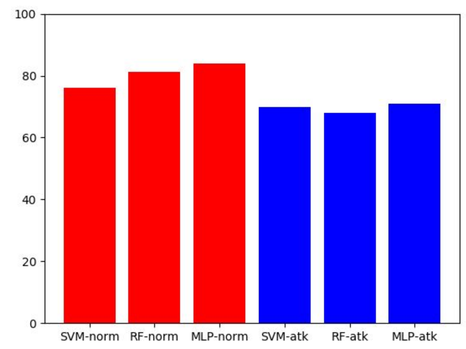
Experimental Results

SVM:
Normal/Abnormal Classification accuracy: 76.101%
Attack Type Classification accuracy: 68.087%
Random Forest:
Normal/Abnormal Classification accuracy: 81.336%
Attack Type Classification accuracy: 68.087%
MultiLayer Perceptron:
Training Normal/Abnormal classification accuracy: 82.81%
Normal/Abnormal Classification accuracy: 83.960%
Attack Type Classification accuracy: 71.075%
False Positive Percentage: 1.4%









Real Time Predictions

The pipeline is utilised to produce real time classifications on the instances produced. The instances classified as attacks are exported into a csv file for further analysis.

```
Normal Behavior
Possible 'Exploits' Attack : added to out.csv for analysis.
Normal Behavior
Possible 'Fuzzers' Attack : added to out.csv for analysis.
Normal Behavior
Normal Behavior
Normal Behavior
Normal Behavior
Normal Behavior
Normal Behavior
Normal Behavior
Normal Behavior
Normal Behavior
```

References:
[1]Mahdi Zamani, Machine Learning Techniques for Intrusion Detection, University of New Mexico, 2013.
[2]Srinivas Mukkamala, Guadalupe Janoski, Andrew Sung, Intrusion Detection: Support Vector Machines and Neural Networks , New Mexico Institute of Mining and Technology, 2002
[3] Nour Moustafa, Abdelhameed Moustafa, Designing an online and reliable statistical anomaly detection framework for dealing with large high-speed network traffic, University of New South Wales, Australia, June 2017.