Karla Muller

DDoS Detection and Classification

INTRODUCTION

Denial of Service attacks have been one of the most common threats in cybersecurity since they first originated in 1996. Panix Internet Service Provider was flooded with SYN requests and brought the services down for several days (Cisco, 2004). Since then, the volume, diversity and complexity of these attacks has grown from a DoS of a couple Gbps to a Distributed Denial of Service attack of 1.2 Tbps.

Prevention of DDoS attacks is now based on a number of different techniques that are mostly based on balancing loads of servers in different data centres, buying more bandwidth, buying protection appliances and software modules from security vendors, network configurations, among others (eSecurity, 2018). However, these techniques only help with the mitigation of specific types of attacks and have no real value when the attack can simultaneously be based on multiple layers and taxonomies. For example, Volume Based attacks differ from Protocol attacks and Application Layer attacks as they target the network in very different ways. A Volume Based attack like UDP floods will happen quite fast while Application Layer attacks will include low-and-slow requests from seemingly legitimate and innocent users (Imperva).

Based on these issues and the complex structures of DDoS attacks, can Machine Learning models predict whether a network packet request is an attack? Can it classify it based on the type of attack?

The goal of this project was to answer this question with the help of a dataset that includes simulations of the most up-to-date DDoS attacks; the Canadian Institute for Cybersecurity's DDoS Evaluation Dataset (CICDDoS2019).

DATA

Data is in 12 csv files and each includes 88 network traffic features (independent variables) and a Label feature describing the flow as benign or attack. This analysis includes 5 of these attacks:

1. MSSQL: Reflective TCP based, targets with reflected, amplified traffic using the Microsoft SQL Resolution Protocol (MC-SQLR)
2. NetBIOS: Reflective TCP/UDP based, allows access from separate computers to shared resources over a local network
3. LDAP: Reflective TCP/UDP based, abuses the Lightweight Directory Access Protocol (LDAP) used for directory services on corporate networks
4. SYN: Exploitation TCP based, floods the system with SYN requests
5. UDP: Exploitation UDP based, floods the system with UDP requests

EDA

Data preprocessing:
Due to the unbalanced nature of these attacks benign network flows were greatly outnumbered (approximately 1:1000). Figure 1.a shows the original data imbalance.

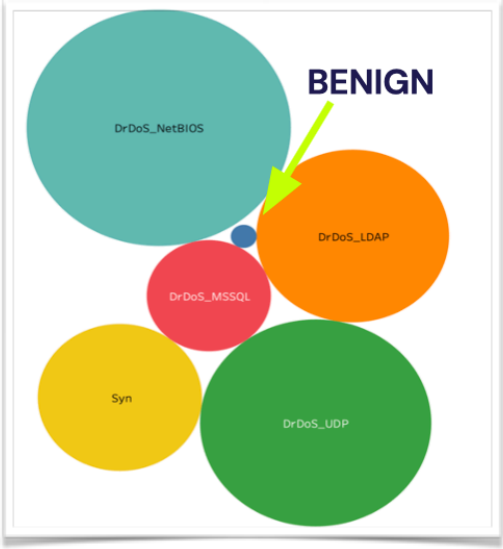

Figure 1. Dataset imbalance after subsampling attacks and oversampling benign data.

Minority class was augmented by taking 100% benign connections of each data set and subsampling attacks by 0.2. Data was subsequently reduced by Random Undersampling the train data attacks to equal the size of benign flows. Both data sets were used for comparison. The subsets of data were shuffled into a main dataframe that was used for analysis.

Data cleaning:
Infinity values for were replaced with maximum values for the variables 'Flow Bytes' and 'Flow Packets'. Additionally, NaN values for 'Flow Bytes' were replaced with median values of each class (non-normal distribution).

Feature engineering:
The number of features were reduced from 88 to 40 by dropping identifier columns and by penalizing the absolute size of the regression coefficients using LASSO. Further exploration after modelling revealed additional identifier features such as Destination Port and Unnamed: 0.1.

MODELLING

Due to its interpretable nature, Decision Tree Classifier was the chosen model to predict attacks and benign flows. GridSearchCV was used on Decision Tree Classifiers with a 5-fold cross-validation and an optimal max depth 10 for balanced train data. The weighted f1-score was 0.99. As the minority class, the benign classification was the most affected by hyperparameter tuning.

	precision	recall	f1-score	support
0	1.00	0.99	1.00	1575
1	0.94	0.97	0.96	36180
2	1.00	1.00	1.00	87197
3	1.00	0.99	0.99	163731
4	1.00	1.00	1.00	63292
5	0.99	1.00	0.99	125386
accuracy			0.99	477361
macro avg	0.99	0.99	0.99	477361
weighted avg	0.99	0.99	0.99	477361

Figure 2. ClassificationReport shows the multi-class main classification metrics.

Model Evaluation and Results:
All DT models resulted in 0.99 accuracy score for both train and test sets. The F1 score 'weighted' parameter was used to evaluate the models. This parameter calculates metrics for each label, and finds their average weighted by the true instances for each label; accounting for data imbalance (sklearn.metrics.f1_score). The classification report is shown in Figure 2.

Figure 3 shows the top 5 features the DT classifier used to make its predictions. The variables were evaluated by the Gini Importance ratio. These variables represent the key identifiers that set apart benign and attack connections to the server. Refer to this diagram for reference on packet request structure (Evans, 2017). The confusion matrix of the Decision Tree Classifier is show in Figure 4.

The model was able to correctly classify among the different types of attacks and benign flows with both high precision and recall. The model lost interpretability as the maximum depth increased.

The second stage of the analysis included the binary classification. With a max tree depth of 20, the DT received a weighted avg of 1.0. However the precision of class 0 (benign flows) decreased to 0.7

	Predictec Class 0	Predicted Class 1
True 0	1575	0
True 1	519	475267

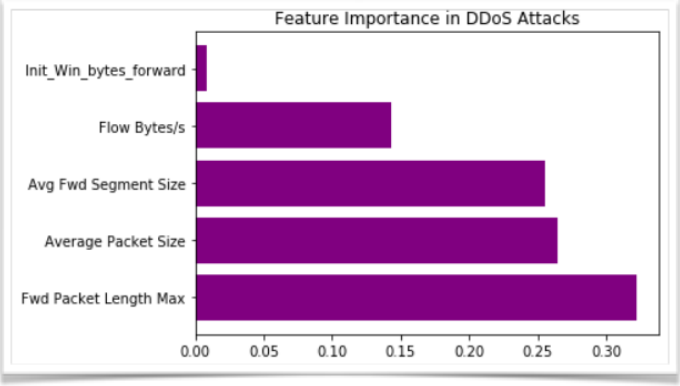

Figure 3. Top 5 variables of Decision Tree Classifier based on the Gini Importance ratio for multi-class classification.

	Predictec Class 0	Predicted Class 1	Predicted Class 2	Predicted Class 3	Predicted Class 4	Predicted Class 5
True 0	1574	0	0	0	1	0
True 1	5	35616	183	106	0	270
True 2	18	164	86961	10	3	41
True 3	47	2000	8	161639	2	35
True 4	12	10	0	0	63267	3
True 5	109	1632	95	8	7	123535

Figure 4. Confusion matrix of DT Classifier shows predicted classes (columns) and true classes (rows) of Benign : 0, MSSQL : 1, LDAP : 2, NetBIOS : 3, Syn: 4, UDP: 5.

An XGB model increased class 0 precision to 0.92. The simplification of the model from multi-class to binary resulted in a significant decrease in precision for class 0 (from 1.0 to 0.75). This may be due to the “spatial” overlap of data points. Figure 5 shows a PCA plot and a t-SNE plot of the binary classification. Pleas see GitHub or notebook for multi-class diagrams.

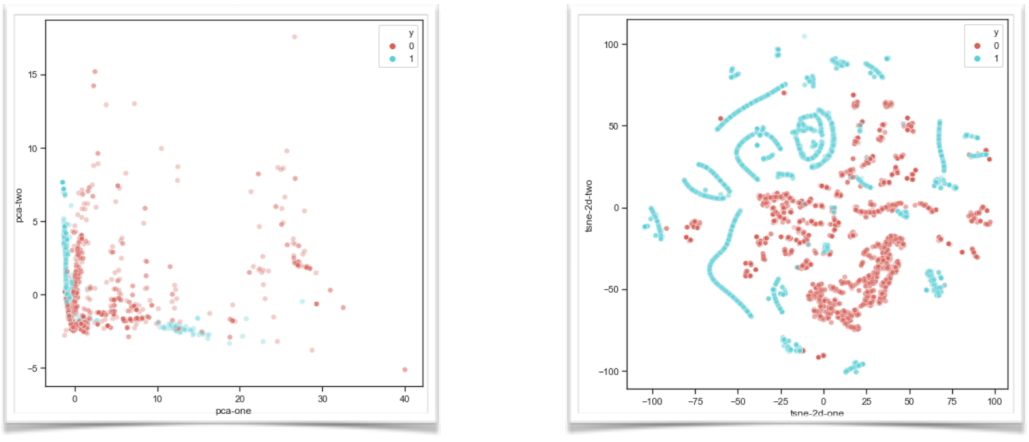

Figure 5a.First two components of PCA (left) and t-SNE (right).

Karla Muller

FUTURE DIRECTION

This model could be used in production as a detection method for DDoS attacks. The future direction of this project aims to pipe network traffic to detect invasions in real-time.

ACCESS

Please refer to this repository for the data and notebooks used for these analyses:
https://github.com/kmuller33/ddos-detect

REFERENCES

"Distributed Denial of Service Attacks - The Internet Protocol Journal - Volume 7, Number 4". *Cisco*. Retrieved 2020-03-27.

Rubens, P. (2018). “How to Prevent DDoS Attacks: 6 Tips to Keep Your Website Safe”. eSecurity. Retrived 2020-03-27.

“DDoS Attacks”. Imperva. Retrieved 2020-03-278.

Evan, J. (2017). “How big can a packet get?”. Retrieved 2020-03-27.

Derksen, L. (2016). “Visualising high-dimensional datasets using PCA and t-SNE in Python”. Retrieved 2020-03-27.S