Power Dispatching Network Attack Identification by XGBoost

Yan WANG[1,*], Mingyu SUN[1], Ning HU[1], Sentao LIU[1] and Juncheng SI[1]

[1] State Grid Shandong DongYing Electric Power Company, DongYing 257091,
Shandong Province, China

E-mail: * zdhkj2019@163.com

Abstract. Power dispatching network is a core of cardinal significance, which always suffers from the network attacks from both inside and outside of the local-area network. The network attack detection process must perform tremendously good, or will disturb the normal business traffic. This paper proposed an automatous method to inspect the net traffic and identify the abnormal traffic only by analyzing several signals rather than probing the network traffic packet.

1. Introduction

The electric power [1] is the vital energy source in national product and people daily life so the importance of electric power production and transmission is beyond question. As the link between electric power production and transmission, electric power dispatching decides the effective and healthy operation of the whole electric net. With the wide deployment of flow monitoring in IP networks, the analysis of the exported flow data has become an important research area. It has been shown that flow data can be used to detect traffic anomalies, DoS attacks, and the propagation of worms [2].

As an important prerequisite for time-critical network operation tasks, many modern network devices support monitoring functions that offer monitoring data with low latency. The export of flow data is such a monitoring function that allows retrieving information about the traffic currently observe data monitoring device, which may be a router or a stand-alone network monitor.

This paper proposed a method to archive the net attack detection and identification by leveraging the XGBoost [3] machine learning method to train a prediction model.

2. Related Work

The anomaly detection is an important data analysis task which is useful for identifying the network intrusions. This paper presents a briefly analysis of four major categories of anomaly detection techniques which include classification, statistical, information theory and clustering [4].

The classification-based network anomaly detection includes four kinds of mainstream methods, support vector machine [5], Bayesian network [6], neural network [7] and other ruled-based methods respectively. The statistical anomaly detection includes mixture model, signal processing technique and principal component analysis (PCA) [8].

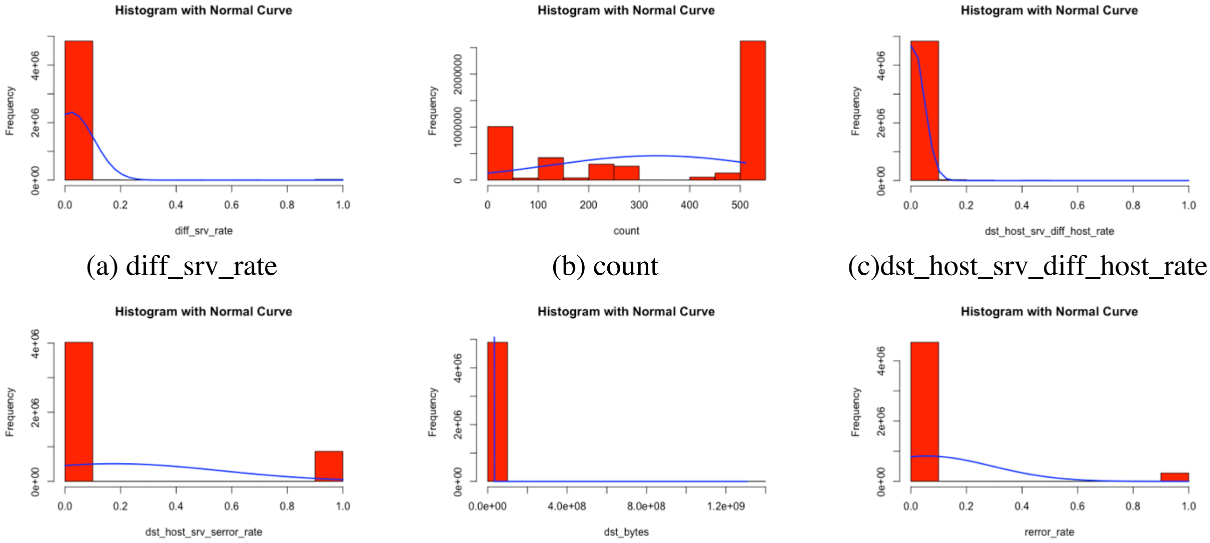
3. Net Attack Multi-Classification

This paper proposed an intrusion detector learning method to detect network intrusions protects a computer network from unauthorized users, including perhaps insiders. The intrusion detector learning task is to build a predictive model (i.e. a classifier) capable of distinguishing between abnormal connections, called intrusions or attacks, and normal connections.

3.1. Net Attack Learning Samples

To build a generally machine learning model for the net attack identification or classification, a standard net attack data sample, which includes a wide variety of intrusions simulated in an inside or a local-area network environment, is audited for the model training process. The raw training data [9] is about 743 megabytes of compressed binary TCP dump data from several weeks of network traffic, which is processed into 4,898,431 connection records. There are 41 features collected for each record, which labeled with a class or a category.

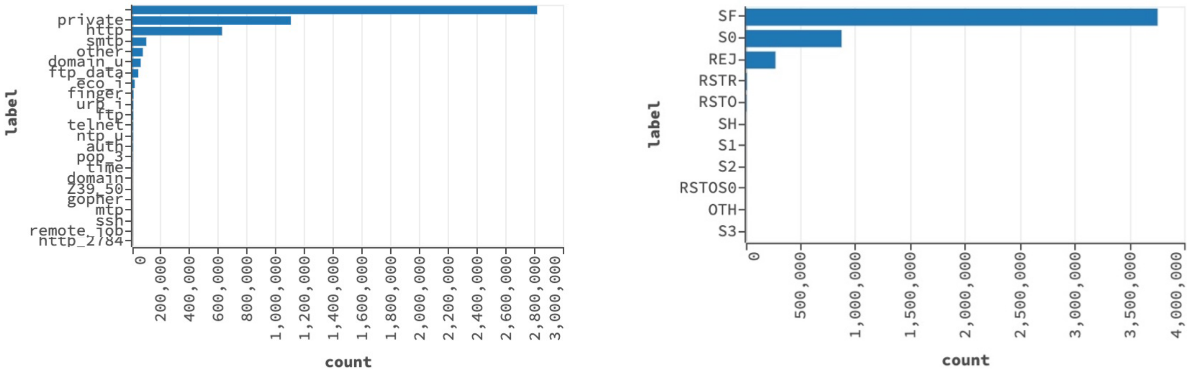
Some feature patterns are discovered through the data analysis process, which are plotted by numerical distributions, as shown in figure 1 and figure 2.



(a) diff_srv_rate (b) count (c)dst_host_srv_diff_host_rate

(d) dst_host_srv_serror_rate (e) dst_bytes (f) rerror_rate

Figure 1: Feature Numerical Distribution



(a) service (b) flag

Figure 2: Feature Categories Distribution

3.2. Model Training Process

The raw data sample is split by the ratio of 3 to 1, that is, the 75% of the data sample combines the training data, and the rest of the data sample becomes the validation data. The XGBoost method is chosen to train the machine learning model. Before feed the train data, the XGBoost model is built with some optimized parameter presetted. By turning down the learn rate from the default 0.3 to a smaller value 0.1, the overfitting situation could be avoided through the training process. Considering the large amount of the sample set, the cross-validation (aka. k-fold) phase is disabled to speed up the training program.

The total training and validating process takes almost one hour to finish, which is performed on a laptop with 8 gigabytes memory mounted and a 2.6 GHz processor driven.

3.3. Validation Metrics Result

Evaluating machine learning algorithm is an essential part of any project. The trained model may give some satisfying results when evaluated using a metric say *accuracy_score,* but may give poor results when evaluated against other metrics such as *logarithmic_loss* or any other such metric. Most of the times classification accuracy is used to measure the performance of the model, however it is not enough to truly judge the model.

The model we built with the chosen sample suffers from a skewed class problem, as is shown in figure 3. There is not really a solution to this problem, but the precision and recall measure could help for evaluating the final model. Precision describes how many of the data records, which got classified as positives, actually are illustrating positive. On the other hand, recall refers to the percentage of correctly classified positive based on the overall number of positives of the data set. The corresponding formulas are Precision = true_positives / (true_positives + false_positives) and Recall = true_positives / (true_postives + false_negatives).

The final validation metrics is presented by the confusion matrix measured by precision and recall, as shown in figure 4. For a better view of the result, a very small number of individual classes are dropped from the confusion matrix, which would cause very minimal negligible influence.

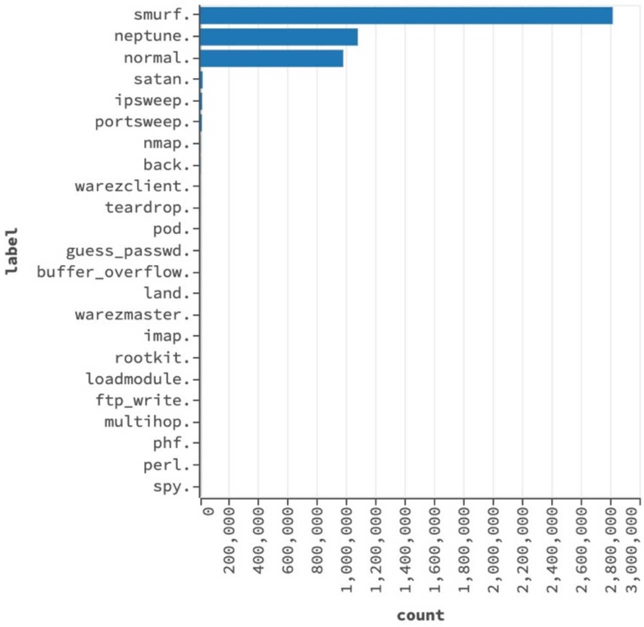
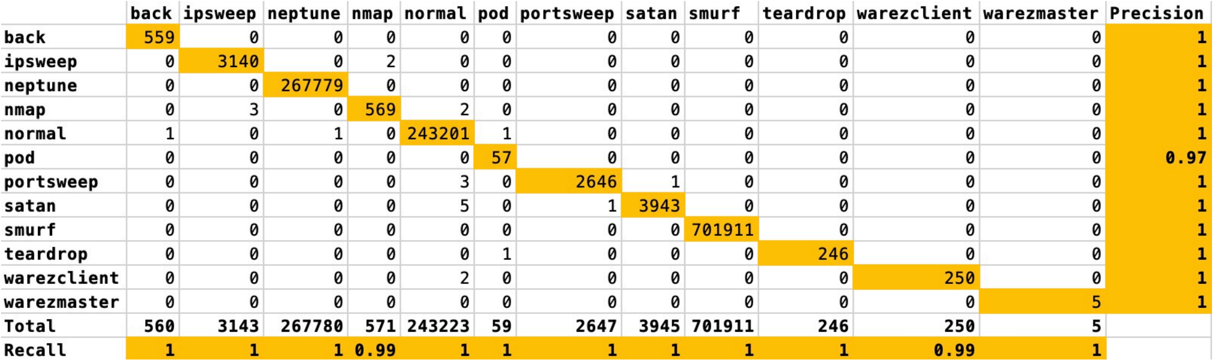


Figure 3: Skewed Class



	back	ipsweep	neptune	nmap	normal	pod	portsweep	satan	smurf	teardrop	warezclient	warezmaster	Precision
back	559	0	0	0	0	0	0	0	0	0	0	0	1
ipsweep	0	3140	0	2	0	0	0	0	0	0	0	0	1
neptune	0	0	267779	0	0	0	0	0	0	0	0	0	1
nmap	0	3	0	569	2	0	0	0	0	0	0	0	1
normal	1	0	1	0	243201	1	0	0	0	0	0	0	1
pod	0	0	0	0	0	57	0	0	0	0	0	0	0.97
portsweep	0	0	0	0	3	0	2646	1	0	0	0	0	1
satan	0	0	0	0	5	0	1	3943	0	0	0	0	1
smurf	0	0	0	0	0	0	0	0	701911	0	0	0	1
teardrop	0	0	0	0	0	1	0	0	0	246	0	0	1
warezclient	0	0	0	0	2	0	0	0	0	0	250	0	1
warezmaster	0	0	0	0	0	0	0	0	0	0	0	5	1
Total	560	3143	267780	571	243223	59	2647	3945	701911	246	250	5	
Recall	1	1	1	0.99	1	1	1	1	1	1	0.99	1	

Figure 4: Validation Metrics – Confusion Matrix

4. Conclusion

In this paper, the XGBoost machine learning method is leveraged to build a model for the net attack detection and identification based on a public net attack data sample. It turns out that the XGBoost model performs a very good result on the given sample, which could be a strong testimony to apply XGBoost method for power dispatching net attack classification.

5. Acknowledgements

This work was supported by "Research on Lightweight Active Immune Technology for Electric Power Supervisory Control System", a science and technology project of State Grid Co.,Ltd in 2019.

References

[1] Chang Y, Chen Y, Lu L, Jia M. Design of Electric Power Dispatching Management System based on Data Mining. In 2015 International conference on Applied Science and Engineering Innovation 2015 May 30. Atlantis Press.

[2] Munz G, Carle G. Real-time Analysis of Flow Data for Network Attack Detection. IEEE; pp. 100–8.

[3] Chen T, Guestrin C. Xgboost: A scalable tree boosting system. InProceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining 2016 Aug 13 (pp. 785-794). ACM.

[4] Ahmed M, Naser Mahmood A, Hu J. A survey of network anomaly detection techniques. Journal of Network and Computer Applications. 2016 Jan;60:19–31.

[5] Agarwal B, Mittal N. Hybrid approach for detection of anomaly network traffic using data mining techniques. Procedia Technology. 2012 Jan 1;6:996-1003.

[6] Friedman N, Geiger D, Goldszmidt M. Bayesian network classifiers. Machine learning. 1997 Nov 1;29(2-3):131-63.

[7] Graves A, Wayne G, Reynolds M, Harley T, Danihelka I, Grabska-Barwińska A, Colmenarejo SG, Grefenstette E, Ramalho T, Agapiou J, Badia AP. Hybrid computing using a neural network with dynamic external memory. Nature. 2016 Oct;538(7626):471.

[8] Jolliffe I. Principal component analysis. Springer Berlin Heidelberg; 2011.

[9] Stolfo J, Fan W, Lee W, Prodromidis A, Chan PK. Cost-based modeling and evaluation for data mining with application to fraud and intrusion detection. Results from the JAM Project by Salvatore. 2000:1-5.