A Human-Machine Collaborative Detection Model for Identifying Web Attacks

Yong Hu[1,2], Bo Li[1,2(✉)], Weijing Ye[2], and Guiqin Yuan[1,2]

[1] School of Computing Science and Engineering, Beihang University, Beijing, China
libo@act.buaa.edu.cn
[2] State Grid Zhejiang Electric Power Company,
Information and Telecommunication Branch, Hangzhou, China

Abstract. Machine learning plays an important part in detecting web attacks. However, it exhibits high false alarm rate due to the lacking of labeled data. Humans perform better than machines in attack recognition, while suffer from low bandwidth. In this paper, we adopt a collaborative detection model, based on machine learning augmented with human interaction to detect web attacks. We leverage human knowledge to continuously optimize the detection model and make machines smarter against fast-changing web attacks. To eliminate the bottleneck of humans, we design an selection mechanism which could recommend most suspicious anomaly behaviors for humans to correct the false decision of machines. In addition, we also define a human involvement ratio, k, to represent how much efforts that human contributes to the collaborative detection model. By tuning k, the model accuracy and human workloads could be effectively balanced. We conduct several comprehensive experiments to evaluate the effectiveness of our model using reallife datasets. The results demonstrate that our approach could significantly improve the detection accuracy compared with traditional machine learning approaches.

Keywords: Web attacks · Collaborative detection · Machine learning

1 Introduction

With the rejuvenation of Artificial Intelligence, researchers tend to use machine learning approaches to detect web attacks. Machine learning approaches could handle network traffic at a high speed without human intervention, and has been applied into some web security systems such as Web Application Firewall (WAF). Anomaly detection is a kind of unsupervised machine learning techniques and has been widely used to web attack detection. Similar to other unsupervised approaches, it also suffers from low accuracy due to the lacking of labeled data. In machine learning domain, it is also referred to as "cold-start" problem which greatly affects the feasibility of applying anomaly detection techniques into production environments.

I. Romdhani et al. (Eds.): CollaborateCom 2017, LNICST 252, pp. 109–119, 2018.
https://doi.org/10.1007/978-3-030-00916-8_11

On the contrary, humans perform much better than machines in identifying web anomalies and attacks. Current security products are still highly dependent on humans to find anomalies and create the corresponding signatures and rules so as to detect similar attacks. The advantage is high accuracy with high detection speed since only known signatures are matched during the detection phase. The bandwidth of human is limited, thus it is impossible for security experts to keep up with the emerging and continuously changing web attacks.

Naturally we adopt a machine-human collaborative detection model to improve the detection performance without sacrificing the detection speed. In this model, human intelligence is crucial to improve detection accuracy of machines. Our main contributions are summarized as follows:

- We introduce a collaborative detection model to identify web attacks from web logs. The model is a continuously updating loop. First, machines identify anomalies and ranking mechanism recommend most suspicious behaviors for human decision. The results are feedback to the model and make the model smarter.
- We carefully analyze the web log datasets and design a feature extraction method to ensure relative high detection accuracy. We design some rules to control how humans are involved, and find a suitable k value, which is used to represent the ratio of human participation in the collaborative model.
- We conduct several experiments to evaluate the effectiveness of our model. The results show that our model can effectively improve the accuracy rate in the case of limited human collaboration. The model performance and human workloads could be well balanced.

Section 3 describes the structure of the model. Section 4 describes the data analyzed by our system. Section 5 accounts for the steps of human participation in the model. Section 6 includes experimental settings, model details and results with analysis. Section 7 sums up conclusions.

2 Related Work

In this paper, our model comes from the active detection model [1] combining the human intuition with machine learning technology. We follow it and make some differences. We simplified the process of unsupervised learning algorithms. Secondly, we provide artificial intuition two opportunities to participate in the detection and pay more attention to the single web logs' information mining with proper data quantization and feature extraction features methods

In terms of machine learning and anomaly detection. [2] focused on the query parameters in the web request and proposed several mathematical models used to detect anomalies; [3] proposed an unsupervised learning method used to detection network anomalies online; [4] proposed a enhanced SVM method used to implement network intrusion detection. Statistics-based methods are primary to discovery distribution information counting on statistical techniques. For the statistical analysis of URI, G.V. [2,14] applied the Gaussian distribution and

Markov model to analyze attribute length, attribute character distribution structural inference, token finder, attribute presence or absence and attribute order.

3 Model

The schematic diagram of this collaborative detection model (CDM) is shown in Fig. 1. CDM is divided into three main stages:

1. The model classifies the data using the unsupervised learning algorithm.
2. Select minor part of the classified data and submit it to the security expert. Investigate and label it with the help of knowledge of analyst.
3. Transport the labeled data back to the model to train the supervised learning algorithm which outputs the final detection result.

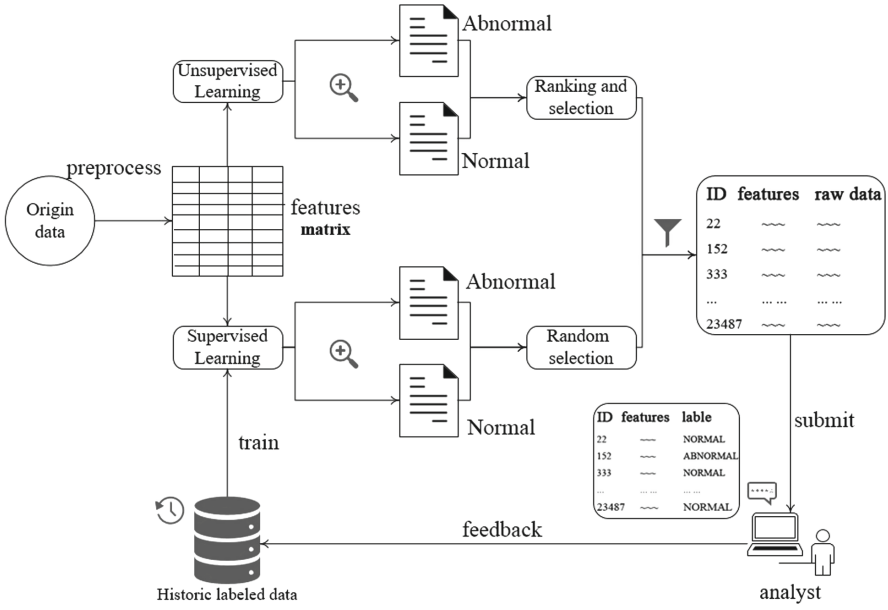

Fig. 1. The schematic diagram of this collaborative detection model

To start with CDM can identify part of the most threatening and false detection positives data, which will be presented to the security experts. Manual analysis can improve the system reliability and recognition rate, and timely detect the latest security threats. Therefore, the CMD owns both a computer's efficient computing power and some capacity the machine temporary lacks.

4 Data Preprocessing

This chapter mainly corresponds to the data preprocessing process in the model.

4.1 Data Characteristic

Our model processes web logs. In this paper, the source data provided by data center comes from the enterprise website. The data is only divided into two categories of normal and abnormal, instead of further classified attack type.

Normal. Most of the source data is normal, its data format follows the generic weblogs' format. The two key fields we adopt are shown in Table 1

Table 1. Some fields of normal data

Name	Type	Description
REQ_URI	VARCHAR(2048)	Request URI
REFERER	VARCHAR(2048)	Request source address
...	...	... ...

Abnormal. Its data format has three additional fields besides the general field listed in Table 1. At least one of them shown in Table 2 is not empty.

Table 2. Additional fields of abnormal data

Name	Type	Description
OFFLINE_ATTACK_TYPE	VARCHAR(128)	Attack type
VUL	VARCHAR(128)	Vulnerabilities
HACK_TOOL	VARCHAR(128)	Attack tool

4.2 Feature Extraction

In the previous study of peers, the REQ_URI and REFERER fields in Table 1 are considered contain many valuable information. Therefore, we decided to use these two key fields to quantize the collected weblogs as our algorithms' input. Get the union of the REQ_URI and REFERER fields as UR to remove the redundant information, and extract the features shown in Table 3. Here are detailed explanations of some fields:

ID 4 Num_token. If a URI path is shown as follow:
www.example.com/login.php?login_attempt=1&l=110
For easier analysis, pure digital and single letter are ignored. Then we get the token set of an URI string:
[www, example, com, login, php, attempt]
So the number of token is 6. Statistical analysis of tokens frequency contributes to anomaly detection.

ID 5 Num_Count_Keyword. We will list a set of key words. Each occurrence of a word in this set lead the statistic plus one.

[select, union, from, and, then, else, count, print, alter, md5, script, php, ini, config, log, mdb, passwd, /etc/passwd]

ID 6 Num_Count_Keychar. We will list a set of key character. Each occurrence of a character in this set lead the statistic plus one.

[Space , { , } , [,] , (,)]

ID 11 Relative Entropy. Related Entropy represents the degree of string confusion. It is calculated:

$$e = -\sum_{i=1}^{m} p_i \log p_i, \quad pi = \frac{n_i}{n}. \tag{1}$$

m is the number of different characters of URI, N is the total number of characters. n_i is the number of i^{th} characters whose frequency is p_i.

Table 3. Extracted features

ID	Name	Type	Description
1	Num_Digit	Integer	Number of Digit in UR String
2	Num_Letter	Integer	Number of Letter in UR string
3	Num_Punctuation	Integer	Number of Punctuation in UR string
4	Num_token	Integer	Number of token in UR string
5	Num_Count_Keyword	Integer	Number of key words count in UR string
6	Num_Count_Keychar	Integer	Number of key character count in UR string
7	Length_UR	Integer	Length of UR
8	Length_Max_Par	Integer	The max length of parameter
9	Length_Min_Par	Integer	The min length of parameter
10	Depth	Integer	The UR path depth
11	Relative Entropy	Float	Relative entropy of UR string

5 Detection Process

5.1 Unsupervised Learning Process

Its main function is coarse particle size anomaly detection, by which the model can select minor data for artificial analysis. This provide the first participatory position of human. Using k-means algorithm, the web logs are divided into two categories. It is impossible for analyst to observe all the data every day, so how to select the data effectively is of great importance. We adopt the key parameter k of the model to represent the human selection ratio.

The normal clustering center point $c^{(n)}$ and anomaly clustering center point $c^{(a)}$ are obtained. And the rank of each web log is calculated based on the vector distance between the single feature vector from its related center point.

Assume that the i^{th} data is marked as normal whose features vector is ${v_i}^{(n)}$, its rank ${r_i}^{(n)}$ is calculated:

$$ {r_i}^{(n)} = \|{v_i}^{(n)} - c^{(n)}\|. \tag{2} $$

Assume that the j^{th} data is marked as abnormal whose features vector is ${v_j}^{(a)}$, its rank ${r_j}^{(a)}$ is calculated:

$$ {r_j}^{(a)} = \frac{1}{\|{v_j}^{(a)} - c^{(a)}\|}. \tag{3} $$

We pick up part of the data from raw data that is marked as normal and abnormal. The choice principle is based on rank values but for different reasons. Assume that the input data size is z, there are n entries marked as normal, with a entries marked as abnormal, our filter ratio is $k(0 < k < 1)$.

For the data marked as abnormal, the model needs to select the top $a \cdot k$ items that have the shortest distance from the abnormal center point. The closer to the center, the greater the threat it may be. Security experts should analysis it carefully. For the data marked as normal, the system needs to select the top $a \cdot k$ items which have the longest distance from the abnormal center point. This part of the data has the greatest probability of mis-detection and some new types of attacks may be missed. The selected data will be analyzed by the security personnel, marked with the corresponding label and added to historical label data set, which is used to train a supervised learning model.

5.2 Supervised Learning Process

In this collaborative detection model, the final detection results of the input raw data are given by this step. In addition to that, a second collaborative detection will be conducted after a supervised learning analysis. Using random forest algorithm, the system can reclassify the raw data and get the final results. The analysts will also conduct a review of supervised learning's result.

Assume that the input data size is z, there are n entries marked as normal, with a entries marked as abnormal, our filter ratio is $k(0 < k < 1)$. This time we randomly selected $a \cdot k$ items from each kind of result (total $2 \cdot a \cdot k$ items) and submit them to the experts to analysis again. According to such a random method, the model can rule out the effects of some accidental factors.

An Accurate Description of k. The humans workload W means how many web logs human need to analyse every day. It is determined by k and the quantity of abnormal results: $N_{a^{(u)}} and N_{a^{(s)}}$ that come from unsupervised learning module and the supervised learning module. Though k is a fixed value, the system can still dynamically determine the actual workload based on the traffic size and threatening situation.

$$ W = 2 \cdot N_{a^{(u)}} \cdot k + 2 \cdot N_{a^{(s)}} \cdot k. \tag{4} $$

6 Experiment

6.1 Experiment Settings

We test total 810,000 normal and 90,000 abnormal web logs. It will be divided into 30 groups, each group has 3,000 items of abnormal data and 27,000 items of normal data, its ratio is 1:9. Each set of data is considered daily web traffic. Our experiment is divided into two parts:

Model Verification. To start with, we only use k-means algorithm to detect the daily data. Then we use CDM to analyze the same data set to verify the feasibility and effectiveness of the model. It is worth noting that the attributes of this set of data (abnormal or normal) are known. The real label of the data is used to simulate the process of experts' detection and to test the validity of the experimental results.

Example of t^{th} day:

1. Preprocess the raw data. Get the features matrix F_t
2. Use the unsupervised model U_t and matrix F_t to mark the input data Get the normal data and abnormal data table with rank score: N_t, A_t. Based on the ratio k, a set of data P_t are selected from N_t and A_t
3. Use the supervised model S_t and matrix F_t to mark the input data. Get the labeled data R_t. Based on the ratio k, a set of data Q_t are selected from R_t
4. Get the union of P_t and Q_t: D_t. The security personnel analysis the data set D_t. Generates a set of data with labels L_t
5. Add L_t to the historical database Z_{t-1}. to get the new database Z_t. Based on the data set Z_t, the new supervised learning model S_{t+1} is trained
6. The final results are R_t

Suitable Selection of *k*. After verifying the CDM model, we repeatedly modify the k to find a reasonable value, expecting to get a reasonable k value, while ensuring the accuracy of the model and maintaining a reasonable manual cooperation workload.

6.2 Results Related to Model Verification

We combine the results of pure unsupervised learning algorithm (UL) and CDM with the form shown in Fig. 2.

For UL: it is not difficult to notice that TPr has always fluctuated between 55% and 70%. The FPr amplitude is greater with big variance value.

Here are such a few points from the results of CDM ($k = 0.1$):

- TPr is maintained at more than 86.5% level, indicating that the model owns excellent exception detection capabilities.
- FPr is maintained at a very low level, up to a maximum of 0.07 except the 2^{nd} day, which indicates the reliability of this model.

– TPr has a spiral upward trend with the development of time. Ultimately, it is maintained at 93% level. While the FPr spiral drops and eventually remains at an ideal level.

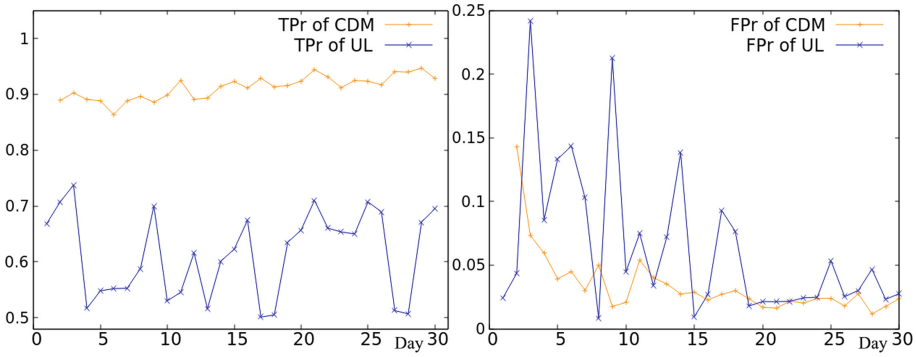

Fig. 2. Comparison of two methods' trends

6.3 Results Related to Selection of k

After setting different k values $(0.06, 0.08, 0.09, 0.10, 0.11, 0.12, 0.14, 0.16)$ for the model, we calculated the mathematical mean of the corresponding values of TPr, FPr, and human collaborative detection workload, as shown in Table 4 and Fig. 3. There are several intuitive results and our observation:

– When k ranges from 0.06 to 0.08, TPr and FPr have several significant changes. This is related to the working principle of the model and the random forest algorithm. Due to the small value of k, the lack of sufficient training samples to separate the two types of data.
– When k is greater than 0.09, TPr is on the rise, but there is still fluctuation. The 2^{nd} round of human participation introduced in Sect. 5 lead to it. In this step, the data is randomly selected. Even if the same k value, each time there will be subtle differences.
– The change of FPr is not obvious while $k > 0.09$.
– The workload increases as k increases. This conforms to our algorithmic design.

We consider that k value around 0.12, able to meet the needs of this model performance. The human collaborative workload data accounts for only 3.3% of the overall test data.

Table 4. Experimental results of different k values for cooperative detection model. The 'Items' column means the number of the weblogs people need to analyze

Day	TPr	FPr	Items	TPr	FPr	Items	TPr	FPr	Items	TPr	FPr	Items
k	0.06			0.09			0.12			0.16		
1	-	-	244	-	-	368	-	-	490	-	-	654
2	0.696	0.058	552	0.853	0.129	904	0.846	0.102	1188	0.855	0.112	1578
3	0.824	0.038	1326	0.897	0.086	2001	0.915	0.099	2678	0.892	0.097	3537
6	0.834	0.026	959	0.862	0.048	1434	0.876	0.047	1907	0.893	0.068	2544
9	0.842	0.010	1222	0.886	0.018	1852	0.898	0.020	2451	0.904	0.019	3230
12	0.847	0.020	542	0.893	0.043	837	0.902	0.041	1119	0.901	0.039	1476
15	0.894	0.016	482	0.934	0.029	746	0.928	0.039	985	0.925	0.032	1291
18	0.884	0.013	739	0.913	0.031	1107	0.922	0.038	1456	0.917	0.023	1928
21	0.920	0.008	556	0.944	0.014	840	0.956	0.014	1117	0.949	0.013	1477
24	0.886	0.010	536	0.928	0.026	818	0.927	0.019	1079	0.927	0.016	1419
27	0.910	0.015	597	0.938	0.025	905	0.943	0.024	1192	0.945	0.023	1567
30	0.903	0.014	573	0.930	0.023	875	0.935	0.023	1156	0.934	0.022	1529
Avg	0.867	0.016	687	0.912	0.034	1047	0.917	0.035	1390	0.918	0.034	1838

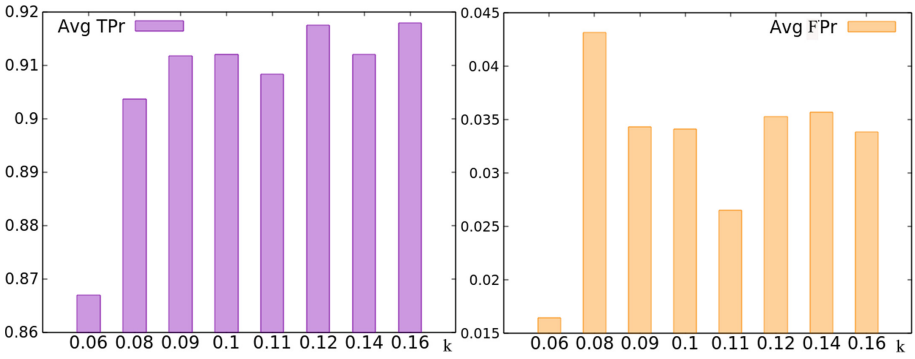

Fig. 3. The average TPr, FPr and workload of different k values

7 Conclusion

Our supporting feature extraction method is useful. Whether in the unsupervised learning model or in the collaborative detection model, this feature extraction method can extract the information in the raw data and detect some of the anomalies within the allowable range of FPr. This collaborative detection model can greatly improve the accuracy of detecting abnormal data with limited human workload. And with the accumulation of data, performance continues to increase and reach a stable state. In this model, there is an appropriate k value

around 0.12 while ensuring the accuracy of the model and maintain a reasonable collaborative workload.

Both to optimize the machine learning algorithm and to propose deeper feature extraction methods are good ideas to improve this detection model. These are part of our future work.

Acknowledgement. The authors gratefully acknowledge the anonymous reviewers for their helpful suggestions. This work is supported by supported by China 863 program (No. 2015AA01A202) and project of Telecommunication Company of State Grid Zhejiang Electric Power Company (5211XT16000A).

References

1. Veeramachaneni, K., Arnaldo, I., Korrapati, V., Bassias, C., Li, K.: AI^2: training a big data machine to defend. In: 2016 IEEE 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), pp. 49–54. IEEE (2016)
2. Kruegel, C., Vigna, G.: Anomaly detection of web-based attacks. In: Proceedings of the 10th ACM Conference on Computer and Communications Security, pp. 251–261. ACM (2003)
3. Lu, W., Traore, I.: A new unsupervised anomaly detection framework for detecting network attacks in real-time. In: Desmedt, Y.G., Wang, H., Mu, Y., Li, Y. (eds.) CANS 2005. LNCS, vol. 3810, pp. 96–109. Springer, Heidelberg (2005). https://doi.org/10.1007/11599371_9
4. Shon, T., Moon, J., Waterman, M.S.: A hybrid machine learning approach to network anomaly detection. Inf. Sci. **177**(18), 3799–3821 (2007)
5. Lewis, D.D., Catlett, J.: Heterogeneous uncertainty sampling for supervised learning. In: Proceedings of the Eleventh International Conference on Machine Learning, pp. 148–156 (1994)
6. Chen, Y., Hwang, K., Ku, W.S.: Collaborative detection of DDoS attacks over multiple network domains. IEEE Trans. Parallel Distrib. Syst. **18**(12), 1649–1662 (2007)
7. Chen, Y., Hwang, K.: Collaborative detection and filtering of shrew DDoS attacks using spectral analysis. J. Parallel Distrib. Comput. **66**(9), 1137–1151 (2006)
8. Yao, D., Yin, M., Luo, J., Zhang, S.: Network anomaly detection using random forests and entropy of traffic features. In: 2012 Fourth International Conference on Multimedia Information Networking and Security, pp. 926–929. IEEE (2012)
9. Zhang, J., Chen, C., Xiang, Y., Zhou, W.: Robust network traffic identification with unknown applications. In: Proceedings of the 8th ACM SIGSAC symposium on Information, Computer and Communications Security, pp. 405–414. ACM (2012)
10. Nadiammai, G., Hemalatha, M.: Effective approach toward intrusion detection system using data mining techniques. Egypt. Inform. Journal. **15**(1), 37–57 (2014)
11. Hodge, V., Austin, J.: A survey of outlier detection methodologies. Artif. Intell. Rev. **22**(2), 85–126 (2004)
12. Bhuyan, M.H., Bhattacharyya, D.K., Kalita, J.K.: Network anomaly detection: methods, systems and tools. IEEE Commun. Surv. Tutorials **16**(1), 303–336 (2014)

13. Ektefa, M., Memar, S., Sidi, F., Affendey, L.S.: Intrusion detection using data mining techniques. In: 2010 International Conference on Information Retrieval and Knowledge Management (CAMP), pp. 200–203. IEEE (2010)
14. Threepak, T., Watcharapupong, A., Assent, I. Web attack detection using entropy-based analysis. In: The International Conference on Information Networking 2014 (ICOIN 2014), pp. 244–247. IEEE (2014)