Empirical Analysis And Validation Of Security Alerts

-- (Two-level alert filtering and one final alert validation)

J. Jeeshitha

Department of Computer Science and Engineering

Anurag Group of Institutions, Hyderabad, Telangana, India

jeeshithacse@cvsr.ac.in

Kavya Koppoju, G. K. Rohit, Aerva Manish Reddy

Anurag Group of Institutions, Hyderabad, Telangana, India

16h61a05e4@cvsr.ac.in, 16h61a05d7@cvsr.ac.in, 16h61a05c1@cvsr.ac.in

Abstract – Empirical Analysis is an evidence based approach that includes stratification of data over studied risk factors. In any security based system evidences are very important in order to make necessary improvements or at least for further enhancement. The developed security platform can be deployed with any existing system like banking, payment wallet, online shopping etc., but most importantly any means of payment. In here, we have deployed it along online shopping payment system to avoid data loss. The pre-filtering before validation will help reduce the burden on Data base administrator.

I. INTRODUCTION

Recent security incidents and data breach episodes that caused credit/debit card credentials leak highlights strong technical advances achieved by the attackers. System administrators cope with security incidents through a variety of monitors, such as intrusion detection systems, event logs, security information and event management systems. Monitors generate large volumes of alerts that overwhelm the operations team and make forensics time-consuming.

Filtering is a consolidated technique to reduce the amount of alerts. In this project we are going to consider a dataset from previously done project (Credit Card Fraud Detection using Random Forest and Majority Voting) and train the dataset using Blacklist Filtering Technique. The project includes a module called "auditor" to manual filter the alerts prior reaching the admin. One final validation from the admin can proceed the transaction.

Our work falls in the category of unsupervised problems, strongly assuming that alerts are repetitive and monotonous. The dataset that we are using also has the same features i.e, monotony and repetitiveness, measured by related work in the security area. Using filter on dataset will reduce repetitiveness and by manual filtering 98% of the data loss can be controlled.

II. RELATED WORKS

Due to its cost efficiency, the controller area network (CAN) is still the most wide-spread in-vehicle bus, and the numerous reported attacks demonstrate the urgency in designing new security solutions for CAN. In this paper, we propose an intrusion detection mechanism that takes advantage of Bloom filtering[1] to test frame periodicity based on message identifiers and parts of the data-field which facilitates detection of potential replay or modification attacks. This proves to be an effective approach since most of the traffic from in-vehicle buses is cyclic in nature and the format of the data-field is fixed due to rigid signal allocation. Bloom filters provide an efficient time-memory tradeoff which is beneficial for the constrained resources of automotive grade controllers. We test the correctness of our approach and obtain good results on an industry-standard CANoe-based simulation for a J1939 commercial-vehicle bus and also on CAN with flexible data-rate traces obtained from a real-world high-end vehicle. The proposed filtering mechanism is straightforward to adapt for any other time-triggered in-vehicle bus, e.g., Flex Ray, since it is built on time-driven characteristics.

Redundant and irrelevant features in data have caused a long-term problem in network traffic classification. These features not only slow down the process of classification but also prevent a classifier from making accurate decisions, especially when coping with big data. In this paper, we propose a mutual information based algorithm[2] that analytically selects the optimal feature for classification. This mutual information based feature selection algorithm can handle linearly and nonlinearly dependent data features. Its effectiveness is evaluated in the cases of network intrusion detection. An Intrusion Detection System (IDS), named Least Square Support Vector Machine based IDS (LSSVM-IDS), is built using the features selected by our proposed feature selection algorithm. The performance of LSSVM-IDS is evaluated using three intrusion detection evaluation datasets, namely KDD Cup 99, NSL-KDD and Kyoto 2006+ dataset. The evaluation results show that our feature selection algorithm contributes more critical features for LSSVM-IDS to achieve better accuracy and lower computational cost compared with the state-of-the-art methods.

Security Operation Centers rely on data triage[3] to identify the true "signals" from a large volume of noisy alerts and "connect the dots" to answer certain higher-level questions about the attack activities. This work aims to automatically generate data triage automatons directly from cyber security analysts' operation traces. Existing methods for generating data triage automatons, including Security Information and Event Management systems (SIEMs), require event correlation rules to be generated by dedicated manual effort from expert analysts. To save analysts' workloads, we propose to "mine" data triage rules out of cyber security analysts' operation traces and to use these rules to construct data triage automatons. Our approach may make the cost (of data triage automaton generation) orders of magnitudes smaller. We have designed and implemented the new system and evaluated it through a human-in-the-loop case study. The case study shows that our system can use the analysts' operation traces as input and automatically generate a corresponding state machine for data triage. The operation traces were collected in our previous lab experiment. 29 professional cyber security analysts were recruited to analyze a set of IDS alerts and firewall logs. False positive and false negative rates were calculated to evaluate the performance of the data triage state machine by comparing with the ground truth.

III. PROPOSED SYSTEM

The proposed system is composed of three phases:

a. Training the dataset using filter.

b. Including an 'Auditor' for manual-filtering the alerts.

c. Validation of alerts by 'Admin'.

a. Training the dataset using filter

Dataset from Credit Card Fraud Detection using Random Forest and Majority Voting is taken and data replications are added for working with and testing this platform. This dataset is then trained using Blacklisting Algorithm[1].

[1] Its a basic access control mechanism that allows through all elements except those explicitly mentioned.

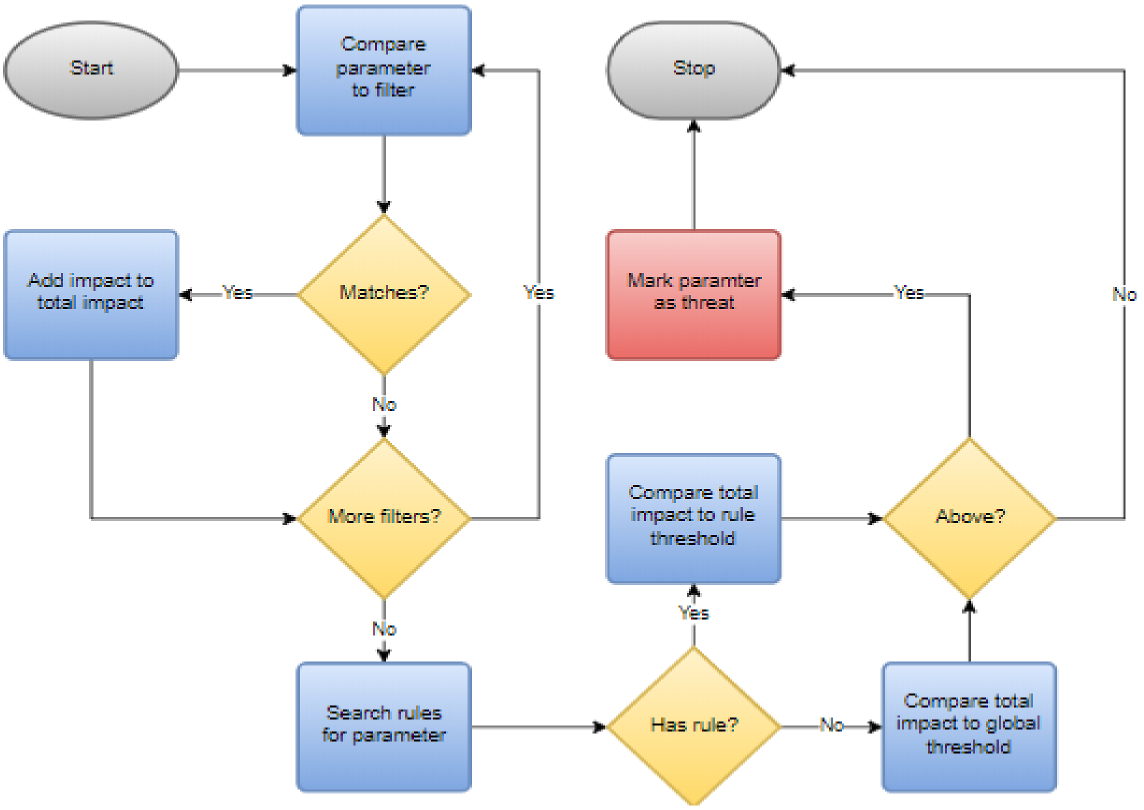
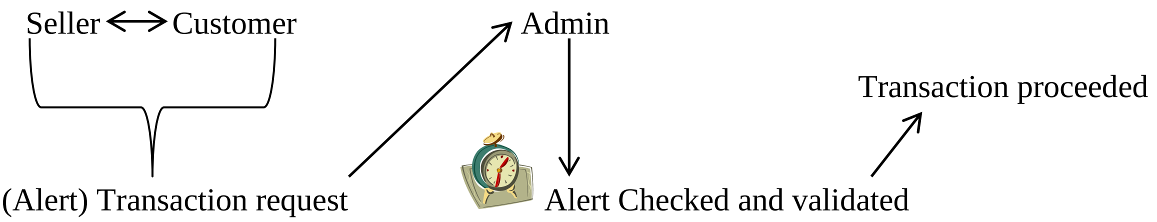

Start
Compare parameter to filter
Stop
Add impact to total impact
Yes
Matches?
Yes
Mark paramter as threat
Yes
No
No
More filters?
Compare total impact to rule threshold
Above?
Yes
No
Search rules for parameter
Has rule?
No
Compare total impact to global threshold

Figure 1. Illustrates how Blacklisting Algorithm works

b. Including an 'Auditor' for manual-filtering the alerts

Here we have deployed this security platform in ecommerce shopping - online payment system. Normally the modules included are

1. Seller
2. Customer
3. Admin

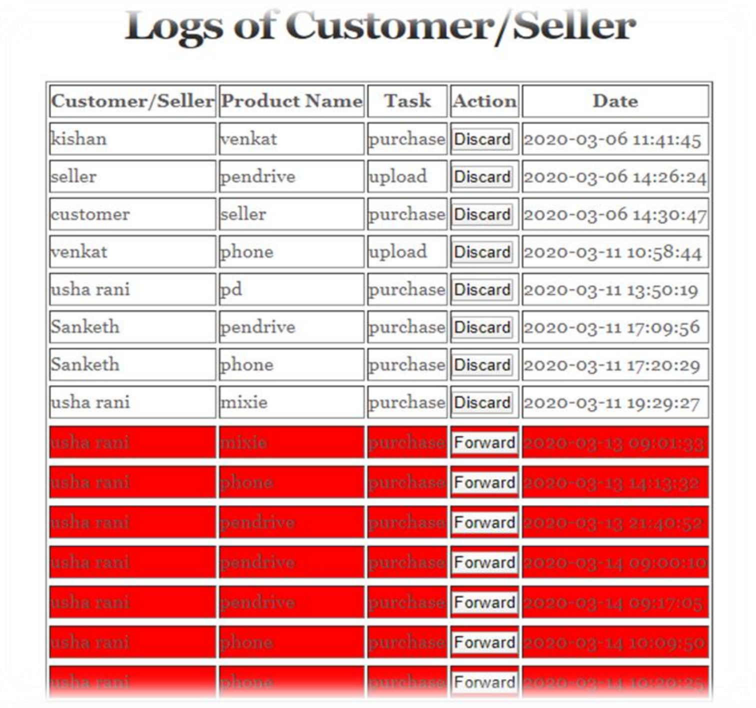


But in here, to avoid the burden on admin, an additional user called as 'Auditor' is given a post. Auditor will be having more privileges than admin. Auditor can keep a track on sellers, customers and most importantly the first copy of transaction log will be received by auditor.

Each transaction log is accompanied by a button (discard/forward) for the auditor to have a control over the transaction.

During re-verification if the auditor finds the transaction to be escaped fraud alert it will be manually discarded by him.

Once discarded by auditor the transaction doesn't reach the admin.



Logs of Customer/Seller

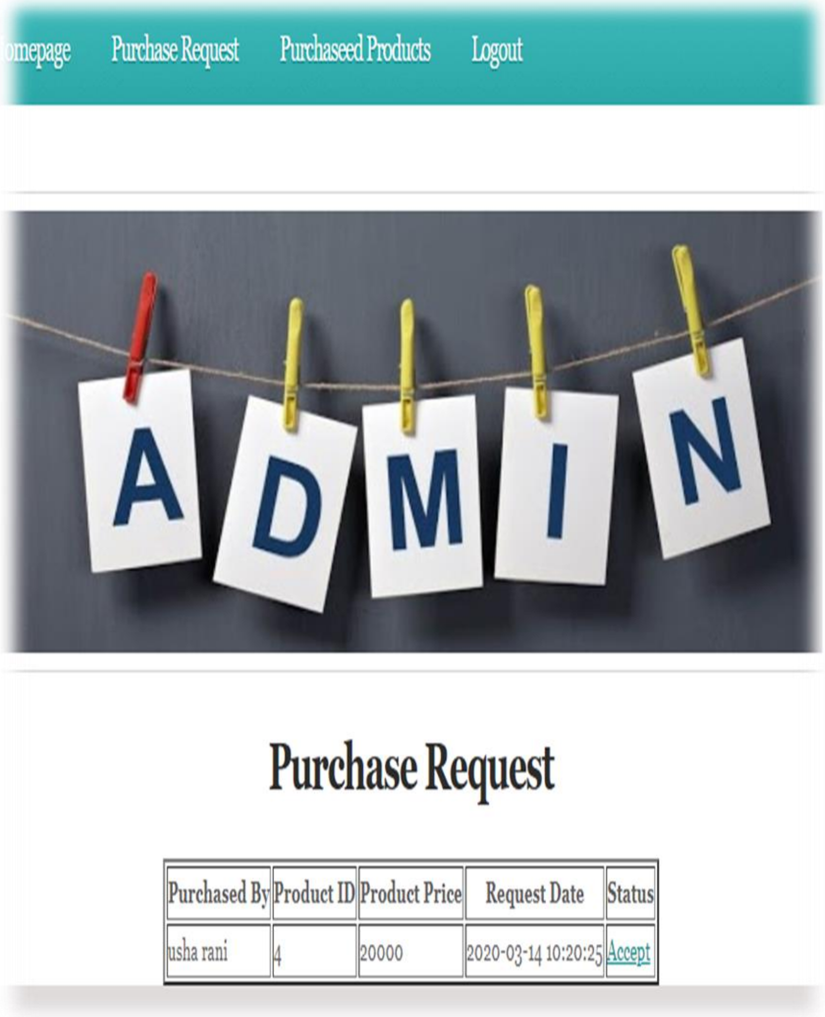
Customer/Seller	Product Name	Task	Action	Date
kishan	venkat	purchase	Discard	2020-03-06 11:41:45
seller	pendrive	upload	Discard	2020-03-06 14:26:24
customer	seller	purchase	Discard	2020-03-06 14:30:47
venkat	phone	upload	Discard	2020-03-11 10:58:44
usha rani	pd	purchase	Discard	2020-03-11 13:50:19
Sanketh	pendrive	purchase	Discard	2020-03-11 17:09:56
Sanketh	phone	purchase	Discard	2020-03-11 17:20:29
usha rani	mixie	purchase	Discard	2020-03-11 19:29:27
usha rani	mixie	purchase	Forward	2020-03-13 09:01:33
usha rani	phone	purchase	Forward	2020-03-13 14:13:32
usha rani	pendrive	purchase	Forward	2020-03-13 21:40:52
usha rani	pendrive	purchase	Forward	2020-03-14 09:00:10
usha rani	pendrive	purchase	Forward	2020-03-14 09:17:05
usha rani	phone	purchase	Forward	2020-03-14 10:09:50
usha rani	phone	purchase	Forward	2020-03-14 10:20:25

Figure 2. Illustrates how an 'Auditor' manually filters/discards alerts

c. Validation of alerts by 'Admin'

The transaction logs or alerts that are forwarded from Auditor to 'Admin' are called Purchase Requests.

The 'Admin' provides one final validation to complete transaction.



Purchase Request

Purchased By	Product ID	Product Price	Request Date	Status
usha rani	4	20000	2020-03-14 10:20:25	Accept

Figure 3. Illustrates how an 'Admin' validates alerts

IV. RESULTS

Applying Blacklisting Algorithm on dataset reduced the monotony and repetitiveness. This made the dataset more concurrently conglomerate that helped in achieving sharp and effective outcomes. 60% of the problem is solved in data level itself and manual filtering improves it even more.

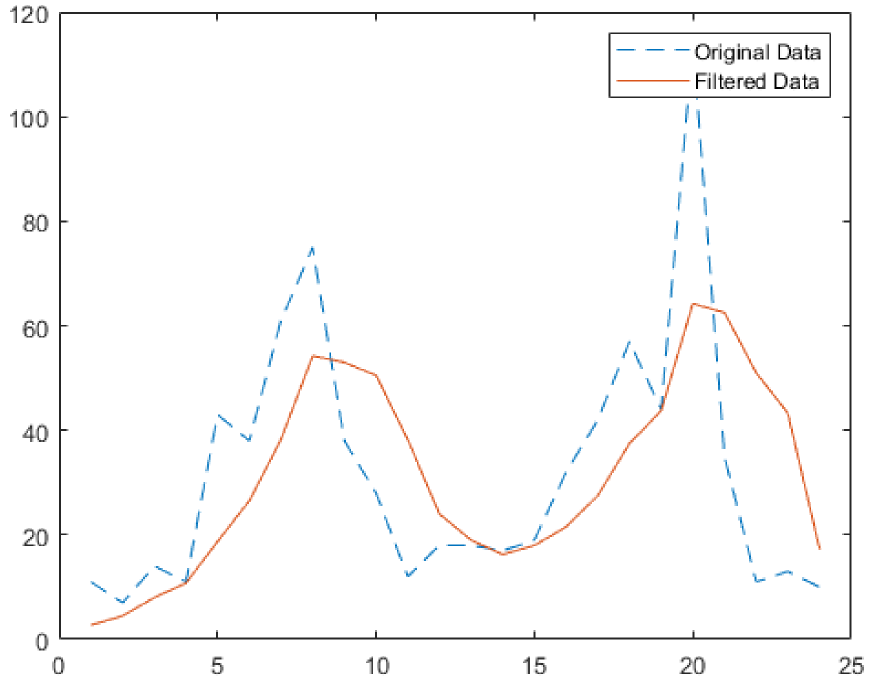


Figure 4. Illustrates Original Data vs. Filtered Data after applying Blacklisting on dataset

V. CONCLUSION

This system reduces consumption of time and provides effective results. On the other hand it helps to know the filters and manual filtering more closely with an ease. This study should be extremely helpful to security practitioners. Our project highlights the challenges underlying the analysis of security datasets and provides measurements on the effectiveness of different filters in a real world setting.

VI. REFERENCES

[1] Bogdan Groza and Pal-Stefan Murvay. Efficient Intrusion Detection with Bloom Filtering in Controller Area Network, IEEE, 2018

[2] Mohammed A. Ambusaidi, Xiangjian He, Priyadarsi Nanda and Zhiyuan Tan. Building an Intrusion Detection System using a Filter Based Feature Selection Algorithm, IEEE, 2016

[3] C. Zhong, J. Yen, P. Liu and R. F Erbacher. Automate cybersecurity data triage by leveraging human analysts’ cognitive process. In International Conference on Intelligent Data and Security (IDS), IEEE, 2017.