A User-Centric Machine Learning Framework for Cyber Security Operations Center

CREATIVE AND INNOVATIVE PROJECT REPORT

Submitted by

G. Santhosh(2016103057)

S. Ben Stewart(2016103513)

P. Udaykumar(2016103622)

College of Engineering, Guindy

ANNA UNIVERSITY: CHENNAI 600 025

AUGUST 2019

ACKNOWLEDGEMENT

This project would not have been possible if not for the tireless efforts by the members of this team in all the divisions in which they were asked to work on. Right from the project ideation till the implementation and the documentation and presentation part. We would also thank our mentors and friends who helped us when the project hit a few roadblocks.

ABSTRACT

To assure cyber security of an enterprise, typically SIEM (Security Information and Event Management) system is in place to normalize security events from different preventive technologies and flag alerts. Analysts in the security operations center (SOC) investigate the alerts to decide if it is truly malicious or not. However, generally the number of alerts is overwhelming capacity to handle all alerts. Because of this, potential malicious attacks and compromised hosts may be missed. Machine learning is a viable approach to reduce the false positive rate and improve the productivity of SOC analysts. In this paper, we develop a user-centric machine learning framework for the cyber security operation center in real enterprise environment. We discuss the typical data sources in SOC, their workflow, and how to leverage and process these data sets to build an effective machine learning system. We use the system using the key repository of information regarding the vulnerabilities that allow intruders to breach computer networks is the National Vulnerability Database (NVD). NVD is a product of the U.S. National Institute of Standards and Technology's (NIST) Computer Security Division and is also sponsored by the U.S. Department of Homeland Security's Computer Emergency Readiness Team (US-CERT). We are implementing the below steps from data massaging, label creation, feature engineering, machine learning algorithm selection, model performance evaluations, to risk score generation.

The above implementation would help other teams with only knowledge of machine learning to get a better understanding of the domain of cyber security and the challenges it provides with the requirement of high accuracy models though the dataset is highly biased. It also helps the teams on the other side of the spectrum who are from cyber security to get to understand how machine learning models can be used to the greater benefit.

TABLE OF CONTENTS

LIST OF FIGURES

1 PROBLEM STATEMENT

To develop a generic model for the cyber security domain because all the machine learning models in this domain have been abstracted as per industry regulations.

2 PROBLEM DESCRIPTION

The cyber security domain has a few implementation for machine learning to be used but they have been abstracted because of the industry regulations. Thus here we try to develop a generic model that works for all datasets in the cyber security domain. The data sets of NVD will be used and the models developed and tested against the CVSS standard .

3 LITERARY SURVEY

Since the domain we have chosen is cyber security that is implemented in the corporate sector all the research work has just the basic ideation and the implementation has been abstracted out due to the confidentiality that is required. The gist that they provide is to get the Lagrangian model to remove the bias in the datasets. An IEEE paper on the same domain was taken as the base. The changes we have intended to implement are to make the algorithm be able to work on any kind of biased data sets in the cyber security domain.

4 INTRODUCTION

The datasets from the industry were hard to find and so we used the NVD provided by the U.S. Department of Homeland Security's Computer Emergency Readiness Team (US-CERT). This had to be used to train a k-means model that is generic enough to support additional parameters. And finally the model would be tested on the golden standard of the Common Vulnerability Scoring System (CVSS) v3.0 standards.

5 WORKFLOW

We will be using the NVD dataset as the base and create the labels for the records so that they could be used for the training process. Next, it would be necessary to get the most important features that would be required to find the anomalies with greater accuracy. Finally, we get to choose the best model required for the use case at hand. This model was tested against the Common Vulnerability Scoring System (CVSS) v3.0 standards as benchmark.

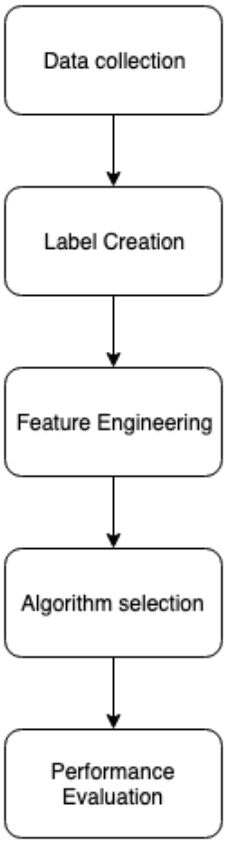

Figure 5.1 General outline of workflow

6 SYSTEM ARCHITECTURE

The below figure gives an overall idea of how the components work together with the pipeline complete. Let us look into detail into all the modules.

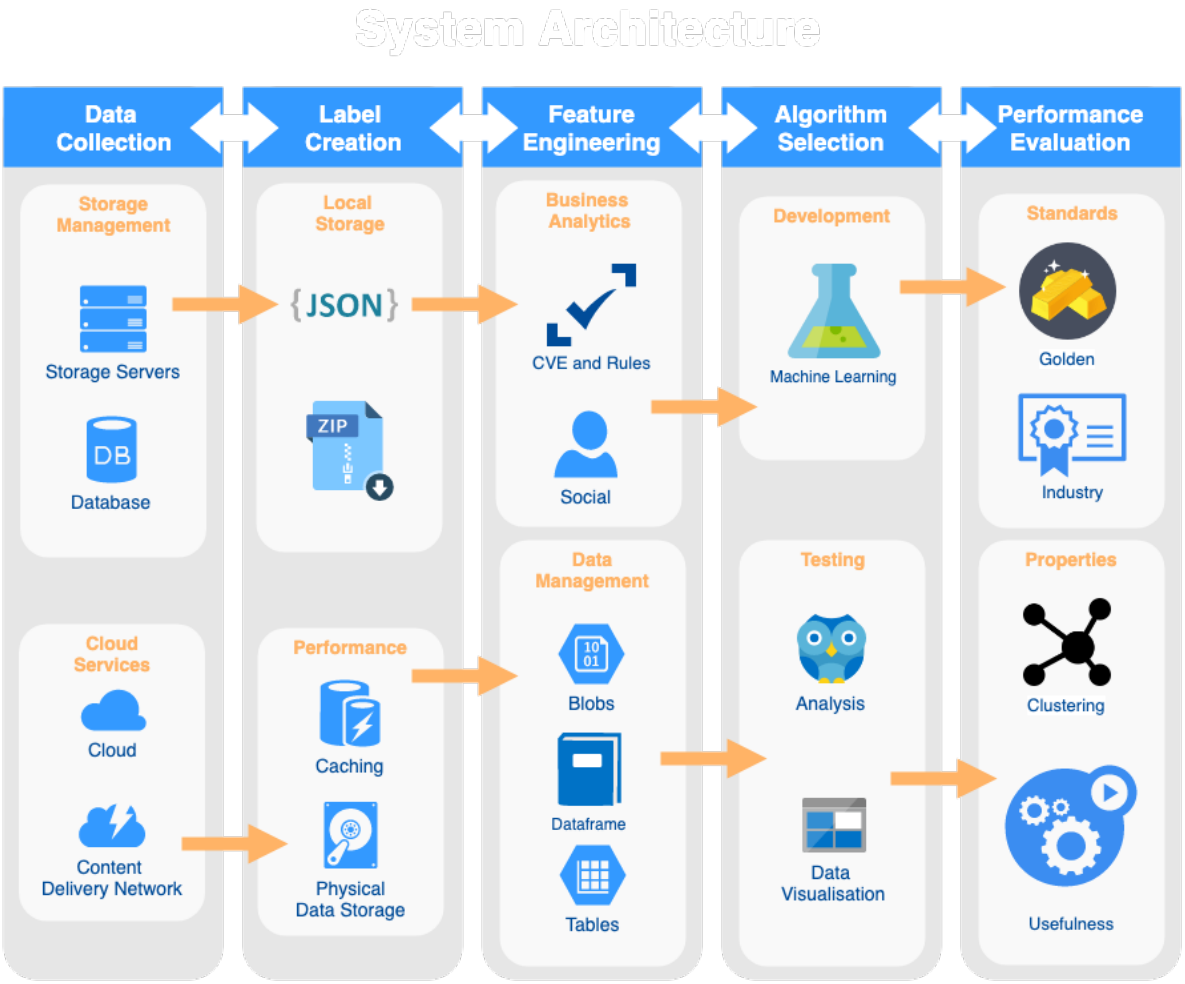

Figure 6.1 Detailed System Architecture

6.1 DATA COLLECTION

The data set provided by the National Vulnerability Database maintained by the U.S. Department of Homeland Security's Computer Emergency Readiness Team (US-CERT) has been used. We used python to build all the modules. These data sets were provided at different endpoints by the organisation for the CDN to reduce the server load. Packages in python were used to hit the server and concatenate the data into a JSON file.

6.2 LABEL CREATION

The data was accumulated as JSON files in accordance to the year of data creation. And as the file size was a bit large we zipped files for later use and loaded them into our local hard disks. The JSON parser module in Python used a custom caching mechanism to provide faster access to the values of the required fields. This component was provided by Pandas and so we stuck to using the same for developing our machine learning models as well.

6.3 FEATURE ENGINEERING

The models had been evaluated based on the Common Vulnerability Scoring System (CVSS) v3.0 standards which is commonly followed in the cyber security domain. But to make the model more generic we also introduced a few parameters for the social factors that would depend on the use case domain. All the input was stored as blobs within dataframes, which in turn were retrieved from the tables that pandas provides.

6.4 ALGORITHM SELECTION

The models had to be developed on the Lagrangian bias removed data sets and had work in all domains like spam calls, phishing emails and inappropriate conversations in social media. As a result they had to be tested against the same CVSS standard. As the output is scalar as with the CVSS standard we could go with the k-means models as they would be efficient and generic enough.

6.5 PERFORMANCE EVALUATION

To test the models the golden standard of Common Vulnerability Scoring System (CVSS) v3.0 standards was used. This gives a score from 0 to 10 based on the ease to exploit the vulnerability and the industry impact it will be having. Thus the above will act as the gold standard. We will also check the information gain from the clustering of the data points and try to arrive at conclusions.

REFERENCES

1. A user-centric machine learning framework for cyber security operations center https://ieeexplore.ieee.org/document/8004902
2. Downloading and unzipping a .zip file without writing to disk https://stackoverflow.com/questions/5710867/downloading-and-unzipping-a-zip-file-without-writing-to-disk
3. Parsing JSON Dataset with Pandas https://www.geeksforgeeks.org/pandas-parsing-json-dataset