**HIGH LEVEL DESIGN(HLD)**

**Network Security With Machine Learning**

**ABHIJIT SEN**

**iNeuron**

Contents:

Document Version Control………………………………………………………………………………….. 3

Abstract……………………………………………………………………………………………………………… 4

1. Introduction……………………………………………………………………………………………… 4

1.1 Why this High-Level Design Document?..................................................... 4

1.2 Scope………………………………………………………………………………………………………. 4

1.3 Definitions………………………………………………………………………………………………… 5

2. General Description………………………………………………………………………………….. 5

2.1 Product Perspective…………………………………………………………………………………. 5

2.2 Problem Statement………………………………………………………………………………….. 5

2.3 Proposed solution…………………………………………………………………………………….. 5

2.4 Technical Requirements…………………………………………………………………………… 5

2.5 Data Requirements………………………………………………………………………………….. 6

2.6 Tools Used…………………………………………………………………………………………….... 6

2.7 Constraints………………………………………………………………………………………………. 7

2.8 Assumptions……………………………………………………………………………………………. 7

3. Design Details………………………………………………………………………………………….. 7

3.1 Process Flow………………………………………………………………………………………...... 7

3.2 Event Log………………………………………………………………………………………………… 7

4. Performance……………………………………………………………………………………………. 7

4.1 Reusability…………………………………………………………………………………………….... 8

4.2 Application Compatibility………………………………………………………………………… 8

4.3 Deployment…………………………………………………………………………………………….. 8

5. Conclusion……………………………………………………………………………………………….. 8

**Document Version Control**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date Issued** | **Version** | **Description** | **Author** |
|  | **V1.0** | **Initial HLD-V1.0** | **ABHIJIT SEN** |

Abstract:

As the cost of data is going cheaper with time, the number of online transactions is proliferating at a rapid pace. Now-a-day almost every transaction is done online which means most of the people are directly connected over the internet. In such a situation it gives a very good breeding ground for the bad black hat guys to launch an attack into our network and steal valuable resources or even destabilize our networks. Traditional approach to thwart network attacks has been the use of Firewalls but often times they are slow in detecting anomalous traffic as compared to the real time streaming data rate. With the advent of machine learning, it’s possible to detect unknown patterns out of the various characteristics of data packets and classify them as either good or bad traffic at a very fast rate. So, in this project I am going to try various machine learning models which can do the packet classification with high degree of accuracy.

1. Introduction
   1. Why this High-Level Design Document?

The purpose of this High-Level document is to add necessary details to current project description to represent a suitable model prior coding. This document is used as a reference manual for how the model interacts at a high-level.

The HLD will

• Presents all design aspects and define them in detail.

• Describe the user interface being implemented.

• Describe the hardware and software interfaces.

• Describe the performance requirements.

• Include design feature and the architecture of the project.

* 1. Scope

The HLD document presents the structure of the system, such as the database architecture, application architecture, and technology architecture. The HLD uses non-technical to middle-technical terms which should be understandable to the administrators of the system.

* 1. Definitions

|  |  |
| --- | --- |
| **Term** | **Description** |
| EDA | Exploratory Data Analysis |
| API | Application Programming Interface |
| IDE | Integrated Development Environment |
| KPI | Key Performance Indicator |

1. General Description

2.1 Product Perspective

The **Network Security With Machine Learning** is a machine learning based Intrusion Detection System which will help us predict if the network traffic is normal and anomalous.

2.2 Problem Statement

To create an AI based Intrusion Detection System using a machine learning model which will help us detect if the network traffic is an attack or normal traffic. The incoming traffic in a network consists of various parameters and traffic will be classified as either good or bad depending on the packet parameters of traffic.

2.3 Proposed solution:

First of all, the various packet parameters of the incoming network traffic is analyzed from the training set to check which parameters are actually important in distinguishing between a good and a bad traffic. The various techniques used for feature selection are manual, statistical, model based etc. and based on the best features, various ML models are designed to see which is performing the best.

2.4 Technical Requirements

The solution can be a cloud-hosted or locally hosted

For accessing this application in cloud, below are the minimum requirements:

• Good internet connection.

• Web Browser.

For training model, the system requirements are as follows:

• +4 GB RAM with i5 processor

• Operation System: Windows, Linux, Mac etc.

• Visual Studio Code / Jupyter notebook

2.5 Data Requirements

Data requirements completely depends on our problem statement.

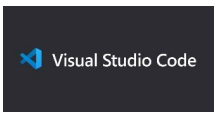
• Comma separated values (CSV) file for both the training and test dataset.

• Each row in the data file consists of the various parameters of a network packet organized as features or columns as 2D.

* There are different types of features: Categorical, Binary and Numerical.

2.6 Tools Used

Python programming language and frameworks such as NumPy, Pandas, Scikit-learn, Matplotlib, Seaborn, Flask,Jupyter Notebook are used to build the whole model.

* VS Code is used as an IDE
* Jupyter Notebook is used for EDA and model testing
* Pandas is an open-source Python package that is widely used for data analysis and machine learning tasks.
* NumPy is most commonly used package for scientific computing in Python.
* Scikit-learn is used for a machine learning.
* Flask is used to build an API.
* GitHub is used as version control system.
* 2.7 Constraints
* This model must be user friendly, as automated as possible and users should not be required to know any of the workings.

2.7 Constraints

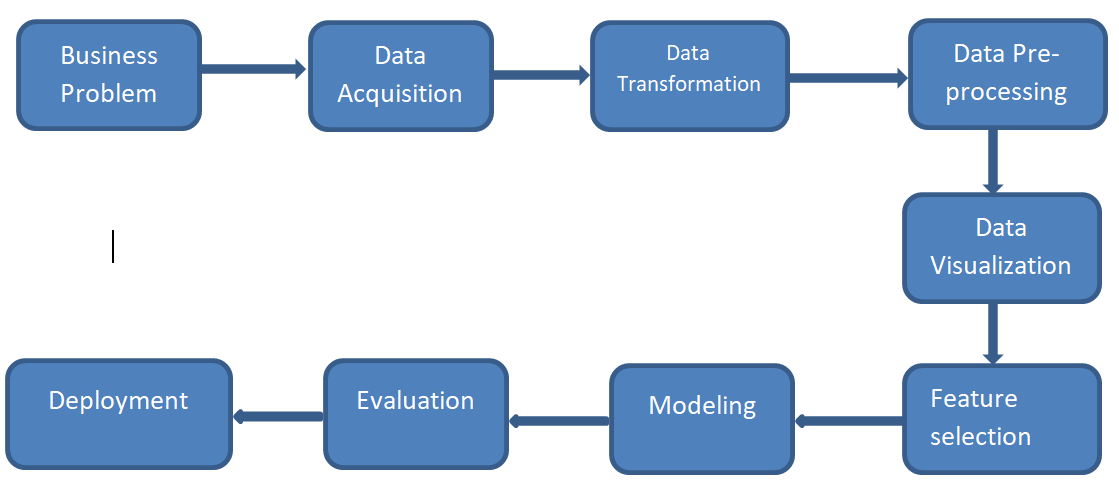
* This model must be user friendly, as automated as possible and users should not be required to know any of the workings.

2.8 Assumptions

The main objective of the project is to develop an API to predict if a network traffic is normal or anomalous based on the various network packet parameters. Machine learning based classification model is used for predicting above mentioned cases on the input data.

1. Design Details:

3.1 Process Flow:



3.2 Event Log

The system should log every event so that the user will know what process is running internally.

Initial Step-By-Step Description:

• The system identifies at what step logging required.

• The system should be able to log each and every system flow.

• Developer can choose logging method. You can choose database logging.

1. Performance
   1. Reusability

The entire solution will be done in modular fashion and will be API oriented. So, in the case of the scaling the application, the components are completely reusable.

* 1. Application Compatibility

The interaction with the application is done through the designed user interface in web, which the end user can access through any web browser

* 1. Deployment:

The deployment is done through Flask API and can be accessed through any Web browser.

1. Conclusion:

This system shows us that the different techniques that are used in order to classify if a network traffic is normal or anomalous. There are various categorical and numerical features that are used to design the predictive model. From the results we could see that Auto-encoder based model has outperformed all the other predictive ML models.