**M S Ramaiah Institute of Technology**

(An Autonomous Institute, Affiliated to VTU)

MSR Nagar, MSRIT post, Bangalore-54

**Design Document**

Title: **Alcatraz, an end-to-end SaaS product for complete computer and network security**

Team:

Akash Gupta 1MS13CS014

Ayush Kumar 1MS13CS034

Bineet Kumar 1MS13CS036

Deepak Jayaprakash 1MS13CS037

1. **Introduction**
   1. **Description of the organization of the document:**

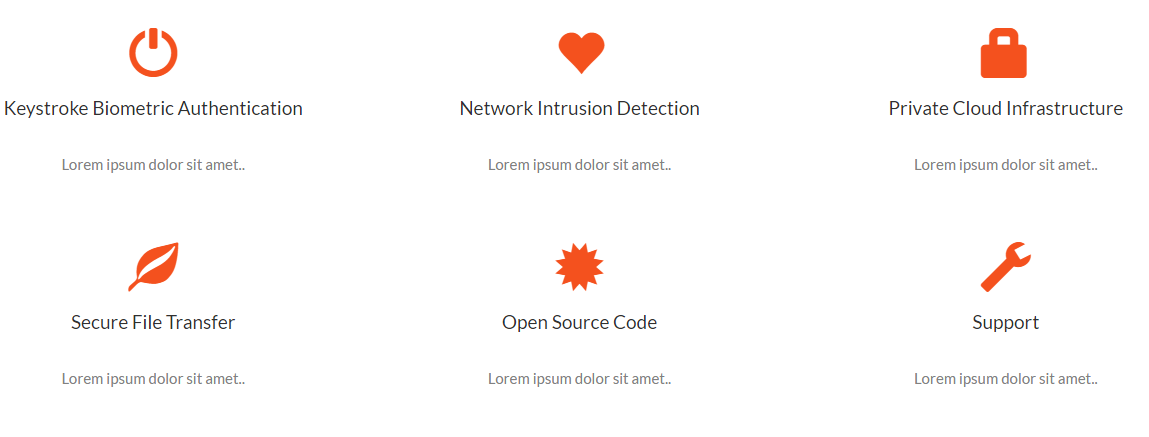
The design document is an overall guide, more of a release notes to the entire product. It contains some detailed description of each feature split module-wise. To make it easy for the end user, the document is organized feature wise so that any developer can only research about the product he/she is interested in, without bothering about the rest of the features.

As the product doesn’t follow the typical object oriented paradigm, there exists no class diagrams, instead there are some sequence diagrams, data flow diagrams, etc.

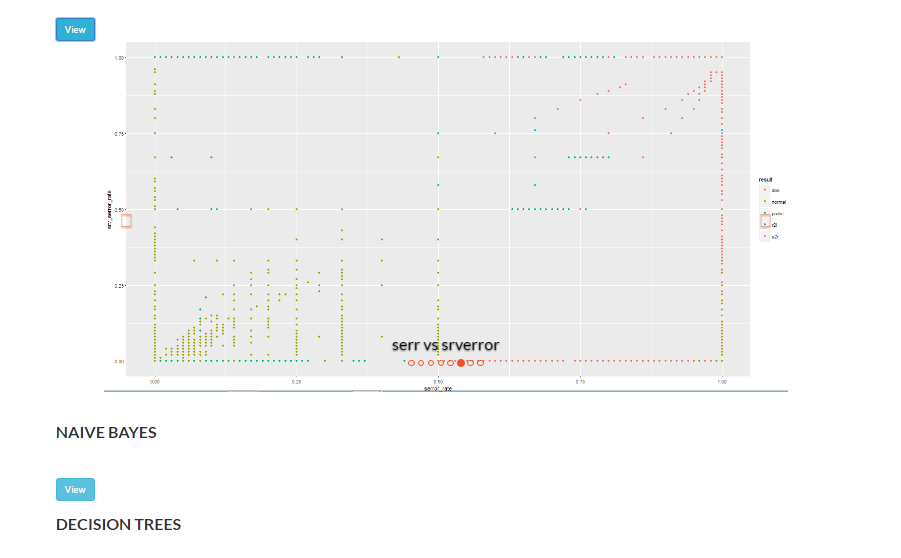
* 1. **Modules in the product:**

The module explanation is given below feature-wise.

1. **UI Module:** This involves the use of front end tech like Bootstrap, jQuery and CSS3. The features include a rich Dashboard for Keystroke demo and also graphs and output classification for Intrusion Detection.



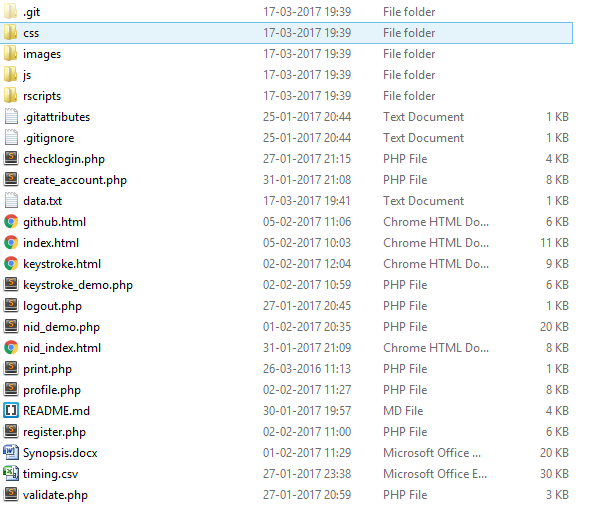
**Fig 1.1: UI landing page**



**Fig1.2: UI for IDS**

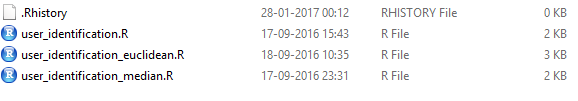
1. **Back end scripts:** After the front end validations and inputs are taken from the user, the data is passed onto the server by using PHP scripts.

In the directory structure shown here, scripts like checklogin.php, logout.php, print.php, profile.php, register.php, validate.php are the backend scripts.



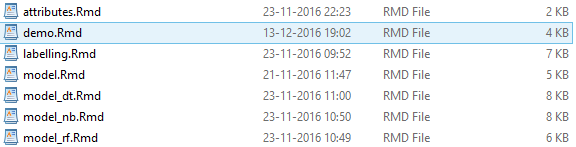
**Fig 1.3: Directory structure**

1. **Machine Learning scripts:** The rscripts used for the classification are the machine learning algorithms embedded onto it. These are the 3 classifiers used in Keystroke Authentication.



**Fig 1.4 ML scripts for Keystroke Authentication**

1. **Databases:** The database used is MySql and is used in conjugation with PHP. The db is used to maintain user accounts and profile information. Along with the db, csv files are edited repeatedly after making front-end JS validations to store some information like keystroke timings, etc
2. **Data mining algorithm module:** The DM algorithms are the classifiers used in NID. The files are as shown below, the steps involved here are selection of attributes, labeling them, building the modules and visualization.



**Fig 1.4: DM classifiers**

1. **Architecture Design**

The architecture for features are as shown below

* 1. **Keystroke Authentication:**

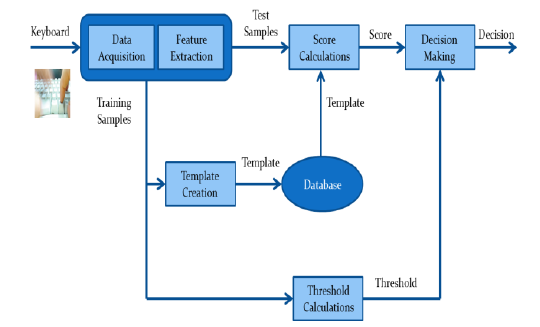
The problem with the existing work and implementations related to keystroke authentication based on static text is that the statistic chosen and the model built are not very accessible and compatible with each other. Therefore we propose an easier and a much simpler model and metric to achieve the desired classification which has better interpretability as shown in **Fig 2.1.**

The whole model can be divided four distinct steps. These are listed as follows:

1. The individual register their name and password with the database. Then the user has to type his username and train the machine for six times.
2. Features are extracted when individuals press and release keys. More specifically the delay between the key-down and key-up time.
3. The algorithm is applied and the threshold is generated based on the variations that the user has done while typing the 6 training set. Hence, the adaptiveness.
4. Calculate the Euclidean distance between training and the test samples to get the user's score.
5. Finally, the user's score is compared against its threshold to make the decision. If the Euclidean measure generated from the test sample is too high when compared to the training set then the user is classified to be an imposter.

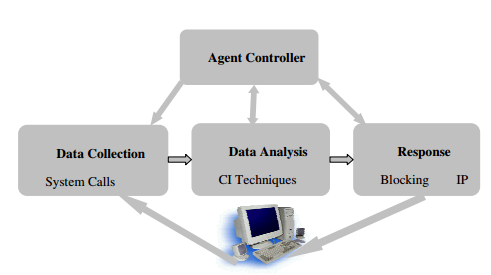
Some key milestones in the architecture:

1. Data and feature extraction
2. Metric selection: Manhattan/Mahalononobis/ Euclidean
3. Decision Making
4. Verification
5. Accuracy Analysis



**Fig 2.1: Architecture for Keystroke Authentication**

* 1. **Network Intrusion Detection:**



**Fig 2.2: Architecture design of an NID**

Majority of the IDS currently in use are either rule-based or expert-system based. Their strengths depend largely on the ability of the security personnel that develops them. The former can only detect known attack types and the latter is prone to generation of false positive alarms.

This leads to the use of an intelligence technique known as data mining/machine learning technique as an alternative to expensive and strenuous human input. These techniques automatically learn from data or extract useful pattern from data as a reference for normal/attack traffic behavior profile from existing data for subsequent classification of network traffic. Identification of important features is one of the major factors determining the success of any learning algorithm on a given task. Feature selection in learning process leads to reduction in computational cost, over fitting, model size and leads to increase in accuracy.

1. **State diagrams**

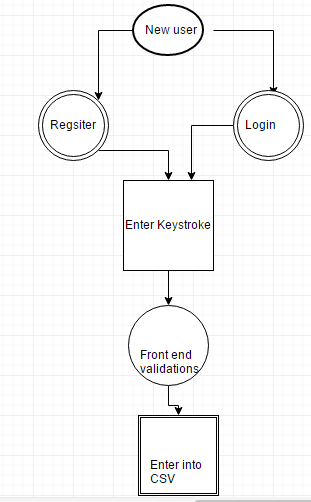


Fig 3.1: Keystroke Authentication registration State diagram

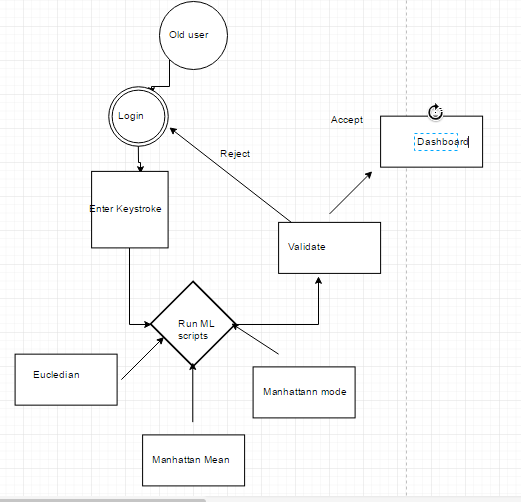
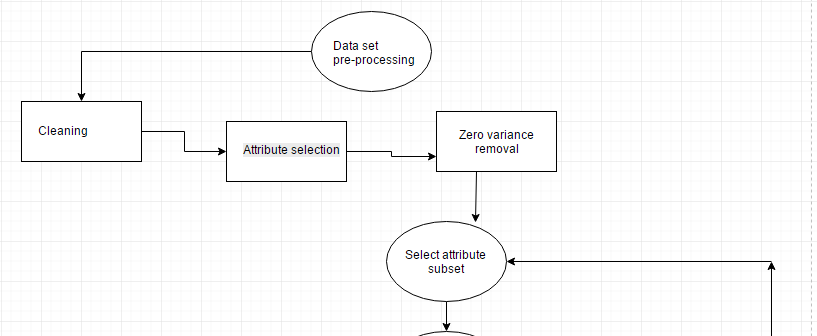


Fig 3.1: Keystroke Authentication ML State diagram



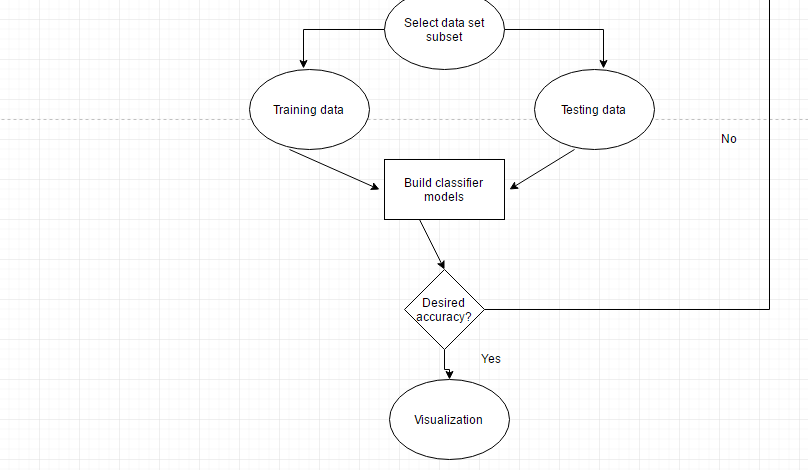


Fig 3.2 NID Data flow