Challenge

**Abstract**

In this challenge you will face real life data in which you will have to complete the entire process of detecting masqueraders. You will have to: (1) examine the data, (2) preprocess it, (3) analyze it, (4) extract features and (5) apply data mining methods that you acquired during the course. Your final task will be to distinguish masqueraders from benign users, from the history of bash commands using data mining methods.

**The Dataset**

<https://drive.google.com/drive/folders/1ApolZoNFToREPuVNfFpaonVuDZMo1K2H?usp=sharing>

The directory **FraudedRawData** contains 40 users’ history bash commands segments. For every UserX, you have a file that contains 15,000 bash commands such that each 100 commands are defined as a segment. The first 5,000 entries (=50 segments) in each file are training entries, i.e., they are guaranteed to be UserX's commands.

The other 10,000 entries (100 segments) contain both the segments of UserX and of masqueraders. 90 of the segments are genuine (i.e., benign) and 10 segments are entered by a masquerader (randomly sorted). Your task is to differentiate the 10 sessions from the other 90.

The file **partial\_labels.csv** provides the key for the rest of the task. It is a 40x150 matrix. Each row represents a user index and each column represents whether the segment of 100 commands has been entered by the user (labeled by 0) or by a masquerader (1). For example, with our counting starting at 0, the commands 6400 - 6500 in file User0 have actually been entered by a masquerader. Commands 1200 – 1300 is guaranteed to have been entered by User0.

For 10 out of the 40 users (User0…User9) - Full classification is given in the **partial\_labels.csv** for you to evaluate your algorithm before you apply it on the other 30 users.

Please ask all your relevant questions in the course forum to make it available for other students. (On moodle2)

**Guiding questions**

* What kind of problem is this (i.e., classification, clustering, regression – could be all)?
* What defines an instance?
* What type of features could we extract?
* What types of commands are in there?
* Is there a meaning to the command length?
* Is there a meaning to several commands in a row?
* What kind of statistical information can I produce?
* What type of criteria can help me evaluate my methods? (Accuracy will probably not help here).

In the **partial\_labels.csv** you were already supplied with 10 users' train and test set full labels. First, try your methods on them and then apply them among the rest.

**Grading**

For each correct prediction of a benign segment you will receive **1 point**. For each correct prediction of a masqueraded segment you will receive **9 points**.

The validation set (public score) contains 10 users from users 10-39, the test set (private score) contains the remaining 20 from users 10-39. Consider the fact that in the test set you have 30 users, and each one of them has 10 masqueraded segments and 90 benign segments. The maximum score is 30x(90x1+10x9) = 5400. Notice that labeling all the instances with a single label (0 or 1) will produce a score of 2700. The test set score is divided by 5400 (maximum score), yielding the **weighted accuracy score**.

Final score is based on the best of 2 of your selected uploads, based on the private test set evaluation. Grading is also by your relative placement in the private leaderboard (70%) and the report (30%).

Besides the dry score of the classification, the work will be also evaluated according to the creativity of the feature engineering and the use of better fitting algorithms.

The 2 highest scoring groups will receive extra credit (5-points and 3-point to the final course grade).

**Submission**

Signup to kaggle and enter the challenge (**once**):

<https://www.kaggle.com/t/d4a53f88ed5b48218977c5739c388752>

Pair to teams of two.

Alter the given **sample\_submission.csv** file with your predicted labels of the segments.

id : UserX\_FromCommands-ToCommands. label : 0 / 1.

Do not make any other changes in the file. Submit your predictions via kaggle.

**Submission is in pairs.**

Additional Submission: the final submission should contain:

* Your kaggle team name.
* Your 2 selected result files of **sample\_submission.csv.**
* A description of your data mining methods and the feature engineering process. Highlight things that you think are creative and out-of-the-box thinking. The file should **be up to 1 page**, use the format of this document for sizing and font. Name it **description.doc/docx**.
* Also, add your source code to the submission in a directory called **code.** You do not have to be organized with your code, it will only be checked in case of the suspicion of cheating. You are encouraged to use other API's and open source libraries, but submit only code that you wrote, even if it does not compile.

Submit all the deliverables in a single zip file entitled id1Number\_id2Number.zip (for example 123456789\_987654321.zip**)** through one of the group members’ Moodle account.

Good Luck!