Lab 4

Machine Learning for Cybersecurity

Due Date - November 12th

The goal of this lab is to introduce basic machine learning concepts as applied to cybersecurity problems. An incomplete Jupyter notebook will be provided as a starting point for each exercise. You will need to turn in the completed Jupyter notebooks (with corresponding output) for credit.

Note: All datasets are provided on canvas. You do not need to get these datasets from the original source.

Lab 4 Setup

For this lab, we will need to install machine learning packages for Python3, as well as Jupyter Notebook to run our notebook files.

Here are the commands needed to install on the class Ubuntu VM:

```
sudo apt install python3-pip
sudo pip3 install numpy pandas sklearn
sudo pip3 install jupyterlab
```

Next, download the accompanying archive of notebook files from Canvas. Extract this into a location of your choosing, browse to the directory on the command line and run the command:

```
jupyter notebook
```

This will start a local jupyter notebook server and open the interface in a browser.

Exercise 1

In this first exercise, you will build a supervised machine learning pipeline for classifying whether or not a particular set of network traffic corresponds to normal traffic or fuzzing traffic. The data comes from this source (https://www.unsw.adfa.edu.au/unsw-canberra-cyber/cybersecurity/ADFA-NB15-Datasets/) which has a good overview of the dataset.

Much of the preprocessing has already been done, but you will have to do some additional processing to get the data ready.

Notebook: **lab4_exercise1.ipynb**

Part 1

For Part 1, load the training and test set using the splits given. Complete the rest of the pipeline (model training and inference) using this split.

Part 2

For Part 2, combine both the given files and generate your own random train/test split. Complete the rest of the pipeline using this split.

Fill in the missing sections of the Jupyter notebook and answer the questions at the bottom of the notebook.

Exercise 2

In this exercise, you will complete a simple supervised machine learning pipeline for classifying different types of TOR traffic. Unlike all other datasets in this lab, this is a multi-class classification problem. The data comes from this source (Lashkari et al. Characterization of Tor Traffic using Time based Features. Proceedings of the 3rd International Conference on Information Systems Security and Privacy, 2017.), and has already been preprocessed for you. For reference, the features extracted for this data are described in section 3.1 of the paper.

Fill in the missing sections of the Jupyter notebook and answer the questions at the bottom of the notebook.

Notebook: **lab4_exercise2.ipynb**

Exercise 3

In exercise 3, you will run a series of experiments on syscall log data. This data is mostly unprocessed. The data is the AFDA Linux Dataset and comes from this source (G. Creech and J. Hu. A Semantic Approach to Host-based Intrusion Detection Systems Using Contiguous and Discontiguous System Call Patterns. IEEE Transactions on Computers, 2013.). In this dataset, each system call trace is provided in a separate text file. System call parameters have been removed, and each system call is replaced by a numeric ID. Thus, an example system call trace will look like this: `6 6 63 6 42 ...`

As before, fill in the missing sections of the Jupyter notebook and answer the questions.

Notebook: **lab4_exercise3.ipynb**

Part 1

In Part 1, you will perform feature extraction on the syscall log data, create a training/test split, and train/test a model.

Part 2

For Part 2, vary the percentage of attack data in the test sets, and analyze the effect on classifier performance.

Exercise 4

In exercise 4, you will run some anomaly detection experiments. You will take some normal data, inject some malicious datapoints corresponding to worm activity, and run isolation forests in order to try to detect the malicious activity. The data comes from the same source as Exercise 1.

Notebook: **lab4_exercise4.ipynb**