Network Security Lab 2

Author

- Group 32
  - 0856004 李家安
  - 0616216 戴翊安
  - 0866020 楊維鈞

How to run the code

1. Preprocess training data

```
python3 ./data.py ${training_data_folder}
```

2. Run the train and predict

```
python3 ./main.py ${predict_data_folder}
```

What model and algorithm we used

We have seperate program to 2 steps

- Pre-process data
  - For each training case (Person N), we collect the data from `Security.xml`, `Sysmon.xml`, `Wireshark.json`.
  - For Security.xml, we pick `EventID`
  - For Sysmon.xml, we pick `OriginalFileName`
  - For Wireshark.json, we pick `ip.dst`
  - save the data we pick into array and save into training_data.py for after usage
- Training data
  - We have import data from `training_data.py`
  - For each dataset, we first do the standlization to ensure there will be no offset from dataset counting.
  - After standlization, we use knn with k = 1 to train the model.
  - For sysmon OriginalFileName, we also think out an rule-base algrothm. We first find out for every user, if there is some program they usually used. And for the predict, if a program also frequently used same as the training dataset, these may be the user.
    - We define frequently used as `Sum(program used count of A) > Sum(program used count of others)`
  - With the result of three knn train and one rule-base algrothm from different data collection, we do the voting for the final result.

Anything interesting you find or problems you encounter in the whole process

- The sample testcase for Security.xml is last of Event. This made us difficult to predict the result from it.
- There are Chinese charactor in Wireshark.json. This made the json file difficult to read and will error on decoding by default.
- The training data is large. It makes my computer out of storage space.
- Wireshark has a lot of feature. To dicided which is useful for us is a lot of works.