Cergy Tech Paris
ING 3 - Option IA Cours Ouverture: AI Applications in CCAM
Open Project

Secil ERCAN

26/04/2024

Attack Detection in VANETs

Vehicular Ad Hoc Networks (VANETs) will play a significant role in the advent of future Connected and Automated Vehicles (CAVs). Indeed, they will allow vehicles and roadside infrastructure to exchange a set of information that could improve both road safety and traffic efficiency. However, these vehicular networks could be vulnerable to many attacks that can affect authentication, message authentication, availability, traceability, and privacy.

In this project, you will detect different attack types in VANETs implementing a suitable Deep Learning method or Federated Learning based on a Machine Learning method.

VeReMi Dataset

Vehicular Reference Misbehavior (VeReMi) was built specifically for testing V2X security in 2018. A simulated dataset, generated using LuST and VEINS (based on OMNET++ and SUMO), was firstly proposed for five attack types which was extended in 2020 with new types.
The extended version of the dataset has 19 attack (attack and malfunction) types:

- Malfunctions
  - Position malfunctions
    - Constant position
    - Constant position offset
    - Random position
    - Random position offset
  - Speed malfunctions
    - Constant speed
    - Constant speed offset
    - Random speed
    - Random speed offset
  - Delayed messages
- Attacks
  - Eventual Stop
  - Disruptive
  - Data Replay
  - DoS
  - DoS Random
  - DoS Disruptive
  - Grid Sybil
  - Data Replay Sybil
  - DoS Random Sybil
  - DoS Disruptive Sybil

It also provides a scenario including all types, namely *MixAll_0024*. For more information about the attack types, please see the following papers.

The paper introducing the first version of dataset:

R. W. van der Heijden, T. Lukaseder and F. Kargl, "VeReMi: A dataset for comparable evaluation of misbehavior detection in VANETs", In: Beyah, R., Chang, B., Li, Y., Zhu, S. (eds) *Security and Privacy in Communication Networks. SecureComm 2018.* Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol 254. Springer, Cham. doi: 10.1007/978-3-030-01701-9_18

The paper introducing the extension:

> J. Kamel, M. Wolf, R. W. van der Hei, A. Kaiser, P. Urien and F. Kargl, "VeReMi Extension: A Dataset for Comparable Evaluation of Misbehavior Detection in VANETs," *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, Dublin, Ireland, 2020, pp. 1-6, doi: 10.1109/ICC40277.2020.9149132.

Data was collected for two different time slots, between 07:00-09:00 and 14:00-16:00. The VeReMi dataset contains one scenario for each attack type and each time slot. For each scenario, the dataset consists of a ground truth file and a set of message log files including two types of messages:

- type = 2 GPS (Global Positioning System)
- type = 3 BSM (Basic Safety Message)

where GPS data provides the information about the local vehicle (receiver) and BSM gives the information about the message received from other vehicles through Dedicated Short Range Communications (DSRC).

A BSM provides the following features:

- rcvTime
- sendTime
- sender
- senderPseudo
- messageID
- pos & noise (in x/y/z axis)
- spd & noise (in x/y/z axis)
- acl & noise (in x/y/z axis)
- hed & noise (in x/y/z axis)

The structure of folders in this dataset can be seen in the following figure:

- Each scenario with .zip files
- Each .zip file with one .json file for ground truth and a set of .json files for each receiver vehicle

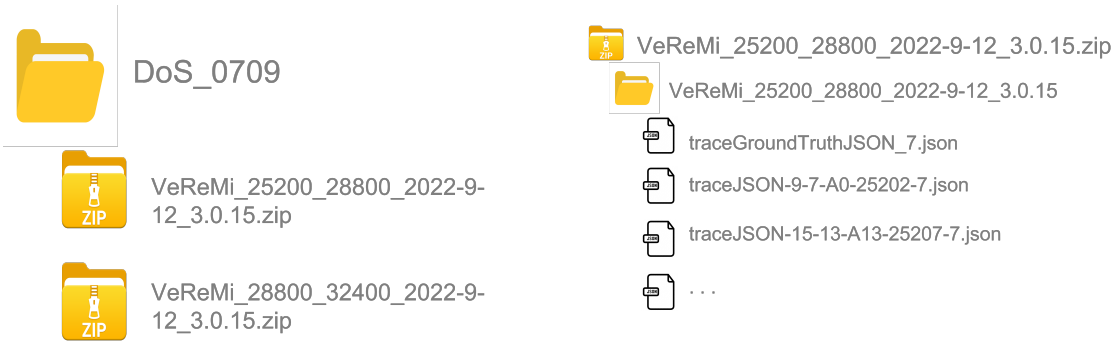

Figure 1: Structure of folders

TODO `You will work on MixAll_0024 dataset for a multi-class classification.`

The mixed scenario *MixAll_0024* consists of 24 .zip files with all attack types.

Download the dataset from the following link: `https://mega.nz/folder/z0pnGA4a#WFEUISyS5_maabhcEI7HQA` (It is enough to download *MixAll_0024*).

Data labelling

In this dataset, we need to extract the labels for each vehicle. The label shows not only whether the vehicle is an attacker or not but also the type of attacker. This information can be deduced from the name of .json file which identifies the receiver by vehicle number, OMNET++ module number, and the attack type. E.g., JSONlog-15-13-A13-25207-7.json refers to the 15th vehicle with OMNET++ module ID 13. A13 refers to the fact that this vehicle is an attacker of type 13. The number following A indicates the attack type.

traceJSON- 15 -13- A13 -25207-7.json
15 : (receiver) vehicle number
13: OMNET++ module ID
A13 : attacker type

Since each .json belongs to a receiver vehicle, we can extract the label of the receiver, however, we need the label of the sender in order to detect whether a sender is an attacker or not. Hence, we need to list the labels of (receiver) vehicles and then *merge* these labels to the main dataset by comparing on *sender ID* to obtain the label of the sender.

Labels for different attack types are listed here: 0 Normal, 1 ConstPos, 2 ConstPosOffset, 3 RandomPos, 4 RandomPosOffset, 5 ConstSpeed, 6 ConstSpeedOffset, 7 RandomSpeed, 8 RandomSpeedOffset, 9 EventualStop, 10 Disruptive, 11 DataReplay, 12 DelayedMessages, 13 DoS, 14 DoSRandom, 15 DoSDisruptive, 16 GridSybil, 17 DataReplaySybil, 18 DoSRandomSybil, 19 DoSDisruptiveSybil.

`TODO` `You will extract labels from the file names.`

Content of the project

- Import the data (including labels)
- Clean the dataset (if necessary)
- Discover the variables
- Propose a suitable Deep Learning method or Federated Learning based on a Machine Learning method for multi-class classification
- Implement the proposed method (choosing the proper parameters)
- Use the accuracy, precision, recall, and F1-score indicators to evaluate the performance of the method
- Provide confusion matrix including all attack types
- Comment on the results

Upload your python file and (maximum five pages) report summarizing the dataset, problem, outputs, and the importance of method you proposed for this dataset/problem.

Deadline for both groups: **May 9, 2024 Thursday 23:59**