A Misbehavior Authority System for Sybil Attack Detection in C-ITS

Joseph Kamel, Farah Haidar, Ines Ben Jemaa, Arnaud Kaiser, Brigitte Lonc, Pascal Urien

A Misbehavior Authority System for Sybil Attack Detection in C-ITS

Joseph KAMEL
IRT SystemX
Palaiseau, France
joseph.kamel@irt-systemx.fr

Farah HAIDAR
Renault
Guyancourt, France
farah.haidar@renault.com

Ines Ben Jemaa
IRT SystemX
Palaiseau, France
ines.ben-jemaa@irt-systemx.fr

Arnaud Kaiser
IRT SystemX
Palaiseau, France
arnaud.kaiser@irt-systemx.fr

Brigitte LONC
Renault
Guyancourt, France
brigitte.lonc@renault.com

Pascal Urien
Telecom ParisTech
Paris, France
pascal.urien@telecom-paristech.fr

***Abstract*—Global misbehavior detection is an important back-end mechanism in Cooperative Intelligent Transport Systems (C–ITS). It is based on the local misbehavior detection information sent by Vehicle's On–Board Units (OBUs) and by Road–Side Units (RSUs) called Misbehavior Reports (MBRs) to the Misbehavior Authority (MA). By analyzing these reports, the MA provides more accurate and robust misbehavior detection results. Sybil attacks pose a significant threat to the C–ITS systems. Their detection and identification may be inaccurate and confusing. In this work, we propose a Machine Learning (ML) based solution for the internal detection process of the MA. We show through extensive simulation that our solution is able to precisely identify the type of the Sybil attack and provide promising detection accuracy results.**

Keywords*—*Sybil Attack, Misbehavior Detection, Cooperative Intelligent Transport Systems, Machine Learning, Cyber-Security

I. Introduction

Cooperative Intelligent Transport Systems (C–ITS) is a mature technology that aims at improving road safety, traffic efficiency and users comfort. Vehicle's On–Board Units (OBUs) and Road–Side Units (RSUs) (also referred to as Intelligent Transport Systems (ITS) Stations (ITS–Ss)) periodically broadcast Vehicle–to–Everything (V2X) messages to advertise their geographical position, speed, heading and other parameters to the neighboring ITS–Ss.

Cyber-security in V2X communications is ensured by the use of a Public Key Infrastructure (PKI), as specified in the European and the American standards. In C–ITS, the PKI is a global entity responsible for the distribution and management of the digital keys use by the vehicles to communicate. Basically speaking, vehicles authenticate themselves to the PKI and get in return a unique long-term digital certificate called Enrolment Certificate (EC). ITS–Ss then use their EC to request several short-term Authorization Tickets (ATs) from the PKI. Thus ITS–Ss typically have one EC and several ATs, also called pseudonym identities. Pseudonyms are frequently changed by ITS–Ss to avoid tracking and guarantee drivers privacy. ITS–Ss use their pseudonyms to digitally sign the V2X messages they broadcast. The PKI prevents external malicious entities from attacking or disrupting the system. However, it does not prevent internal entities (i.e. ITS–Ss that are registered at the PKI) from sending false information in their V2X messages. Misbehavior detection is a promising technology that enables detection of potentially misbehaving ITS–Ss within the system by checking the V2X messages data.

The misbehavior detection process is performed at two levels: local detection at the ITS–S level and global detection at the central Misbehavior Authority (MA). Local detection consists of performing plausibility and consistency checks on the received V2X message data [1]. The results of these checks are then analyzed by a set of misbehavior detection algorithms. If an anomaly is detected, the ITS–S reports it by sending a Misbehavior Report (MBR) [2] to the MA. Global detection starts after the reception of these MBRs by the MA. The MA is in charge of collecting the MBRs coming from the ITS–Ss and deciding whether the reported ITS–Ss are actually misbehaving or not (e.g. it may be possible that an ITS-S unintentionally sends false information because its sensors are faulty). This decision is achieved by analyzing the reports using a set of algorithms (anomaly detection, deep learning).

In this work, we focus on Sybil attacks. A Sybil attack takes place when an ITS–S takes advantage of its available pool of pseudonyms and uses them simultaneously to disturb the system: it periodically broadcasts V2X messages and signs them with different pseudonyms. The pseudonyms used for the Sybil attack are valid which complicates the MA detection.

In this paper, we propose a misbehavior detection process at the MA, which is able to identify and detect both Sybil and other types of attacks. It is based on advanced Machine Learning (ML) algorithms. In addition, we evaluate our solution by integrating it in both European Telecommunications Standards Institute (ETSI) and Institute of Electrical and Electronics Engineers (IEEE) C–ITS standard architecture.

The remainder of the paper is organized as follows: Section III details the Sybil attack and its variations. Section II

978-1-7281-3885-5/19/$31.00 ©2019 IEEE

discusses the related works. Section IV presents the global C-ITS system architecture. Section V details our proposed software architecture for decision at the MA. Section VI provides the simulation settings and parameters we used to evaluate our proposal. Section VII presents and discuss the obtained results. Finally, section VIII concludes this work and gives some perspectives.

II. Related Works

Sybil attack was first introduced by Douceur in [3]. Due to the important damages it may cause in C–ITS systems, researchers have proposed several detection approaches.

Pouyan et al. [4] present three methods for local Sybil attack detection. The resource testing method assumes that a radio network entity can not send and receive on the same channel at the same time. This detection method is not valid in vehicular networks because attackers may have multiple channels to send and receive messages. The Position verification method assumes that a vehicle can be localized at only one position at the same time. The encryption and authentication based methods assume that using a PKI is enough to detect Sybil attack. In our analysis, we consider that a legitimate entity with valid key materials can perform a Sybil attack.

Hao et al. [5] propose a protocol that detects Sybil nodes in a cooperative way by examining the consistency between the vehicles positions and those of their neighbors. The idea is based on detecting the sudden appearance of a vehicle or of multiple vehicles as well as on evaluating the number of neighbors. When a vehicle detects locally that a neighbor is potentially malicious, it broadcasts a warning message to have the confirmation of other neighbors that an attack is occurring. When the number of vehicles that confirm that an attack is occurring is greater than a threshold, the identified vehicle may be quarantined for a certain period of time or reported to the authority. We believe that cooperative detection systems are not reliable because the attacker takes part of the community and could distort the detection procedure. Moreover, it requires an honest majority to work properly.

Ghaleb et al. [6] propose a local misbehavior detection model based on artificial neural network. Some features are used to decide if a vehicle is misbehaving or not. In our opinion, local detection is not sufficient as it is based only on captured information by the vehicle. A global system with access to more misbehaviour reports is required to improve the detection system.

Our analysis of the existing works show that most of the proposed solutions for C–ITS focus on local Sybil attack detection. We believe that a global detection is crucial and still not studied well in the context of C–ITS. In our work, we propose a complete and generic solution for the global detection at the MA level considering multiple types of Sybil attacks.

III. The Sybil Attack

Vehicles communicate using the IEEE 802.11p network, also known as the ITS-G5 network. All vehicles periodically broadcast V2X messages using the 5.9 GHZ frequency band. Each message contains the vehicle's pseudonym (a temporary identity) and several kinematic information (position, velocity, heading, etc..). The C–ITS PKI delivers to vehicles one long term certificate and several short term certificates, called pseudonym certificates. These certificates are used to sign the V2X messages. Vehicles frequently change their pseudonym to avoid tracking and protect their privacy. Each vehicle uses a single pseudonym certificate for a certain time period to sign its generated V2X message. To ensure the ability of vehicles to continuously send V2X messages, it is necessary that several valid pseudonyms are simultaneously available. The European Commission recommends the use of a maximum pool of 100 valid pseudonym certificates [7]. When a ITS–S is low on available pseudonyms, it sends requests to the PKI to refill its pool with new certificates. Notice that vehicles should not use more than one pseudonym certificate during a certain period of time to sign their messages. However, a misbehaving vehicle may intentionally use multiple valid pseudonym certificates at the same time, which results in a Sybil attack.

Depending on the attackers objective, this attack may take different forms. In this work, we classify it into 4 categories:

1) S1 Traffic Congestion Sybil: As shown in figure 1, the attacker uses valid pseudonyms to simulate multiple ghost vehicles. Vehicles within the communication range of the malicious vehicle receive the fake messages and conclude that a congestion occurs on the road. The attacker intelligently calculates the kinematic data for the ghost vehicles such that the fake messages have plausible and coherent contents.

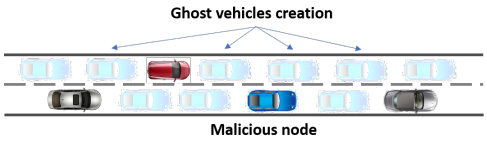


Fig. 1: S1: Traffic congestion Sybil

2) S2 Data replay Sybil: This attack consists on reporting legitimate vehicle as malicious. The attacker chooses a victim vehicle and creates messages containing positions broadcasted by the victim vehicle. As shown in figure 2, the attacker sends at time t=1 a message containing the same position (X1) as the victim vehicle. One of the hardest challenge of the detection system is to know which node is the real one (the victim) and which one is the ghost one. In this case, there is a good probability that the victim vehicle is reported as attacker.

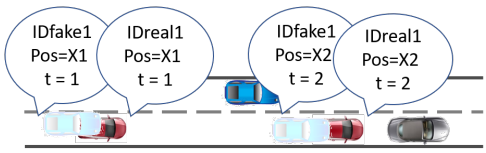


Fig. 2: S2: Data replay Sybil

3) S3 Dos Random Sybil: As shown in figure 3, the attacker creates messages with random data (e.g., the position is not on the road). The attacker uses a different pseudonym for every sent message. The motivation behind such attack could be to overwhelm the misbehavior detection algorithms of neighboring ITS–S.

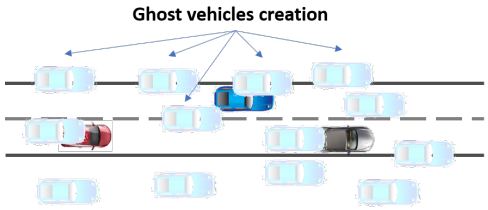


Fig. 3: S3: Dos Random Sybil

4) S4 Dos Disruptive Sybil: This attack is a combination between S3 and S2. As shown in figure 4, the attacker uses a different pseudonym for each message but does not fill them with random data. Instead, the transmitted data is based on the ones received from the neighboring vehicles. The difference between S4 and S2 is that S4 does not follow one victim, the attacker is trying to disturb the system with sudden appearance of vehicles. For example, the attacker send at time t=1 a message containing a position (pos=X1), and at time t=2 a message containing another position (pos=X2) which is the position of another vehicle. The motivation of the attacker could be the degradation of the safety system quality thus decreasing the reliability of the exchanged information.

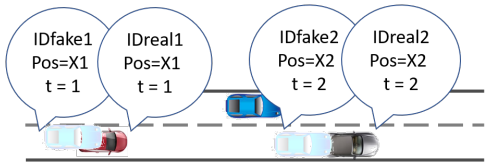


Fig. 4: S4: Dos disruptive Sybil

IV. Proposed System Model

A. The misbehavior detection process

We propose a misbehavior detection process that consists of the following operations (Figure 5):

1) The misbehavior detection: OBUs and RSUs locally detect a potential misbehaving entity. The ITS–S will keep checking the plausibility and the consistency of several mobility information in the V2X message until one check fails the tests. These local detection checks are detailed in section IV-B.
2) The misbehavior reporting: when an ITS–S detects a malicious behavior, it sends a MBR to alert the MA about the existence of a malicious entity in the network. The MA is a central authority located in the cloud, which is in charge of receiving and processing the MBRs. Notice that the reporting protocol and format are specified in our previous work [2].

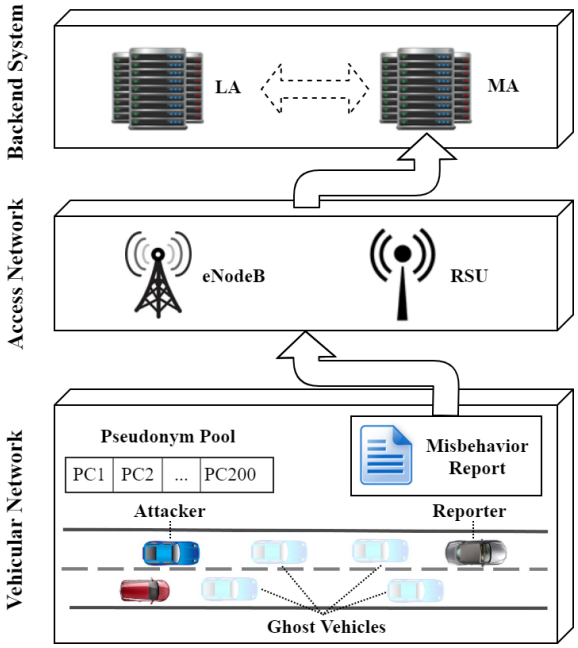


Fig. 5: System model

3) The misbehavior investigation: the MA processes the received MBRs in order to detect the type of the reported misbehavior. The global detection of Sybil attack requires linking between several pseudonyms to identify the original attacker generator. The American architecture integrates a Linkage Authority (LA) entity whose function is to provide the results of linking several pseudonyms based on a straightforward association between them. However, a similar function does not exist in the European standards. This is why, without prior knowledge on pseudonyms association, we specified an LA-like function based on ML technique to link several pseudonyms. This operation is detailed in section V.

B. Local Detection Checks

The misbehavior detection process is largely based on checks performed by the ITS–Ss. Therefore, these checks should contain relevant and sufficient information for the detection process. In this work, we aggregated and implemented the checks used in multiple local detection works [8]. However, the implemented checks does not return a binary value, instead a plausibility factor is calculated as described in our previous work [9]. For more details on the implementation of the detectors, all the implementations are open-source on github [10]. Here is a summary of all the selected local detectors:

- *Range plausibility:* The position of the sending ITS–S is inside of the ITS–S maximum radio reception range.
- *Position plausibility:* The position of the sending ITS–S is at a plausible place (e.g. on a road, without overlaps

of physical obstacles, etc.).

- *Speed plausibility:* The speed advertised by the sending ITS–S is less than a predefined maximum threshold.
- *Position consistency:* The distance separating two consecutive beacons from the same ITS–S is less than a predefined maximum threshold.
- *Speed consistency:* The speed difference between two consecutive beacons from the same ITS–S is a plausible acceleration or deceleration.
- *Position speed consistency:* The distance separating two consecutive beacons from the same ITS–S is consistent with the advertised speed.
- *Beacon frequency:* The time separating two consecutive beacons from the same ITS–S is compliant with the standards.
- *Position heading consistency:* The position angle separating two consecutive beacons from the same ITS–S is consistent with the advertised heading.
- *Intersection check:* The beacon from two different ITS–Ss must not have intersecting positions.
- *Sudden appearance:* The beacon of a suddenly appearing ITS–S within a certain close range must not have a preset positive speed.
- *Kalman Filter Tracking:* The beacon information of the ITS–S is tracked with a Kalman filter [11]. The advertised beacon information must not diverge from the predicted information as proposed in [12]. The calculation implementation is open source [10].

V. Proposed Misbehaviour Authority Investigation Process

We propose a MA system architecture (see figure 6). The proposed architecture consists of three main phases: *General Misbehavior Type Detection*, *Pseudonym Linkage* and *Sybil Type Detection*. The MA system takes an MBR as input and returns the predicted attack type as output. In the first phase we start by detecting misbehavior types related to one single pseudonym identity. This detection is effective against misbehavior types that are non-Sybil. However, this detection fails against attacks that makes use of multiple pseudonyms. To address this problem we propose the pseudonym linking schemes of phase two. In the second phase we attempt to link the pseudonyms related to the same physical reported ITS–S. If no link is found the process is complete and the misbehavior type is returned. If a link is found then a Sybil attack is suspected and the linked pseudonyms are candidates for Sybil attack type detection in phase three. In this third phase the linked pseudonyms are treated as one and the evidences collected from all the linked pseudonyms is used in a specific Sybil type detection process. The predicted Sybil misbehavior type is returned and the process is complete.

A. Phase 1: General Misbehavior Type Detection

The goal of this phase is to detect as accurately as possible the type of misbehavior related to one pseudonym. This is accomplished using the following steps:

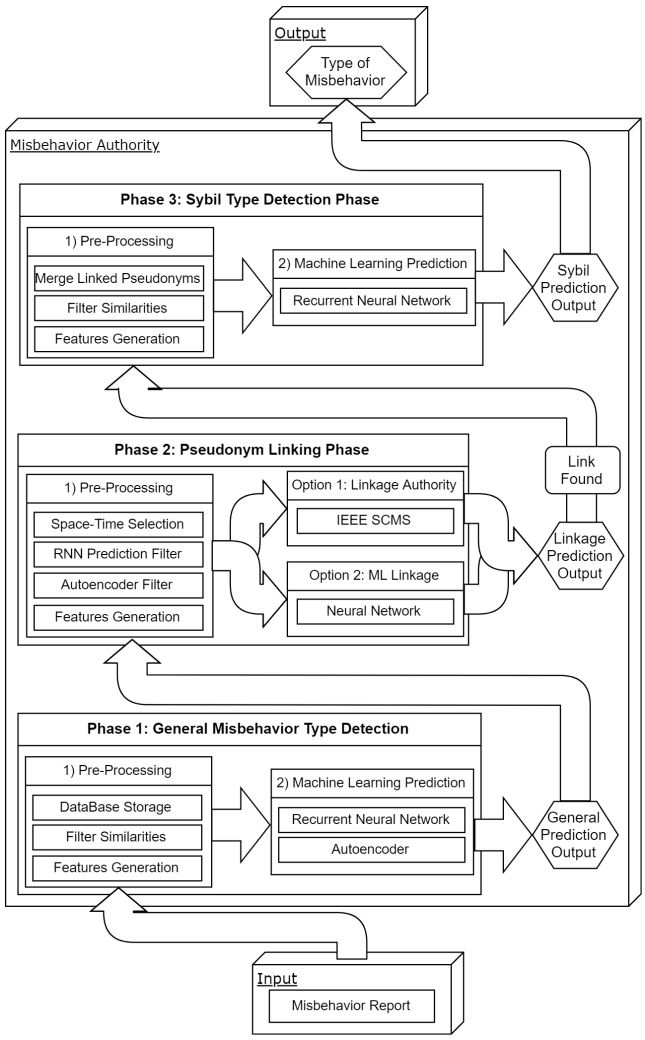


Fig. 6: Global Detection System Architecture

1) Pre-processing :
   - Database Storage: We start by adding the reports to a spatial database. This enable us to do fast and efficient geographic queries.
   - Filter Similarities: We aggregate similar data from multiple reports (e.g. several ITS–S detecting the same implausibility and sending the same evidences). We found this filtering to significantly improves the detection speed and quality.
   - Feature Generation: We extract key information from the collected evidence in our database. These information, also called features, are then used by the ML-based detection algorithm to determine the type of misbehavior. The quality of the extracted features are crucial to the robustness and accuracy of the detection. We evaluated different sets of features through cross validation on our relatively large data-set and verified a set of 30 features:
     – The local detection checks done on V2X messages described in section IV-B.

- The $Position$, $Speed$, $Accel$, $Heading$ and $Time$ of the last beacon.
- The $\Delta Position$, $\Delta Speed$, $\Delta Accel$, $\Delta Heading$ and $\Delta Time$ between the last 2 beacons.
- The $\Delta Time$ between the last 2 received reports.
- The number of time this evidence has been received (e.g. the number of filtered reports data).

2) Prediction:
- Autoencoder: We first compress the previously created features using an autoencoder. Although this step may not be important for the detection of a non-Sybil attack, the result is useful for the pseudonym linkage Phase.
- Recurrent Neural Network (RNN): We provide the previously calculated and compressed features by the auto-encoder to an RNN. The choice of an RNN was made due to the temporal relation between the received reports. We tested different simple models and determined that the Long Short-Term Memory (LSTM) has a good performance in our use case. Hence, for our testing purposes we use an LSTM. However, more rigorous experimenting is needed to test different models, thus more complex and elaborate models could be proposed in the future.

B. Phase 2: Pseudonym Linking

The goal in this phase is to link the pseudonyms coming from the same vehicle as accurately as possible. However, in order to be compliant with both US and European C-ITS Systems, we explore two options for pseudonym linking.

1) Pre Processing:
- Space-Time selection: In this step we use the spatial database to recall all the reports within a range and time of the reporter node. We propose doing this for processing efficiency reason, e.g. it prevents having to test all the previously received pseudonyms and limits the detection to the target region.
- Prediction Filter: We use the output prediction of the RNN to filter reports with diverging predictions. If the RNN detects the same type of attack for two different pseudonyms in the same region and type, we consider them candidates for linking test. Otherwise, the pseudonyms are discarded.
- Autoencoder Distance Filter: We use the output prediction of the auto-encoder to filter reports with diverging compressed features. We calculate distances between the compressed features of the recalled and current pseudonym. We exclude the pseudonyms with compressed features far from the one.
- Linkage Features Generation: Similarly to the first feature generation step, we need to extract the relevant information form the selected pair of pseudonyms. These features are used by the ML algorithm to determine if the reported pseudonyms are linked or not. Therefore, from each pair of reports we extract and similarly validate the following set of features:
  - The difference between all the previously calculated features of the two latest received report of each pseudonym.
  - The euclidean distances between the reporter ITS–S position and broadcasted position of the reported pseudonym for both selected pseudonyms.
  - The euclidean distances between the reporter ITS–S position of one pseudonym and the broadcasted position of the other reported pseudonym.
  - The absolute difference between the two latest RNN predictions of the selected pseudonyms.

2) Linking:
- Linkage Authority (Option 1): The US architecture supports a LA. This enables us to do straightforward linking between the selected pseudonyms. No ML-based prediction is needed.
- ML-based linking (Option 2): The European architecture lacks a LA. To cope with this issue we propose using a ML-based solution. The goal of this solution is to determine, using the previously calculated features, if two reported pseudonyms are generated by the same physical ITS–S. For testing purposes we use an Multi-Layer Perceptron (MLP), which is the classical type of neural networks. However, more rigorous experimenting is needed to propose more complex solutions.

C. Phase 3: Sybil Type Detection

This algorithm activates if a link is found in the previous phase. The goal is to detect the type of Sybil attack related to the number of linked pseudonyms in the previous phase.

1) Sybil Algorithm Pre-Processing:
- Merge Multiple Linked Pseudonyms: In this step we prepare a new database entry where we merge the evidence data of the multiple linked pseudonyms.
- Filter Similarities: Similarly to the previous filter, we aggregate similar data from the new database entry. This also improves the prediction performance.
- Sybil Features Generation: We extract from the new and filtered database entry the key detection information. These features are the indications used by the ML algorithm to determine the type of Sybil attack. We create the same features used by the general algorithm described in the first phase phase. Additionally, we add two specific feature to the Sybil type detection:
  - The number of linked pseudonyms.
  - The number of reports in the new database entry.

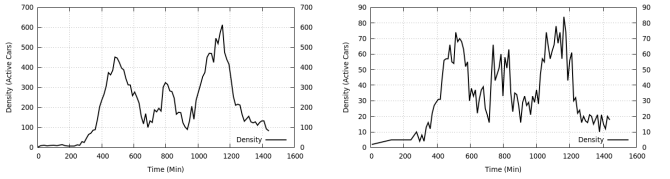
2) Prediction:
- Recurrent Neural Network: Similarly to the general prediction algorithm, we provide the previously calculated features to an RNN. We also use the LSTM for testing purposes. The Model and the ML algorithms and hyper-parameters should be investigated further.

Finally the output of the MA algorithm will be the *Sybil Attack Type* if a pseudonym link is found and the *General Misbehavior Type* otherwise.

VI. Simulation settings and scenarios

In order to evaluate our proposed solution, we use the F^{2}MD framework [13]. F^{2}MD is a VEINS extention, VEINS [14] is an open source framework for vehicular network simulations. VEINS is based on OMNeT++ and SUMO, a network simulator and road traffic simulator respectively. We use the Luxembourg SUMO Traffic (LuST) scenario for the vehicle traces [15]. LuST is a synthetic data set generated with SUMO and validated with real data, provided by the vehicular lab of the university of Luxembourg [16]. We use different sections of the scenario for the training part and testing part of our ML algorithms (Figure 7). The train scenario is $6.51km^2$ and of peak density of $104.5Vehicle/km^2$. The test scenario is $1.61km^2$ and of peak density of $67.4Vehicle/km^2$. The topology of the scenarios consists of a downtown area, with residential roads and main arterial roads linked to highways. In total, the train scenario contains $82,146$ vehicles with $301,082,858$ exchanged V2X messages and $5,209,072$ transmitted MBRs. The test scenario contains $24,663$ vehicles with $17,051,860$ exchanged V2X messages and $294,160$ transmitted MBRs. Both scenarios have an attacker rate of 5%. For further technical details, the source code of our VEINS extension along with all the configuration details of the simulated scenario are published on github [10].

(a) Test Network (b) Train Network



(c) Test Vehicle Density (d) Train Vehicle Density

Fig. 7: Simulation Scenario: Part of Luxembourg city

In both scenarios we implement the attacks described in section III. Additionally, we implement a set of other types of misbehavior in order to increase the complexity of the classification. We extracted from the literature a set of possible misbehavior types [17]: (1) *Fixed Position Offset:* the vehicle broadcasts its real position with a fixed offset, (2) *Random Position Offset:* the vehicle broadcasts its real position with a random offset limited to a max value, (3) *Fixed Speed:* the vehicle broadcasts the same speed each beacon, (4) *Fixed Speed Offset:* the vehicle broadcasts its real speed with a fixed offset, (5) *Random Speed Offset:* the vehicle broadcasts its real speed with a random offset limited to a max value.

VII. Results and analysis

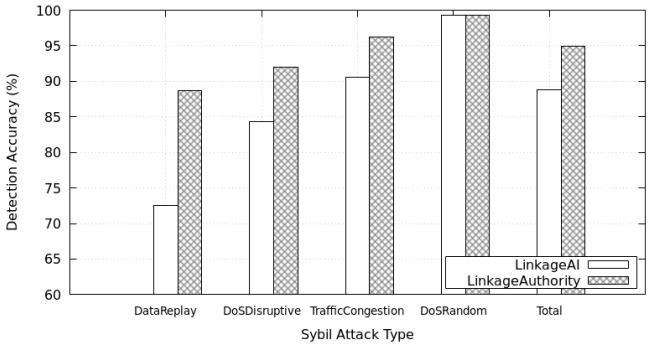


Fig. 8: Detection Accuracy By Type of Linkage

Figure 8 shows the results of detection accuracy of the Sybil attacks by linkage type. The detection accuracy is the ratio of the true classified reported vehicles over all the reported vehicles. The first result we notice is that the total detection for Sybil attacks types using a LA is at 94.97%, whereas it's only at 88.83% using the Linkage AI model. This is an expected result as the AI prediction is uncertain compared to the absolute information provided by the LA. We also notice that the detection accuracy difference between the two linkage types is proportional to the general detection accuracy for each type of Sybil attack. This is due to the prediction output of the first phase. Attacks that are difficult to classify, are less likely to be linked by the AI Linkage. Especially since the classification output of the first phase is used as an input feature for the AI model. This problem however is not present using the LA.

TABLE I: AI Linkage Evaluation Results

$Precision$	$Recall$	$Accuracy$	$Fallout$	$Specificity$	F_1Score
95,6%	89,6%	96,4%	1,3%	98.7%	92.5%

Table I shows the Evaluation Results of the AI Linkage Mechanism. The evaluation metrics are detailed in our previous publication [9]. As this system is replacing the LA, a high confidence in a perceived linkage is needed before it is considered. This shows clearly in the results as the $Precision$ is significantly higher than the $Recall$. Consequently, the lower $Recall$ (with respect to the LA) results in the lower detection accuracy perceived in Figure 8.

Figure 9 shows the detection accuracy of the attacks by the number of the received reports. In other words, it shows the number of reports needed for an accurate detection.

First, we notice that the detection accuracy for the *Data Replay Sybil* and *Dos Disruptive Sybil* attacks require more reports to converge than for the *Traffic Congestion Sybil* and

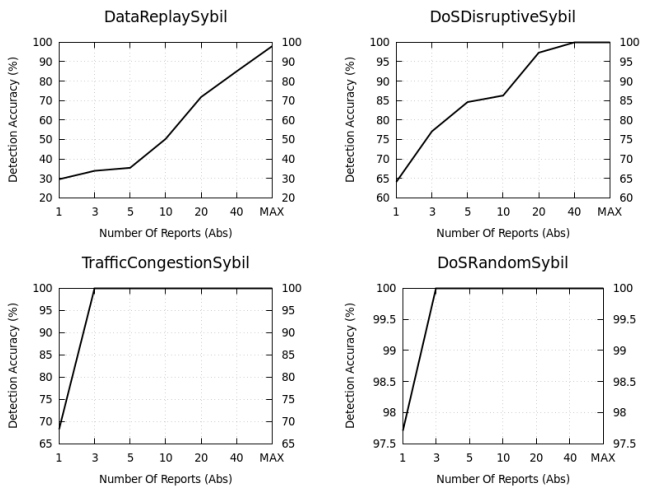


Fig. 9: Detection Accuracy by Number of Received Reports

Dos Random Sybil. The reasoning for that is both the former attacks cause the local vehicles to simultaneously report other genuine vehicles alongside the attacker. These false positive reports adds a significant amount of noise to the data. Therefore, more data is required to sort the genuine pseudonyms from the attacker pseudonyms. We also notice that the *Data Replay Sybil* attack requires more information than the *Dos Disruptive Sybil* to converge. This is a consequence of the former intelligently generating a realistic path instead of just replaying data incoherently.

Additionally, we notice that *Traffic Congestion Sybil* has a relatively low detection rate with one report. However, even though the attacker tries to intelligently remain within the plausible range, the detection then quickly converges. This is due to the lack of the simultaneously falsely reported genuine vehicles. The information is clean from false positives thus multiple reports are analyzed much more efficiently.

Finally, the *Dos Random Sybil* attack does not cause false positives neither is it within the plausible ranges. As a result it is easily detected even with evidence from only one report.

VIII. Conclusion and Future Work

Sybil attacks are a dangerous threat that can significantly deteriorate the C–ITS system quality and lead to catastrophic road accidents. MisBehavior Detection is the proposed solution against those types of attacks. The current MisBehavior Detection architecture consists of a local component done on every vehicle and a global component in the cloud. Local misbehavior detection is well treated in the literature. However, the global component is not as mature.

In this paper we propose a global misbehavior detection mechanism for C–ITS. More precisely, we proposed a MA architecture specifically robust against Sybil attacks. This is achieved using machine learning analysis on the Misbehavior Report and pseudonym linking on the global level. We propose an implicit Machine Learning (ML) based linking or direct linking using the IEEE Linkage Authority. We show through extensive simulations that overall detection rate for various types of Sybil attacks is relatively high.

Future works, includes plans of deployment and testing of variants of this solution on the C–ITS field tests in France. Currently, we are still developing and refining and the different components of the proposed architecture.

Acknowledgment

This research work has been carried out in the framework of the Technological Research Institute SystemX, and therefore granted with public funds within the scope of the French Program *Investissements d'avenir*.

References

[1] R. W. van der Heijden, S. Dietzel, T. Leinmüller, and F. Kargl, "Survey on misbehavior detection in cooperative intelligent transportation systems," *IEEE Communications Surveys Tutorials*, vol. 21, no. 1, pp. 779–811, Firstquarter 2019.

[2] J. Kamel, I. Ben Jemaa, A. Kaiser, and P. Urien, "Misbehavior reporting protocol for c-its," in *2018 IEEE Vehicular Networking Conference (VNC)*, Dec 2018, pp. 1–4.

[3] J. R. Douceur, "The sybil attack," in *Peer-to-Peer Systems*, P. Druschel, F. Kaashoek, and A. Rowstron, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 251–260.

[4] A. Pouyan and M. Alimohammadi, "Sybil Attack Detection in Vehicular Networks," in *Computer Science and Information Technology 2.4*, 2014, pp. 197 – 202.

[5] Y. Hao, J. Tang, and Y. Cheng, "Cooperative sybil attack detection for position based applications in privacy preserved vanets," in *IEEE Global Telecommunications Conference - GLOBECOM*, Dec 2011, pp. 1–5.

[6] F. A. Ghaleb, A. Zainal, M. A. Rassam, and F. Mohammed, "An effective misbehavior detection model using artificial neural network for vehicular ad hoc network applications," in *2017 IEEE Conference on Application, Information and Network Security (AINS)*, Nov 2017, pp. 13–18.

[7] European Commission (EC), "Security Policy & Governance Framework for Deployment and Operation of European Cooperative Intelligent Transport Systems (C-ITS)," *Cooperative, connected and automated mobility (CCAM)*, pp. 1–36, December 2017.

[8] S. So, P. Sharma, and J. Petit, "Integrating plausibility checks and machine learning for misbehavior detection in vanet," *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 564–571, 2018.

[9] J. Kamel, A. Kaiser, I. Ben Jemaa, P. Cincilla, and P. Urien, "CaTch: a confidence range tolerant misbehavior detection approach," in *2019 IEEE Wireless Communications and Networking Conference (WCNC) (IEEE WCNC 2019)*, Marrakech, Morocco, Apr. 2019.

[10] J. Kamel, "Github repository: Framework for misbehavior detection (f^2md)," 2019. [Online]. Available: https://github.com/josephkamel/f2md

[11] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, p. 35, 1960.

[12] A. Jaeger, N. Bißmeyer, H. Stübing, and S. A. Huss, "A novel framework for efficient mobility data verification in vehicular ad-hoc networks," *International Journal of Intelligent Transportation Systems Research*, vol. 10, no. 1, pp. 11–21, Jan 2012. [Online]. Available: https://doi.org/10.1007/s13177-011-0038-9

[13] Framework For Misbehavior Detection (F^2MD). (2019) F^2MD website. [Online]. Available: https://www.irt-systemx.fr/f2md

[14] C. Sommer, R. German, and F. Dressler, "Bidirectionally coupled network and road traffic simulation for improved ivc analysis," *IEEE Transactions on Mobile Computing*, vol. 10, no. 1, pp. 3–15, Jan 2011.

[15] L. Codeca, R. Frank, and T. Engel, "Luxembourg sumo traffic (lust) scenario: 24 hours of mobility for vehicular networking research," in *IEEE Vehicular Networking Conference (VNC)*, Dec 2015, pp. 1–8.

[16] VehicularLab. University of luxembourg. [Online]. Available: http://vehicularlab.uni.lu

[17] J. Petit and R. Ansari, "V2X Validation Tool," https://bitbucket.org/onboardsecurity/dsrcvt, BlackHat 2018.