







Article

Misbehavior-Aware On-Demand Collaborative Intrusion Detection System Using Distributed Ensemble Learning for VANET

Fuad A. Ghaleb ^{1,2,*}, Faisal Saeed ³, Mohammad Al-Sarem ³, Bander Ali Saleh Al-rimy ^{4,*},
Wadii Boulila ^{3,5}, A. E. M. Eljialy ⁶, Khalid Aloufi ³ and Mamoun Alazab ⁷

¹ School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia, Johor 81310, Malaysia

² Department of Computer and Electronic Engineering, Sana'a Community College, Sana'a 5695, Yemen

³ College of Computer Science and Engineering, Taibah University, Medina 344, Saudi Arabia; fsaeed@taibahu.edu.sa (F.S.); mohsarem@gmail.com (M.A.-S.); wadii.boulila@riadi.rnu.tn (W.B.); koufi@taibahu.edu.sa (K.A.)

⁴ Faculty of Business and Technology, UNITAR International University, Selangor 47301, Malaysia

⁵ RIADI Laboratory, National School of Computer Sciences, University of Manouba, Manouba 2010, Tunisia

⁶ Department of Information Systems, College of Computer Engineering and Sciences, Prince Sattam bin Abdulaziz University, Al-Kharj 11942, Saudi Arabia; ae.mohammed@psau.edu.sa

⁷ College of Engineering, IT & Environment, Charles Darwin University, Northern Territory 0810, Australia; mamoun.alazab@cdu.edu.au

* Correspondence: abdulgaleel@utm.my (F.A.G.); bnder321@gmail.com (B.A.S.A.-r.)

Received: 8 July 2020; Accepted: 25 August 2020; Published: 1 September 2020



Abstract: Vehicular ad hoc networks (VANETs) play an important role as enabling technology for future cooperative intelligent transportation systems (CITSs). Vehicles in VANETs share real-time information about their movement state, traffic situation, and road conditions. However, VANETs are susceptible to the cyberattacks that create life threatening situations and/or cause road congestion. Intrusion detection systems (IDSs) that rely on the cooperation between vehicles to detect intruders, were the most suggested security solutions for VANET. Unfortunately, existing cooperative IDSs (CIDSs) are vulnerable to the legitimate yet compromised collaborators that share misleading and manipulated information and disrupt the IDSs' normal operation. As such, this paper proposes a misbehavior-aware on-demand collaborative intrusion detection system (MA-CIDS) based on the concept of distributed ensemble learning. That is, vehicles individually use the random forest algorithm to train local IDS classifiers and share their locally trained classifiers on-demand with the vehicles in their vicinity, which reduces the communication overhead. Once received, the performance of the classifiers is evaluated using the local testing dataset in the receiving vehicle. The evaluation values are used as a trustworthiness factor and used to rank the received classifiers. The classifiers that deviate much from the box-and-whisker plot lower boundary are excluded from the set of the collaborators. Then, each vehicle constructs an ensemble of weighted random forest-based classifiers that encompasses the locally and remotely trained classifiers. The outputs of the classifiers are aggregated using a robust weighted voting scheme. Extensive simulations were conducted utilizing the network security laboratory-knowledge discovery data mining (NSL-KDD) dataset to evaluate the performance of the proposed MA-CIDS model. The obtained results show that MA-CIDS performs better than the other existing models in terms of effectiveness and efficiency for VANET.

Keywords: misbehavior detection; vehicular ad hoc network; VANET; collaborative intrusion detection system; distributed ensemble learning

1. Introduction

Vehicular ad hoc networks (VANETs) are considered an enabling technology for the future cooperative intelligent transportation systems (CITSs) that improves road safety and traffic efficiency as well as provides passenger comfort [1,2]. With the enhanced development of the standards of the Internet of things (IoT), there are different new applications, such as Internet of vehicles (IoV) in which vehicles become the Internet carrier [3–5]. Vehicles in VANETs cooperate and share their sensor information that are enabling a wide range of applications for making safer roads and efficient transportation and providing cheaper internet connectivity [5–7]. However, the VANET environment is highly dynamic with rapidly changing topology, in which the vehicles are varying in speeds and density, which hinders the seamless exchange of the information among vehicles. The problem exacerbates as vehicles run in harsh environment where the communication and sensing quality is adversely affected by the surrounding dynamic and noisy environment. This harsh vehicular environment makes monitoring user activities in VANETs a challenging task, which opens the door for many types of attacks. Moreover, the decentralized nature of VANET makes it vulnerable to several types of attacks such as active interfering, passive eavesdropping, and others [2,8–10]. Cybercriminals can disturb VANET operations and launch many types of attacks that might lead to accidents, congestions, and disruption of the network activities. Therefore, security is a major concern of VANET due to the potential consequences on people lives and economical activities.

Many solutions have been proposed to protect vehicles from being a target of cyberattacks. Cryptographic techniques such as digital signature, authentication, and encryption have been widely used as a first line of defense to prevent many types of external attacks. However, these preventive measures are inadequate for protection against the insider attacks. Due to the cooperative nature of VANET, malicious nodes or intruders can still perform malicious activities such as denial of service, vehicle hijacking, information leakage, manipulation of information, the sharing of misleading information, etc. Therefore, intrusion detection systems have been proposed as the second line of defense to detect and thwart the intrusion malicious vehicles [11–14]. However, due to inherent features of VANETs and the harsh and dynamic environment, the traditional intrusion detection systems (IDSs) that were designed for other wireless networks such as wireless LAN and WSN are not directly applicable for VANETs [2,12–15]. The high mobility, varying density, and network size introduce new vulnerabilities and challenges when applying IDSs on VANETs [11,16].

Recently, there have been several attempts to design IDSs for VANETs [11,17,18]. Different approaches of IDS solutions have been suggested for VANETs including anomaly, signature-based, hybrid, etc. Different IDS architectures also have been proposed such as centralized, cluster, decentralized, distributed, cooperative, and collaborative IDSs. However, due to the cooperative nature of VANET, many of the recent proposed IDSs rely on the collaboration between vehicles to detect the intruders [2,8,16]. In the cooperative IDS (CIDS), vehicles share knowledge related to their detection experiences to help vehicles in the vicinity to detect the intruders more accurately. For instance, authors in [2] found that distributed machine learning is an appropriate scalable method for collaborative detection in VANETs and is used for improving the detection accuracy by sharing knowledge and classifying adversarial behaviors using local datasets. In addition, authors in [8] proposed a multi decision intelligent detection model that considers the wireless and mobile nature of VANET to increase the detection accuracy and reduce the overhead. Although such a type of cooperation can be effective for VANETs, it is vulnerable to the misbehaving vehicles that share false or fake knowledge about known or unknown attacks. Vehicles may share misleading information to degrade the detection efficacy. Therefore, a misbehavior-aware (MA) IDS solution is needed to identify the misbehaving vehicles and exclude them from the set of the collaborators. Moreover, existing CIDS models rely on a simple voting scheme (majority win scheme). Unfortunately, such a scheme is vulnerable to colluding attacks such as botnet, where attackers collude to send misleading information and disrupt the IDS system. Therefore, improving the detection performance of the CIDS models and protecting VANETs from cybercrimes is an essential security requirement in VANETs.

To this end, this paper proposes a misbehavior-aware on-demand collaborative IDS model (MA-CIDS) using distributed ensemble learning. Each vehicle uses its local data to build a local IDS classifier based on a distributed random forest algorithm. Then, on demand, each vehicle sends its locally trained IDS to the vehicles in its vicinity. Unlike existing ensemble models, the final decision of the proposed MA-CIDS model is taken using an improved and robust weighted voting scheme. To design a robust weighted voting system, the performance measures in terms of precision and recall of the locally trained IDS are shared among vehicles. These measures are obtained by testing the locally trained IDS classifier based on a testing dataset in each vehicle. These measures are used as a belief factor to weight the output of the classifier in the voting system. To reduce the contribution of the suspicious vehicles in the voting and remove the misbehaving vehicles from the sets of collaborators, each vehicle evaluates the performance of the IDS classifiers received from neighboring vehicles using its testing dataset. The evaluation results are used to achieve two tasks. Firstly, they are used to penalize the belief factor of the neighboring vehicle and adjust the contribution of the neighboring vehicle in the voting system. Secondly, they are used as input features for the box-and-whisker plot method to detect the misbehaving vehicles. Vehicles that deviate much from the lower boundary of the box-and-whisker plot are excluded from the set of the collaborators. Finally, to construct the MA-CIDS model, each vehicle constructs its own ensemble of weighted random forest-based classifiers, which contains both the locally and remotely trained classifiers. The outputs of the classifiers are aggregated using a robust weighted voting scheme.

The rest of the paper is organized as follows. The related work is reviewed in Section 2. The model description is elaborated in Section 3. Section 4 explains the conducted experiments and provides the results. The results are discussed and analyzed in Section 5. Finally, Section 6 concludes the study and provides the future work.

2. Related Work

Securing VANETs has attracted great interest of many researchers during the last years [7,17,19–26]. VANET is vulnerable to many security issues that can disrupt the functionality of these applications. The intrusion detection system for VANETs aims to detect internal as well as external attacks with high accuracy [16,27,28]. According to the previous used methods, IDSs can be categorized into signature-based IDS, anomaly-based IDS, and others [29]. Recently, several approaches have been proposed, among which the machine learning-based approaches are the most promising [2,21,28,30]. However, most pure IDSs produce a large number of false positives and low detection accuracy. In the literature, several researchers investigated the ability to use machine learning (ML) for intrusion detection. ML has demonstrated promising results in the field of IDS. Several authors have applied ML-based models to learn complex patterns and behavior from the collected data. Unlike rule-based IDS [31–34] and expert system-based IDSs [35–38], which cannot detect emergence intrusions, ML-based IDSs enable vehicles to extract distinct features either from new coming messages and intrusions.

Machine learning methods were applied widely to solve IDS issues in different networks. For instance, an early study by [21] applied the random forest (RF) method to build automatically patterns of intrusions. Then, intrusions were detected by matching the network activities against the built patterns. To evaluate the performance of this model, the authors used the knowledge discovery data mining (KDD)'99 dataset. To handle the problem of imbalanced data, both the downsampling and oversampling methods were applied. After that, the experiment was carried out on the WEKA environment using 66% samples as a training set and 34% as a testing set. The experimental results showed that this approach achieved a high-detection rate of 94.7% with a low false-positive rate of 2%. Similarly, authors in [2] applied an RF classifier to detect intrusion behaviors among high-speed traffic data. The authors adapted RF to the Apache Spark distributed processing system. The obtained results showed that the framework can enhance the real-time detection of network intrusion with a large capacity and high speed. In [36], the combination of the k-nearest neighbor (K-nn) method with a genetic algorithm was used. The author tested the model using a handmade

dataset with 35 packet-based features. The training dataset contained well-balanced instances (600 normal instances and 600 attack instances), whereas the testing set contained 100 instances for both class labels. Thirty random chromosomes were generated for the initial population and trained using the training set. After training the model, all 35 features were fed into the k-nn algorithm. According to the experiments, the best overall accuracy of known attack was 97.42% for which only the top 19 features were considered and an accuracy of 78% with the top 28 features of unknown attack was obtained. In addition, Al-Jarrah et al. [39] investigated the impact of feature selection techniques on the performance of RF. For this purpose, they combined RF with forward and backward ranking features' selection techniques. In terms of the used dataset, they filtered out the original KDD'99 and deleted the redundant data. Several preprocessing techniques including normalization, discretization, and balancing techniques were applied. To find the most important features among the 41 features of the original KDD'99 dataset, they applied forward selection ranking (FSR) and backward elimination ranking (BER) algorithms with an RF classifier. The experimental results showed that the RF-FSR technique was suitable for large-scale network IDSs. In addition to RF and K-nn methods, the Support Vector Machine (SVM) was used in [40], the incremental SVM was used in [41], and the naïve Bayes and decision trees were used in [42] to detect malicious network intrusions. An extensive review of the applied data mining techniques in IDSs can be found in [18,29,43].

In addition to ML techniques for IDS, many hybrid IDSs have been proposed. Kim et al. [44] proposed a two-stage hybrid IDS method that hierarchically integrates a misuse detection model and an anomaly detection model. Firstly, the misuse detection model was developed based on the C4.5 decision tree classifier. Then, several models of one-class SVM were used for the split subsets. The results showed enhancements in terms of the detection rate and false positive rate comparing to the conventional methods. In addition, Al-Yaseena et al. [45] proposed a multi-level hybrid IDS, which employs both SVM and extreme learning machine to efficiently detect known and unknown attacks. To improve the performance of the used classifiers, they also proposed a modified k-means algorithm to produce a small and representative training dataset. The findings show that the approach reduced the training time and improved the overall performance of IDS. The hybridization of k-means and decision trees, namely C4.5, was also applied in [46] where the model was applied for classifying anomalous and normal activities in a computer system. In addition, Thaseen and Kumar [47] applied the chi-square feature selection to reduce the dimensionality and to find the optimal subset among all the attributes. Then, the data with the selected attributes were used for training multi-class SVM. In [29], the authors proposed a hybrid IDS based on the stacking ensemble of C5.0 and one-class SVM. This model was evaluated using network security laboratory-knowledge discovery data mining (NSL-KDD) and Australian Defence Force Academy datasets (ADFA) datasets. The results showed an enhancement in terms of detection rate and false-alarm rates.

Recently, several works have been published related to ML for intrusion detection in VANET. For instance, Shams et al. [48] proposed an approach combining the promiscuous mode for data collection and SVM for IDS in VANET. They aimed to analyze data to create a trust value for vehicles on the network as trust aware SVM-based IDS. The main idea was to guarantee that vehicles within the network have a complete idea about activities of their next hop, which will help to maintain high-performance in case of attacks. In [49], the authors designed an ML-based intrusion detection approach to detect intruders globally and locally in VANETs. They used an ANN technique to secure cluster heads, a light-weight SVM to identify malicious multi-point relays. Results depicted that the used approach was more robust and trust-worthy compared to existing ML-based techniques. In addition, Zhou et al. [50] designed a distributed collaborative IDS based on invariant to detect betray attacks in VANET. This approach was based on four steps: (1) distributed framework to store the collected big data; (2) reputation-based communication method to ensure reliable communication using a global reputation state, traffic density, and link life; (3) analysis of dynamic behavior to discover the normal driving characteristics of vehicles; and (4) stochastic petri net was used to design the system

states and their evolution. Results depicted good performances in detecting attacks compared with existing methods.

3. The Proposed MA-CIDs Model

In this section, the architecture of the proposed MA-CIDS model is described. Figure 1 illustrates the components of the proposed model. As shown in Figure 1, the MA-CIDS encompasses four main phases, namely the individual IDS construction phase, neighboring IDS classifiers and metadata exchanging phase, misbehavior evaluation phase, and collaborative classifier construction phase. The output of each phase is used as input to the next phase. The detailed description of each phase is presented in the following subsections.

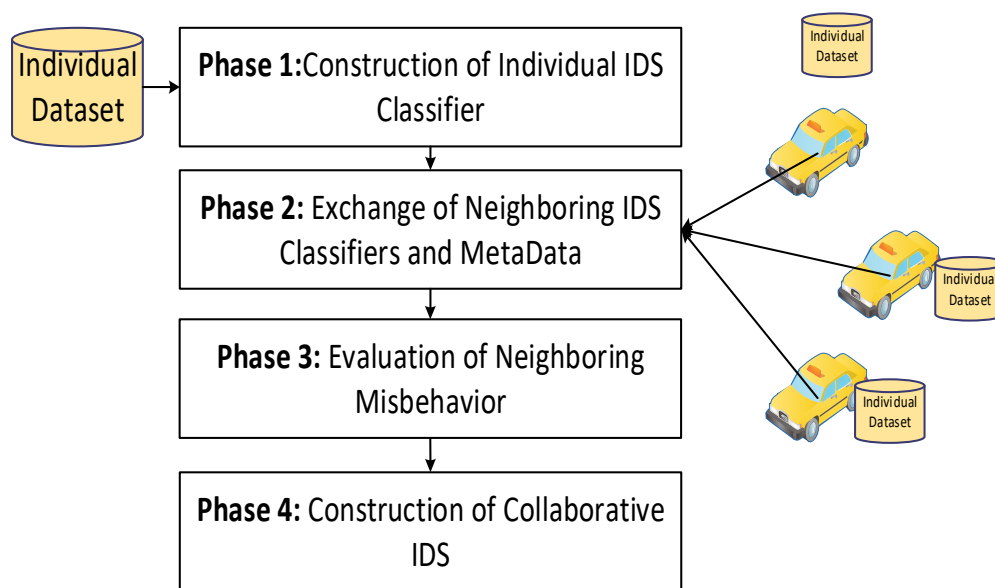


Figure 1. Architecture of the misbehavior-aware on-demand collaborative intrusion detection system (MA-CIDS) model.

3.1. Phase 1: Individual IDS Construction

In this phase, each vehicle (subject vehicle) constructs its local IDS classifier using local data collected by monitoring and auditing its network activities as well as the neighboring vehicles' activities. The data contains attributes extracted from packets' headers and the communication protocols used through the communication. Because the data is collected in a harsh communication environment and contains both categorical and numerical data, the preprocessing is needed. The data is preprocessed by removing incomplete records, encoding categorical data, and then standardization. Then, each vehicle uses a feature selection algorithm to select the more important features. Then, each vehicle split the collected and preprocessed data into two sets, one for local model training and the other for testing. Next, a machine learning algorithm, namely the random forest algorithm, is used to construct an ensemble of local classifiers. Random forest (RF) was selected among many algorithms due to its robustness to noisy data and good fit with even non-linear data such as VANET data. In addition, RF showed its superior performance compared to other classifiers for VANET data as reported by many researchers [2,7,14,21]. Finally, the performance of the trained local classifiers, also called the individual IDS classifier, was evaluated using the testing datasets. The performance evaluation metrics such as accuracy, precision, recall, and F1 score are used as the metadata for the locally trained classifier. A vehicle decides to use or share its locally trained classifier based on the achieved performance on the testing dataset. Figure 2 shows the methods used to construct the IDS classifier.

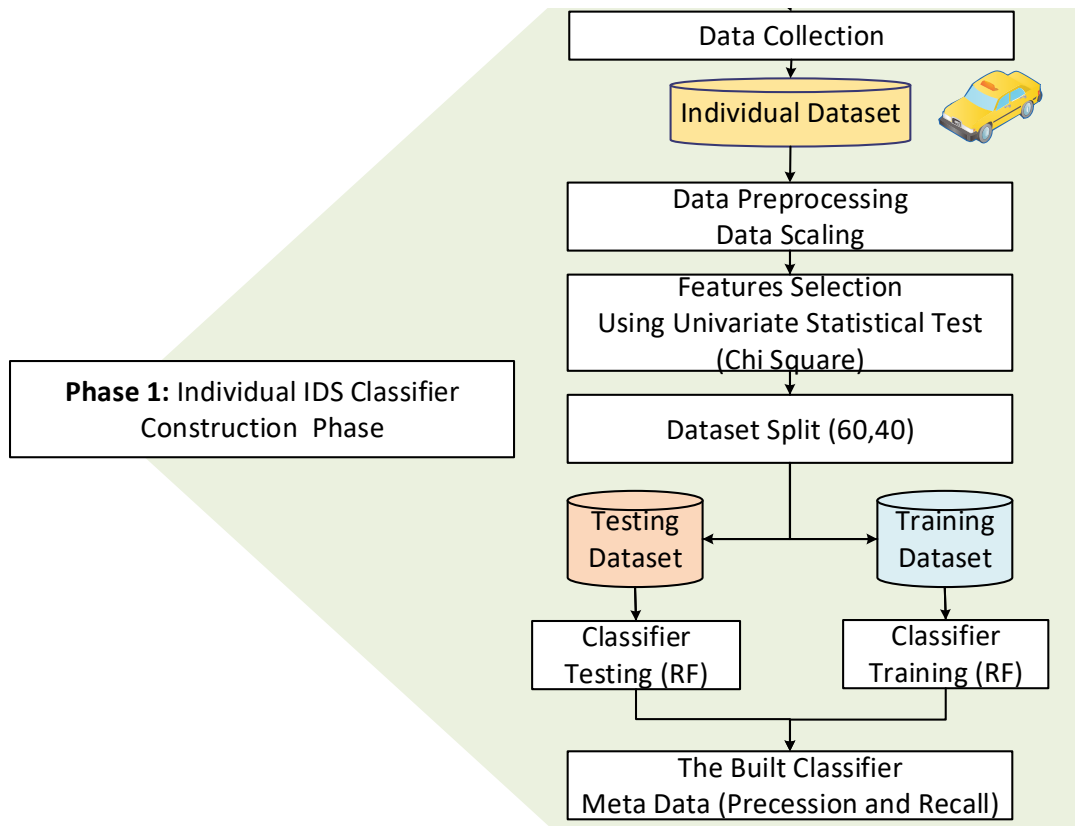


Figure 2. Sharing decision algorithm.

3.2. Phase 2: Neighboring Classifiers and Metadata Exchanging

In this phase, the collaboration among neighboring vehicles is established. A vehicle communicates with the vehicles in their vicinity in one-hop communication. Each vehicle shares the trained classifier as well as the metadata with the vehicles in its vicinity. To avoid communication overhead, an on-demand sharing strategy is proposed. Algorithm 1 shows the proposed on-demand IDS classifier sharing algorithm, which is used to exchange the locally trained IDS classifiers and their metadata among one-hop communication vehicles. The symbols that are present in Algorithm 1 are described in Table 1. It is on-demand because vehicles send a request for collaboration if the performance of their classifiers fall under a specific threshold. The algorithm suppresses the misbehaving vehicle that sends too many sharing requests in a small period of time. Each vehicle decides whether it needs to collaborate and update its IDS classifiers or not. The decision of updating the IDS model is taken based on the performance of the collaborative IDS classifier that is constructed in phase 4. If it is needed, then the vehicle sends a message asking for the collaboration. Each vehicle in the vicinity receives the collaboration request that is issued by the subject vehicle and uses Algorithm 1 to decide whether to respond by sharing its locally trained IDS classifier and its metadata or not.

Algorithm 1: On-Demand IDS Classifier Sharing Algorithm

Input: $IDS_{local(s)}, N_{req}, T_{elabsed}, N_{coll}, T_{Ncoll}, F_{score}, T_{Fscore}$

Output: $D_{local(s)}$.

Sender Vehicle

- 1: **if** $N_{coll} < T_{Ncoll}$ **and** $F_{score} < T_{Fscore}$ **do**
- 2: *broadcast (collaboration request message)*

Receiver Vehicle

- 1: $D_{local(s)} = False$
 - 2: **if** *request for colobration is recieved* **do**
 - 3: $A_{(v_i)} \leftarrow audit(req_{id} ++, vehicle_{id}, time)$
 - 4: $t_{(v_i)} \leftarrow get_time_elabsed(A_{(v_i)}, vehicle_{id}): v_i \in V$
 - 5: $req_{count} \leftarrow get_request_count(A_{(v_i)})$
 - 6: **if** $req_{count} < N_{req}$ **and** $t_i < T_{elabsed}$ **Then**
 - 7: $D_{local(s)} = True$
 - 20: **return** $D_{local(s)}$
-

Table 1. Symbols description.

Symbol	Description
IDS_{local}	The local trained classifier
$T_{elabsed}$	Time threshold for resending the local classifier
N_{req}	Threshold of number of sharing requests per area
t_i	Elapsed time since last approved sharing
req_{count}	number of sharing requests per area
$A_{(v_i)}$	Set of the number sharing requests
$D_{local(s)}$	Sharing decision true or false

3.3. Phase 3: Neighboring Misbehavior Evaluation

In this phase, each vehicle evaluates the received local IDS classifiers from neighboring vehicles using its local testing dataset. The precision $p_{test(i)}$, and the recall $r_{test(i)}$, obtained by testing the neighboring classifier IDS_i on the local testing dataset of the subject vehicle are used as penalty value of the neighboring classifier. Meanwhile, F1 score $f_{test(i)}$ resulted from evaluating the neighboring classifier IDS_i using the local testing dataset is used as misbehavior indication. The precision $p_{test(i)}$ is used as the penalty if the instance tested negative by the neighboring classifier of vehicle i , while the recall $r_{test(i)}$ is used as the penalty if the instance tested positive. Then, each vehicle ranks the neighboring classifiers by multiplying the claimed performance, namely the precision $p_{neighbor(i)}$ and the recall $r_{neighbor(i)}$, that are reported by the neighboring vehicle, by the penalty factor, namely the precision $p_{test(i)}$, and the recall $r_{test(i)}$, that are resulted from applying the neighboring classifier on the local testing dataset. Finally, the box-and-whisker plot is computed to detect the misbehaving vehicles. The box-plot is a non-parametric statistical tool that can summarize a statistical variable without the need to know its underline distribution. If the data point is located outside the box plot boundaries, it is considered inconsistent with neighboring values. Thus, a classifier with an F1 score of the local testing dataset that fall under the lower adjacent value of the box-and-whisker plot is considered a misleading classifier and should be removed from the set of collaborators $C_{neighbors}$. However, the lower adjacent value (LL) of the box-and-whisker plot decreases as the number of intruders increases, yielding to their inclusion in the set of celebrators. Nevertheless, their impact on the overall decision is low due to their low F1 scores, which are used as weights in the final decision. Algorithm 2 illustrates the neighboring misbehavior evaluation algorithm. The symbols that are presented in Algorithm 2 are described in Table 2.

Algorithm 2: Misbehavior Evaluation Algorithm

Input: $IDS_{neighbors}, P_{neighbors}, R_{neighbors}$
Output: $C_{neighbors}$
1: $\forall IDS_i \in IDS_{neighbors}$ do
2: $p_{test(i)}, r_{test(i)}, f_{test(i)} \leftarrow test(IDS_i)$
3: $p_{neighbor(i)} \leftarrow get(IDS_i, P_{neighbors}) : p_{neighbor(i)} \in P_{neighbors}$
4: $r_{neighbor(i)} \leftarrow get(IDS_i, R_{neighbors}) : r_{neighbor(i)} \in R_{neighbors}$
5: $p_i = p_{test(i)} * p_{neighbor(i)}$ for normal class
6: $r_i = r_{test(i)} * r_{neighbor(i)}$ for abnormal class
7: $w_i \leftarrow (p_i, r_i)$
8: $C_{neighbors} \xleftarrow{append} (IDS_i, w_i, f_{test(i)})$
9: Box – Plot ($F_{test} : F_{test} \in T_{neighbors}$) = $\begin{cases} \mu = (Q_1 + Q_3)/2 \\ IQR = (Q_3 - Q_1) \\ UL = Q_3 + 1.5 IQR \\ LL = Q_3 - 1.5 IQR \end{cases}$
10: $\forall IDS_i \in C_{neighbors}$ do
11: if $f_{test(i)} < LL$ Then
 $C_{neighbors} \xleftarrow{remove} (IDS_i, p_i, r_i, f_{test(i)})$
12: return $C_{neighbors}$

Table 2. Symbols description.

Symbol	Description
$IDS_{neighbors}$	Set of all received classifiers
$P_{neighbors}$	The corresponding set of all precisions of the $IDS_{neighbors}$ as reported by collaborative vehicles
$R_{neighbors}$	The corresponding set of all recalls as reported by collaborative vehicles
IDS_i	The classifier shared by vehicle i
$p_{test(i)}, r_{test(i)}, f_{test(i)}$	The precision, recall, and F1 score of the IDS_i , respectively.
F_{test}	The corresponding set of F1 scores of the $IDS_{neighbors}$ as tested by receiver vehicle
$C_{neighbors}$	Set of all evaluated $IDS_{neighbors}$ with their ranks ($IDS_i, p_i, r_i, f_{test(i)}$)
μ	The mean of the box-and-whisker plot
UL, LL	The upper adjacent value, and lower upper adjacent value of the box-and-whisker plot
Q_1, Q_3, IQR	Q_1, Q_3 , are the first and third quartile, and IQR is the entire quartile range

3.4. Phase 4: Collaborative IDS Construction

In this phase, the construction of the collaborative IDS classifier is described. The construction is achieved into two steps. Firstly, each vehicle uses the set of collaborators $C_{neighbors}$ obtained from the previous phase to construct an ensemble-based classifier. Secondly, each classifier is given a pair of weights $w_i (p_i, r_i)$ calculated in the previous phase (see lines 5–7 in Algorithm 2), p_i is used as a weight if the instance tested negative (normal), and r_i is used if the instance tested positive (intruder). The final classification output of the ensemble classifier is expressed as follows:

$$D_f = \begin{cases} 0 & \frac{\sum(p_i \times (1-d_i))}{\sum p_i} > \frac{\sum(r_i \times d_i)}{\sum r_i} \\ 1 & \frac{\sum(p_i \times (1-d_i))}{\sum p_i} \leq \frac{\sum(r_i \times d_i)}{\sum r_i} \end{cases} \quad (1)$$

where D_f is the final decision by the ensemble MA_CIDS, d_i is the decision made by the classifier received from the neighboring vehicle i , p_i is the weight of the classier sent by vehicle i when the output is negative ($d_i = 0$), and r_i is the weight of the classier sent by the vehicle i when the output is positive

($d_i = 1$). Initially, when there are no collaborators to construct the ensemble of classifiers, vehicles use their own locally trained classifiers to detect the intruders.

4. The Experimental Design and Results

This section describes the setup of the experimental environment in which the implementation of the proposed model and techniques was conducted. Then, the experiments and dataset used in this study were described in detail. The performance metrics were also explained. The experimental results of each technique were presented, including the comparison with the previous studies.

4.1. The Experimental Environment Setup

In this study, the simulation of urban mobility (SUMO) was used to simulate vehicle mobility. SUMO is computer software that is used to generate vehicular traffic, and by which vehicles' speed, types, and behavior and density can be configured. Five traffic scenarios with different density were created. In each scenario, different traffic density in terms of vehicles per kilometer was used, which was selected from the following list {10, 20, 30, 40, and 50}. Random vehicle types, speed, and behavior were used in each scenario to simulate vehicle mobility along 5 km road length with two lanes and maximum vehicle speed set to 80 km/h. The generated vehicle trajectories were replayed under the Python programming environment. The NSL-KDD was used to represent vehicle network-traffic. The NSL-KDD is currently the best available dataset for benchmarking of different network based IDSs in VANET [51–55]. Table 3 illustrates the types of attacks that were presented in the datasets. The NSL-KDD dataset was divided into small chunks and distributed among vehicles. Each vehicle divided its local dataset into a training set and testing set, 60% for training and 40% for testing. Then, each vehicle extracted the importance features and used them to train the machine learning-based classifier (this study used RF, XGBboost, and SVM algorithms). Vehicles exchanged their trained classifiers and the models' metadata using a simulated network protocol, namely 802.11p [56,57]. The transmission range was set to 1000 m in an omnidirectional [56].

Table 3. Symbols description.

Attacks in Dataset	Attack Type
Denial of Service (DoS)	Back, Land, Neptune, Pod, Smurf, Teardrop, Mailbomb, Processtable, Udpstorm, Apache2, Worm
Probe	Satan, IPswEEP, Nmap, PortswEEP, Mscan, Saint
R2L	Guess_password, Ftp_write, Imap, Phf, Multi, hop, WareZmaster, Xlock, Xsnoop, Snpmpguess, Snpmpgetattack, Httpptunnel, Sendmail, Named
U2R	Buffer_overflow, Loadmodule, Rootkit, Perl, Sqlattack, Xterm, Ps

4.2. The Performance Measures

To evaluate the performance of the proposed collaborative IDS model (MA-CIDS), six performance measures were used, namely, classification accuracy, precision, recall (the detection rate), F1 score, false positive rate (FPR), and false negative rate (FNR). These measures are commonly used by researchers to evaluate the intrusion detection systems in VANET [50]. The classification accuracy is the percentage of the truly classified instances of the testing datasets. The precision measures the correctly classified normal instances over the total normal instances in the testing datasets, while the recall metric measures the correctly classified intruders over the total number of the intruders in the testing dataset. The F1 score measures the overall performance of the model in terms of trade-off between precision and recall. If both precision and recall are high, the F1 score is high. False positive rate (FPR) measures the percentage of normal instances that are incorrectly classified, while the false negative rate (FNR) is the percentage of the intruders that are incorrectly classified.

4.3. Experimental Results

In this section, the performance of the proposed misbehavior-aware collaborative IDS model (MA-CIDS) in terms of the classification accuracy, FPR, FNR, and F1 score is presented. MA-CIDS encompassed ensemble IDS classifiers that were trained with local datasets and shared by neighboring vehicles. Each vehicle divided its local dataset into a training set and testing set, 60% for training and 40% for testing. Then, each vehicle extracted the importance features and used them to train machine learning-based classifiers. Each vehicle used the testing set to measure the classification performance of each classifier based on the extracted features. The testing dataset was used for two purposes. The first was to evaluate the performance of the locally trained classifier and the second was to evaluate the performance of the neighboring shared classifiers. The classification outputs of all classifiers were aggregated using a weighted average function. The performance of each classifier on the local testing data, namely the precision and recall, were used as weights for both the normal and anomaly class, respectively. The weights of the shared classifiers were penalized by multiplying them by the precision and recall that were obtained by testing those classifiers on the host testing dataset.

Figure 3 and Table 4 shows the comparison results between the proposed misbehavior-aware collaborative IDS model (MA-CIDS) and traditional cooperative IDS model (CIDS) [51–55] in the presence of misbehaving vehicles which shared a misleading IDS classifier. It was assumed that 10% of the participants were misbehaving vehicles. The random forest algorithm, the gradient boosting algorithm (XGBoost), and support vector machines were tested as the base classifiers for the MA-CIDS and also used to compare with the traditional CIDS model. It was observed that the classification accuracy, FPR, FNR, and F1 score of the proposed MA-CIDS were higher than the traditional CIDS for RF, XGBoost, and SVM classifiers. The MA-CIDS (RF), which used the RF algorithm as the base classifier, outperformed both MA-CIDS (XGBoost) and the MA-CIDS (SVM), in which the XGBoost and SVM algorithms were used as the base classifiers, respectively. Although the MA-CIDS (SVM) had lower FNR (0%), its FPR was high (11%) compared with the MA-CIDS (RF) that achieved low FNR (2%) and FPR (4%). Although both ensemble learning-based algorithms, i.e., MA-CIDS(RF) and MA-CIDS (XGBoost), had similar false positive rates (FPR; 4%), MA-CIDS (RF) achieved lower false negative rate FNR (2%) as compared to the MA-CIDS (XGBoost).

Figure 4 shows a detailed comparison of the impact of the number of collaborators on the performance of all tested models. It could be observed that the classification accuracy (in Figure 4a), FPR (in Figure 4b), FNR (in Figure 4c), and F1 score (in Figure 4d), of the proposed MA-CIDS were higher than the traditional CIDS for both RF and SVM classifiers. The performance of all tested models improved as the number of collaborators increased. The MA-CIDS (RF) beat all other tested IDS models. The performance in terms of F1 score of CIDS model that was built using the RF algorithm CIDS (RF) and CIDS (SVM) rapidly increased as more participants were collaborating in the IDS model, while the performance of CIDS model built using the gradient boosting algorithm CIDS (XGboost) slightly increased when the number of participants increased.

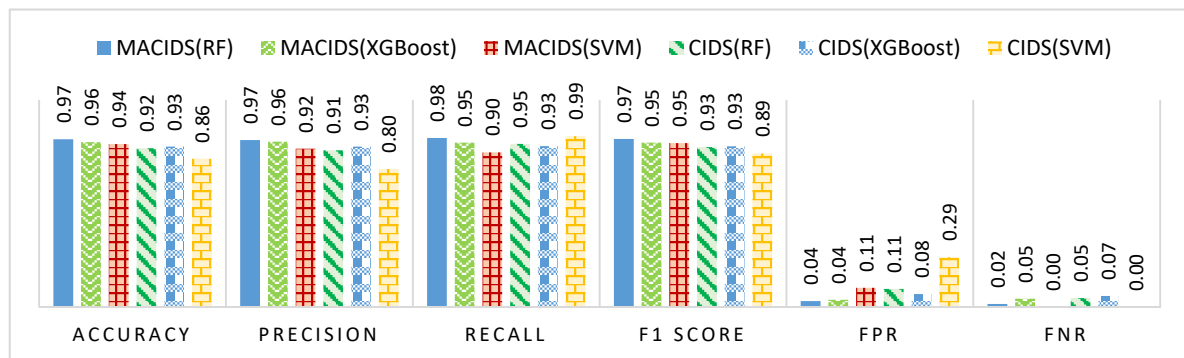


Figure 3. Comparisons of the detection performance.

Table 4. Details of the detection performance.

Tested Model	Precision	Recall	F1 Score	FPR	FNR	
MA-CIDS (RF)	0.97	0.97	0.98	0.97	0.04	0.02
MA-CIDS (SVM)	0.94	0.92	0.90	0.95	0.11	0.00
MA-CIDS (XGBoost)	0.96	0.96	0.95	0.95	0.04	0.05
CIDS (RF)	0.92	0.91	0.95	0.93	0.11	0.05
CIDS (SVM)	0.86	0.80	0.99	0.89	0.29	0.00
CIDS (XGBoost)	0.93	0.93	0.93	0.93	0.08	0.07

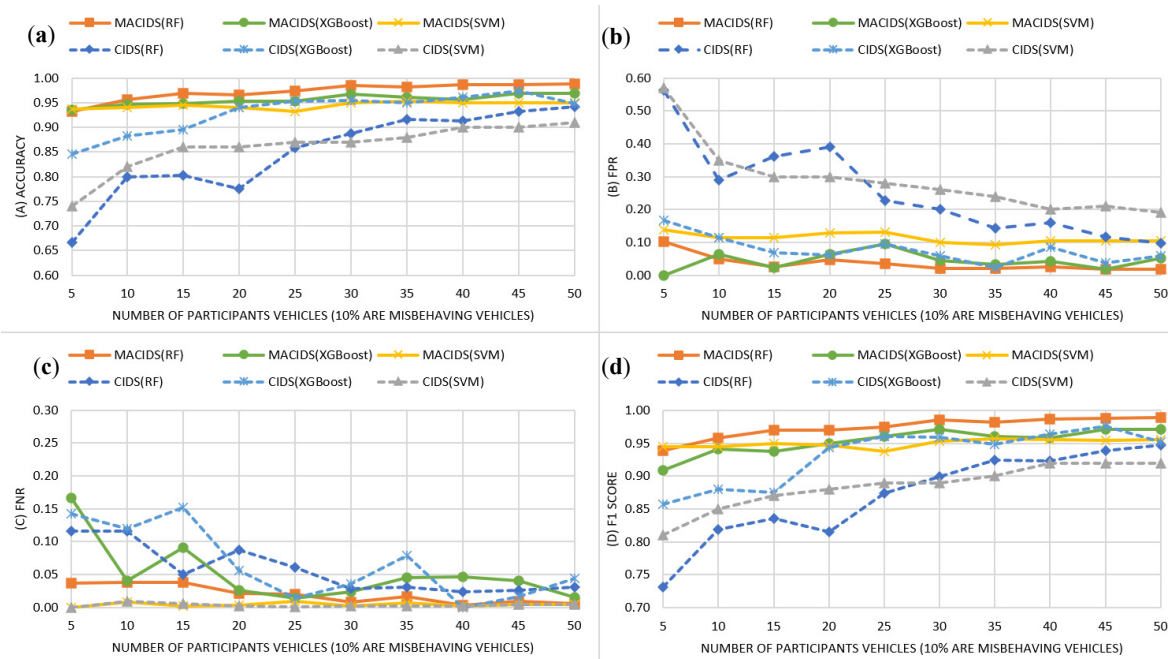


Figure 4. Impact of number of collaborators on the detection performance in terms of (a) Accuracy, (b) False Positive Rate, (c) False Negative Rate, and (d) F1 Score.

To evaluate the impact of increasing the percentage of misbehaving vehicles on the performance of the proposed MA-CIDS(RF) model, experiments with four scenarios were conducted. In each scenario, the number of collaborators was set to one of four numbers (10, 20, 30, 40), and the percentage of misbehaving vehicles was increased from 10% to 40%, with a 10% increment in each run. Figure 5 illustrates the impact of the percentage of misbehaving vehicles on the performance. The X-axis of all sub-figures represents the percentage of misbehaving vehicles, while the Y-axis is the corresponding performance measure, namely the classification accuracy (in Figure 5a), FPR (in Figure 5b), FNR (in Figure 5c), and F1 score (in Figure 5d). It can be observed in Figure 5 that as the percentage of the misbehaving vehicles increased, the performance of the model decreased. However, the model became more robust to the misbehaving vehicles as the number of collaborators increased. For example, in the scenario with 50 collaborators, the overall performance in terms of F1 score was highest among all tested number of collaborators. Nevertheless, the performance slightly decreased when the percentage of misbehaving vehicles increased. The overall performance in terms of F1 score decreased from 99% to 98% when the percentage of misbehaving vehicles were gradually increased from 10% to 40%.

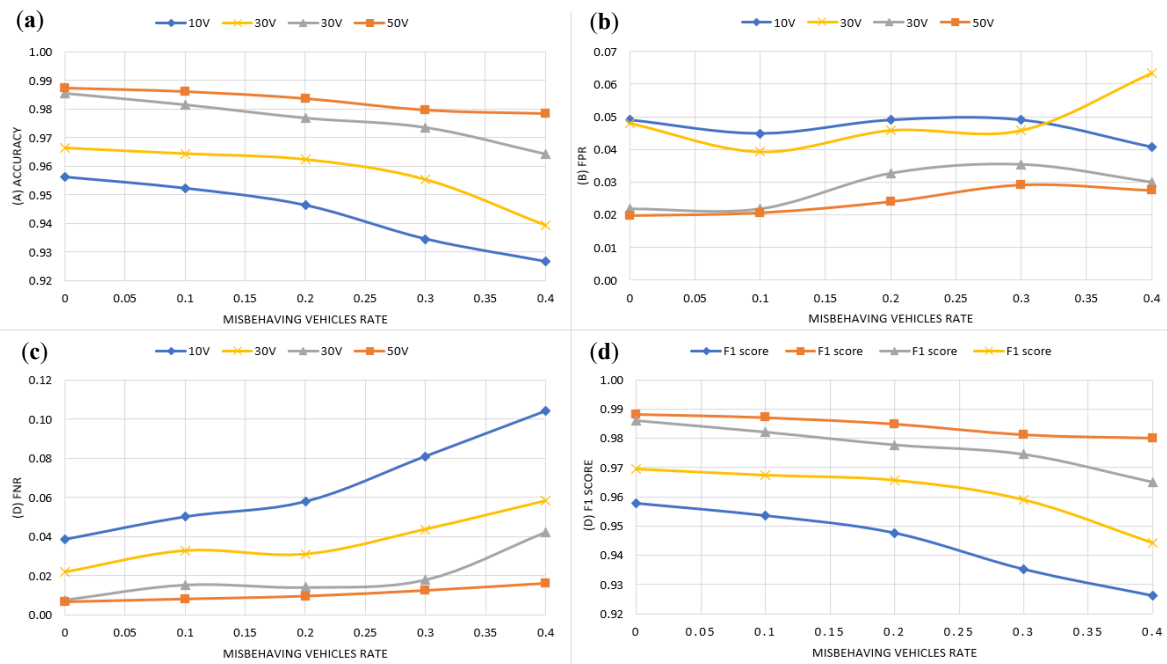


Figure 5. Impact of increasing the percentage of misbehaving vehicles on the detection performance in terms of (a) Accuracy, (b) False Positive Rate, (c) False Negative Rate, and (d) F1 Score.

5. Analysis and Discussion

In this section, the performance of the proposed misbehavior-aware collaborative IDS model (MA-CIDS) is discussed, and the robustness and the reliability under a dynamic environment is analyzed. In terms of model design, the proposed collaborative IDS model shares its local trained classifier as well as its metadata, namely the precision and recall, with vehicles in the vicinity. The metadata (the precision and recall) are obtained from the evaluation of the classifier on the testing dataset in the subject vehicle. As opposed to the existing cooperative IDS models that exchange their classification outputs with the neighboring vehicles, the neighboring vehicle shares their trained classifiers. Exchanging the classifiers is more efficient than frequently sharing the classification output in terms of communication overhead. Given that VANET works in a harsh and dynamic environment, exchanging much data leads to congestions and communication overhead which severely impacts the detection performance. The proposed MDS design is also effective in terms of detection performance as the concept of ensemble classifiers whose decisions are aggregated by a voting mechanism. To make the IDS robust to misbehaving vehicles who share malfunction classifiers or manipulated information (e.g., in case of botnets attacks and colluding attacks), the voting that is proposed in the MA-CIDS model mechanism was designed to penalize the weights of the shared classifiers and reduce their contribution in the final decision. Moreover, the classifiers that have a high contradiction between the reported and tested performance are excluded from the final decision. Figure 3 and Table 4 present the performance gained by the proposed MA-CIDS and the traditional CIDS model.

As shown in Figure 3 and Table 4, the performance of the proposed MA-CIDS model outperforms the traditional CIDS model. The main drawback of the conventional CIDS is that it is vulnerable to the misbehaving vehicles that perform colluding attacks, potent attacks, and Sybil attacks, which share manipulated classification outputs and misleading experiences. Thus, in the presence of 10 percent of misbehaving vehicles, the conventional CIDS models produce high false alarms (11% and 23%) for the RF and SVM classification algorithms, respectively. The overall performance in terms of the F1 score of the conventional CIDS models is 93% for RF and 89% for SVM, which are relatively low compared to the proposed MA-CIDS models that are 97% for RF and 95% for SVM. This is because the proposed model independently evaluates the collaborators using weighted and misbehaving-aware

voting systems, while the CIDS relies on simple voting-based decision making (the majority win voting scheme), which is vulnerable to many types of attacks.

It can also be observed from Figure 3 and Table 4 that the performance of the RF-based models is better than that of SVM models. This is because RF is an ensemble-based method that is robust to complex data. Moreover, RF encompasses several independent decision trees and not one complex decision tree. Furthermore, the SVM algorithm is complex, and many parameters need to be carefully tuned, such as the selection of the appropriate kernel, regularization, and hyper-parameters to fit the dataset in hand. As can be noticed from Figure 4 that SVM works better than RF with small datasets when the number of participants is less than 30. Moreover, the RF-based model is more sensitive to the amount of training data. As can be seen in Figure 4a,d, the overall performance (Figure 4d) and the classification accuracy (Figure 4a) achieved by the SVM is higher than that of RF when the training dataset is small. However, the RF-based model outperforms the SVM when the number of collaborator vehicles increases. The random forest algorithm works based on the concept of the power of crowds, so when the number of the independent classifiers increases, better performance is achieved. Even in the absence of misbehaving vehicles, as can be seen in Figure 5, when the number of misbehaving vehicles is set to zero (see the corresponding values of the X-axis is 0 in Figure 5), the overall performance is improved as the number of collaborators increases.

In the presence of misbehaving vehicles, the decision making of the ensemble model is slightly affected by the percentage of misbehaving vehicles. As can be observed in Figure 5, in all studied scenarios, when the number of misbehaving vehicles increases, the overall performance slightly declines. Moreover, the performance of the conventional CIDS is sensitive to the number of misbehaving vehicles. Even 10% of misbehaving vehicles can severely impact the performance, as can be observed in Figure 3 and Table 4. Accordingly, the proposed MA-CIDS model is more robust to misbehaving vehicles and well protected from the colluding attacks.

Figure 6a shows the performance of the proposed MA-CIDS under different traffic scenarios, while Figure 6b illustrates the corresponding time needed to construct the ensemble classifiers. As can be seen in Figure 6a, the performance in terms of F1 score slightly improves as the traffic density increases. The MA-CIDS with gradient boosting algorithm MA-CIDS (XGBoost) tends to have a fluctuant behavior, while the other studied classifiers, RF and SVM, are more stable. The MA-CIDS (RF) achieves the best performance under all studied scenarios. As can be observed in Figure 6b, the time needed to construct the collaborative model exponentially increases as the traffic density increases. However, this will not violate the detection time requirement as the performance of the proposed model is not affected much by the traffic density (see Figure 6a). Thus, in the case of high traffic density, a vehicle can construct the model with a lower number of collaborators. That is, not all neighboring vehicles need to be included in the construction of the collaborative model as the performance will not be significantly improved. In contrast to the case of low traffic density or in the absence of collaborators, a vehicle uses its last constructed collaborative model or the locally trained model to detect the intruders. To sum up, the proposed MA-CIDS (RF) achieves the best accuracy and it is the most robust for the VANET environment among all the studied classifiers.

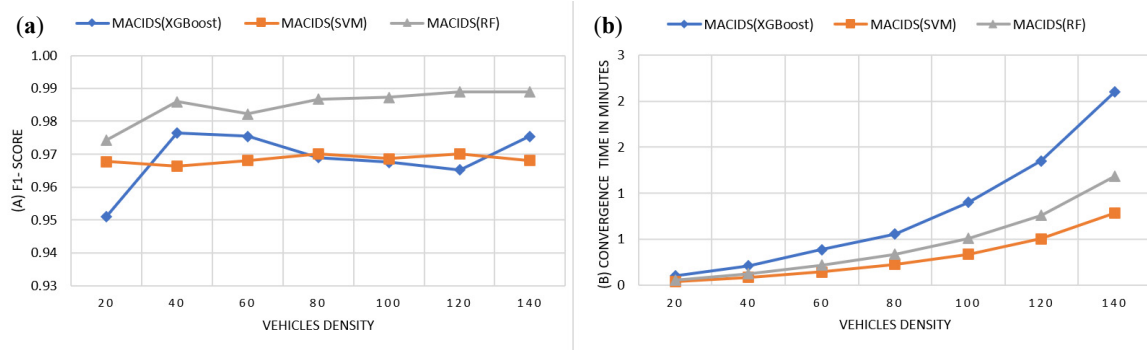


Figure 6. Impact of increasing the vehicle density on (a) the detection performance and (b) convergence time.

6. Conclusions

In this paper, a misbehavior-aware collaborative intrusion detection system (MA-CIDS) is proposed using distributed ensemble learning to improve the efficacy of the VANET CIDS models. An efficient sharing scheme is presented to improve the shared knowledge and reduce communication overhead. Vehicles on-demand share their locally trained classifiers using a random forest algorithm associated with the classifier performance measures. The performance of the shared classifiers is evaluated using a local testing dataset in the received vehicle and is used as the trustworthiness factor. The classifiers that deviate much from the box-and-whisker plot lower boundary are excluded from the set of the collaborators. Finally, vehicles construct ensembles of weighted random forest-based classifiers encompassing both the locally and remotely trained classifiers. The outputs of the classifiers are aggregated using a robust weighted voting scheme. Extensive simulations were conducted by utilizing the network security laboratory-knowledge discovery data mining (NSL-KDD) dataset to evaluate the performance of the proposed MA-CIDS model. The overall performance in terms of F1 score was 97% with a 4% false positive rate compared to the existing CIDS model, which achieved a 93% F1 score with an 11% false positive rate. The obtained results show that MA-CIDS performs better than the other existing models in terms of effectiveness and efficiency for VANET. In the future, the collaborative IDS model will be investigated with both supervised and unsupervised machine learning techniques.

Author Contributions: Conceptualization, F.A.G. and M.A.; Methodology, F.A.G., M.A.-S. and B.A.S.A.-r.; Software, F.A.G.; Validation, F.A.G. and F.S.; Formal analysis, F.A.G., F.S., M.A.-S., B.A.S.A.-r., K.A. and W.B.; Investigation, M.A.-S., M.A. and W.B.; Data curation, F.S., M.A.-S., B.A.S.A.-r. and W.B.; Writing—original draft, F.A.G., F.S. and M.A.-S.; Writing—review & editing, F.A.G., F.S., M.A.-S., B.A.S.A.-r., W.B. and A.E.M.E.; Visualization, F.A.G., F.S. and M.A.-S.; Supervision, F.S., K.A. and M.A.; Project administration, F.S. and M.A.; Funding acquisition, A.E.M.E. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: Our deep gratitude is extended to the Ministry of Higher Education (MOHE), Malaysian International Scholarship (MIS), and Cybersecurity Research Lab, School of Computing, Faculty of Engineering at the Universiti Teknologi Malaysia (UTM) for their unlimited support throughout this study.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Pathan, A.S.K. (Ed.) *Security of Self-Organizing Networks: MANET, WSN, WMN, VANET*; CRC Press: Boca Raton, FL, USA, 2016.
2. Zhang, H.; Dai, S.; Li, Y.; Zhang, W. Real-time Distributed-Random-Forest-Based Network Intrusion Detection System Using Apache Spark. In Proceedings of the 2018 IEEE 37th International Performance Computing and Communications Conference (IPCCC), Orlando, FL, USA, 17–19 November 2018; pp. 1–7.

3. Gasmi, R.; Aliouat, M. Vehicular Ad Hoc NETWORKS versus Internet of Vehicles—A Comparative View. In Proceedings of the 2019 International Conference on Networking and Advanced Systems (ICNAS), Annaba, Algeria, 26–27 June 2019; pp. 1–6.
4. Chiti, F.; Fantacci, R.; Gu, Y.; Han, Z. Content sharing in Internet of Vehicles: Two matching-based user-association approaches. *Veh. Commun.* **2017**, *8*, 35–44. [[CrossRef](#)]
5. Ji, B.; Zhang, X.; Mumtaz, S.; Han, C.; Li, C.; Wen, H.; Wang, D. Survey on the Internet of Vehicles: Network Architectures and Applications. *IEEE Commun. Stand. Mag.* **2020**, *4*, 34–41. [[CrossRef](#)]
6. Santamaria, A.F.; Sottile, C.; De Rango, F.; Voznak, M. Road safety alerting system with radar and GPS cooperation in a VANET environment. In *Wireless Sensing, Localization, and Processing IX*; International Society for Optics and Photonics: Bellingham, WA, USA, 2014; Volume 9103, p. 91030G.
7. Ghaleb, F.A.; Maarof, M.A.; Zainal, A.; Al-Rimy, B.; Alsaeedi, A.; Boulila, W. Alrimy Ensemble-Based Hybrid Context-Aware Misbehavior Detection Model for Vehicular Ad Hoc Network. *Remote Sens.* **2019**, *11*, 2852. [[CrossRef](#)]
8. Wahab, O.A.; Mourad, A.; Otok, H.; Bentahar, J. CEAP: SVM-based intelligent detection model for clustered vehicular ad hoc networks. *Expert Syst. Appl.* **2016**, *50*, 40–54. [[CrossRef](#)]
9. Al-Rimy, B.A.S.; Maarof, M.A.; Alazab, M.; Alsolami, F.; Shaid, S.Z.M.; Ghaleb, F.A.; Al-Hadhrami, T.; Ali, A.M. A Pseudo Feedback-Based Annotated TF-IDF Technique for Dynamic Crypto-Ransomware Pre-Encryption Boundary Delineation and Features Extraction. *IEEE Access* **2020**, *8*, 140586–140598. [[CrossRef](#)]
10. Al-Rimy, B.; Maarof, M.A.; Shaid, S.Z.M. Crypto-ransomware early detection model using novel incremental bagging with enhanced semi-random subspace selection. *Future Gener. Comput. Syst.* **2019**, *101*, 476–491. [[CrossRef](#)]
11. Sharma, S.; Kaul, A. A survey on Intrusion Detection Systems and Honeypot based proactive security mechanisms in VANETs and VANET Cloud. *Veh. Commun.* **2018**, *12*, 138–164. [[CrossRef](#)]
12. Liang, J.; Chen, J.; Zhu, Y.; Yu, R. A novel Intrusion Detection System for Vehicular Ad Hoc Networks (VANETs) based on differences of traffic flow and position. *Appl. Soft Comput.* **2019**, *75*, 712–727. [[CrossRef](#)]
13. Ghaleb, F.A.; Maarof, M.A.; Zainal, A.; Al-Rimy, B.A.S.; Saeed, F.; Al Hadhrami, T. Hybrid and Multifaceted Context-Aware Misbehavior Detection Model for Vehicular Ad Hoc Network. *IEEE Access* **2019**, *7*, 159119–159140. [[CrossRef](#)]
14. Azab, A.; Layton, R.; Alazab, M.; Oliver, J. Mining malware to detect variants. In Proceedings of the 2014 Fifth Cybercrime and Trustworthy Computing Conference, Auckland, New Zealand, 24–25 November 2014; pp. 44–53.
15. Tzeng, S.-F.; Horng, S.-J.; Li, T.; Wang, X.; Huang, P.-H.; Khan, M.K. Enhancing Security and Privacy for Identity-Based Batch Verification Scheme in VANETs. *IEEE Trans. Veh. Technol.* **2017**, *66*, 3235–3248. [[CrossRef](#)]
16. Kumar, N.; Chilamkurti, N. Collaborative trust aware intelligent intrusion detection in VANETs. *Comput. Electr. Eng.* **2014**, *40*, 1981–1996. [[CrossRef](#)]
17. Engoulou, R.G.; Bellaïche, M.; Pierre, S.; Quintero, A. VANET security surveys. *Comput. Commun.* **2014**, *44*, 1–13. [[CrossRef](#)]
18. Agrawal, S.; Agrawal, J. Survey on Anomaly Detection using Data Mining Techniques. *Procedia Comput. Sci.* **2015**, *60*, 708–713. [[CrossRef](#)]
19. Lin, X.; Sun, X.; Ho, P.-H.; Shen, X. GSIS: A Secure and Privacy-Preserving Protocol for Vehicular Communications. *IEEE Trans. Veh. Technol.* **2007**, *56*, 3442–3456. [[CrossRef](#)]
20. Daza, V.; Domingo-Ferrer, J.; Sebé, F.; Viejo, A. Trustworthy Privacy-Preserving Car-Generated Announcements in Vehicular Ad Hoc Networks. *IEEE Trans. Veh. Technol.* **2008**, *58*, 1876–1886. [[CrossRef](#)]
21. Zhang, J.; Zulkernine, M.; Haque, A. Random-Forests-Based Network Intrusion Detection Systems. *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.* **2008**, *38*, 649–659. [[CrossRef](#)]
22. Huang, D.; Misra, S.; Verma, M.; Xue, G. PACP: An Efficient Pseudonymous Authentication-Based Conditional Privacy Protocol for VANETs. *IEEE Trans. Intell. Transp. Syst.* **2011**, *12*, 736–746. [[CrossRef](#)]
23. Shen, A.-N.; Guo, S.; Zeng, D.; Guizani, M. A lightweight privacy-preserving protocol using chameleon hashing for secure vehicular communications. In Proceedings of the 2012 IEEE Wireless Communications and Networking Conference (WCNC), Paris, France, 1–4 April 2012; pp. 2543–2548.
24. Liu, J.K.; Yuen, T.H.; Au, M.H.; Susilo, W. Improvements on an authentication scheme for vehicular sensor networks. *Expert Syst. Appl.* **2014**, *41*, 2559–2564. [[CrossRef](#)]

25. Li, W.; Song, H. ART: An Attack-Resistant Trust Management Scheme for Securing Vehicular Ad Hoc Networks. *IEEE Trans. Intell. Transp. Syst.* **2015**, *17*, 960–969. [[CrossRef](#)]
26. Chaubey, N. Security Analysis of Vehicular Ad Hoc Networks (VANETs): A Comprehensive Study. *Int. J. Secur. Appl.* **2016**, *10*, 261–274. [[CrossRef](#)]
27. Daeinabi, A.; Rahbar, A.G.; Khademzadeh, A. VWCA: An efficient clustering algorithm in vehicular ad hoc networks. *J. Netw. Comput. Appl.* **2011**, *34*, 207–222. [[CrossRef](#)]
28. Sedjelmaci, H.; Senouci, S.M. An accurate and efficient collaborative intrusion detection framework to secure vehicular networks. *Comput. Electr. Eng.* **2015**, *43*, 33–47. [[CrossRef](#)]
29. Khraisat, A.; Gondal, I.; Vamplew, P.; Kamruzzaman, J.; Alazab, A. Hybrid Intrusion Detection System Based on the Stacking Ensemble of C5 Decision Tree Classifier and One Class Support Vector Machine. *Electronics* **2020**, *9*, 173. [[CrossRef](#)]
30. Maglaras, L.A. A novel distributed intrusion detection system for vehicular ad hoc networks. *Int. J. Adv. Comput. Sci. Appl.* **2015**, *6*, 101–106.
31. Jha, S.K.; Hassan, M. Building agents for rule-based intrusion detection system. *Comput. Commun.* **2002**, *25*, 1366–1373. [[CrossRef](#)]
32. Li, L.; Yang, D.-Z.; Shen, F.-C. A novel rule-based Intrusion Detection System using data mining. In Proceedings of the 2010 3rd International Conference on Computer Science and Information Technology, Chengdu, China, 9–11 July 2010; Volume 6, pp. 169–172.
33. Patel, S.K.; Sonker, A. Rule-Based Network Intrusion Detection System for Port Scanning with Efficient Port Scan Detection Rules Using Snort. *Int. J. Futur. Gener. Commun. Netw.* **2016**, *9*, 339–350. [[CrossRef](#)]
34. Parameshwarappa, P.; Chen, Z.; Gangopadhyay, A. Analyzing attack strategies against rule-based intrusion detection systems. In Proceedings of the 19th International Conference on Distributed Computing and Networking-Workshops ICDCN'18, Varanasi, India, 4–7 January 2018; pp. 1–4.
35. Barbará, D.; Jajodia, S. (Eds.) *Applications of Data Mining in Computer Security*; Springer Science & Business Media: Berlin, Germany, 2002; Volume 6.
36. Yin, C.; Huang, S.; Su, P.; Gao, C. Secure routing for large-scale wireless sensor networks. In Proceedings of the International Conference on Communication Technology Proceedings, 2003. ICCT 2003, Beijing, China, 9–11 April 2003; Volume 2, pp. 1282–1286.
37. Çam, H.; Ozdemir, S.; Nair, P.; Muthuavinashiappan, D.; Sanli, H.O. Energy-efficient secure pattern based data aggregation for wireless sensor networks. *Comput. Commun.* **2006**, *29*, 446–455. [[CrossRef](#)]
38. Su, M.-Y. Real-time anomaly detection systems for Denial-of-Service attacks by weighted k-nearest-neighbor classifiers. *Expert Syst. Appl.* **2011**, *38*, 3492–3498. [[CrossRef](#)]
39. Al-Jarrah, O.; Siddiqui, A.; ElSalamouny, M.; Yoo, P.; Muhaidat, S.; Kim, K. Machine-Learning-Based Feature Selection Techniques for Large-Scale Network Intrusion Detection. In Proceedings of the 2014 IEEE 34th International Conference on Distributed Computing Systems Workshops, Madrid, Spain, 30 June–3 July 2014; pp. 177–181.
40. Rani, M.S.; Xavier, S.B. A Hybrid Intrusion Detection System Based on C5. 0 Decision Tree Algorithm and One-Class SVM with CFA. *Int. J. Innov. Res. Comput. Commun. Eng.* **2015**, *3*, 5526–5537. [[CrossRef](#)]
41. Yi, Y.; Wu, J.; Xu, W. Incremental SVM based on reserved set for network intrusion detection. *Expert Syst. Appl.* **2011**, *38*, 7698–7707. [[CrossRef](#)]
42. Amor, N.B.; Benferhat, S.; Elouedi, Z. Naive Bayes vs decision trees in intrusion detection systems. In Proceedings of the 2004 ACM symposium on Applied computing—SAC'04, Nicosia, Cyprus, 14–17 March 2004; pp. 420–424.
43. Salo, F.; Injadat, M.; Nassif, A.B.; Shami, A.; Essex, A. Data Mining Techniques in Intrusion Detection Systems: A Systematic Literature Review. *IEEE Access* **2018**, *6*, 56046–56058. [[CrossRef](#)]
44. Kim, G.; Lee, S.; Kim, S. A novel hybrid intrusion detection method integrating anomaly detection with misuse detection. *Expert Syst. Appl.* **2014**, *41*, 1690–1700. [[CrossRef](#)]
45. Al-Yaseen, W.L.; Othman, Z.A.; Nazri, M.Z.A. Multi-level hybrid support vector machine and extreme learning machine based on modified K-means for intrusion detection system. *Expert Syst. Appl.* **2017**, *67*, 296–303. [[CrossRef](#)]
46. Muniyandi, A.P.; Rajeswari, R.; Rajaram, R. Network Anomaly Detection by Cascading K-Means Clustering and C4.5 Decision Tree algorithm. *Procedia Eng.* **2012**, *30*, 174–182. [[CrossRef](#)]

47. Thaseen, I.S.; Kumar, C.A. Intrusion detection model using fusion of chi-square feature selection and multi class SVM. *J. King Saud Univ. Comput. Inf. Sci.* **2017**, *29*, 462–472. [[CrossRef](#)]
48. Shams, E.A.; Rizaner, A.; Ulusoy, A.H. Trust aware support vector machine intrusion detection and prevention system in vehicular ad hoc networks. *Comput. Secur.* **2018**, *78*, 245–254. [[CrossRef](#)]
49. Zeng, Y.; Qiu, M.; Ming, Z.; Liu, M. Senior2Local: A Machine Learning Based Intrusion Detection Method for VANETs. In Proceedings of the Computer Vision, Tokyo, Japan, 10–12 December 2018; Springer Science and Business Media LLC: Berlin, Germany, 2018; pp. 417–426.
50. Zhou, M.; Han, L.; Lu, H.; Fu, C. Distributed collaborative intrusion detection system for vehicular Ad Hoc networks based on invariant. *Comput. Netw.* **2020**, *172*, 107174. [[CrossRef](#)]
51. Zhang, T.; Zhu, Q. Distributed Privacy-Preserving Collaborative Intrusion Detection Systems for VANETs. *IEEE Trans. Signal Inf. Process. Netw.* **2018**, *4*, 148–161. [[CrossRef](#)]
52. Zhang, T.; Zhu, Q. Differentially Private Collaborative Intrusion Detection Systems for VANETs 2020. *arXiv* **2020**, arXiv:2005.00703.
53. Schmidt, D.A.; Khan, M.S.; Bennett, B.T. Spline-based intrusion detection for VANET utilizing knot flow classification. *Int. Technol. Lett.* **2020**, *3*, e155. [[CrossRef](#)]
54. Schmidt, D. Knot Flow Classification and its Applications in Vehicular Ad-Hoc Networks (VANET). Master's Thesis, East Tennessee State University, Johnson City, TN, USA, 2020.
55. Gao, Y.; Wu, H.; Song, B.; Jin, Y.; Luo, X.; Zeng, X. A Distributed Network Intrusion Detection System for Distributed Denial of Service Attacks in Vehicular Ad Hoc Network. *IEEE Access* **2019**, *7*, 154560–154571. [[CrossRef](#)]
56. Uzcategui, R.; De Sucre, A.J.; Acosta-Marum, G. Wave: A tutorial. *IEEE Commun. Mag.* **2009**, *47*, 126–133. [[CrossRef](#)]
57. Zhang, L.; Wu, Q.; Solanas, A.; Domingo-Ferrer, J. A Scalable Robust Authentication Protocol for Secure Vehicular Communications. *IEEE Trans. Veh. Technol.* **2009**, *59*, 1606–1617. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).