Vehicle Behavior Analysis to Enhance Security in VANETs

Robert K. Schmidt[§], Tim Leinmüller[§], Elmar Schoch[¶], Albert Held[*] and Günter Schäfer[‡]
[§]DENSO AUTOMOTIVE Deutschland GmbH, Germany, { r.schmidt | t.leinmueller } @denso-auto.de
[¶]Institute of Media Informatics, Ulm University, Germany, elmar.schoch@uni-ulm.de
[*]Daimler AG, Research and Development, Ulm, Germany, albert.held@daimler.com
[‡]Telematics/Computer Networks Research Group, Technische Universität Ilmenau, Germany, guenter.schaefer@tu-ilmenau.de

***Abstract*—Vehicular Ad-Hoc Networks (VANET) will help to improve traffic safety and efficiency. By exchanging information between each other, vehicles can warn drivers or even prepare for a dangerous situation, i.e. engage pre-crash functionalities like airbag preloading. The decision how to react on information received from other vehicles always has to be made locally. For the security of the system, i.e. to prevent misuse or distortion, each vehicle must be able to evaluate its surrounding independently.**

In this paper, we propose a framework for behavior analysis of other vehicles in the vicinity to approach this problem. By combining the output of multiple behavior analysis modules, each vehicle is assigned a trustworthiness value which may be additionally exchanged among all vehicles, building up reputation. Based on this information, vehicles are classified into trustworthy, untrustworthy or neutral. Applications, for example, may then take this trust rating into consideration in order to react appropriately on incoming information.

I. Introduction

Enabling vehicles to communicate with each other via an ad-hoc network provides many advantages. An extensive summary of potential VANET applications can be found in [1]. In this paper we consider mainly applications related to traffic safety and efficiency. In terms of road traffic safety, a vehicle can look much farther ahead than local sensors (radar or lidar) can. This comprises warnings about a congestion behind a curve or warnings about a critical road condition like black ice. Another group of applications, which is often called cooperative awareness applications relies on accurate knowledge of the positions of neighboring nodes. In order to let each vehicle know of its surrounding, the movement is documented by frequently sending out position information plus speed and heading via so-called beacon packets, also called beacons. This way, vehicles can detect critical situations like blind road crossings for example.

To prevent or at least attenuate false warnings, Vehicular Ad-Hoc Networks need to be secured against injection of falsified data. Especially, the position information on each vehicle has to be reliable since it is needed for all safety and traffic efficiency related applications and for routing. Misbehavior in terms of wrong position information is most likely to disturb the whole system, i.e. safety applications and their resulting decisions. The origin of falsified position information can be on the one hand inaccurate readings from the GPS receiver. On the other hand, there might be someone that does this maliciously, e.g. forging positions to deceive applications.

It is important to note that common cryptographic mechanisms, as proposed for instance in [2] or [3], do and can not target such a problem. When each vehicle signs every message sent with its corresponding key, authenticity and integrity of the messages are achieved, but the content itself may be forged [4]. Even if only vehicles are allowed to participate in the system that received security credentials from a trusted third party (i.e. certified keys from a PKI), misuse is still feasible, e.g. by compromised software installations etc. Hence, mechanisms are needed that analyze and evaluate the validity of the information included in the message, e.g. compared to the history of the vehicle and current context.

In this paper, we propose the VEhicle Behavior Analysis and Evaluation Scheme (VEBAS). The underlying hypothesis is that a behavior analysis of vehicles indicates trustworthiness and untrustworthiness of messages they send. With behavior we refer to all observable information on a vehicle, in particular its past, present and even future movements and its communication activities. The basis of the behavior analysis are the previously mentioned beacon packets, containing vehicle position and movement information. By receiving a sequence of beacon messages from a vehicle, the receiving vehicles are provided with a sufficient amount of data that allows for a meaningful analysis. The result of this analysis leads to the per-vehicle behavior evaluation.

Additionally, these evaluation results might be shared with other vehicles, building a reputation system. Including a node's reputation in the local evaluation process can provide a more detailed view of the current situation, obvious to the problem of additional security risks. In both cases, VEBAS has to be a system that runs continuously. When a safety message is received, time constraints do not allow for a just-in-time behavior analysis. The evaluation result (obviously only based on previous behavior) of the message sending node must already be available.

The remainder of the paper is organized as follows. Section II states system assumptions and details the requirements for the behavior analysis system. A survey on related work in section III shows that the demanded requirements have not been met yet. In section IV, we introduce our behavior evaluation approach. The following discussion in section V outlines

that our scheme is well-suited for the dynamic changes in VANETs. We then conclude in section VI and point to further research on the components.

II. Preliminary Considerations and Requirements Analysis

In this section we define requirements on the behavior evaluation system in general and to the reputation system in particular. These requirements represent important issues in our opinion. Mainly, they result from a previous analysis of characteristics in VANETs, shown in [4]. Defining and explaining these requirements later guides the design of our approach.

A. System Assumptions

Our approach applies on a Vehicular Ad-Hoc Network. Basically, the communicating entities are vehicles. They broadcast their current position via Beacon messages. The assumed fixed interval for these messages is 500 ms. The communication is ad-hoc, i.e. we do not consider the presence of any communication infrastructure like store-and-forward entities at fixed locations. When communicating, vehicles use a constant identifier. For privacy reasons, in other works, e.g. [5] there are changing identifiers suggested. However, this is out of the context of this paper but will considered in follow-up papers.

The authenticity of messages is ensured by means of signatures generated by ECDSA-256 [6] algorithm for example. The corresponding public keys are assumed to be self-signed. By using these keys as identifier, a series of messages can be assigned to one specific vehicle.

B. Requirements on the Behavior Evaluation System

In line with the characteristics, we require our approach VEBAS to be:

- *Decentralized:* Vehicles have to be able to evaluate their surrounding independently.
- *Fair:* The result of the evaluation has to be meaningful. As long as there is not enough evidence for either trustworthiness or untrustworthiness, the vehicle remains neutral.
- *Dynamic:* Once there is enough evidence, the system must react immediately. In addition, the system should not keep this grading but be flexible enough to react quickly on behavior change, i.e. misbehavior.
- *Manageable:* The framework has to allow for easy integration of different analysis modules. Furthermore, they should be configurable differently, e.g. regarding their reliability, importance and output frequency and output permanence.

Following, more detailed requirements on the reputation system are stated and explained.

C. Requirements on the Reputation System

To protect the overall evaluation system from attacks via manipulated ratings from other vehicles, we define the requirements as listed below. Beyond general issues of reputation systems as summarized for instance in [7], we derive additional requirements for VANETS.

- *The evaluation system has to cope properly with message loss.* A loss of a beacon of an honest vehicle should not lead to a negative rating since the communication might have become unreliable.
- *The quality of evaluation must be independent on the different traffic scenarios.* Due to different traffic conditions, local evaluation capabilities may be strongly limited. To improve the view of the neighborhood in terms of trustworthiness, the evaluation system must allow exchange of local positive ratings.
- *Attackers should be detected properly.* Only a misbehaving vehicle should be detected by means of a bad reputation, i.e. a negative value. At best, the attacker should get a complete distrust. In other words, the system should minimize false positives, i.e. honest vehicles should not be falsely rated bad by negative sensors and vice versa malicious vehicles should not be falsely rated good by positive rating sensors.
- *Quick reaction on an attacker.* Additionally to detecting attackers correctly, this should happen as quick as possible.
- *The exchange of ratings should not allow an attacker to know when he is distrusted.* If the attacker has carried out a position forging or while he is doing so, he should not find the optimal point of time when he becomes distrusted and hence could change his identity to start from neutral rating.
- *No trust distribution "loops".* The exchange of reputation should be limited to local ratings. The aggregation of local and cooperative ratings should be not be sent out to prevent falsely increased trust values caused by loops. Hence, a one-level reputation system is required.

Summarizing, many aspects in the design of our system have to be considered. In the following, the defined requirements serve to compare existing approaches and motivate the design of a new approach.

III. Related Work

Research on security in VANETs has already produced different approaches that are related to our idea of behavior analysis. We summarize these approaches in the following section. Their basic idea is to detect misbehavior, often called malicious data.

Golle et al. [8] propose a scheme to detect malicious data by finding outlying data. If there is data that does not fit to the current view of the neighborhood, it will be marked as malicious. The current view, in turn, is established cooperatively. Vehicles share sensor data with each other. An adversary model helps to find explanations for inconsistencies.

The main goal is to detect multiple adversaries or a single entity carrying out a sybil attack. This work, however, does not provide concrete means to detect misbehavior but only mentions a "sensor-driven detection".

Raya et al. [9] combine a solution of immediate revocation of certificates of a "misbehaving" vehicle and formulate a detection system for misbehavior. It is assumed that the PKI is not omnipresent and hence the need for an infrastructure-assisted solution. They use timestamped, signed messages and trusted components (hardware and software) as well. The basic idea behind the autonomous solution is to evaluate deviation from normal behavior of vehicles, while they always assume an honest majority. By using *clustering techniques*, they are able to differ between normal and abnormal behavior, and hence detecting attackers. Once, misbehavior is detected, a revocation of a certificate is indicated over a base station, a vehicle connects to. This revocation is then distributed to other vehicles and the Certification Authority itself. Again for our work, an absence of a PKI is assumed. Therefore, mainly the detection of misbehavior is of interest. The applied technique is a clustering of behavior. As a first criticism, the assumption of an honest majority of vehicles does not hold as identities may be generated arbitrarily.

Both works lack an explicit distinction of evidence for trustworthy and untrustworthy behavior with respect to quantity. For example, insufficient evidence for positive behavior should result neither in a positive nor negative evaluation result but remain neutral. Hence, the corresponding requirement in section II is not met.

It is attempted to find misbehavior. The other way round, vehicles that have not been detected as malicious entities are found trustworthy, implicitly. In this work, we want to find explicit evidence for correct, i.e. trustworthy behavior in parallel with detecting untrustworthy behavior. We must also admit that there may be vehicles which are not to be evaluated completely since there is not enough evidence. Furthermore, this should be done independently, we do not even allow assistance of some infrastructure.

In our previous works, we provided contributions in the fields of trust in VANETs in general [10] and position verification in VANETs in [11], [12], [13], [14]. For instance, in [12], we proposed a basic position verification system designated to evaluate the cooperativeness of vehicles regarding geographic routing in VANETs. We make use of upper limitations of distance to message sender for acceptance of messages as well as overhearing mechanisms.

Additionally, changes in movement and density of vehicles are analyzed combined with a map-based verification. Together with the exchange of neighbortables, they build a trust value by aggregating the modules' output. As a result, we suggest the Acceptance Range Check as a good means to detect random position forging. In this work, the different schemes for position verification are adapted to *Behavior Analysis Modules*. Now, the goal is to establish a general framework to comprehend the different schemes.

In the next section, we elaborate on our approach VEBAS, complying with the requirements defined before.

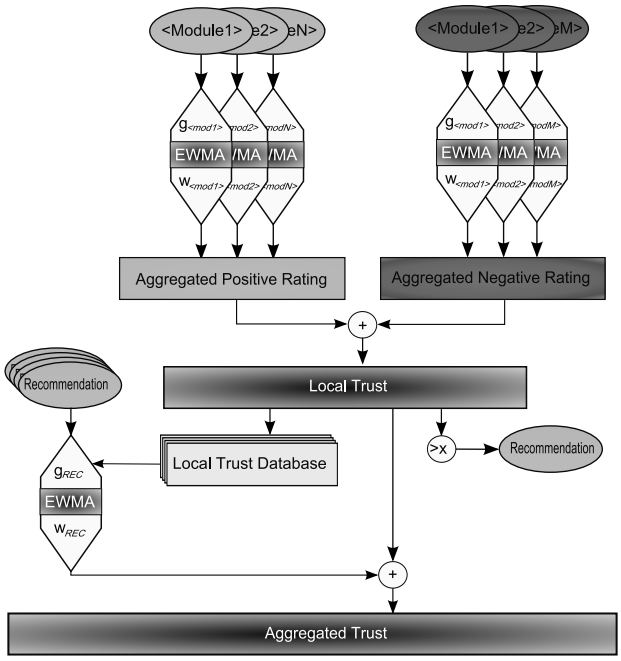

Fig. 1. Structural view on VEBAS

IV. Behavior Evaluation Scheme

In the following, the evaluation framework is presented. As mentioned before, through an analysis of the very recent history of every neighboring vehicle, each vehicle is enabled to evaluate the trustworthiness of its surrounding. The components shown in figure 1 are explained step by step in the following.

The trust value expresses either trust, distrust or uncertainty. This is reflected by the three equidistant ranges of length 1. The final trust value r lies in the interval

$$r \in [-2; +1]$$

We now describe the system component by component, starting with the behavior analysis module as a basis for the evaluation of behavior. Then, the aggregation and aging of the outputs of the modules are shown. In the next step, the first trustworthiness decision is expressed as r_{local}, i.e. the evaluation based on locally available data. The second part of the system comprises the exchange of reputation information in terms of Recommendations. Its aggregation outputs the so-called aggregated trust r_{agg}. How the recommendations are processed is part of the last subsection.

A. Behavior Analysis Modules - Basic Modules

In order to evaluate the behavior of a vehicle, the analysis is split into different modules. Each module checks for a specific property. The modules themselves are divided into positive-rating modules and negative-rating modules, depending on their output value, i.e. $r_{<mod>} \in \{-1, 1\}$. The notation for each module is given as an equation, e.g. depending on the

Symbol	Description
i	The vehicle where analysis module runs on
j	The inspected vehicle
$B_j^1 \ \ldots \ B_j^n$	The current set of received beacons from vehicle j
$d_i(j)$	Distance from analyzing vehicle to inspected vehicle
$d_{<mod>}$	Module specific distance threshold where $<mod>$ is the module's abbreviation
d_{TX}	Mean transmission range
$r_{<mod>}$	Module output upon inspecting a beacon
$\{\ w,\ g,\ \alpha\ \}_{<mod>}$	Parameters for weighting (w) and aging ($\alpha,\ g$) of compound modules' output
$\overline{r}_{<mod>}\ (t)$	The current compound module output at time t
r_{local}	Current evaluation value of inspected vehicle based on autonomous modules (j is left out for simplicity)
r_{agg}	Current evaluation value of the inspected vehicle comprising outputs of all modules

Fig. 2. Table of modules' input data

distance to the inspected j, i.e. $d_i(j)$. The different symbols are comprehended in Table 2.

In the following, we describe selected positive modules at first, summarizing approaches of previous work and introduce new modules. Subsequently, some negative-rating ones are explained.

1) Positive-Rating Modules:

- Movement Analysis (MA+)
- Sensor-Proofed Position (<X>PP)
- Minimum Distance Moved (MDM)

a) Movement Analysis: As mentioned before, we separate some sensors into positive and negative rating modules. This is the case the movement analysis module. In this section we define the positive part of the movement analysis, denoted as (MA+) whereas the negative part (MA-) is defined in the next section. By explicit separation, we extend our previous movement analysis mechanisms found in [11]. MA+ comprises checks on e.g. valid average velocity, acceleration and heading of the vehicles. More detailed, this means that the average speed between two positions may not be higher than $250km/h$. The negative counterpart gives a negative rating is this average speed is above $300km/h$.

b) Sensor-Proofed Position: Sensor-Proofed Position stands for different means to measure distances to neighboring vehicles. Depending on the chosen hardware sensor, different scopes arise. Ultra-Sound provides only small scope whilst Radar (RPP) or Lidar (LPP) may even measure greater distances with appropriate accuracy. Another important module is the Minimum Distance Moved module. It analyzes if a vehicle has evidently moved within a certain period of time. Following, we want to outline the RPP and MDM. Details on the Movement Analysis can be found in our previous work [12].

c) Radar-Proofed Positon: This module is consulted once there should be a vehicle in direct line of sight. As the local radar sensor may only measure a line-of-sight distance to the next obstacle, further processing is needed. To verify a position, a vehicle has to measure its distance to the vehicle that has reported its position. The result has then to be compared with the result of the respective radar sensor tolerating some measurement inaccuracy, due to movement, message latency, GPS inaccuracy, etc. If the radar sensor proves this distance, it gives a positive rating. Further discussion is needed to realize a negative rating version. For our model, we simplify this module to verify distances to vehicles in front of us and to the rear for a distance of 150 meters at maximum and to the side only in the next adjacent lane direct next to us.

d) Minimum Distance Moved: MDM is a means to proof movement of vehicles. Once the vehicle has consistently moved for a particular distance d_{MDM}, where $0 < d_{MDM} \leq c * d_{TX}$ with c denoting a tolerance parameter, this module outputs a rating.

$$r_{MDM} = 1 \ \mid \ d_i(t(B_j^1), t(B_j^n)) \geq d_{MDM}$$

The parameter d_{MDM} should be higher than twice the transmission range, i.e. $d_{MDM} \geq 2 * d_{TX}$, to ensure a sensor output after driving through the transmission area of the sender.

As we see in figure 1, the MDM module is not weighted or aged. It serves as a proof for the basic criterion, namely the movement along a minimum distance to cope with stationary attackers[1]. MDM outputs a $+1$, changing the rating from $r = -1$ to $r = 0$. The Movement Analysis is always positive as long as the vehicle provides a consistent movement pattern. Once there is radar contact, RPP rates positive additionally.

2) Negative-Rating Modules: On the other hand there are the following negative rating modules, that are actually responsible to detect misbehavior.

- Acceptance Range Threshold (ART)
- Movement Analysis (MA-)
- Map-Proofed Position (MPP)
- Sudden Appearance Warning (SAW)
- Maximum Beaconing Frequency (MBF)

a) Acceptance Range Threshold: This module serves to detect unreasonable high distance to a position claimant. Assuming an upper boundary for transmission range, position forging may be detected that way. For example, receiving a position information of a vehicle claiming to be two kilometers away.

b) Movement Analysis: as a negative rating module demonstrates the distinction of cases. The positive version checks for reasonable physical limitations. The negative version outputs when there is reasonable misbehavior. The range

[1]We consider the stationary attacker, e.g. an attacker with a laptop located on a bridge to be one of the most likely threats and hence one of the greatest risks.

between both is reserved for position inaccuracies[2], i.e. unintentional misbehavior.

Another negative rating module is MPP which returns a negative rating if a position is not found on a valid road. However, this module has to be set up with caution since the available map may be outdated. A vehicle coming from a country road may also occur which should in both cases not result in negative rating.

To extend previous work, i.e. to focus more on behavior, the last two modules have been newly developed and hence are introduced in the following.

c) Sudden Appearance Warning: This module detects a sudden appearance of a vehicle in our very vicinity.

Normally, a vehicle first appears at the boundary of the transmission range but not directly next to our own position. Hence, if the first received beacon of a vehicle contains a position that is nearer than d_{SAW}, the module rates negatively.

$$r_{SAW} = -1 \mid d_i(B^1) < d_{SAW}$$

For this module, it is to discuss how sensitive it is regarding message loss. In other words, how often do vehicles appear within a small radius for the first time and how much does this differ this from an abnormal behavior? In this case, also practical experiences are needed.

d) Maximum Beaconing Frequency: MBF detects a violation of the common maximum beaconing frequency.

A vehicle flooding fresh beacons has to be taken care of because it may use it for a faster increase of the trustworthiness.

$$r_{MBF} = -1 \mid t(B^t, B^{t-1}) < \Delta t_{MBF}$$

where $\Delta t_{MBF} < \Delta t_B$. For example, Δt_{MBF} may be chosen as $0.9 * \Delta t_B$. Additionally, to cope with message loss, a (tight) upper boundary is not defined as it may not be assumed to receive beacons frequently.

Summarizing, we have now defined the analysis basis. The outputs have to be further processed and stored. This is achieved by an output aging function, described in the next subsection.

B. Module Output Aging Function

In the following, the design of the fading influence of the modules over time is developed. A simple means to consider the past of a module's output is to keep a weighted moving average. Since we do not have to access specific past ratings, we only need the recent rating $r_{<mod>}$ and the last module output average $\overline{r}_{<mod>}\ (t-1)$. We further want to age exponentially. Hence, we use the Exponentially Weighted Moving Average (EWMA) for the aging of the modules' outputs. This method has been proposed before in MANETs [15]. In our work, in turn, it serves for integrating all single ratings of the Basic Module into one continuous average $\overline{r}_{<mod>}$:

$$\overline{r}_{<mod>}\ (t) = \alpha \cdot r_{<mod>}(t) + (1-\alpha) \cdot \overline{r}_{<mod>}\ (t-1)$$

[2]These tolerance values have to be chosen carefully, since the attacker could make use of them.

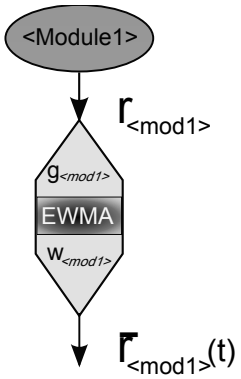

Fig. 3. Basic Module output $r_{<mod1>}$ and Compound Module output $\overline{r}_{<mod1>(t)}$

where $\overline{r}_{<mod>} \in [\,0, 1)$ and $\overline{r}_{<mod>}\ (0) = 0$. The α denotes the weight of the recent module output. The doubled reciprocal value $\frac{2}{\alpha}$ will be following called "aging factor", roughly denoting the number of inputs to return from a high average (=1) to a low value, i.e. nearly zero. This aging component added to each basic module again allows for an appropriate adjustment. In this case, we are enabled to tune the module's contribution to the final rating over time, i.e. how long a single output lasts.

C. Aggregation of Module's Output - The Compound Module

Since the limit of the above described equation is -1 or $+1$ for positive modules and negative modules, respectively, the resulting value has to be weighted again to ensure a proper mapping to the final trust value in the range of $[-2; +1]$. This is simply denoted as $w_{<mod>} \in [\,0, 1]$ for each module. As we demanded before, the positive and negative aggregated ratings are kept separately. Finally, the sum of all positive and negative modules is built to form the local trust r_{local}.

The previously described aging by means of EWMA requires modification by adding another parameter. A slower fading (a smaller value of α) also decreases the immediate influence of a single output of the basic module. This demands another functionality: Each module should have a gain $g_{<mod>}$. In other words, $g_{<mod>}$ defines, how strong the influence of a new module output is. For example, for $g = 5$ the EWMA is executed five times for each non-zero module rating.

To clearly distinguish between both, basic and compound module, figure 3 shows the difference. The basic module is the behavior analysis module itself. If its result is put in the moving average and finally weighted, it represents the output of the Compound Module.

D. Recommendations

This is the part of the system that is responsible for distributing evaluation on other nodes.

For this system a cautious approach is chosen to reduce the risk of possible misuse of reputation. The reputation value will only consist of local ratings, i.e. the reputation

aggregation is based on first-hand information [7]. Local rating in this case refers to the rating that is calculated without considering received recommendations, i.e. r_{local}. Including the recommendations leads to the final evaluation r_{agg}.

In other words, the so-called recommendation on a vehicle reflects a positive vote, broadcasted to all vehicles in the surrounding. They, in turn, determine whether to consider this as a rating. It is the idea not to consider recommendations blindly. They are only used for trust calculations if the recommending vehicle and the recommended vehicle already have a positive local rating. This prevents that an attacker could easily build multiple forged vehicles with a positive rating through forged cooperative trust.

As seen in figure 1, the considered recommendations are aggregated in a separate EWMA. This is weighted and added to the local trust value. The local rating of the recommending vehicle serves as temporal gain for this recommendation. Hence, the variable gain for a recommending vehicle $i = 1$ with $r_{1_{local}} = 0.5$ and a defined maximum gain $g_{R_{max}} = 10$, the gain is $g_{R_1} = 10 * 0.5 = 5$.

For our system, we have chosen a proactive strategy for sending out recommendations, as we do not want to have additional messages in terms of recommendation requests.

Once there is a vehicle that has reached a predefined local trust level, a recommendation is issued. Repeating a recommendation is done frequently, however with a globally specified maximum frequency[3]. It is noted that the frequency should not be too low. Instead, repetition of a recommendation only serves to reach new vehicles or at least vehicles where the recommending and the recommended vehicle have passed MDM. Another argument for a lower frequency of recommendation is the additional communication effort. Each message consists of the vehicle ID and if there are many recommended vehicles and many recommender, this could lead to an exhaustion of the channel bandwidth. Hence, the interval should be high enough, say every five seconds which is a tenth of the beacon rate.

E. Final Aggregation of Local Trust and Recommendations

Following the final procedure to build up trust relations, the local and aggregated trust is explained, referring to figure 1. There are mainly three phases:

1) Generate local rating r_{local} by combining the results from the positive and negative rating modules on a vehicle j.
2) Aggregate r_{local} with the weighted recommendations to establish a final rating r_{agg}.
3) Periodically update these values, e.g. upon reception of a new beacon or a recommendation.

F. Trustworthiness Thresholds

This is the final stage of the evaluation system. The output of the system, i.e. r_{local} and r_{agg}, only becomes meaningful when thresholds are applied. The setup of the thresholds is associated with the outputs and aggregation of the compound modules' outputs which, in turn, rely on the basic modules' outputs.

The idea behind multiple thresholds is to fulfill different trustworthiness constraints related to the different "consuming" applications that consult our system for example. An application demanding more evidence for trustworthy behavior selects the highest threshold which is also only shortly maintained if there is no frequent refreshment via positive ratings by the basic modules.

[3]This common frequency may also be checked by a separate analysis module to detect misbehavior.

V. Discussion

In this section, we discuss the various components and their relation. When applying the framework, it is necessary to understand that the trust-establishing ability of the system relies on an appropriate adjustment of system parameters. We will deal with this in the following.

Results from some of the ideas for behavior analysis modules are available from our previous work in [16]. Further simulation studies on the total system will follow in our future work. Its results strongly depend on the configuration of the framework, which should not be guided only by simulation results but by a theoretical discussion on security issues.

A. Basic and Compound Module Setup

As the total outcome of the system is founded on the behavior analysis modules, their selection, setup and relationship are a crucial component of the overall system. The distinction how to detect misbehavior and how to detect evidently trustworthy behavior is basically done by the corresponding category of modules.

The positive rating modules apply to the latter. We have presented some mechanisms to do so in section IV-A1. In our opinion, a minimum behavior analysis to find honest behavior should at least comprise three characteristics which must be fulfilled together ("AND") to find a vehicle *trustworthy*:

1) Consistency of broadcasted movement, i.e. Movement Analysis
2) Some verified positions, proofed by local sensors (Radar, Lidar, etc.)
3) Long-term movement, to build up trust in vehicles moving with us for a long time

On the other hand, detection of misbehavior has been described in section IV-A2. One negative or few negative ratings should lead to a total output of *untrustworthy* ("OR"):

1) Inconsistency of movement data beyond a tolerance of inaccuracy
2) Non-reasonable positions
3) Communication activities beyond specification, e.g. increased beaconing interval

Once, one of the mentioned criteria is met, the corresponding basic module gives an output. Within the compound module, this value is further processed. First, it is amplified by the gain g before it is added to the weighted moving average as seen in figure 3. The output of the compound module is

then adjusted by the corresponding module's weight w. The procedure within the compound module is in line with the process of discounting described in [7]. The framework further allows to set different parameters. As also proposed by the authors of [7], a different discounting of positive and negative ratings is henceforth feasible. In other words, a negative rating should not be forgotten quickly in contrast to a positive rating. The idea behind that is that an attacker may quickly come up after building up trustworthiness. Hence, trustworthiness has be refreshed frequently while untrustworthiness has not.

Positive ratings may be extended by cooperative means, i.e. the recommendations. However, they should only be considered to find trustworthy vehicles. It has to be ensured that recommendations do not outperform negative local ratings.

B. Cooperative Trust - Recommendations

The recommendations serve to increase the trust level of already positively rated vehicles. If the local trust level is reached, a recommendation is sent out. The level, in turn, is only reached if there were sufficient ratings from the compound modules. The recommendations therefore should increase the number of trusted vehicles in the surrounding.

In contrast to the all other modules, the recommendations require additional communication effort. The corresponding message includes the ID of the recommended vehicle and the ID of the recommending vehicle plus the signature.

Depending on the deployed algorithm and the key length, the size of the message may reach 100 to 150 bytes. It has to be taken care that the channel will not be overloaded by the recommendations temporarily and, in general, to leave enough bandwidth for the actual warning messages. One means to do so is to define a maximum frequency between resending a recommendation. Here, we propose an interval of five seconds. These seems to be suitable for standard traffic density, i.e. free flow. In other scenarios, near to traffic congestion, it might be preferable to switch to a lower frequency or apply a back off strategy if particular vehicles have been recommended multiple times recently.

The recommendations procedure has to be considered with caution. An attacker duplicating himself may start a sybil attack by rating all its instances positively by giving frequent recommendations. Especially, if the attacker is driving on a road and behaves normally. Once he has enough trust of other honest cars he can start the attack and sends out recommendations on his generated "virtual cars". As we described before, we suggest only to consider particular recommendations, e.g. positive votes of vehicles that are locally rated positively, recommending vehicles where the evaluation system has also returned a positive rating.

Concluding, the recommendations offer a trade-off regarding security. On the one hand, there is the additional threat of creating forged positive ratings. But on the other hand, they provide an increased view by also assessing other views. The threat is countered by protecting and detecting forgery, e.g. by filtering recommendations and detecting violations of a specified maximum sending interval.

C. Message Loss Tolerance

As the system only relies on single broadcasts of position information and does not need to receive a position information exactly each Δt_B seconds, the system can tolerate message loss. All position information has to be consistent even if there is one ore more beacons missing. Lost messages only result in slower trust increase. The same holds for recommendations not being properly received.

An attacker would also have no advantage from sending beacons irregularly. The problem of message loss has been taken into account during system design, since there is no negative rating module that inspects the compliance of frequent beacon receptions. On the other hand, in the final design, there has to be a means to prevent a higher beacon frequency to prevent a higher output of the positively rating modules.

D. Final Decision on Trustworthiness

As a first step, we propose three trustworthiness thresholds and one untrustworthiness threshold. So, we have low, medium and high trustworthiness options with $r_{agg} \geq \{0.4,\ 0.8,\ 0.95\}$. On the other hand, we believe that it is better to check for untrustworthy vehicles by looking at the local rating r_{local} to quickly react on misbehavior. Hence, we are independent on misleading or outdated recommendations. Being in line with that, the system classifies vehicles as untrustworthy if $r_{local} \leq -1.25$.

The different trustworthiness thresholds also offer different options for applications. However, defining security and trustworthiness constraints for applications is out of scope of this paper.

Thresholds defined in that way implies neutral vehicles to be represented by a rating in between, i.e. not above lowest trustworthiness and not below untrustworthiness threshold.

VI. Conclusion

This paper introduces a distributed Vehicle Behavior Analysis and Evaluation Scheme (VEBAS). The scheme comprises a framework for behavior analysis modules on which an evaluation of neighboring vehicles regarding trustworthiness is performed. This system is able to distinguish between three classes: Trustworthy, Untrustworthy and Neutral vehicles. In other words, it detects misbehavior, especially intentional misbehavior and honors evident honest behavior and also preserving a class of vehicles that can not be analyzed due to insufficient (sensor) information. The system is further independent from applications and the data it analyzes is movement-related data. It is imaginable, however, to have behavior analysis modules that inspect the context of an application message and evaluate it additionally.

The behavior analysis modules have been partly adapted from previous works. In this artcle we also defined new modules as some characteristics have not been addressed in the related work. One module, Minimum Distance Moved, especially focuses on detection of a stationary attacker which we assume to be the greatest risk. We have further described and selected appropriate modules and their relationship. To

increase the number of trust relations we make use of exchanging ratings (recommendations) among neighboring vehicles, i.e. a reputation system. The handling of the recommendations follows the idea of only considering first-hand information, preventing "self-reinforcement" of reputation [7]. We have further discussed that the communicational overhead implied by the system is still low and that there is no additional danger to traffic safety.

Based on this article we will study different aspects in more detail. Especially, the integration of vehicle's sensors like radar or lidar have to be investigated. In combination with a sophisticated attacker model and few-evidence scenarios, i.e. low traffic density, extensive simulation studies of the system will follow, providing also quantative data on the evaluation of the framework. Additionally, conducting additional field tests may also be helpful to find dynamic adjustments of parameters to provide even better performance by creating an automated scenario-based system setup. We also plan to investigate cooperative means in more detail. On the one hand, these are cooperative analysis modules. On the other hand, more sophisticated trustworthiness dissemination approaches are of interest, similar to Eigentrust [17] and PageRank [18].

REFERENCES

[1] "Car to Car Communication Consortium Manifesto - Overview of the C2C-CC System," C2C-CC, Tech. Rep., Aug. 2007.

[2] J.-P. Hubaux, S. Čapkun, and J. Luo, "The Security and Privacy of Smart Vehicles," *IEEE Security and Privacy*, vol. 4, no. 3, pp. 49–55, 2004. [Online]. Available: http://lcawww.epfl.ch/Publications/luo/HubauxCL04.pdf

[3] M. Gerlach, H. Rechner, and T. Leinmüller, "Security framework for vehicular applications," in *Third International Workshop on Vehicle-to-Vehicle Communications 2007 (V2VCOM 2007)*, 2007.

[4] T. Leinmüller, E. Schoch, and C. Maihöfer, "Security issues and solution concepts in vehicular ad hoc networks," in *Proceedings of the Fourth Annual Conference on Wireless On demand Network Systems and Services (WONS 2007)*, Obergurgl, Austria, Jan. 2007. [Online]. Available: http://www.leinmueller.de/publications/lsm07solutionconcepts.pdf

[5] F. Doetzer, "Privacy Issues in Vehicular Ad Hoc Networks," in *Workshop on Privacy Enhancing Technologies*, Cavtat, Croatia, May 2005. [Online]. Available: http://wwwspies.in.tum.de/personen/doetzer/publications/Doetzer-05-PrivacyIssuesVANETs.pdf

[6] D. B. Johnson and A. J. Menezes, "Elliptic Curve DSA (ECSDA): An Enhanced DSA," in *SSYM'98: Proceedings of the 7th conference on USENIX Security Symposium*. Berkeley, CA, USA: USENIX Association, 1998, pp. 13–13.

[7] S. Buchegger, J. Mundinger, and J.-Y. L. Boudec, "Reputation Systems for Self-Organized Networks: Lessons Learned," in *IEEE Technology & Society Magazine*, 2007.

[8] P. Golle, D. Greene, and J. Staddon, "Detecting and Correcting Malicious Data in VANETs," in *Proceedings of the First ACM Workshop on Vehicular Ad Hoc Networks (VANET)*. Philadelphia, USA: ACM Press, Oct. 2004. [Online]. Available: http://crypto.stanford.edu/~pgolle/papers/vanet.html

[9] M. Raya, P. Papadimitratos, I. Aad, D. Jungels, and J.-P. Hubaux, "Eviction of Misbehaving and Faulty Nodes in Vehicular Networks," *IEEE Journal on Selected Areas in Communications, Special Issue on Vehicular Networks*, 2007.

[10] P. Wex, J. Breuer, A. Held, and T. Leinmüller, "Trust issues for vehicular ad hoc networks," in *67th IEEE Vehicular Technology Conference (VTC2008-Spring), Marina Bay, Singapore*, 2008.

[11] T. Leinmüller, E. Schoch, and F. Kargl, "Position Verification Approaches for Vehicular Ad-Hoc Networks," *IEEE Wireless Communications, Special Issue on "Inter-Vehicular Communications"*, vol. 13, no. 5, pp. 16–21, Oct. 2006.

[12] T. Leinmüller, E. Schoch, F. Kargl, and C. Maihöfer, "Improved Security in Geographic Ad-Hoc Routing through Autonomous Position Verification," in *VANET '06: Proceedings of the third international workshop on Vehicular Ad-Hoc Networks*. New York, NY, USA: ACM Press, 2006, pp. 57–66.

[13] T. Leinmüller and E. Schoch, "Greedy Routing in Highway Scenarios: The Impact of Position Faking Nodes," in *Proceedings of Workshop on Intelligent Transportation (WIT 2006)*, 2006.

[14] T. Leinmüller, E. Schoch, F. Kargl, and C. Maihöfer, "Influence of Falsified Position Data on Geographic Ad-Hoc Routing," in *Proceedings of the second European Workshop on Security and Privacy in Ad-hoc and Sensor Networks (ESAS 2005)*, Jul. 2005.

[15] S. Buchegger and J. L. Boudec, "A Robust Reputation System for Mobile Ad Hoc Networks," 2003. [Online]. Available: citeseer.ist.psu.edu/buchegger03robust.html

[16] T. Leinmüller, E. Schoch, F. Kargl, and C. Maihöfer, "Improved security in geographic ad hoc routing through autonomous position verification," in *VANET '06: Proceedings of the 3rd international workshop on Vehicular ad hoc networks*. New York, NY, USA: ACM Press, 2006, pp. 57–66. [Online]. Available: http://www.leinmueller.de/publications/vanet06-AutonomousPositionVerification.pdf

[17] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, "The Eigentrust Algorithm for Reputation Management in P2P Networks," in *WWW '03: Proceedings of the 12th international conference on World Wide Web*. New York, NY, USA: ACM Press, 2003, pp. 640–651.

[18] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank Citation Ranking: Bringing Order to the Web," Stanford Digital Library Technologies Project, Tech. Rep., 1998. [Online]. Available: citeseer.ist.psu.edu/page98pagerank.html