Security Concept with Distributed Trust-Levels for Autonomous Cooperative Vehicle Networks

Tobias Madl[1]

***Abstract*— The newly proposed cooperative intelligent transportation system (cITS) is a big step towards completely autonomous driving. It is a key requirement for vehicles to exchange crucial information. Only with exchanged data, such as hazard warnings or route planning each vehicle will have enough information to find its way without a driver. However, this data has to be authentic and trustworthy, since it will directly influence the behavior of every vehicle inside such a network. For authentic messages, public key infrastructure (PKI)-based asymmetric cryptography mechanisms were already proposed by different organizations, such as the European Telecommunications Standards Institute (ETSI). The second crucial information of trustworthiness is still missing. In this paper, a new security concept is presented, which introduces a trust-level for each vehicle to enable an assessment, whether data is trustworthy or not. Besides, a Pretty Good Privacy (PGP)-inspired certificate administration is proposed to manage the certificates and their affiliated trust-level. The new concept mitigates sybil attacks and increases the speed of data processing inside vehicles.**

I. INTRODUCTION

Currently, autonomous driving is one of the main goals for vehicular development. Some computer-controlled driving levels are already achieved, for example, highway pilots, lane assistants, or other mostly highway-related assistant systems. However, utterly autonomous driving, as defined in Society of Automotive Engineers (SAE) J3016 [1], especially inside cities or other traffic-heavy areas, continues to be a very demanding task for a single-vehicle. An example can be seen in Figure 1. It shows a dangerous situation, where a pedestrian crossing is hidden behind a building and not detectable for the vehicle that intends to turn right. Thus, vehicle networks for exchanging sensor data, detected obstacles, or other crucial information, such as route planning are required and have already been proposed by different authors [2]–[4].

These cITS or vehicular ad-hoc networks (VANETs) are very promising for completely autonomous driving in every next-generation vehicle. These connections also enable many misuse cases and attack potential. Attacks, such as an enforced right of way functionality or other manipulative behaviors, were predicted by security specialists when these networks were introduced [6]. Therefore, the driver cannot be assumed to be available as ”the ultimate fallback to ensure safety, in contrast to the assumptions underlying the ISO 26262 functional safety standards, which assume that the driver is indeed the final guardian of safety” stated by Petit et al. [7]. It also applies to the vehicle’s behavior, such as selected routes, driving maneuvers, or hold actions. When these are influenced, even obvious, the passengers will not be able to interfere with the vehicle’s decisions.

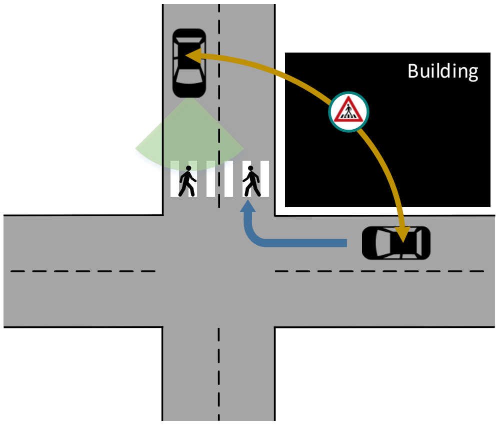


Fig. 1: Cooperative communication for autonomous driving (adapted from [5])

Trust between the participants is one of the most essential elements to work properly since these cITS are mainly cooperative networks. Consequently, it is one of the most valuable assets of the system. There are very different approaches and definitions to establish trust between members of the network. Some concepts locally restrict the range and only trust other vehicles that traveled some time together [8]. Others trust nobody per default and evaluate multiple data from different members to sort out trustworthy and untrustworthy sources [9]. However, no matter the mechanism used, it must be guaranteed that no manipulation is possible, because this can completely deny the proper functionality of the cooperative network.

In this study, the proposed security concept heavily relies on certificates and a PKI. The ETSI TS 103 097 standard and the IEEE 1609.2 standard for the U.S. specify how certificates should be used in automotive vehicle networks to achieve integrity and authenticity. For example, it is specified, that each sent message has the signature and certificate attached, which enables broadcast authentication as the authenticity can be checked by everyone without exchanging any further messages [10]. This ETSI standard proposed, and certificate-based system is extended in functionality with our proposed concept. However, the feasibility can be taken for granted, since its functionality was already proven and analyzed. The connected ETSI TS 102 940 and ETSI TS 102 941 define

[1] Fraunhofer Institute AISEC,Lichtenbergstraße 11, 85748 Garching (near Munich), Germany tobias.madl@aisec.fraunhofer.de

the communication security architecture with its security management and the trust and privacy management for Intelligent Transport Systems. It is also the concept's foundation until the point where some adaptions or extensions are made. The differences do not negatively influence the security of the proposed ETSI standard concept. Besides, the standard explains the need for authentic messages. The reason is that the other vehicles and members of the network rely on the broadcasted information. It could also result in harmful situations or a complete network failure, if the messages could easily be manipulated or anyone could send unauthenticated data.

The newly invented and here presented concept of this paper focuses on creating a new trust relationship between the members of the network that is based on short-living certificates by a central node. This certificate is required to participate in the network and send signed messages to the other members. Newly introduced is the idea, that each certificate has a trust-level that is continuously adjusted and stored on a central node. It will be the root of trust, which provides the certificates for the members. When invalid data, e.g., due to intentional manipulation or sensor malfunction, is detected by a network member, the trustworthiness of the assigned certificate is decreased, and the owner is more ignored inside the network, if the invalid/incorrect data remains present. When the trust-level falls below a certain threshold, the certificate is revoked: the vehicle can no longer participate in this network until it receives a new certificate in the next cycle of the short-living certificates. The main reason for this restrictive measure is to prevent flooding attacks or any additional malicious traffic, which would be forwarded or broadcasted by other network participants.

The remainder of this paper is organized as follows. Section II presents an overview of the related work on trust-based security concepts for cITS and vehicle networks. Section III describes the system model and security concept with all its features and tasks. Section IV presents an initial trust gain/decrease formula with examples. Section V performs a security analysis of the proposed concept. Finally, Section VI concludes the paper and outlines future work.

II. Related Work

One approach for securing vehicle networks is misbehavior detection. It proposes that, in general, all vehicles are equally trustworthy and exchange messages with each other. This happens until any misbehavior is detected and reported. For a more in-depth analysis of these misbehavior detection algorithms, the well-organized overview by van de Heijden et al. [10] is recommended. Some of these detection algorithms only work locally on a single vehicle, and some are globally applied to the whole network. These measures are reactive security mechanisms and require a successful detection of an attack to perform and react properly. The general premise of reactive security is that the time it takes an attacker to perform damage is greater than the amount of time it takes to detect and react to a problem. However, it is not unlikely that either the attack detection is delayed or the attack is not detected. Thus, as vehicle networks are a very time-critical domain, it is likely that these mechanisms will not prevent most of the damage and should only be used with proactive security mechanisms.

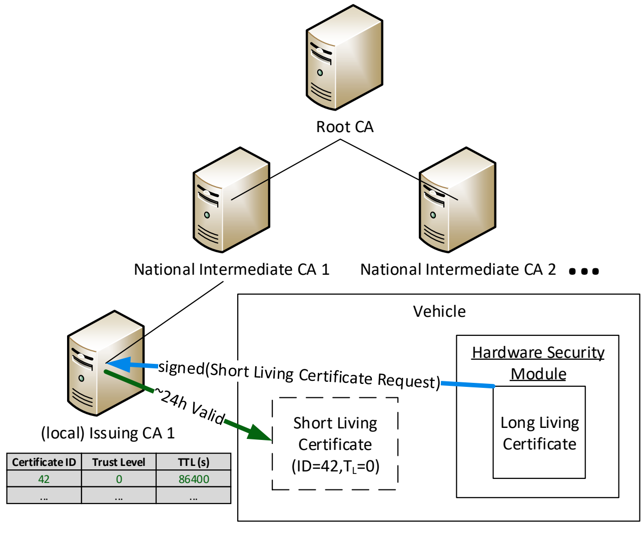
Other approaches that are more similar to the proposed approach, are trust-based algorithms. One of them is an Anti-Attack Trust Management Scheme (AATMS) [11] proposed by Zhang et al. Their approach is a TrustRank algorithm inspired system, based on the Bayesian inference to calculate the local trust of vehicles based on historical interactions. A small set of seed vehicles according to the local trust and social factors are selected, to evaluate each vehicle's global trust. They simulated their approach and showed that it is possible to identify trustworthy and untrustworthy vehicles inside the VANET, even under malicious attacks. However, this approach requires historical interactions to work efficiently, which requires big data storage. Besides, the whole trust network collapses if some seed vehicles are chosen poorly.

Another idea was proposed by Guo et al. named TROVE [12]. All messages received by other nodes are evaluated and compared to define trust in the presence of a given event in a context. It is fed back in a reinforcement learning (RL) model, continually updating the trust evaluation function. It is a locally stored trust-level that is continuously updated by received and verified data. Thus, even if it is possible to sort out untrustworthy data by the presence of more than 50 % malicious nodes, it gets more unlikely, since the vehicle cannot double-check each data by itself.

Hu et al. [13] proposed a different approach called REPLACE. This mechanism selects platoon vehicles, which are the most trusted ones in a close area or a vehicle column. These are the ones that heavily influence the driving behavior of the surrounding cars. However, these vehicles must be driven by real drivers. The trust in these vehicles is provided by the accompanying vehicles by reporting good or bad behavior. It could be an interim solution during the transition from partly autonomous to fully autonomous vehicles. Here, the root of trust is a well-selected platoon vehicle, which is only verified by the few surrounding cars.

A different approach for trust-levels of vehicles was proposed by Kiening et al. [14], which defines different trust assurance levels (TALs) ranging from zero to four. These levels depend on the initial intelligent transportation system (ITS) station inside each vehicle and their evaluated assurance level by a security workgroup. In result, vehicles with higher TALs are more trustworthy than vehicles with lower ones. In case of a security breach of a vehicle its assigned TAL will be reduced to zero until the vulnerability is resolved and fixed. This concept is a good way do define an initial trust-level for different vehicles inside cITS and could even be extended with the approach presented in this paper.

Thus, it can be concluded that none of the existing security concepts for cooperative vehicle networks have a centrally stored trust-level, which is decentralized, increased, or decreased by the network participants, to create a trustful network. In the following sections, a new approach will be


Root CA
National Intermediate CA 1
National Intermediate CA 2 •••
Vehicle
Hardware Security Module
signed(Short Living Certificate Request)
~24h Valid
(local) Issuing CA 1
Short Living Certificate (ID=42,T$_L$=0)
Long Living Certificate

Certificate ID	Trust Level	TTL (s)
42	0	86400
...	...	...

Fig. 2: Public key infrastructure

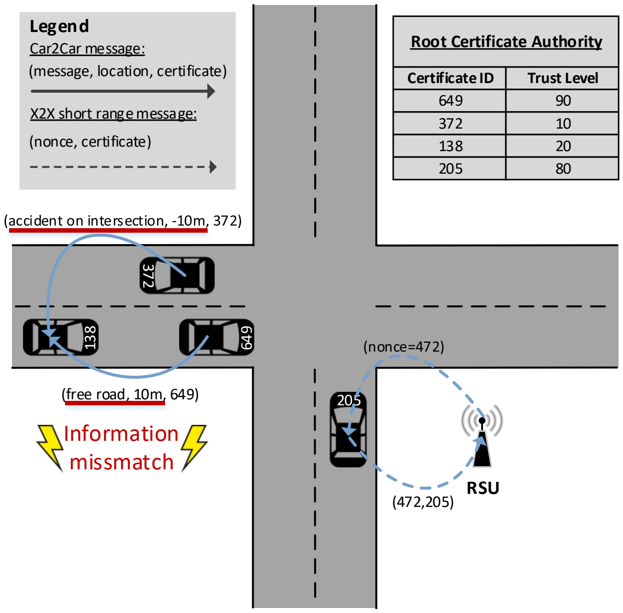
presented, which enables this kind of trustful network and supports the vehicles in their autonomous driving task by creating an additional filter option for information received from other network participants. The proposed approach will create new network protection against various common attack vectors, especially against sybil attacks.

III. System Model

We now present the new proactive security concept for trust establishment in cooperative automotive vehicle networks. The proposed concept's main feature is the introduction of short-living, anonymous certificates, which are managed by a central certificate authority (CA) combined with a dynamic trust-level (e.g. Figure 2). The proposed concept tries to filter out which information a vehicle should use for its decision-making. For example, to consider a single autonomous vehicle in this network, it receives much different information. First, it analyzes and fuses all of its sensor data, such as cameras, radars, ultrasonic sensors and lidars. Next, it receives information like cooperative awareness messages (CAMs) or decentralized environmental notification messages (DENMs) from many different surrounding vehicles, transferring varying and possibly even conflicting information. The task is to sort out which information is accurate and trustworthy and which is not. The major problem is that some information is not verifiable by the vehicle itself, since it is not in reach of any onboard sensors. Thus, the only way to check this information is to compare all the received data and draw conclusions. The vehicle must discover which data could be correct, which is very likely wrong, irrelevant, or not checkable. This process is very tedious and time-consuming since it requires much computational power in the vehicle itself to evaluate this received data continually.

A. Basic Concept

The proposed solution to the presented problem is a trust-level attached to the transferred signature. The trust-level enables an ordering of the most trustworthy information to the less trustworthy data. With this additional information, a vehicle can trust the provided data without exceedingly trying to verify it. There will still be basic checks and a fusion with the own sensor data: however, very trustworthy data can now be taken as fact and used to derive additional route planning and behavior. Thus, this additional information, which is requested from the backend, such as the root CA or one of its sub-CAs, will increase the amount of trustworthy data to optimize autonomous driving without additional computation power. An example for this data exchange can be seen in Figure 3. Here, the vehicle with the certificate ID 649 is highly trustworthy and messages surrounding vehicles that it has detected a free intersection. On the other hand, the vehicle with certificate ID 372 advertised a conflicting event of an accident: thus, a blocked intersection. However, this vehicle's trust-level is very low: it will be practically ignored, as vehicle 649 is significantly more trustworthy. In this example, both vehicles 649 and 138 will continue passing the intersection, if no other trustworthy event tells otherwise or they detect any obstacle.



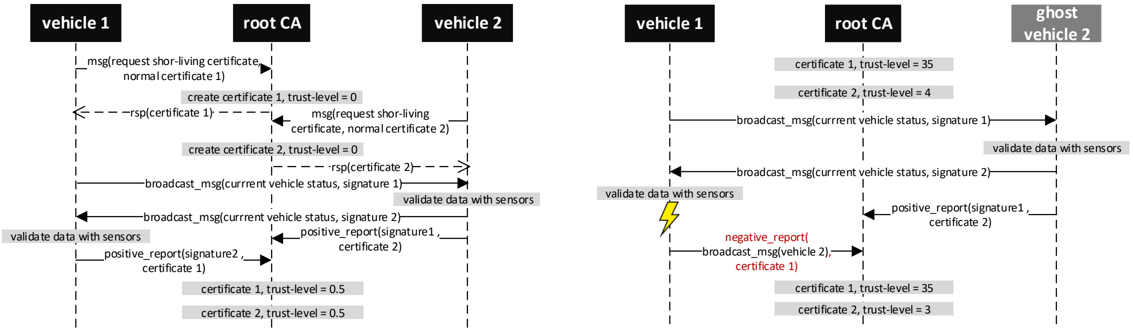
Root Certificate Authority

Certificate ID	Trust Level
649	90
372	10
138	20
205	80

Fig. 3: Negative handshake of a vehicle and a RSU handshake example

B. Advantages

The main advantage of this concept is the trust-level. It enables each participating vehicle a ranking, which information is the most trustworthy and what is the less. This advantage expresses in two ways. First, when the data has a high trust-level, it can be trusted directly, and only minimal to no validity checks must be executed. This significantly increases the speed of the sensor fusion process and thus speeding up the decision-making. But, the final implementation of the sensor fusion is up to the vendors and does not influence the trust-level itself. It is only an additional possibility of filtering or ranking the received data. Next, untrustworthy data can be


vehicle 1
root CA
vehicle 2
msg(request shor-living certificate, normal certificate 1)
create certificate 1, trust-level = 0
rsp(certificate 1)
msg(request shor-living certificate, normal certificate 2)
create certificate 2, trust-level = 0
rsp(certificate 2)
broadcast_msg(currrent vehicle status, signature 1)
validate data with sensors
broadcast_msg(currrent vehicle status, signature 2)
positive_report(signature1 , certificate 2)
validate data with sensors
positive_report(signature2 , certificate 1)
certificate 1, trust-level = 0.5
certificate 2, trust-level = 0.5
vehicle 1
root CA
ghost vehicle 2
certificate 1, trust-level = 35
certificate 2, trust-level = 4
broadcast_msg(currrent vehicle status, signature 1)
validate data with sensors
broadcast_msg(currrent vehicle status, signature 2)
validate data with sensors
positive_report(signature1 , certificate 2)
negative_report(broadcast_msg(vehicle 2), certificate 1)
certificate 1, trust-level = 35
certificate 2, trust-level = 3

(a) Short-living certificate request and mutually positive handshake of two vehicles

(b) Negative handshake of a vehicle

Fig. 4: Handshake examples

ignored entirely or seen as unimportant, reducing the amount of data that must be evaluated. Last, mostly trustworthy vehicles will be used for decision-making, even if they build not the majority of surrounding vehicles. Thus, sybil attacks, which require a majority of attackers in currently surrounding network members, will not work anymore since all untrustworthy data is ignored. Another advantage of the trust-level is the possibility to exclude malfunctioning or evil members from the network continuously; however, with a grace period to possibility join the network again after 24 hours.

C. Technical Architecture

In the default state of autonomous driving, the vehicle broadcasts valuable information to inform other vehicles about the current position, heading, or any special events. Each message or message block is signed with the short-living certificate of the original vehicle. If a new vehicle now enters the broadcast area, both vehicles will notice messages with a yet unknown signature. Both try to validate the received information and detect the real presence of the newly detected vehicle with their sensors. If this validation is successful, the new vehicle's message, which was used to validate its authenticity, will be signed with their certificate and sent to the backend. It now confirms that the message's reporter trusts the other sender. However, the trust-level will only be increased if both vehicles positively report to the backend, ensuring the interest to report to the backend. An exception to this rule are RSUs. In the proposed concept, RSUs are defined as an additional vehicle network component that have more trust than normal vehicles and perform handshakes with the network participants. Their main tasks are verifying the presence of the handshaking partners and the broadcast of important information. Thus, vehicles do not require to report to the backend since only the RSU will file a positive or negative report.

On the other hand, if the vehicle detects an inconsistency with the broadcasted data or the simple non-existence of the vehicle, then a short report, which contains the reason for the report, the abnormal message, and the certificate of the probable evil sender, will be sent to the backend. If the backend receives multiple reports with the same conflicting message from many vehicles, it will directly heavily decrease the reported vehicle's trust-level, since it was probably a direct attack against the network.

Backend: Like the ETSI TS 103 097 standard proposes, there is one root certificate authority, which is a central and possible worldwide organization like the Internet Assigned Numbers Authority (IANA). This CA has sub-CAs that are distributed to countries, regions, and cities. These CAs centrally store a certificate for each single-vehicle, corresponding to the car's vehicle identification number (VIN). When the autonomous car is manufactured, the first certificate will be created and installed. Each time the vehicle is inspected from the OEM, this certificate will be exchanged to maintain the current state of the art. The certificate must be exchanged every three years either by the OEM or by a CA trusted sub-authority. The load will be distributed, like the Domain Name System (DNS) service to guarantee high availability and low latency. Additionally, the computational complexity does not increase with the count of network participants, which enables good scalability. This can also be seen in the proposed equations in Section IV.

Handshake: There are two ways the vehicles can prove their authenticity to increase their trustworthiness. One can be observed on the right side of Figure 3 on the previous page. In this case, a vehicle passes an RSU. The RSU currently broadcasts a short living nonce (only valid for a few minutes or seconds) on a low range communication channel, e.g. Bluetooth. Then, the vehicle signs this nonce with its certificate and broadcasts it on the normal communication channel. This message is again received by the RSU, approving the local presence of the vehicle.

The second method is already described in Section III-C and can be observed in Figure 4. The negative case is also described in Figure 4. In this example, the trust-level of the attacker's vehicle is only reduced by one. However, this threshold could be increased depending on how many reports were filed against the harmful vehicle. This public-key cryptography and signing method is a well-tested and widely used mechanism to transfer authenticated messages and derive the trustworthiness of the sending author [15]. Thus, this concept can be applied to the automotive domain, as seen in the example.

D. Privacy

Privacy is one of the biggest concerns in the autonomous driving domain, since it enables tracking of each vehicle, everywhere, at any time. Thus, various existing measures are combined with proposed methods to protect this valuable asset.

Short-living certificates: To increase the privacy of every participant in a vehicle network, it is often suggested to introduce pseudonym certificates [16], [17]. The main difference in the proposed approach is the addition of an attached trust-level to these anonymized certificates. Therefore, these certificates are no longer just an access restriction for participating in the network, but also a value if other network participants trust the messages that are signed with it or not. If attackers could obtain such trusted certificates, many misuse-cases would be possible again. Thus, these certificates' lifetime will drastically be reduced to around 24 hours to prevent these attacks. Besides, each vehicle must prove its authenticity every day using the above-described handshakes since the trust-level is not transferred and will be lost when the certificate expires. To prevent tracking of the root CA, different approaches of anonymously requesting these short-living certificates were also already proposed by Khodaei, Brecht et al. [16], [17].

Key storage: The main certificate, introduced in Section III-C is one of the most sensitive components inside the system since it is required in order to receive the temporary anonymous certificates. These long-living certificates must be stored securely, e.g., in hardware security modules (HSMs), to prevent extraction or manipulation. In the best case, these HSMs can contact the root CA if any tampering is detected, resulting in a central invalidation of the stored certificate. However, even if an attacker denied this invalidation, the HSM destroys its secrets if any manipulation is detected. Thus, no further short-living certificates can be issued.

Certificate revocation: Each time a new certificate is detected, the vehicle connects to the backend and requests the certificate's trust-level. The trust-level is either locally stored or if the backend answers with a revocation response, the certificate is stored in the local revocation list. If a certificate is not seen for some time, the vehicle is likely not around anymore. The local trust-level will be deleted. Otherwise, this locally held trust-level will be refreshed every few minutes, ensuring that every revocation is detected as soon as possible, but the network is not excessively used.

E. Staged Rekeying

The newly introduced short-living certificates, which should have a lifespan of around 24 hours can create a significant workload for the PKI infrastructure if all vehicles would synchronously issue a new certificate at a certain time. Distributing them over the whole day, when the vehicle is first used, has the disadvantage of creating a new tracking possibility. To prevent this, the certificates should still be issued in bulk. They are divided into some hour slots to validate the certificates from 22 to 26 hours. Thus, the load gets split. Besides, not all surrounding vehicles are suddenly not trustworthy anymore. The newly created certificate has a randomized time inside the defined boundaries independent of the previous certificate.

F. Full Anonymization

In addition to the proposed concept, it is possible to perform a fully anonymized certificate distribution. The origin certificates can be issued in bulks and allow an anonymous certificate issuing, which is already introduced by Khodaei, Brecht et al. [16], [17]. The certificate is now used to issue the short-living certificates, without any possibility to identify the actual vehicle or driver behind the issuer. However, in the future the problem could arise that a long-living certificate could not be revoked since there is no matching between the vehicle and the anonymous certificate. So if there would ever be the requirement to revoke the long-living certificate, cameras at RSUs could be used to identify vehicles and their belonging certificates, to revoke them.

IV. Trust Gain/Decrease

This section focuses on the calculation of the individual trust-level for each vehicle. It is essential to define how trust increases or decreases, how different handshakes affect both partners, and when a vehicle is defined as trustworthy. This is probably by far not the best possible formula since this is the first proposal of these values and formulas. However, it meets the stated requirements, which are listed as follows:

- Many positive handshakes required to achieve a high trust-level
- Fast decrease of sender's trust, if evil actions or wrong data/information are observed
- RSUs are generally trusted, and handshakes with RSUs influence the trust-level stronger than normal network participants
- HSMs with a valid certificate is a network participation requirement, and it will be assumed that a normal attacker will not have access to more than 99 functioning HSMs
- Handshakes between two vehicles are only possible once during the lifetime of the short-living certificate
- Vehicles can be excluded from sending data if the trust-level falls below a certain threshold

A. Initial Proposed Formula

The here presented values and equations implement a first unpolished and optimized solution to the above stated requirements. These values and calculations have to be adjusted to simulation or real-world evaluations in order to gain the most value from this new concept.

Initially, every newly issued certificate will start with a trust-level of zero, which can also be seen as zero percent trustworthiness. It means that 100 is the maximal trust-level, representing the most trustworthy certificate. As the third requirement states, it is assumed that a normal attacker does not have access to more than 99 functioning HSMs; thus, less than 100 possible malicious certificates. The reason for this assumption is the amount of work and evil activity that is required to gain access to so many devices. For example, as mentioned earlier, vehicles on a junkyard often do not have a valid certificate anymore; thus, cannot be used by an attacker. The small number of vehicles, which still have a working HSM with a valid certificate, is probably only a very small number of scrapped vehicles. Therefore, the only other source of gaining access to working HSMs with valid certificates would be stealing working vehicles. However, realistically, stealing 100 vehicles just to get some advantages in the cooperative network or executing a locally restricted denial of service seems disproportionate.

In result, the maximal trust boundary ($T_{\max}$) can be defined:

$$T_{\max} \in \mathbb{R} \quad T_{\max} = 100 \tag{1}$$

The starting trust-level is zero, but there must be a state where a vehicle could be excluded from sending data to the network, as stated in the requirements. Thus, there is the possibility for an attack by sending early negative handshakes to exclude others from the network. However, the lower boundary still requires 20 consecutive negative handshakes without a single positive interaction to exclude the victim from the network. With the above attack requirements, this is very likely one of the much more difficult ways to deny a single vehicle's proper functionality. As result, the lowest boundary is defined as follows:

$$T_{\min} \in \mathbb{R} \quad T_{\min} = -10 \tag{2}$$

The trust-level T_{L} can now be defined as follows:

$$T_{\mathrm{L}} \in \mathbb{R} \quad T_{\min} \leq T_{\mathrm{L}} \leq T_{\max} \tag{3}$$

With the above constrains, the following rules for the trust-status T_{S} can be concluded:

$$T_{\mathrm{S}} = \begin{cases} \text{trustworthy}, & \text{if } 50 \leq T_{\mathrm{L}} \\ \text{not trustworthy}, & \text{if } -10 < T_{\mathrm{L}} < 50 \\ \text{excluded}, & \text{if } T_{\mathrm{L}} = -10 \end{cases} \tag{4}$$

The calculation of the new trust-level $T_{\mathrm{L,new}}$ for a positive handshake, can be calculated as follows, which is dependent on the maximal trust-level $T_{\max}$ and $H_{\max}$ describing the number of handshakes that is required to reach the maximal trust-level. For the proposed approach, $H_{\max}$ is equal to 200 since at least 100 individual handshakes are required to reach the minimal trustworthiness with $T_{\mathrm{L}} = 50$ (shown in equation 4). Thus, to reach $T_{\max}$ two times the number of handshakes is required.

$$T_{\mathrm{L,new}} = \min(T_{\mathrm{L,old}} + T_{\max}/H_{\max}, T_{\max}) \tag{5}$$

Using this equation, the delta of a positive handshake can be calculated:

$$H_{\Delta+} = T_{\max}/H_{\max} = 0.5 \tag{6}$$

The decrease of the trust-level in case of a negative report/handshake should be higher, than an increase to satisfy the above requirements. However, with above assumptions (attacker controls less than 100 functioning HSMs) an attacker could not exclude a full trustworthy ($T_{\mathrm{L}} = T_{\max}$) vehicle:

$$H_{\Delta-} = 2 \cdot H_{\Delta+} \tag{7}$$

The final result is the following equation, which is used to calculate $T_{\mathrm{L,new}}$ for either a positive or negative report:

$$T_{\mathrm{L,new}} = \begin{cases} T_{\mathrm{L,new}} = \min(T_{\mathrm{L,old}} + H_{\Delta+}, T_{\max}), \\ \text{if positve report} \\ T_{\mathrm{L,new}} = \max(T_{\mathrm{L,old}} - H_{\Delta-}, T_{\min}), \\ \text{if negative report} \end{cases} \tag{8}$$

For an RSU handshake or report, it is seen as an even more powerful trust gain or loss. The same rule applies to this kind of handshake, which states that between an RSU and another network entity, only one handshake per certificate lifetime is possible. Besides, the number of RSUs compared to the vehicles in the network is much smaller, leading to the possibility of higher production expenses per unit. In return, they can be built more tamper-proof against attacks and are better monitored for failure or manipulation. Thus, the following delta values are proposed:

$$H_{\Delta+,\text{RSU}} = 5H_{\Delta+} \tag{9}$$

$$H_{\Delta-,\text{RSU}} = 5H_{\Delta-} \tag{10}$$

With these numbers, an attacker would have to gain control over 20 RSUs to create a minimal trustworthy vehicle or 22 RSUs to exclude a fully trustworthy vehicle from the network. Furthermore, the attacker would have to bypass more security mechanisms, such as a monitoring system, which monitors these devices for any manipulation. Therefore, this attack is again utterly disproportionate to the goal an average attacker tries to achieve.

B. Example

In this section, some positive and negative simulations for trust-level progress are shown in figure 5. The x-axis represents the number of handshakes H, and the y-axis represents the current trust-level T_{L}. The first and the last example are two linear cases, where only positive or only negative handshakes are reported. The second example shows

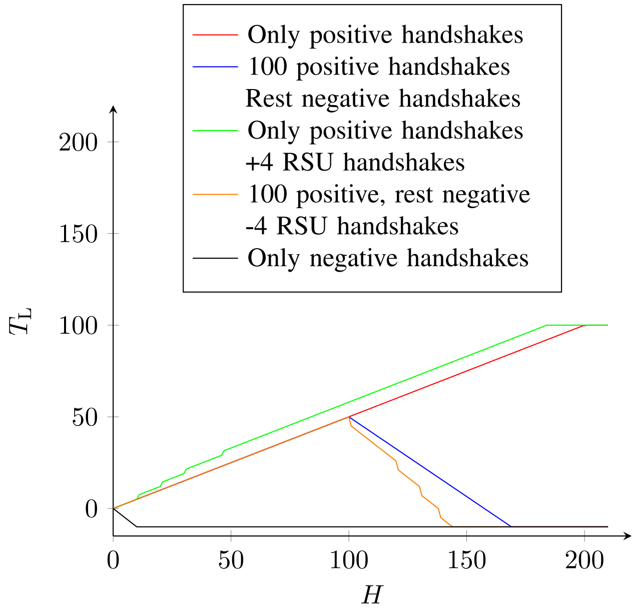


Fig. 5: Trust-level progress examples

a vehicle that initially operates normally and gets positive handshakes. After 100 positive handshakes were detected, the vehicle now sends wrong information and only receives negative handshakes anymore. The vehicle also loses trust faster, as it initially gained it. The third example is only an optimistic handshake simulation, but this time with additional RSU handshakes. In contrast to the first example, the vehicle achieves $H_{\max}$ faster, as it positively performs handshakes with four RSUs. The fourth example is similar to example 2. However, with the negative RSU reports, the vehicle loses its trust much faster and is already excluded before hitting the 150 handshake mark. Thus, it can be stated that with these proposed values, the bad or attacking vehicles will be excluded very timely, even if no RSUs are around.

V. Security Analysis

To analyze the robustness of the proposed security concept, different attacks and their impacts will be discussed in this section. As a comparison, the attacks will also be applied to the original cITS without the additional trust-level concept.

A. Sybil Attack

In this attack scenario, most of the surrounding network participants are evil (>50% evil). However, in the default cITS environment, these vehicles would heavily influence all surrounding vehicles' decision-making, as the majority verifies their data. Thus, the attack would be successful and could lead to dangerous situations and manipulated driving behavior.

In contrast, if the minority of good vehicles have high trust-levels, all surrounding low trustworthy and potentially evil vehicles are mainly ignored by the sensor fusion. Thus, even if there are a majority of malicious nodes, the benign participants with high trust-levels are not influenced by them. In the case of no other trustworthy participant around, the system behaves the same as it does without a trust-level. However, this is an absolute rare edge case, since 100 % of the surrounding nodes must be evil or untrustworthy.

In addition to even more reduce the likelihood of sybil attacks, handshakes between pseudonym certificates can only occur once a day. Thus, it is not possible to make two vehicles trustworthy by constantly handshaking each other. Attack mitigation for this additional security could be the creation of multiple fake vehicles and then handshaking each other to gain trust and execute a sybil attack. This is mitigated by requiring real hardware in the form of a HSM, which contains the private key of the vehicle, to participate in the automotive network. Thus, the attack costs and efforts for a sybil attack outweigh the possible benefits arising from this kind of attack since it only temporarily gives the right of way or possibly redirects traffic.

B. On-Off Attack

In this attack, an attacker behaves normally most of the time and then for a short period, it does evil activities. In general, these attacks are not easy to detect, and a normal vehicle network is prone to such malicious activities.

At first sight, the addition of trust-levels does not look very helpful in this kind of attack since the vehicle behaves normally during this time, and it would also gain trust. However, the multiple on-off switches between good and bad behavior will be more difficult since a detected attack will heavily decrease the trust-level, which takes more time to rebuild. Thus, in the end, this attack is not completely mitigated but at least slowed down since every time the attacker tries to execute an attack, it has to gain a high trust-level in advance.

C. RSU Attack

This attack is not directly targeted towards vehicles but RSUs. In a default vehicle network, it could lead to manipulated data transferred by the RSU or some other attacks like denial of service.

The above described trust-level concept generates an additional threat of a hijacked trustworthy handshake partner, which could eventually lead to falsely marked trustworthy vehicles. A hijacked RSU is a much stronger attack and influence in the proposed concept. However, two security mechanisms arise to mitigate this additional vulnerability and increase the effort to an unworthwhile level. First, these trustworthy RSUs should contain an HSM like the vehicles for their private certificate, which detects any manipulation. Next, the same rule for handshakes applies to RSUs as to vehicles, which defines that a handshake between two certificates in the network can only happen once. Thus, even a hijacked RSU cannot mark a vehicle trustworthy just by itself; it only increases the trust more than a normal vehicle. In the end, the effort to hijack enough RSUs to mark a vehicle trustworthy is probably higher than the benefit from its outcome, which would mitigate this additional threat.

D. Trust-level Specific Attacks

By introducing additional functionality in most cases, new threats are also introduced. However, if the concept is sound, these threats only pose no to little danger to the whole system, and its additional functionality outweighs the residual risk.

One of such new threats is the creation of trusted ghost vehicles. In the default network, ghost vehicles also pose a thread, but if many of them could be generated to automatically handshake them trustworthy, this would even increase the impact of this manipulation. Thus, in the first level of security, HSMs are required, which reduces the risk significantly, since real hardware is required to simulate a vehicle. Now, an attacker could recover HSMs from scrap yards to create some ghost vehicles. However, the deployed private keys on these HSMs have an expiry date, which is most likely bound to the vehicle's general inspection every few years. This date is often already passed or very close to passing if the vehicle is placed on a scrap yard. The HSM is then rendered useless since the certificate is not renewed.

The next attack that focused on this trust-level is manipulating a handshake with an RSU. For example, one vehicle could contain multiple HSMs to simulate multiple vehicles performing handshakes, gain trust-levels, and later be used for sybil attacks. In this case, another security mechanism arises. The RSU only performs handshakes with vehicles, it can identify, thus, if multiple vehicles try to handshake at the same time and position, they will be denied.

Finally, there is still the possibility that a normal vehicle drops under a certain trust-level and is excluded from the communication. Thus, it does not result in a vehicle break down, but rather exclusion from sending messages to the network. It means that the vehicle still receives data and continues driving, but it will not be able to communicate anymore. In the end it loses its ability to communicate any route planning and other beneficial communication rights.

E. Summary

In summary, the newly proposed concept is a different approach, which for the first time, enables mitigation for sybil attacks. The introduction of the PKI infrastructure already increased the security level of cITS. However, with the additional requirement of trust, attacks against the cooperative vehicle networks become considerably more unlikely since the required attack effort far outreaches the generated benefit for an attacker.

VI. Conclusion

This paper proposes a new security concept that extends the ETSI proposal for security in-vehicle networks. It consists of a trust-level with a short living certificate for each vehicle. The trust is gained with PGP-like handshakes and lost when malicious, invalid, or impossible data or messages are observed. The proposed method enables a strong defense against sybil attacks or most evil nodes in the vehicle's surroundings. Besides, it increases the sensor fusion performance, behavior, and route planning as it reduces the validation work and automatically filters out untrustworthy data. It also proposes an initial formula for trust and handshake values, which could be used for first implementations. Finally, the robustness of the proposed concept against various attacks was also discussed and compared with the initial variants without trust-level. In future work, this concept will be implemented in a simulation framework and evaluated to obtain the final thresholds of the new approach. Thus, this concept increases the security of the cITS and probably boost each autonomous vehicle's performance when being a member of this network.

References

[1] SAE On-Road Automated Vehicle Standards Committee, "Taxonomy and definitions for terms related to on-road motor vehicle automated driving systems," *SAE Standard J*, vol. 3016, pp. 1–16, 2014.

[2] S. Yousefi, M. S. Mousavi, and M. Fathy, "Vehicular ad hoc networks (vanets): challenges and perspectives," in *2006 6th International Conference on ITS Telecommunications*. IEEE, 2006, pp. 761–766.

[3] S. Zeadally, R. Hunt, Y.-S. Chen, A. Irwin, and A. Hassan, "(VANETs): status, results, and challenges," *Telecommunication Systems*, vol. 50, no. 4, pp. 217–241, 2012.

[4] R. Nagel, S. Eichler, and J. Eberspacher, "Intelligent wireless communication for future autonomous and cognitive automobiles," in *2007 IEEE Intelligent Vehicles Symposium*. IEEE, 2007, pp. 716–721.

[5] Car 2 Car Communication Consortium and others, "Car 2 car communication consortium website," 2020. [Online]. Available: https://www.car-2-car.org/about-c-its/

[6] M. N. Mejri, J. Ben-Othman, and M. Hamdi, "Survey on VANET security challenges and possible cryptographic solutions," *Vehicular Communications*, vol. 1, no. 2, pp. 53–66, 2014.

[7] J. Petit and S. E. Shladover, "Potential cyberattacks on automated vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 546–556, 2014.

[8] R. Sugumar, A. Rengarajan, and C. Jayakumar, "Trust based authentication technique for cluster based vehicular ad hoc networks (VANETs)," *Wireless Networks*, vol. 24, no. 2, pp. 373–382, 2018.

[9] M. Feiri, J. Petit, R. K. Schmidt, and F. Kargl, "The impact of security on cooperative awareness in VANET," in *2013 IEEE Vehicular Networking Conference*. IEEE, 2013, pp. 127–134.

[10] R. W. van der Heijden, S. Dietzel, T. Leinmüller, and F. Kargl, "Survey on misbehavior detection in cooperative intelligent transportation systems," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 779–811, 2018.

[11] J. Zhang, K. Zheng, D. Zhang, and B. Yan, "AATMS: An Anti-Attack Trust Management Scheme in VANET," *IEEE Access*, vol. 8, pp. 21 077–21 090, 2020.

[12] J. Guo, X. Li, Z. Liu, J. Ma, C. Yang, J. Zhang, and D. Wu, "TROVE: A Context-Awareness Trust Model for VANETs Using Reinforcement Learning," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6647–6662, 2020.

[13] H. Hu, R. Lu, Z. Zhang, and J. Shao, "REPLACE: A reliable trust-based platoon service recommendation scheme in VANET," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 2, pp. 1786–1797, 2016.

[14] A. Kiening, D. Angermeier, H. Seudie, T. Stodart, and M. Wolf, "Trust assurance levels of cybercars in v2x communication," in *Proceedings of the 2013 ACM workshop on Security, privacy & dependability for cyber vehicles*, 2013, pp. 49–60.

[15] P. R. Zimmermann, *The official PGP user's guide*. MIT press Cambridge, 1995, vol. 5.

[16] M. Khodaei, H. Jin, and P. Papadimitratos, "SECMACE: Scalable and robust identity and credential management infrastructure in vehicular communication systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 5, pp. 1430–1444, 2018.

[17] B. Brecht and T. Hehn, "A security credential management system for V2X communications," in *Connected Vehicles*. Springer, 2019, pp. 83–115.