Security Overhead and its Impact in VANETs

Sebastian Bittl, Karsten Roscher, and Arturo A. Gonzalez
Fraunhofer ESK
Munich, Germany
{sebastian.bittl,karsten.roscher,arturo.gonzalez}@esk.fraunhofer.de

Abstract—Vehicular ad hoc networks (VANETs), often called Car2X communication systems, are about to enter the mass market in upcoming years. They are intended to increase traffic safety by enabling new safety critical driver assistance systems. This also means that strong security mechanisms are required to safeguard communication within VANETs. However, standardized security mechanisms lead to significant overhead in terms of bandwidth requirement and delay. Prior work has focused on reducing the overhead by advanced strategies for pseudonym and authorization authority certificate exchange. However, we find that this is not enough to enable reliable message exchange in VANETs. Various other sources of overhead caused by security mechanisms in VANETs are identified in the provided analysis. Thereby, we find cross layer and cross message dependencies. In combination with the non-fragmentation property of VANET messages, such dependencies are discovered to lead to massive dropping of packets due to maximum size violations at low protocol layers. Thus, we develop a method for cross layer on demand content assembling for VANET messages, which can avoid the size limit violations without preventing individual layers from disseminating their variable length data sets.

I. INTRODUCTION

In the wake of upcoming large scale deployment, Vehicular ad-hoc networks (VANETs) are gaining increased attention. Applications based on such communication systems promise to enhance safety of driving by enhanced awareness of the environment of participating ITS-Ss (Intelligent Transport Systems - Stations). Important progress has been made by standardization within the European ETSI ITS and US Wireless Access in Vehicular Environments (WAVE) frameworks. Security of these systems is of particular concern, due to the proposed safety critical use cases and wireless information exchange.

A security system based on digital signatures has been developed with great similarities between ETSI ITS and WAVE [1], [2]. Thereby, each disseminated message is signed individually by its sender. Each ITS-S is equipped with so called pseudonym certificates (PSCs), which are used to sign outgoing messages. The used PSC is changed frequently to avoid tracking. Hence, on demand distribution of PSCs among cooperating ITS-S is required. To allow validation of PSCs, these are part of a certificate chain with authorization authority certificates (AACs) validating PSCs. Such AACs are secured via root certificates, which are known by each ITS-S. AACs are distributed on demand between ITS-Ss, but many ITS-Ss share the same AAC, e.g., all vehicles from the same manufacturer.

Unfortunately, developed security systems have been shown to introduce significant overhead into VANETs. Thereby, severe performance degradation in terms of channel load and average communication distance have been discovered [3], [4]. Thus, different mechanisms to reduce the overhead of security mechanisms have been studied. However, prior work has focused on dedicated sub-topics, e.g., PSC exchange, regarding performance of ITS security. No in detail study of all aspects of introduced overhead and possible dependencies between them has been published yet. Especially, cross layer and cross message type performance has been hardly considered. Thus, we provide an analysis of these important topics in this work.

Thereby, we find that uncoordinated inclusion of optional data sets on different protocol layers, in combination with large size optional data sets and lack of message fragmentation support, leads to a high probability of violating rigid maximum packet size limits, even for standard ETSI ITS messages. Currently, ITS-G5 limits packets to about 650 bytes payload of the access layer [5]. Affected packets are discarded at the access layer, which prevents ITS-S from sending them. To overcome the found design weakness we propose a generic cross layer content aware message assembling method.

The further outline is as follows. Firstly, Section II provides an overview of related work. In Section III an analysis of different sources of overhead in regard to VANET security is provided. Based on the obtained results, Section IV outlines a denial of service attack emerging from uncoordinated cross layer data set inclusion. Afterwards, possible countermeasures to the found design weakness are discussed in Section V. Finally, Section VI provides a conclusion about achieved results alongside with possible topics of future work.

II. RELATED WORK

An overview of currently standardized VANET security features, especially based on ETSI ITS is provided in [6]. ETSI ITS as well as WAVE use a security envelope to secure each message transmitted by an ITS-S. Thereby, the network layer payload (excluding the basic so called common header) is handed over to a cross layer security entity which embeds this payload into an envelope consisting of multiple dedicated header and trailer fields [1], [2].

Evaluation of the performance impact of message size increase caused by overhead from security mechanisms on the ETSI ITS network level showed a significant influence [7]. However, only protocol layers up the network layer have been considered and higher layers have not been looked at.

Different sources of overhead caused by ITS security mechanisms on the performance of VANETs have been identified. An overview is given in the following alongside with references to prior work in the individual domains.
1) Bandwidth requirement increase or restriction of data size for other protocol layers by
   a) data encoding affecting the encoded size of the envelope [8],
   b) inclusion of extra data sets, e.g., on demand included PSC, [3], [4], [9]–[12],
   c) digital signature algorithm [6].
2) Delay build up by
   a) channel access (rises with channel load / ITS-S’s bandwidth requirement) due to used CSMA-CA,
   b) pure transmission time,
   c) authentication delay either from signature verification [6] or from discarded packets due to missing security parameters (see also point 3),
   d) data encoding affecting processing time [8].
3) Cryptographic packet loss, covered for CAMs [11], [12],
4) Storage space [13] and
5) Pseudonym changes.

The broadcast methods used for typical VANET message dissemination (excluding hardly used single hop uni-cast) use no re-transmissions due to packet loss. This holds for ETSI ITS Cooperative Awareness Messages (CAMs) and Decentralized Environment Notification Messages (DENMs) as well as for WAVE Basic Safety Messages (BSMs). Thus, this kind of delay does not occur for the greatest share of messages in current VANET approaches.

In prior work the different sources of overhead were mostly studied separately. However, there are clearly mutual influences between them. Thus, Section III provides a detailed analysis of the different given sources of overhead and their cross influences on each other.

III. ANALYSIS OF OVERHEAD CAUSED BY ITS SECURITY

In the following the different aspects of security caused overhead mentioned in Section II are looked at in detail.

A. Impact of ITS Security on Message Length

It has been shown that increased data length of the security envelope leads to significant increases in channel load and thereby decrease in communication range [3], [4]. Two main sources of data length increasing have been identified. The first one is applied message encoding rule, e.g., binary or ASN.1. A significant impact on achievable message size has been shown, with so called binary XML encoding being able to reduce message size compared to standardized binary encoding [8].

The second and even more significant source of message length variation caused by the security envelope is inclusion of optional data fields, which is mainly the PSC [3], [4], [8]. Additionally, the so called PSC request list is also of varying length, but even its maximum size is much smaller than the size of a PSC [1]. Thus, different schemes for efficient distribution of PSCs within VANETs have been studied [3], [4], [9]–[12].

Currently, ETSI ITS and WAVE use ECDSA for signing of messages [1], [2]. Usage of message recovery signatures has been suggested to reduce message size, e.g., by German Bundesamt für Sicherheit in der Informationstechnik (BSI), but no further work on this topic has been done so far.

Prior work has concentrated on channel load increase due to increased message size caused by security overhead. Another aspect not considered so far is hard limitation of usable message share by other layers.

Typically no fragmentation of higher layer data, e.g., facility layer CAMs, is done at the network layer in VANETs. Thereby, it could be ensured that the overall packet does not violate maximum packet size restrictions enforced by the access layer [5], [14]. In contrast, this is a standard procedure, e.g., for IP-based networks. Instead, VANET protocols specify a maximum packet size, e.g., ETSI ITS via distributed congestion control (DCC) rules, at the access layer and any packet exceeding the limit is dropped [5]. To avoid such kind of message drops, a dedicated layer would have to make sure to restrict its data size to always allow lower level layers to include their maximum amount of data, e.g., protocol headers. However, higher layer entities should not care about the underlying communication technology (e.g., DSRC or LTE-A) and the corresponding maximum message length, which may vary in dependence of the used radio technology.

ETSI ITS DCC rules for the ITS-G5 access layer limit the maximum channel access time. Thereby, the maximum message size at this layer is restricted to about 650 bytes [5].

An in depth analysis of current ETSI ITS systems using ITS-G5 shows that facility layer entities, e.g., the CAM facility, do not ensure to always leave enough spare capacity within messages to lower layers. Moreover, there is no interface from whom any other layer above the access layer could obtain the information on how much spare capacity it has to leave for lower layers. Especially, inclusion of the PSC by the security entity is critical as shown by experiments conducted with the ezCar2X framework [15]. Thereby, we obtain that a CAM including a maximum size low frequency container (in addition to mandatory data fields) yields a CAM using 389 bytes leaving only 261 bytes of spare capacity to lower layers. Thereby, all optional data fields in the low frequency container have been used, as allowed by the corresponding standard. In case the security entity includes its PSC, it requires to use 221 bytes. Also taking into account other layers’ headers this leads to significant problems as outlined in the following.

1) Data Fields and Size Ranges: The following standard ETSI ITS protocol stack entities use variable length data sets:

   1) Facility layer containing among others
      a) CA basic service generating CAMs [16],
      b) DEN basic service generating DENMs [17].
   2) GeoNetworking [14],
   3) security entity [1],
   4) applications using BTP.

More facility layer entities will probably be added in the future. Sizes of data fields used by these entities are given in Table I.

<table>
<thead>
<tr>
<th>Facility Layer Entities</th>
<th>Size (bytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA basic service</td>
<td>389</td>
</tr>
<tr>
<td>DEN basic service</td>
<td>221</td>
</tr>
<tr>
<td>GeoNetworking</td>
<td>261</td>
</tr>
<tr>
<td>Security entity</td>
<td>221</td>
</tr>
<tr>
<td>Applications using BTP</td>
<td>389</td>
</tr>
</tbody>
</table>

For CAMs, a message with 42 bytes holds only mandatory data. 389 bytes contain mandatory data and a low frequency container, and up to 428 bytes are used when all data fields are set in high frequency, low frequency and public transport
The payload size required by DENMs are even higher compared to the ones for CAMs, except of the case of a minimum size DENM. However, a minimum size DENM does not make sense to be used in practice, as it only holds a management container and no information about the event itself.

Two figures are given for a common size DENM in Table I. The smaller one is obtained from a DENM holding situation container (LLC) and Medium Access Control (MAC)) are constant, for CAMs (single hop broadcast, always used for CAMs) and GBC (geo-broadcast, always used for DENMs) as well as on the access layer (Logical Link Control (LLC) and Medium Access Control (MAC)) are constant, for DENMs the size is always 233 bytes at least and for all remaining message types 230 bytes at least [8]. Each of the given values for a CAM can be extended by up to 20 bytes in case the optional so called certificate request list with up to six entries is present. This list is not used for other message types. To obtain minimal and common values the mandatory header fields for the security envelope and the certificate have been used.

Moreover, the ETSI ITS standard for security data formats allows to include many more data fields in a certificate, e.g., extra validity restrictions [1]. These have been additionally used to obtain the worst case size in Table I. Thereby, two ITS service specific permission (one for CAM and one for DENM, 2 · 4 bytes) and a location restriction defined via a polygonal region (98 bytes) have been used. All other location restrictions yield less message size.

2) Consequences of the Found Data Sizes: The found worst case size of the security envelope of CAMs clearly shows a design weaknesses of the system. With a security envelope of size 576 bytes it is not even possible to send a BTP packet without any payload, as the sum of all protocol overheads already exceeds the maximum message size. A main contributor is found by the polygonal region validity restriction consisting of 12 geographic positions, with 4 bytes each. Thus, this kind of validity restriction should be removed as it renders the remaining system useless, unless the allowed maximum messages size is significantly increased.

As one can see from the data in Table I, layer two to four overhead for CAMs commonly sums up to either 178 bytes without included certificate, 303 bytes in case of an included PSC and even 436 bytes in case of an included certificate chain.

As calculated above, in case of a present low frequency container a CAM can leave only 261 bytes to lower layers, which clearly only works in case the security entity does not include a certificate. However, current standards do not specify any kind of data length aware certificate inclusion. Hence, violation of the message length limit and thereby discarding of the messages at access layer level can occur.

Moreover, in case of a special vehicle role, e.g., public transport, the corresponding extra container (public transport container) is present in the CAM. This will lead to a situation in which the CAM as generated by the facility layer is even longer than the mentioned 389 bytes. Thus, always exceed the message size limit at the access layer.

In case of low CAM generation rate of 1 Hz, all the mentioned data fields (CAM containers and PSC) will always be present in each generated message according to present standards [16]. This can lead to a situation in which the ITS-S will never be able to send any CAM, as all generated CAMs are dropped at its access layer. Thus, this clearly has to be regarded a severe design weakness of the current ETSI ITS
In presence of an attacker who wants to perform a denial of service (DOS) attack on a VANET the situation is even worse, as explained in detail in Section IV. ITS-S behavior in both cases clearly shows that the approach which has been used so far has to be replaced with an improved alternative. Thus, such an alternative is developed in Section V.

Basically, three different possibilities exist to overcome the found message length problem without changing the sporadically included content itself. Theses are

1) increasing the maximum allowed message size at the access layer by either
   a) increasing the maximum air time $T_{\text{air}}$ (as defined in [5]) of a single message, or
   b) increasing the fixed transmit data rate of the control channel at the physical layer, or
2) allow fragmentation of high level messages,
3) to coordinate inclusion of sporadically included data set between different protocol layers to share message content resources more efficiently.

Possibilities 1a or 1b would significantly change the characteristics of a VANET on the wireless channel. With increased $T_{\text{air}}$ the probability of packet collisions and the channel load caused by a single transmitted packet both rises alongside. Increasing the transmit data rate on the physical layer will clearly make the system less robust against common challenges of wireless communication, like signal distortion. Thus, both mechanisms can be expected to significantly reduce the communication range of ITS-Ss. Hence, we do not recommend to use them.

Fragmentation in VANETs is studied in [20] assuming packet reception is acknowledged by receivers. However, this is not the case in broadcast mode of current ETSI ITS and WAVE approaches. Reference [21] studies the trade off between long packages with no or low amount of fragmentation and many short packages resulting from massive fragmentation. Thereby, it is found that optimal package length depends on traffic conditions and should be smaller than 1000 bytes for typical traffic densities. Thus, significantly increasing $T_{\text{air}}$ in VANETs seems infeasible to overcome the found package length issue. Moreover, the influence of fragmentation on delays for information dissemination and thus on cooperative awareness quality of ITS-Ss have not been studied. However, increasing the amount of packet transmissions by fragmentation will increase the channel load on the highly bandwidth restricted single control channel. Thus, inclusion of fragmentation support into current VANET approaches seems infeasible to overcome the found package length issue.

Instead, a cross layer content aware message assembling strategy as suggested as countermeasure no. 3 is recommended. Hence, such a strategy is introduced in Section V.

### B. Cryptographic Packet Loss

Cryptographic packet loss is caused by discarding of received packets due to missing cryptographic parameters for their verification. In VANETs using sporadic distribution of PSCs, i.e., the PSC is not contained in every single message, the unavailability of a PSC is regarded a the only source of such packet loss in prior work.

However, with the introduction of on demand distribution of not only the PSC but also the authorization authority certificates (AACs) used to secure the PSCs another source of cryptographic packet loss has been introduced. Even if the PSC is available, it can still be unusable due to a missing corresponding AAC, which makes it impossible for the receiver to verify the PSC. Thus, also the message signed by the PSC cannot be verified. Hence, the message is discarded. This has not been considered in prior work.

The impact of the found second source of cryptographic packet loss can be significant, not only for CAMs but also for other messages due to cross message dependencies as outlined in greater detail in Section IV-B.

### C. Impact of ITS Security on Data Reception Delays

In general delay caused by the security entity will lead to delayed delivery of received data to higher level entities, like other low level delay sources. Clearly, cooperative awareness of the vehicular environment will suffer from such delays. Thus, ADAS build on top of this data source will have to deal with less up to date data.

The channel access mechanism in VANETs typically follows the CSMA-CA scheme, e.g., in ITS-G5 as well as 802.11p, which are used in ETSI ITS and WAVE, respectively. Increased channel load is known to significantly increase channel access time for systems using CSMA-CA. Thus, the message size increase due to security mechanisms outlined in Section III-A can be expected to also increase the data transmission delay caused by increased channel access time. Channel access delay is studied for VANETs in detail in [22].

Pure transmission time of packets increases alongside with their size. However, due to relatively low message size on the physical layer, this delay can be considered negligible in comparison to the other given sources of communication delay.

Authentication delay due to runtime of message verification algorithms is still an issue as outlined in [6]. However, delay occurring due to discarded packets (see also Section III-B) is a far bigger issue, as the minimum delay caused by a discarded CAM is 100ms, i.e., the minimum sending interval between two messages of the same kind. With higher repetition intervals, the delay can be significantly bigger, e.g., up to 1s for CAMs (neglecting additionally possible packet loss).

### D. Storage Space

An ITS-S needs to store confidential cryptographic material, e.g., private keys for its PSCs. Thus, a hardware security module (HSM) is required to protect such data from unauthorized access [6], [18]. However, the costs for such an HSM typically rise significantly alongside with required storage space. General requirements on the used HSM can be found in [23].
However, prior work has focused on the need to store fixed size own cryptographic material within the HSM. This assumes that cryptographic parameters of other ITS-Ss, e.g., public keys contained in received PSCs, do not need to be stored in a tamper proof manner. However, the PSC is not contained in each CAM. Thus, receivers have to store it at least temporarily. With important progress being made in the area of in-vehicle attacks (see e.g. [24] and references within), buffered PSCs should be protected to prevent an attacker from modifying them. By doing so, the attacker could foil the input verification of the protocol stack and pass arbitrary data to higher level protocols. This should clearly be avoided.

Thus, received PSCs and AACs should be buffered within the HSM. With a vehicular environment of about 200 vehicles within communication range in high density traffic scenarios [6], the HSM should at least provide the possibility to store this amount of PSCs. Moreover, a maintenance strategy for the PSC buffer will be required. Typically timeout based removing of entries is used for neighborhood tables within VANETs. The PSC buffer can be regarded as being some kind of such a neighborhood table, thus we suggest timeout based removing of entries. In case the buffer has reached its maximum size, the entry for which most time has passed since a message signed with it was received should be replaced, when a new PSC has to be stored. This assumes that a node from whom no message has been received for the longest time has the highest chance of having moved out of communication range.

Buffer maintenance is required at least with each received message. Hence, some extra computational effort will be caused increasing the overhead of the security functionality. To determine optimal parametrization of the outlined strategy for commonly encountered traffic scenarios future work is required, as such evaluation is not in the focus of this work.

E. Pseudonym Changes

To protect privacy of drivers, the PSC used by an ITS-S is changed frequently. Different strategies for when and how to change the PSC have been developed [25], [26]. However, additional overhead introduced into the security system has hardly been looked at, except from the need to provide the different PSCs to each ITS-S (see also Section III-D).

In case of a pseudonym change, the considered ITS-S will always include the PSC in the next sent CAM [1]. Thus, one can expect that cryptographic packet loss (see also Section III-B) caused by pseudonym changes is low. However, all ITS-Ss not receiving the first CAM after the pseudonym change, e.g., due to collisions on the wireless channel, will have to perform the standard PSC acquiring procedure as for other newly detected nodes. Thus, also cryptographic packet loss will occur for them with high probability. A side effect of PSC changing and emission is that all receivers will detect a new node, which is regarded as an implicit PSC request [1], [12]. Hence, all receivers will include their PSC in their next CAM. However, such inclusion is not required as the ITS-S changing its PSC already knew about the PSCs of its neighbors before. Thus, all the PSC inclusions of the neighbors, which did not change their PSC, have to be regarded as pure overhead.

To avoid the unnecessary PSC transmissions, other ITS-Ss would have to know that the new node appeared due to a PSC change. However, explicitly signaling this information would provide an attacker tracking ITS-Ss with even more information. The uncertainty whether an ITS-S is really new or just changed its PSC would be totally removed. Thus, privacy of drivers would probably suffer from such an approach. Hence, we do not recommend to use it.

F. Cross Layer Overhead Reduction

The network and facility layer of ETSI ITS both use IDs being derived from the pseudonym ID. Thus, such IDs to not need to be stored in corresponding data sets. The receiver’s security entity already identifies the sender. This information can be passed to upper layers together with other sender information like service specific permissions (SSPs) [17]. Thus, IDs could be removed from network and facility layer data sets. Thereby, the network layer header can be shortened by six bytes and facility layer messages by up to four bytes (depending on the chosen address, due to UPER ASN.1 encoding [27]).

This would tighten dependencies between protocol layers, i.e., the network layer would have to adjust its payload in dependence of a (not) used security mode. However, current standards specify to secure all messages and derive all IDs on the distinct layers from the security entity’s pseudonym ID.

IV. OVERHEAD LEADING TO DENIAL OF SERVICE ATTACK

Various kinds of attacks on VANETs have been found [28], [29]. Thereby, denial of service attacks caused by attackers misusing dedicated VANET features are of particular concern. For example, forced PSC inclusion or even forced certificate chain inclusion has been shown to significantly increase channel busy ratio worsening communication conditions seriously [12]. In contrast to jamming, the attacker can target areas even bigger than his own communication range.

A VANET can still basically work in the presence of increased channel load. While communication range will decrease and collisions on the wireless channel will occur more frequently [3], [4], basic message exchange within close surrounding of ITS-Ss can still be expected to work.

In contrast, the uncoordinated nature of optional data set inclusion throughout the different ITS-S’s protocol layers, in combination with above described message size increase attacks, can strip an ITS-S from the possibility to send dedicated data sets altogether. The reason for this is above described behavior of current ETSI ITS and WAVE facility layer entities. These do not guarantee to generate messages short enough to allow lower level security functionality to use its worst case size security envelope at all times. As the security entity is not required to check for length of its generated output, it can exceed the maximum message size defined at the access layer. The attacks on certificate (chain) dissemination force the security entity to use the maximum length security envelope for every CAM. Thus, every data set from higher layers which is not short enough to be combinable with this long security envelope will be dropped by the access layer.
As outlined in Section III, all messages containing the low frequency container (distributed with 2 Hz frequency) and a security envelope including a certificate chain will be dropped by the ITS-G5 access layer. Thus, an attacker can prevent ITS-Ss from distribution of CAMs including a low frequency container. This clearly affects applications using data from this dedicated container, e.g., the exterior light status.

### A. Presence of Extra Optional CAM Containers

Moreover, other optional containers within CAMs apart from the low frequency container, encapsulated in the so-called special vehicle container which holds, e.g., a public transport container, are included with the same frequency of 2 Hz. As they are all managed by the same CA service, the timeouts for including every one of them will happen at the same time. Thus, they will always be included at once in a CAM. Hence, the CAM will be even longer compared to the case of only including the low frequency container, which already lead to dropping of the messages by the access layer.

The described behavior of the communication stack enables an attacker carrying out the attack on certificate chain dissemination to force a targeted ITS-S to never send CAMs containing optional containers. All such CAMs will be dropped at the access layers of the attacked ITS-Ss. Thus, the attacker can carry out a DOS attack on all applications, which require data from optional CAM containers. For example, a traffic light management system using the public transport container within CAMs to prioritize access of public transport vehicles to crossings can be put out of order.

For low CAM emission frequencies of 1 or 2 Hz the situation is even worse. Each CAM will include all optional containers. Thus, all of them will be to long to allow inclusion of the certificate chain under the security entity. Thus, all messages will be dropped by the access layer and the ITS-S will not be able to send out any message. This does not only affect the system in presence of an attack, but it will also permanently prevent regular emission of the certificate chain in case of low CAM generation frequency. Therefore, affected ITS-Ss are unable to distribute their AAC. Thus, this has to be considered as a major design problem of the current ETSI ITS system.

### B. Cross Message Dependency

Regarding cross message dependencies, the lack of a possibility to distribute AACS, found in Section IV-A, is a major problem. All messages except of CAMs are assigned to one of the security profiles DENM or generic. These to profiles do not allow AAC distribution, but require PSC inclusion.

Thus, other ITS-Ss can only receive data secured by these profiles in case of successful exchange of CAMs in advance. However, such exchange is not possible in some cases as outlined before. Thus, in such cases no exchange of any message is possible between affected ITS-Ss.

## V. CROSS LAYER AWARE MESSAGE ASSEMBLING

In order to enable cross layer aware message assembling, knowledge about message size limitations has to be available to all layers together with the minimum part of a message required by each layer. Thus, the ITS management entity should collect the requirements from the individual layers and disseminate it to all layers. Thereby, the available message size is to be obtained from the access layer chosen by the network layer. In a hybrid communication scenario dissemination technology can be chosen for each message individually. Thus, the available message size may also vary between packets.

In order to enable this approach each layer has to provide information about the minimum size of data it has to include in a message (\(m_{\text{message type}}^{\text{layer}}\)). However, this limit may vary between individual packets. To keep the individual layers clearly separated, this information should be available through the cross layer ITS management entity. For example, it should provide the facility layer CA basic service information about the minimum data size required by lower layers for the next CAM to be sent. The facility layer does only need to know about this sum, so the individual composition of the reserved data size should be abstracted by the management entity.

Within ETSI ITS and WAVE the minimum amount of data size required by a protocol layer only depends on the message type, e.g., GeoNetworking uses seven different but fixed formats/sizes of the extended header [14]. Fortunately, no cross layer dependencies between the presence of optional fields on the different layers exists. For example, the PSC is included independently of the presence of optional containers in a CAM or DENM. Thus, once an entity has triggered generation of a new message, the management entity can gain knowledge about lower layers message part consumption requirements just based on the message type. One should note that, this may not be true for other protocol stacks, which significantly increases the effort of determining lower level requirements.

We discuss two different approaches to the content and length aware message assembling problem. The first one is a simple top down approach, while the second suggestion uses bottom up reservation of message shares.

### A. Strict Top Down Approach

The simplest approach is to just have each layer take its share of message size without taking variable length of lower layers into account. Thereby, it has to make sure that the remaining unused share of the message is at least long enough to hold the minimum length data fields from lower layers. Thus, lower level entities have to live with the space left over by higher level entities.

As a drawback, this approach can lead to starvation of lower layers. In case high layers always only leave the minimum required message share to lower layers, these lower layers will only be able include their minimum data set. Thus, distribution of extra (configuration) data is not possible, even in case this would be required to support further communication.

Regarding ETSI ITS, this approach will lead to the following scenario. A CAM generation frequency of 1 or 2 Hz will always lead to the inclusion of all used optional containers. Thereby, at least the low frequency will be included by all types of ITS-Ss. Thus, the security entity will never be able to include the certificate chain and dissemination of authorization authority certificates will not work at all. Hence, an ITS-S which cannot...
verify the PSC of another ITS-S, due to missing knowledge of the corresponding AAC, will never achieve awareness of that other ITS-S on the facility layer. However, the ITS-S will be able to send out messages in contrast to the case without maximum length awareness being currently standardized.

Due to the discovered drawback of this mechanism, another approach which uses length reservation from lower layers at higher layers is discussed in the following.

B. Top Down Approach with Bottom Up Reservation

Basically, this approach works like the one from Section V-A. However, we add the possibility for lower protocol levels to reserve space within messages at higher levels.

As lower level layers, e.g., the network layer, should not be required to know about the existence of specific higher layers, they should not reserve spare message length directly at this higher layers. Instead, we propose a publisher and subscriber mechanism run by the ITS management entity as follows. Each layer, except of the highest level one, which needs more than the minimum amount of message share informs the management entity about its required data length. The management entity will then inform all subscribers about this kind of request. All layers which add content to a message have to subscribe at the management entity to be informed about such announcements.

After having received a data length reservation announcement, the higher level entities should make sure that they leave the requested spare data size to lower layers. One could try to introduce some kind of prioritization into that procedure. Thereby, the higher layers could ignore the reservation requests in some cases. However, this introduces dependencies between different layers which will be hard to maintain, especially in case of hybrid communication scenarios with a common higher layer and multiple lower layers, e.g., at the access layer level.

1) Message Part Allocation Algorithm: For the design of the reservation algorithm, one has to take into regard when functionalities at different layers know which amount of data they are about to sent in the next message. Thereby, one can differentiate two characteristics, which are

1) on the fly decision, i.e., the decision is only made when a new message gets assembled, or
2) asynchronous decision, i.e., the decision can be done in advance, e.g., triggered by dedicated received messages.

In case only strategy no. 1 is used, the system will be equal to the one in Section V-A. Higher level entities just use as much as they want to do and lower levels have to cope with the remaining spared message part.

Therefore, strategy no. 2 should be used whenever possible. This means, that whenever a criteria for including optional data gets fulfilled, e.g., a timeout for including cyclically distributed data happens, the corresponding required message parts gets reserved at higher layers.

One can assume that an entity which is not able to transmit its optional data in the current message will try to do so again in the next one. Additionally, a communication connection is typically build up in a bottom up nature, i.e., the link has to be maintained at the network layer to allow message exchange at higher layers. Thus, a failed message part allocation should be repeated straight after the message for which the failure occurred was sent at the lowest layer. Thereby, the lowest layer is allowed to perform its reservation first. Thus, the probability of successful message part allocation will be high, as other entities will have this possibility later.

To avoid that the asynchronous reservation mechanism blocks the bottom up reservation mechanism from working, such requests should only be allowed in case no message sending is currently due at layers below the requester.

The above described message assembling algorithm is illustrated in Figure 1 for the case of a CAM.

As displayed in Figure 1, after a CAM from a formerly unknown node is received, the security entity should send its PSC in the next CAM. Thus, it requests to allocate the corresponding size for the next CAM at the management entity.

A typical inclusion sequence of the low frequency container in side CAMs and the PSC within the security envelope in case of used cross layer coordination is illustrated in Figure 2. Thereby, the intention to include optional data in the next message is given by a dashed line. Massive dots show the points in time at which a new CAM gets assembled.

Fig. 2. Cooperative inclusion of sporadically distributed data used by CA basic service (upper part) and security entity (lower part).
In the example from Figure 2, an ITS-S starts up at $t = 0$. The CA basic service and the security entity wish to include their optional data in the first CAM. The facility layer can do so first ($t = 100\text{ms}$), while the security entity has to wait for the second generated CAM ($t = 200\text{ms}$). At $t = 600\text{ms}$ the CAM includes the optional data from the facility layer again. A message part reservation by the security entity happens at $t = 1030\text{ms}$, which delays inclusion of optional data at the facility layer for one CAM. Thus, the PSC is sent at $t = 1100\text{ms}$.

2) Starvation of High Layers: Unfortunately, with this approach starvation of higher levels is possible, e.g., in case of a present AAC dissemination attack CAMs can never include optional containers. Thus, all the information from that containers will not be disseminated and applications relying on them will not be able to work. However, this approach ensures that all communication up to a certain level (with in ETSI ITS the network layer security entity) will always be possible, which is a property not ensured by the proposal from Section V-A.

As described above, higher layers could reject message part allocations from lower layers to enforce inclusion of their own data sets. However, we propose to effectively limit data sizes at lower layers instead.

3) Summary: By using cross layer content aware assembling of ETSI ITS messages dropping of packets at the access layer can be avoided. Thus, the weaknesses of the standardized approach found in Section III-A2 and IV can be overcome.

VI. CONCLUSIONS AND FUTURE WORK

Future safety critical ADAS based on VANETs require efficient and reliable security mechanisms. However, these introduce many sources of extra overhead. Especially, current approaches not using cross layer coordinated message assembling have been shown to lead to various problems. These include denial of service attack vulnerabilities and failure of message transmission even in standard use cases, due to restrictive message size limits and lack of message fragmentation support.

To overcome the found design weaknesses we propose mechanisms to coordinate message assembling between protocol layers. Thereby, coordination is performed by the already existing cross layer ITS management entity. This mechanism can avoid the found design weaknesses of current VANETs.

Future work can study mechanisms to overcome the found weaknesses in general VANETs by developing usable message fragmentation support.

REFERENCES

[18] C2C-CC Basic System Standards Profile, CAR 2 CAR Communication Consortium Std. 000 042, Rev. 1.0.5, Jan. 2014.