A Security Analysis of the ETSI ITS Vehicular Communications*

Alexandru Constantin Serban[1,2], Erik Poll[1], and Joost Visser[1,2]

[1] Radboud University, Nijmegen {a.serban, erikpoll}@cs.ru.nl
[2] Software Improvement Group, Amsterdam {a.serban, j.visser}@sig.eu

Abstract. This paper analyses security aspects of the ETSI ITS standard for co-operative transport systems, where cars communicate with each other (V2V) and with the roadside (V2I) to improve traffic safety and make more efficient use of the road system. We focus on the initial information exchange between vehicles and the road side infrastructure responsible for authentication and authorisation, because all the security aspects for these interactions are regulated in the ETSI ITS standards. Other services running in vehicular networks are open to choose application-specific security requirements and implement them using features from the ETSI ITS standard. We note some possibilities for replay attacks that, although they have limited impact, could be prevented using simple techniques, some of which are directly available in the ETSI ITS standard.

Keywords: Intelligent vehicles · Security · Access Control.

1 Introduction

Adding cognitive intelligence to vehicles is considered to be the next evolution step in order to improve traffic efficiency and safety. One of the first abilities to be deployed for this is communication. Through communication, vehicles can exchange traffic updates with other vehicles or the road-side infrastructure to enhance their context awareness for more efficient and safer use of the road. . Applications which involve an exchange of information between two or more vehicles are called *co-operative driving applications*. A wide range of acronyms describe the communication between vehicles and other entities. *Vehicle-to-vehicle* (V2V) allows vehicles to talk with others and relay information in real time. *Vehicle-to-infrastructure* (V2I) allows vehicles to communicate with static stations such as traffic lights or weather stations. *Vehicle-to-everything* (V2X) incorporates all types of communication and serves as a generic acronym that will be used throughout this paper.

An example of co-operative driving is *platooning*; a scenario in which a string of vehicles autonomously follow a leader, by sharing acceleration and steering

* This research was funded by NWO through the i-CAVE project.

information. It has been shown that platooning can increase traffic efficiency and highway throughput by minimising the distance between vehicles [1, 2, 3]. Moreover, the feasibility of platooning was demonstrated through the Grand Co-operative contests [4, 5].

Another example of co-operative driving application is broadcasting of traffic events and emergency messages in highway settings. For example, a stationary vehicle at a potentially dangerous location can periodically broadcast a warning message to other vehicles, announcing its location and state. Other traffic participants can then use this information to plan new manoeuvres and avoid traffic disruptions.

The systems for co-operative applications support two communication models: a vehicle can exchange messages with other vehicles (V2V) or with the road-side infrastructure (V2I). V2V can be used in scenarios such as platooning or event broadcasting, while V2I allows a broader range of services such as authentication, regional updates distribution, or infotainment content delivery.

To meet fundamental security and privacy requirements, a complex software architecture and communication protocols are needed. Security plays a crucial role in co-operative applications because a security breach can easily lead to human casualties. Therefore, the international standard developing organisations have worked on technical standards intended to implement a unique and secure communication protocol that spans a broader region.

In Europe, the communication architecture and protocols are conceived and standardised by the *European Telecommunications Standards Institute* (ETSI), through the ETSI *Intelligent Transport Systems* (ITS) series of standards [6, 7, 8, 9, 10, 11, 12]. The acronym ETSI ITS will be used to indicate this collection of standards. When we target a specific document, we will explicitly mention its number (e.g. ETSI ITS 731). ETSI ITS is inspired by the IEEE 1609 family of standards developed and adopted in the US.

Previous work has shown that security requirements are often not met by early versions of communication protocols (e.g. think of SSL 1.0 or SSH 1.0). Moreover, since standard descriptions are complex and (unclearly) written in natural language, software implementations are often flawed [13, 14, 15].

The goal of this paper is to analyse the security requirements of ETSI ITS communication standard for co-operative vehicles. We focus on the initial communication between a vehicle and the road-side infrastructure that precedes a vehicle's access to a vehicular network. Several weaknesses that allow message replay and can lead to *denial-of-service* (DoS) attacks are identified. While the impact of DoS attacks on traffic safety is low, this paper shows the recurrent issue of protocol specifications failing to meet security requirements is perpetuated in the automotive industry as well.

The rest of the paper is organised as follows. Section 2 provides background information about the ETSI ITS communication model and software architecture. Section 3 presents the results of our security assessment. Related work is presented in Section 4, followed by conclusions and future research in Section 5.

2 Background

ETSI ITS publishes a collection of standards for co-operative driving applications. They are divided in three stages that address different concerns. Stage 1 introduces the 'macro' economical and strategic requirements and 'micro' system and standard requirements [7]. Stage 2 gives a detailed specifications of interaction patterns between vehicles and roadside infrastructure [8, 9, 10, 11, 12]. Stage 3 provides a mapping to concrete IEEE 1609 message types and presents custom extensions [16, 6]. This is the description closest to implementation.

The ETSI ITS communication stack is very similar to the OSI model, as illustrated in Figure 1, with the OSI stack in grey to the right of the ETSI ITS layers in colours.

In fact, ETSI ITS extends the OSI model by only adding two orthogonal layers: management and security, which provide cross-layer services to all levels of the stack. The security layer provides services to ensure confidentiality, integrity and availability. The management layer implements all operations that support certificate management, a necessary step for secure communication [9]. All these services can be independently accessed by any other layer.

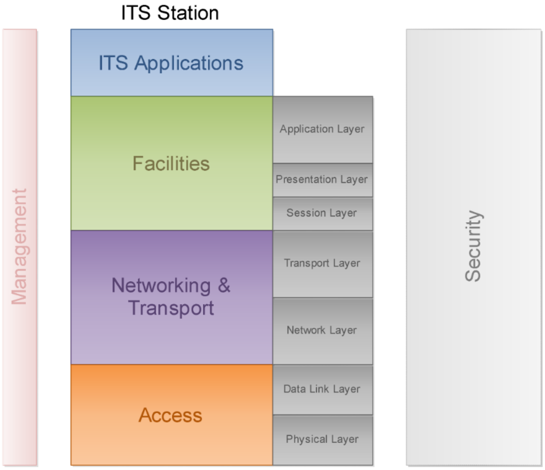

Fig. 1. The ETSI and OSI communication stacks [9].

Co-operative driving applications are deployed at the application layer, i.e. the highest layer in Figure 1, and make use of the underlying communication facilities. ETSI ITS does not specify security requirements for applications, but only provides the infrastructure that applications can use to meet their security requirements.

Overall, we distinguish between two actors involved in communication:

1. *Infrastructure stations*: communication stations that do not act in a personal role and provide communication support.
2. *ITS Stations* (ITS-S): functional entities that act in a personal or public interest and correspond to personal or public assets such as cars, ambulances, communication poles, toll payment booths, etc.

Moreover, we distinguish two steps in the communication protocol:

1. Access Control: an exchange of information between an ITS-S and the infrastructure in order to prove identity (authentication) and gain access to a specific service (authorisation), and
2. Service consumption: communication between ITS-S in order to exchange traffic or infotainment information.

This paper focuses on the access control patterns because the security requirements for different services are not standardised and are set by the service providers.

An overview of the access control flow is depicted in Figure 2. In order to get access to the communication infrastructure and services, a vehicle must, at first, contact an *Enrolment Authority* (EA) and *authenticate*. The EA answers with a set of pseudonymous certificates that help preserve the true identity of a vehicle and, thus, the owner's privacy. In this case, the EA resembles the road registration authority and it's able to validate that a vehicle can be trusted to function correctly within the network.

The next step is to request permission to access a service. For this, a vehicle contacts an *Authorisation Authority* (AA) using one of the pseudonymous certificates representing a temporary identity. In response, it receives a set of certificates, one for each requested service. Finally, a vehicle uses such a certificate received from the AA to access a service. In this case, the AA *authorises* a vehicle to use a service.

The ETSI ITS standards describe requests similar to database functions *create, read, update* and *delete* (CRUD) for all certificates provided by the EA or the AA. For each operation, the message exchange is provided as a stage 2 description in [8] and as a stage 3 mapping to IEEE 1609 in [16].

As general security requirements, ETSI states that it is necessary to ensure that data can not be linked to any individual, so that no personally identifying information is leaked when using the services. Moreover, ETSI requires that no authorised party are allowed to deduce the location or identity of an ITS station by analysing communication traffic which flows to and from an ITS vehicle.

In general, a security analysis includes, but is not limited to, an assessment of confidentiality, integrity, availability or freshness of information during an exchange between two parties. The following section investigates these aspects for the ETSI ITS communication.

3 Findings

This section presents an analysis of the communication for access control and identity management as specified by ETSI ITS. Three entities are involved, il-

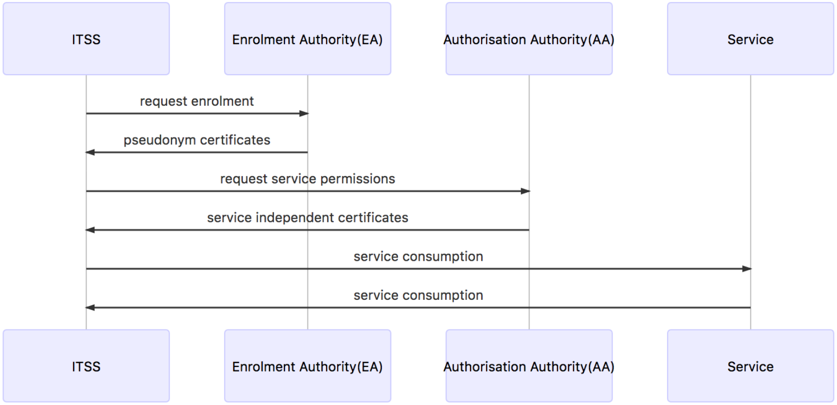

Fig. 2. Access control flow diagram in ETSI ITS vehicular networks.

lustrated in Figure 2: (1) a vehicle, referred to as ITS-S; (2) the enrolment authority, or EA; and (3) an authorisation authority, or AA. Initially a vehicle request enrolment certificates from the EA. Afterwards, using one of the enrolment certificates, it requests permission to access a set of services from the AA and, finally, it gets to access a service.

The communication with service providers is not covered in this report because security requirements are service specific and not standardized. This model is called *verify-on-demand* (VoD): each service can individually request security checks from the security layer. The set of operations analysed are: request certificates (create), update certificates (update) and certificate revocation (delete).

We identify some weaknesses due to the lack of a cryptographic nonce in some communication requests. A nonce is as an arbitrary number specific only to one request, which ensures that old request bodies can not be used in replay attacks. It is usually implemented as a random number. It can also be implemented by a counter, but predictability of the nonce may then introduces weaknesses.

3.1 Communication of a Vehicle with an EA

The communication with an EA serves the purpose of issuing enrolment credentials for a vehicle. The EA can validate that a vehicle is trusted to function correctly and can access the network.

Obtain Enrolment Credentials. The *obtain enrolment credentials* request is initiated by a vehicle when it has no enrolment credentials for an operational region or at the beginning of its life cycle.

For this requests, the protocol ensures confidentiality, integrity, availability and freshness. However, this request is still worth mentioning because, unlike for the next requests discussed, here freshness is guaranteed by the use a cryptographic nonce, given as a *network challenge* in the request's body.

Update Enrolment Credentials. The *update enrolment credentials* request is initiated by a vehicle when it determines that the enrolment credentials can not be used (e.g. when a certificate expires).

For this request the protocol ensures confidentiality, integrity and availability. However, freshness is not guaranteed because there is no nonce in the request body. This means that an attacker can replay the same message again and again. An attacker who can eavesdrop on a request that results in a *reject* response, can re-use this response in future requests. To execute this attack, the attacker has to eavesdrop on the communication between a vehicle and an EA and understand that a vehicle requests a credential update. Later, the attacker can replay the same response. The attack assumes a vehicle will pass the same road segment – where the attacker was eavesdropping – when requesting credentials update or that an attacker can span a wider operational range. This assumption limits the impact of this replay attack.

Remove Enrolment Credentials. The *remove enrolment credentials* request is initiated by a vehicle when it leaves an operational region or when it wants to discard a pseudonym.

For this request the protocol ensures the same security requirements as the update enrolment credential request, i.e. confidentiality, integrity and availability, but again it fails to ensure freshness. The reason is the same: there is no nonce in the request body. An attacker can use the attack scenario as in Section 3.1. However, the probability that a vehicle passes the same road segment when requesting an enrolment revocation is small, given that the frequency of revocations is not high and often not correlated with the frequency of a vehicle passing through the same road segment.

3.2 Communication of a Vehicle with an AA

The communication with an AA occurs with higher frequency than the communication with an EA. The reason is that a vehicle will request authorisation tickets for a service before every usage. Moreover, the access to services expires faster than the enrolment credentials.

Similar to Section 3.1, all the requests made to AA include no nonce, allowing replay attacks. The impact and ways of mitigation are discussed in the following sub-section.

3.3 Discussion: Impact and Mitigations

To execute the replay attacks described above, an attacker needs to be in the vicinity of one or multiple static infrastructure stations. Since the frequency with which a vehicle requests the same information from a particular station is low, the impact of the replay attacks is also low. However, the findings presented in this paper illustrate a recurring problem with protocols specifications: it is not clear if the risk of these attacks is known and accepted or if the designers are unaware of it. In other words, it is not clear which security requirements are intended to be met by the various protocol requests. Such confusion can contribute to implementation flaws that can later have high impact [17].

The ETSI ITS specs do contain protection mechanisms against replay attacks, but omits them for some requests. These mechanisms are the inclusion of a cryptographic nonce (the so-called network challenge) in the request body, a sequence number, or a timestamp. We recommend to future developers to be aware of these mechanisms and use them appropriately.

A replay attack which contains certificates in the request's body can be partially protected against, namely if the implementation checks the certificate's expiry date. This way the implementation can detect replays. However, note that this does not work if the roadside replays 'permission denied' responses, as these do not contain any expiry date. Developers may not be aware of such mitigations or, even if they are aware, may simple forget to implement them. It is therefor important to mention these mitigations explicitly in the standard, so that developers can consider implementing them.

4 Related Work

Security in *vehicular ad-hoc networks* (VANET) attracted some early attention from researchers. However, since ETSI ITS is fairly new, with late edits still rolling on, there is little literature addressing it.

Bittl and Roscher analysed the complexity of VoD schemes in VANET [18]. Their analysis shows that VoD leads to a significant number of extra cross layer dependencies. Thus, the overall complexity of the ETSI ITS protocol stack is increased, while the separation of dedicated communication layers is reduced. Moreover, the number of interfaces that have to be protected against malformed inputs from wireless attacks is increased.

Bittl also analysed the security mechanisms from ETSI ITS and identified three main weaknesses [17]. Firstly, end-to-end multi-hop communication is not supported. This results in a single-hop distribution of certificates. Secondly, pseudonym management requires a dedicate start-up strategy after node start-up. Thirdly, basic data sets of time and position are acquired from sources lacking security mechanisms and are used in a partly inconsistent way.

Closest to our work is the research by Nowdehi and Olovsson [19]. They implemented the ETSI ITS 103 097 [6] *SecuredMessage* service and found it difficult, given the complexity of the specifications. They noticed they ended up with bugs in their implementation due to these complexities, and found that another open source implementation that they tested contained very similar bugs, suggesting that the SecuredMessage format is inherently tricky to implement Specifically, they criticise the specification for being very liberal and overly permissive – e.g. by allowing multiple payloads in a single message, each of which may then be encrypted or not, and allowing additional *HeaderFieldTypes* not specified in the security profile. Complexity and unclarity in protocol specifications is the root cause behind many security vulnerabilities, as highlighted by the LangSec paradigm [20], so we fully agree with the recommendations of Nowdehi and Olovsson to remove these unwarranted complexities from the standard.

Research into privacy aspects of ETSI ITS has led to a proposal for a privacy improvement using shorter-lived certificates that are issued beforehand and then activated later over a low-bandwidth channel, e.g. using SMS [21].

5 Conclusions and Future Research

We have presented an initial security analysis of the ETSI ITS communication protocol for VANET, focused on the access control communication patterns. The analysis uncovered several ways to perform replay attacks inside VANET. However, the power of these attacks is smaller than in other cases (e.g. a bank application, where a payment replay can cause a bigger damage).

Some weaknesses can be mitigated by extending the message types described by ETSI ITS. Replay protection mechanisms are specified in the standard, however, it is not clear how to implement them for access control requests. The use of a cryptographic nonce, or a special message type that includes timestamps or a sequence number will solve the problems described in this paper.

As Nowdehi and Olovsson [19] and others (e.g. [13, 14, 15]) show, protocol specifications are easy to miss-interpreted leading to flawed and possibly insecure implementations. Testing specification conformance, however, is no straightforward job, because specs are written in natural language and often omit important details. As future research we suggest a formal description and analysis of the ETSI ITS protocol. Moreover, the implementations could be tested through fuzzing [15] or state machine learning [22].

Security plays a crucial role in a series of changes in the automotive industry where there is critical impact on traffic safety [23]. Since systems security is often defined as an arms-race, designing, adapting and implementing a protocol is a never ending process.

References

1. A. Al Alam, A. Gattami, and K. H. Johansson, "An experimental study on the fuel reduction potential of heavy duty vehicle platooning," in *13th International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2010.
2. R. Janssen, H. Zwijnenberg, I. Blankers, and J. de Kruijff, "Truck platooning: Driving the future of transportation." `http://publications.tno.nl/publication/34616035/dLIjFM/janssen-2015-truck.pdf`, 2015. TNO Whitepaper.
3. A. Davila, E. del Pozo, E. Aramburu, and A. Freixas, "Environmental benefits of vehicle platooning," tech. rep., SAE Technical Paper, 2013.
4. C. Bergenhem, S. Shladover, E. Coelingh, C. Englund, and S. Tsugawa, "Overview of platooning systems," in *19th ITS World Congress*, 2012.

5. R. Kianfar, B. Augusto, A. Ebadighajari, U. Hakeem, J. Nilsson, A. Raza, R. S. Tabar, N. V. Irukulapati, C. Englund, P. Falcone, *et al.*, "Design and experimental validation of a cooperative driving system in the grand cooperative driving challenge," *IEEE Transactions on Intelligent Transportation Systems*, 2012.
6. ETSI, "ETSI TS 103 097 (V1.1.1) - security header and certificate formats." http://www.etsi.org/deliver/etsi_ts/103000_103099/103097/01.02.01_60/ts_103097v010201p.pdf, 2017.
7. ETSI, "ETSI TR 102 638 (V1.1.1) - vehicular communications; basic set of applications." http://www.etsi.org/deliver/etsi_tr/102600_102699/102638/01.01.01_60/tr_102638v010101p.pdf, 2009.
8. ETSI, "ETSI TS 102 731 (V1.1.1) - security services and architecture." http://www.etsi.org/deliver/etsi_ts/102700_102799/102731/01.01.01_60/ts_102731v010101p.pdf, 2010.
9. ETSI, "ETSI TS 102 940 (V1.1.1) - its communications security architecture and security management." http://www.etsi.org/deliver/etsi_ts/102900_102999/102940/01.01.01_60/ts_102940v010101p.pdf, 2012.
10. ETSI, "ETSI TS 102 941 (V1.1.1) - trust and privacy management." http://www.etsi.org/deliver/etsi_ts/102900_102999/102941/01.01.01_60/ts_102941v010101p.pdf, 2012.
11. ETSI, "ETSI TS 102 942 (V1.1.1) - access control." http://www.etsi.org/deliver/etsi_ts/102900_102999/102942/01.01.01_60/ts_102942v010101p.pdf, 2012.
12. ETSI, "ETSI TS 102 943 (V1.1.1) - confidentiality services." http://www.etsi.org/deliver/etsi_ts/102900_102999/102943/01.01.01_60/ts_102943v010101p.pdf, 2012.
13. D. Kaloper-Mersinjak, H. Mehnert, A. Madhavapeddy, and P. Sewell, "Not-quite-so-broken TLS: Lessons in Re-Engineering a Security Protocol Specification and Implementation," in *24th USENIX Security Symposium*, 2015.
14. E. Poll and A. Schubert, "Verifying an implementation of SSH," *WITS'07*, 2007.
15. J. De Ruiter and E. Poll, "Protocol state fuzzing of TLS implementations," in *USENIX Security Symposium*, 2015.
16. ETSI, "ETSI TS 102 867 (V1.1.1) - stage 3 mapping for IEEE 1609.2." http://www.etsi.org/deliver/etsi_ts/102900_102999/102940/01.01.01_60/ts_102940v010101p.pdf, 2012.
17. S. Bittl, "Towards solutions for current security related issues in ETSI ITS," in *Communication Technologies for Vehicles*, pp. 136–148, Springer, 2016.
18. S. Bittl and K. Roscher, "Feasibility of Verify-on-Demand in VANETs," 2016.
19. N. Nowdehi and T. Olovsson, "Experiences from implementing the ETSI ITS SecuredMessage service," in *Intelligent Vehicles Symposium (IV'18)*, IEEE, 2014.
20. E. Poll, "LangSec revisited: input security flaws of the second kind," in *5th Workshop on Language-Theoretic Security (LangSec'18), Security and Privacy Workshops (SPW)*, IEEE, 2018.
21. E. R. Verheul, "Issue First Activate Later certificates for V2X - combining ITS efficiency with privacy." https://eprint.iacr.org/2016/1158, 2016.
22. P. Fiterău-Broştean, T. Lenaerts, E. Poll, J. de Ruiter, F. Vaandrager, and P. Verleg, "Model learning and model checking of SSH implementations," in *Proceedings of the 24th ACM SIGSOFT International SPIN Symposium on Model Checking of Software*, ACM, 2017.
23. A. C. Serban, E. Poll, and J. Visser, "Tactical safety reasoning. a case for autonomous vehicles.," in *International Workshop on Connected, Automated and Autonomous Vehicles (Ca2V)*, IEEE, 2018.