Cryptographic Security for Vehicular Controller Area Networks

Extended abstract

Due to the growing complexity of modern automobiles, vehicular communication became an essential topic in the automotive industry. A wide range of solutions (wired buses as well as wireless approaches) were adopted to fulfill the communication needs of the automotive systems. The rapid evolution of these came at the cost of introducing a series of new possible attack surfaces, e.g., multimedia devices, wireless channels such as Bluetooth.
Until recently, the security of vehicular systems was not considered to be a major concern. This, however, changed as a series of attacks were reported by the scientific community. The increasing number of reported vulnerabilities and their impact on driver/passenger safety underline the importance of the subject and call for the development of secure communication inside vehicles.
Assuring the security of modern cars is not a straight-forward task. Buses used for building in-vehicle networks were designed to offer reliable communication, thus they are fitted with efficient mechanisms for error detection and error recovery. However, they are lacking support for assuring basic security objectives. Obviously, the best solution, from the security point of view, would be to devise a dedicated bus with built-in mechanisms for assuring security at the network layer rather than by the application layer. As such an approach would involve longer time for introduction and high costs (for creating specifications of the new protocol, producing corresponding transceivers and testing them properly) it is not considered as a viable option in the short term. In order to keep costs down and avoid the long process of adopting new protocols, currently used buses could be retrofitted with security mechanisms implemented at the application level.
Implementing security on microcontrollers used to build ECUs without increasing costs can be challenging as they are resource-constrained devices in what regards available memory and maximum working frequency. Due to low tolerance margins most of the automotive microcontrollers employed by the industry would likely support only algorithms based on lightweight cryptography. Employing microcontrollers enhanced with cryptographic co-processors can enhance the use of public key cryptography but this might not be acceptable as automotive component manufacturers try to keep production costs low to provide affordable end-products.
In this context, the purpose of this thesis is to address the subject of authenticated in-vehicle communication.

Background and motivation

Controller Area Network (CAN) is a differential serial broadcast bus which allows communication speeds of up to 1 Mbaud. CAN was adopted by all major car manufacturers and became a standard for in-vehicle communication in the automotive industry.
No intrinsic security mechanism was included in the CAN protocol specification apart from the 15 bit CRC that only assures data integrity. This makes CAN vulnerable to simple sniff and replay attacks. Thus, if the authenticity or confidentiality of the transmitted data has to be assured this must be implemented at a higher level.
The design of the CAN arbitration mechanism makes it susceptible to DoS attacks. As stated before, messages with lower IDs have higher priority. Therefore, a malicious node could continuously send a message with id 0x000 to prevent other nodes gaining access to the bus. In addition, by sending several well-directed error flags, an attacker could disconnect target CAN nodes by using CAN automatic fault localization and confinement mechanisms. These problems have no solution except from redesigning the bus architecture. Nevertheless, since all these cause a DoS which is the logical equivalent of cutting the wires they are not considered as relevant as the sniff and replay attacks that can insert adversary packages at will.

Many new features were introduced to modern automobiles with the purpose of increasing reliability, safety and user experience, e.g.: passive keyless entry, immobilizers, multimedia, tire pressure monitoring, telematics, vehicle diagnostics. However, recent research has shown that some of these systems have vulnerabilities that can be exploited by parties looking for financial gains or even having the malicious intent to cause accidents. In-vehicle networks evolved as a consequence of the continuously increasing system complexity. Unfortunately, once infiltrated, these networks provide little to no protection against attacks. With the increase in the complexity of vehicular embedded systems comes an increase in the overall code size and, inevitably, a greater value of the intellectual property. This is why vehicle manufacturers have focused on devising methods for preventing unauthorized software changes in the marketed components. However, it is common knowledge that chip tuning services can be bought from unauthorized individuals for the majority of car makes and models.
The impact that existing vulnerabilities may have when they are used by attackers, to endanger driver safety or to induce financial losses, underline the necessity of increasing the focus on the security of these systems. In order to counteract possible attacks it is essential to have a clear idea on who the adversaries are, what they can do and what is their purpose.

Improving algorithm performance using chip-specific features

To illustrate the performance achievable on automotive specific embedded platforms we evaluate a series of hash functions. Our choice is motivated by the ubiquitous nature of hash functions in security mechanisms. Several practical scenarios in which hash functions are involved can be imagined, e.g., software validation, embedded communications, etc. In particular firmware updates in embedded platforms (which require cryptographic hash functions for the protection of intellectual property, data integrity or non-repudiation) can directly benefit from performance improvements. Notably, digital signatures are employed to ensure that only an authentic firmware is programmed on a certain embedded device. Verifying signatures on a constrained embedded device can be a time consuming task especially as the size of the applications is continuously increasing. The bigger the size of data to be flashed, the longer it will take to compute its hash value needed for signature verification, consequently deploying the framework on thousands of devices delays component delivery for days or even longer. Thus minimizing the overhead of security mechanisms on the production process is beneficial. At the very least, secure communication between embedded devices relies on secure gateways that share secret keys and ultimately rely on MACs , i.e., keyed hashes.
The performed performance analysis in illustrates the constraint nature of some automotive-specific microcontrollers by evaluating their performance in relation to hash functions. The execution speed is bounded by the CPU architecture and reduced frequencies that characterize these platforms. Memory is also an issue as it is usually too limited to hold both the ECU firmware and a memory consuming cryptographic protocol. The performance can be enhanced when using devices equipped with multiple cores or cryptographic co-processors. Still, the overall performance of automotive microcontrollers is still too low to enable the deployment of mechanisms commonly use in computer networks without affecting system timing constraints. System constraints can be compensated by devising lightweight cryptographic primitives.

Application layer authentication

TESLA-based CAN authentication

As a first approach we look at the well known TESLA broadcast authentication protocol. The main argument for choosing a TESLA like protocol in our research is that there is no better solution to perform broadcast authentication without secret shared keys or public key primitives. Also, there is no result so far, to best of our knowledge, that points out clear technical limits on using TESLA like protocols on CAN networks. Thus, we provide clear experimental results on two automotive microcontrollers located somewhat on the extremes of computational power in terms of memory and bus speed: a Freescale S12 equipped with an XGATE coprocessor and an Infineon TriCore.

The results presented here are relevant as the authentication delay is critical for control scenarios. This is different to the usual sensor-network scenario where TESLA like protocols are frequently used because in sensor networks other constraints are more prevalent. For example, energy consumption is a critical issue in sensor networks, but usually for ECUs inside a car this is not a main concern since controllers do not strongly rely on small batteries. The most critical part, in control systems where this protocol is mostly used, is the authentication delay, i.e., how fast a packet can be deemed as authentic. In particular we must assure that a node, if the bus is not taken by a higher priority message, is able to transmit the message and the message can be checked for authenticity as soon as possible. This condition is initially limited by the computational power, but as checking for authenticity can happen only as soon as the disclosure delay expires and the next element of the chain is committed, this also depends on the structure of the chain which is determined by the amount of memory available, and also by the bandwidth. While in sensor networks the disclosure interval is usually in the order of tens or hundreds of milliseconds here we bring this delay lower by 2 to 3 orders of magnitude. Depicting an optimal protocol setting in this context is not straight forward and we study several trade-offs.

One-time signatures

Here we explore the possibility of using one-time signatures for assuring broadcast authentication at the application layer of CAN based. We find the enhanced Merkle and HORS signatures to offer different trade-offs, the first is more efficient in terms of memory, while the second is more efficient in terms of signature size and verification time. Indeed, with the HORS signature we exhibit good improvements in the authentication delay. The enhanced Merkle signature also has certain advantages. More concrete, the size of the messages is quite small in most broadcast scenario since CAN frames carry small data from sensors and actuators and this signature allows message recovery, thus small messages can be embedded in the signature. Finally, both signature schemes can be efficiently paired with time synchronization to reduce the overhead to re-initialize the public keys, which would otherwise require expensive authentication trees.

LiBrA-CAN

LiBrA, the third approach that we employed is based on splitting keys between groups of nodes. The proposed protocol is efficient when the number of nodes is low and the corrupted nodes are in minority. We expect this to be the case in many auto-motive scenarios where, although the number of ECUs may be high, the numbers of manufacturers from which they come may not be high and distributing trust between several groups is an acceptable solution. If the number of nodes is too high we see only two resolutions: public key cryptography (with the drawback of high computa- tional requirements, at least 2 orders of magnitude) or TESLA like protocols with the drawback of authentication delays.

Physical layer authentication

When the application layer approach is not a viable alternative, the physical layer may be used to provide source authentication. The methodology we described here proved to be usable in distinguishing the source of messages without any modifications on the software that the node is running or of the network that it is part of. This may set a distinct perspective on assuring broadcast authentication in CAN networks, an environment where cryptographic techniques are currently absent and will be hard to implement due to the various constraints. Besides detecting intrusions, this technique may also be useful for forensic purposes as event data recorders are closer to be mandatory for newly produced automobiles.

Conclusions

Several solutions for providing authenticated communication over CAN buses have been studied. Some of them focus on the application layer and can provide efficient authentication but at the cost of a higher bus load. When the

increasing the busload is an issue, the physical layer can be used to extract unique signal signatures in order to achieve sender identification.
A decision on what protocol should be used in real-world vehicular applications can be taken only by manufacturers and by means of consortium. The results presented here open road in this direction by proposing new protocols or trade-offs and giving clear hints on the constraints and performances that can be achieved.
While the approaches presented here were tested on CAN buses they could also be used on other buses such as FlexRay or BroadR-Reach (emerging Ethernet physical layer standard designed for use in automotive applications).
The results presented here were published in a series of papers co-authored by the author of this thesis.